Grounding Everything: Emerging Localization Properties in Vision-Language Transformers

Walid Bousselham[1,2] Felix Petersen[3] Vittorio Ferrari[4] Hilde Kuehne[1,2,5]
[1]University of Bonn, [2]Goethe University Frankfurt, [3]Stanford University, [4]Synthesia.io, [5]MIT-IBM Watson AI Lab

Abstract

Vision-language foundation models have shown remarkable performance in various zero-shot settings such as image retrieval, classification, or captioning. But so far, those models seem to fall behind when it comes to zero-shot localization of referential expressions and objects in images. As a result, they need to be fine-tuned for this task. In this paper, we show that pretrained vision-language (VL) models allow for zero-shot open-vocabulary object localization without any fine-tuning. To leverage those capabilities, we propose a Grounding Everything Module (GEM) that generalizes the idea of value-value attention introduced by CLIPSurgery [17] to a self-self attention path. We show that the concept of self-self attention corresponds to clustering, thus enforcing groups of tokens arising from the same object to be similar while preserving the alignment with the language space. To further guide the group formation, we propose a set of regularizations that allows the model to finally generalize across datasets and backbones. We evaluate the proposed GEM framework on various benchmark tasks and datasets for semantic segmentation. GEM not only outperforms other training-free open-vocabulary localization methods, but also achieves state-of-the-art results on the recently proposed OpenImagesV7 large-scale segmentation benchmark. [1]

1. Introduction

Vision-Language models, trained on large-scale web-based datasets such as WIT-400M [29], LAION400M [30], or metaclip-400M [35] with image-text supervision only, have so far shown a remarkable set of capabilities. These models such as CLIP [29], OpenCLIP [30], BLIP [15], or recently MetaCLIP [35] exhibit the ability to generalize to a broad range of downstream tasks like zero-shot image classification [6, 12, 29], visual question answering [13], action recognition [38, 40], image captioning [15, 16], and view synthesis [11]. However, models trained with image-level objectives such as contrastive loss, image-text matching, or image captioning struggle to maintain their zero-shot capabilities for tasks related to visual localization [17, 42]. Even worse, when prompting such models, e.g., for specific objects, they often exhibit an inverse vision-language relation: the prompt embedding has a larger distance from image patches containing the object compared to patches of the background, as shown in Figure 1.

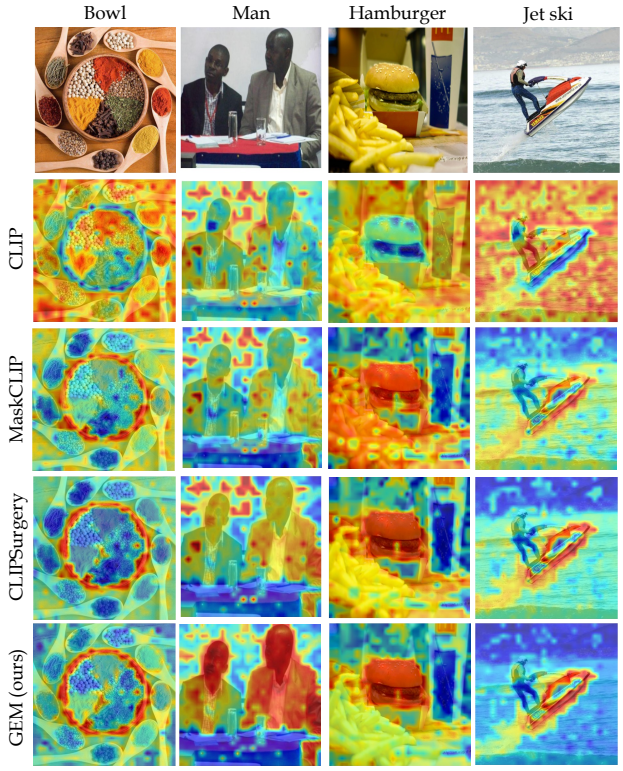


Figure 1. Qualitative results of training-free methods: given a text prompt, the similarity of each image token with the prompt is calculated (red:high, blue:low). The proposed GEM method provides improved grouping and alignment compared to other approaches.

In order to leverage vision-language models (VLMs) to localize objects in an open-vocabulary setting, different sets of approaches have been proposed. The first line of work trains models to detect or segment regions in an im-

[1]Code is available at https://github.com/WalBouss/GEM.

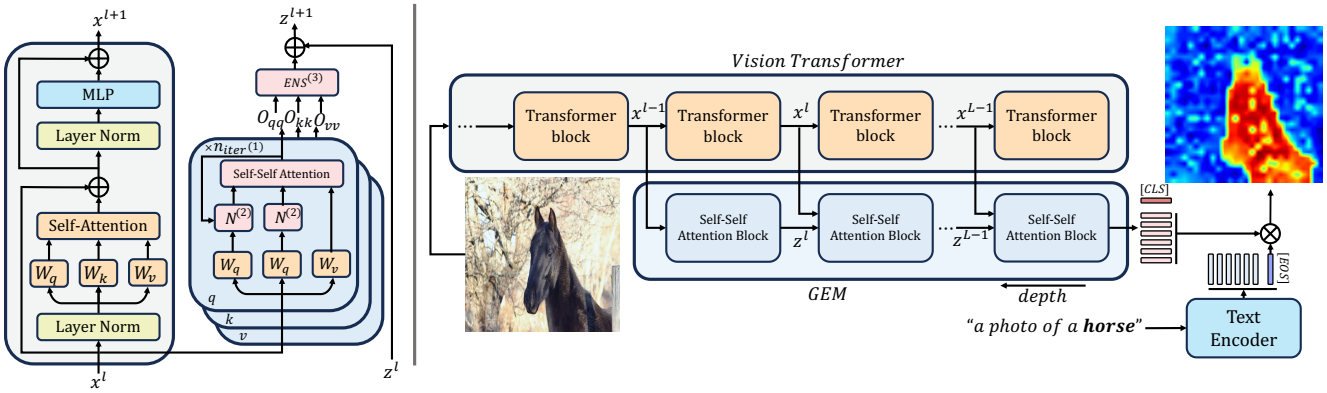


Figure 2. **Grounding Everything Module architecture:** Left: overview of the proposed generalized self-self attention block including n_{iter} iterations$^{(1)}$ and L^2 normalization $N^{(2)}$. The output of the q-q, k-k, and v-v projection is ensembled before applying the skip connection$^{(3)}$. Right: the output of self-self attention blocks is aggregated in parallel to the vision transformer in an alternative pathway. The localization is obtained by the dot product between the patch token output of the GEM and the CLS embedding of the text encoder.

age and then uses the vision-language information to label those regions, e.g., OVSeg [18] or OpenSeg [10]. A second line of work starts from a pretrained vision-language backbone and fine-tunes the model to improve localization, e.g., PACL [26] or SegCLIP [24]. Contrastingly, a third line of work recently emerged that focuses on leveraging the inherent localization capabilities of models trained on image-level objectives without requiring annotations or retraining, namely MaskCLIP [42], CLIPSurgery [17] and very recently SCLIP [33]. These training-free approaches try to process visual patches of the original model in a way that keeps them aligned to the language space. MaskCLIP showed that removing the MLP of the last layer avoids the vision-language inversion (see Figure 1). CLIPSurgery extends the pretrained ViT backbone of the CLIP model by a so-called "surgery pathway" which accumulates the value-value attentions of the original backbone over several layers. While adding the surgery pathway leads to significant performance improvements, it has been an open question how this mechanism improves the overall processing.

In this paper, we extend CLIPSurgery by analyzing the properties that result in the observed characteristics, connecting them to clustering algorithms, and eventually enforcing them within a new, generalized self-self attention architecture. While there are CNN-based VLMs, the focus in this work is given to VLMs using ViTs as image encoder.

First, we show that the value-value attention can be generalized to any self-self attention as key-key, query-query, or value-value representations show similar characteristics. Practically, we show that any form of self-self attention increases similarity among groups of similar tokens, compared to the standard q-k attention. To control the group formation, we propose a set of regularizations: first, we L^2 normalize the projected vectors; second, we combine this with an adaptive temperature τ for the proposed self-self attention operation, showing that the combination of those two elements results in good performance across all setups without the need for hyperparameter tuning. Third, we show that repeating the self-self attention several times further increases the group formation. Finally, we ensemble over all self-self attention types to allow for an integration of all cues. An overview of the resulting Grounding Everything Module (GEM) architecture is shown in Figure 2.

We evaluate the proposed method on two challenging tasks, open-vocabulary zero-shot semantic segmentation and zero-shot point prediction. For the first task, we leverage PascalVOC [9], PascalContext [25], as well as the ADE20K [41] dataset. For the second task, we employ the large-scale OpenImages V7 [1] dataset with almost 6K annotated classes. In all cases, we show improved results over all current training-free methods [17, 42] and competitive results in comparison to other approaches that require some form of fine-tuning [24, 36, 37]. We show that training-free methods in general and the proposed approach in particular are superior to all other approaches on the zero-shot point prediction on the OpenImages V7 dataset, reporting state-of-the-art results on this challenging task.

We summarize our contributions as follows: (1) Inspired by Li et al. [17], we show that self-self attention can be used as a technique for training-free open-vocabulary referential expression localization and segmentation based on pretrained vision-language models. (2) We propose the Grounding Everything Module (GEM) as a combination of self-self attention together with a set of regularizations that allows to generalize over a range of VL models and datasets. (3) We provide an in-depth evaluation of our model and training-free methods in general, showing that they keep up or even outperform fine-tuned methods on large-scale open-vocabulary localization tasks.

2. Related Works

The success of large-scale vision-language models like CLIP has sparked interest in leveraging their abilities for tasks like open-vocabulary object localization.

Given the lack of ad-hoc localization properties of VLMs, one line of approaches focuses on localization first, e.g., by training a region-proposal detector or a segmentation network [14]. They then use the respective VLMs as a form of post-process labeling by computing the correlation of the respective regions with the text prompt. A representative example is OpenSeg [10], which fine-tunes a model using class-agnostic masks and image-text pair data based on ALIGN [12]. Similarly, OVSeg [18] consists of one segmentation model trained to generate class-agnostic masks in an open-vocabulary fashion, and one CLIP model adapted to classify these masks. MaskCLIP$^{(3)}$ [7] adopts a similar strategy by using a Class-Agnostic Mask Proposal Network followed by a visual encoder based on CLIP to both refine the mask prediction and classify it. By relying on a localization model with a closed set vocabulary, i.e., not trained on a web-scale dataset with a large vocabulary, the classification performance is focused on the vocabulary of that model. Recently, GroundingSAM was proposed as a combination of GroundingDINO [22] and SAM [14]: GroundingDINO is a model that leverages various sources of region-level supervision, such as masks and bounding boxes available for different vision tasks to train a general-purpose localizer, and SAM generates segmentation masks from the bounding boxes generated by GroundingDINO. Combining the supervision from various tasks allows these models to be trained on millions of samples with fine-grained supervision, thus, achieving good performance for a large set of tasks.

Alternatively, some works propose to adapt the VLM architecture and training process to favor the emergence of localization. SegCLIP [24] and GroupViT [36] modify the ViT architecture by interleaving regular transformer blocks with grouping blocks that allow the grouping of semantically similar tokens into learnable group tokens used to compute the contrastive loss with the text. Similarly, ViL-Seg [20] and OVSegmentor [37] use online clustering and Slot Attention [23], respectively, for grouping visual features into semantically coherent clusters and, in addition, exploit self-supervision [4, 5, 31] for refinement. ReCo [21] leverages a retrieval process to obtain finer supervision and PACL [26] trains a decoder on top of CLIP with a grounding loss. While these methods use image-caption pairs as supervision, they require heavy filtering of the dataset, like extracting common nouns, which makes the dataset lose its free-form text characteristic. Thus, such approaches do not fully benefit from the VLM's large-scale characteristics.

Some methods refrain from training and instead adapt pretrained VLMs to make them work on fine-grained localization tasks. MaskCLIP [42] proposes discarding the Multi-Layer Perceptron (MLP) of the last layer of the vision transformer and utilizing the final value projection to extract dense patch-level features. Building upon this concept, CLIPSurgery [17] introduces a novel pathway called the "surgery pathway" that operates in parallel with the original vision transformer (ViT) backbone of the CLIP model. It employs value-value instead of query-key attention and aggregates the output of multiple layers via residual connection. Following [42], here, the value-value attention is directly used without a subsequent MLP. To localize an object based on an input label or referential expression, the distance is computed between the token output of the last layer and the respective text embedding. This work builds upon this line of work and not only extends the value-value attention to a normalized self-self attention but also provides an in-depth analysis of the inner workings of the self-self attention mechanism.

3. Grounding with Self-Self Attention

In the following, we introduce the Grounding Everything Module (GEM) by first generalizing the concept of value-value attention [17] to a broader set of projections as self-self-attention and introduce an iterative extension that, together with a temperature regularizer, allows to control the formation of groups of visual features. Second, we consider the connection of the proposed self-self attention (and also CLIPSurgery's value-value attention) to clustering, showing in simulations that it acts as a form of soft clustering.

3.1. GEM: Grounding Everything Module

Self-Self Attention. We first review the concept of value-value attention, showing that, while it allows connecting features from the same semantic region, the same properties can be observed for key-key or query-query projections. CLIPSurgery [17] defines value-value attention as:

$$Attn_{vv} = \text{softmax}(V \cdot V^T), \quad O_{vv} = Attn_{vv} \cdot V \tag{1}$$

with $V = xW_v \in \mathbb{R}^{n \times d}$, where x are the n patch/token outputs by a ViT layer, each of dimension d, W_v is the learned value weight matrix of the original ViT backbone, and O_{vv} is the output of the value-value surgery block.

As a first step, we replace the value projection by either the query or the key projection taken from the original pathway. We, therefore, introduce a generalized self-self attention $Attn_{ss}$ as extension of the value-value attention as:

$$Attn_{ss} = \text{softmax}(xW_{proj} \cdot (xW_{proj})^T) \tag{2}$$

with $x \in \mathbb{R}^{n \times d}$ again representing the patch/token outputs by a ViT layer, and W_{proj} being any projection matrix of the respective ViT layer $W_{proj} \in \{W_v, W_q, W_k\}$. We evaluate the performance for each projection in Table 1 on the Pascal VOC and Pascal Context datasets (for evaluation details see Section 4.1 and see Section 6 evaluation on more backbones). The query-query and key-key attentions both lead to the same or improved performance compared to value-value attention. Compared to regular self-attention (query-

key attention) as used in the CLIP baseline, any self-self attention improves performances significantly. In Section 3.2, we discuss that this can be attributed to self-self attention increasing the similarity of already similar patch tokens, thus leading to cluster formation.

Normalization and Adaptive Temperature. In the self-self attention setting, projected tokens with larger norms disproportionately influence other tokens, regardless of their similarity with other visual tokens. We therefore propose an L^2-normalization for each projected token before computing self-self attention. We can further guide the cluster formation by introducing a temperature τ in the softmax formulation of the self-self attention $Attn_{ss}$ as:

$$\text{softmax}(a, \tau) = \frac{\exp\left(a_i \cdot a_j^T / \tau\right)}{\sum_l \exp\left(a_i \cdot a_l^T / \tau\right)} \tag{3}$$

where, $\cdot$ is the dot product operation. Assuming a zero-shot setting without access to labeled training or validation data, we aim to fix the temperature τ for the self-self attention so that it performs well without requiring hyperparameter tuning. Therefore, we propose an adaptive temperature using the average norm of the visual tokens before projection times the temperature originally used to train ViT as

$$\tau = \frac{n \cdot \sqrt{d}}{\sum_i ||x_i||_2}, \tag{4}$$

where n is the number of visual tokens and d the dimension of tokens, respectively. This combination of normalization and adaptive temperature improves the group formation and thus the localization as shown in Table 1. Further details on temperature ablation are available in Section 4.3.

Iterative Self-Self Attention. We propose to iteratively apply the proposed normalized self-self attention to facilitate the gradual refinement of the cluster formation of semantically related visual tokens. More formally, given input visual tokens denoted as $x \in \mathbb{R}^{n \times d}$ and a projection matrix $W_{proj} \in \mathbb{R}^{d \times d}$, the k-th iteration of our iterative self-self attention is described as:

$$\begin{aligned} p^0 &\leftarrow \frac{x W_{proj}}{\|x W_{proj}\|_2} \\ \tilde{p}^k &\leftarrow \text{softmax}(p^{k-1} \cdot (p^{k-1})^T, \tau) \cdot p^{k-1} \\ p^k &\leftarrow \frac{\tilde{p}^k}{\|\tilde{p}^k\|_2} \end{aligned} \tag{5}$$

where p^0 is the is the normalized projection input to the self-self attention operation, p^k is the output of the k-st application of the self-self attention as described in Eq. 5, multiplied with the output of the $k-1$th iteration and divided by its norm. After K iterations of self-self attention, the output (for the W_{proj} projection), denoted O_{ss}, is obtained by applying the assignment to the values since they are trained to carry semantic information:

$$O_{ss} = \text{softmax}(p^K \cdot (p^K)^T, \tau) \cdot V \tag{6}$$

Practically, we found that one additional iteration, (i.e., applying the self-self attentions once to the projected tokens as in Eq. 5 and once on the values V as in Eq. 6) is sufficient for most cases. We therefore, fix the iterations to one throughout the paper and provide an ablation in Section 4.3.

qkv-Ensemble. We finally ensemble the iterative self-self attention applied to the query, key, and value projections to integrate the information brought by the different projections. The output of the qkv-ensemble attention is:

$$O_{qkv} = \frac{O_{qq} + O_{kk} + O_{vv}}{3} \tag{7}$$

where O_{qq}, O_{kk}, O_{vv} are the outputs based on the projection matrices W_q, W_k, W_v. Table 1 shows the improvement by ensembling over the three normalized projections.

Projection	Norm.+Temp.	VOC	Context
CLIP	-	10.4	7.7
v-v	✗	41.9	30.5
k-k	✗	43.9	31.0
q-q	✗	43.8	30.8
qkv	✗	43.1	30.7
v-v	✓	44.4	31.9
k-k	✓	44.8	32.0
q-q	✓	44.7	31.5
qkv	✓	45.1	32.3

Table 1. mIoU for v-v, k-k, and q-q attention and qkv ensemble on PascalVOC and PascalContext with and without L^2-Norm and adaptive temperature (see Table 6 in the SM for more backbones).

3.2. Self-Self Attention for Clustering

Practically, self-self attention calculates the similarity between every pair of visual tokens. These similarities are then employed in the transformer as weights in a weighted sum operation used to update the tokens. As a result, tokens are updated with a weighted sum of tokens, with more weight on more similar tokens, converging to a respective mean representation corresponding to a cluster center. To illustrate this effect, we conducted a simulation based on a set of 20 d-dimensional random Gaussian vectors representing the input tokens x and a random linear projection as W_{proj}. We iteratively apply the proposed self-self-attention including normalization and with different temperature parameters on the 20 vectors. As shown in Figure 3, this process leads to a clustering of the 20 vectors using self-self attention. Moreover, with higher temperature, as well as with more iterations, there are fewer and larger

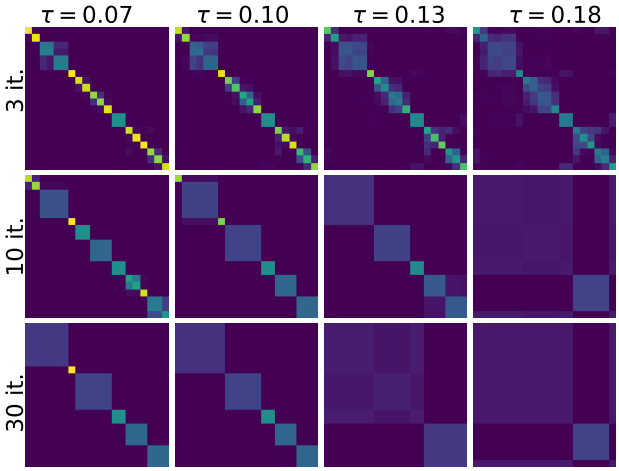


Figure 3. Evaluation of self-self attention for different numbers of iterations and temperature on a set of 20 random vectors (dim=5). As the number of iterations / temperature τ increase, self-self attention forms larger clusters.

clusters, while fewer iterations and a lower temperatures enforce more and smaller clusters. In practical scenarios, complex datasets with many classes per image might benefit from a less clustered feature space, consequently requiring fewer iterations. While we do not need to backpropagate through the clustering, as we perform no training, we want to remark that it is indeed differentiable, paving the way for research on fine-tuning GEM models. Moreover, the illustrated random-projection based clustering process (Eq. 5) is relevant for the field of differentiable relaxations [2, 27, 28], where other differentiable clustering algorithms, e.g., based on spanning-forests [32], have been proposed.

We can further connect this behavior to the Lipschitz constant of the used projections to the self-self attention's grouping effect. More formally, in finite dimension, any linear operator is Lipschitz continuous and, under the l_2 norm, its Lipschitz constant is given by the spectral norm of the weight matrix—i.e., the largest singular value of the weight matrix. Let $W_{proj} \in \mathbb{R}^{d_1 \times d_2}$ denote the weight matrix of the linear projection and C its Lipschitz constant, we have:

$$\forall x_1, x_2 \in \mathbb{R}^{d_1}, ||x_2 W_{proj} - x_1 W_{proj}||_2 \leq C||x_2 - x_1||_2$$
$$C = \max_{||x||_2 \neq 0} \frac{||xW_{proj}||_2}{||x||_2} \tag{8}$$

For the self-self attention to reinforce the similarity of tokens already close to each other (i.e., representing the same object), we need the self-self attention projection to pull these tokens closer to each other. In other words, the linear projection must be a contraction, i.e., $C < 1$. Conversely, a Lipschitz constant that is too small will result in unrelated tokens being mixed together. For the here analyzed models, we validated the Lipschitz constant across all projections as follows: $C_{value} = 0.51 \pm 0.073$, $C_{key} = 0.63 \pm 0.091$ and $C_{query} = 0.66 \pm 0.104$. Moreover, the similarity between tokens (i.e., grouping) in the self-self attention is further enforced by doing multiple (per head) parallel projections, all with a Lipschitz constant < 1, as seen in value-value, query-query, or key-key projections. Hence, tokens that are similar under all the projections will share information.

For a comparison of the proposed self-self attention to Kmeans, we refer to Table 8 in the SM.

4. Evaluation

4.1. Setup

Datasets. **PascalVOC** [9] provides segmentation masks for 20 classes in natural images, focusing on common objects like cats, dogs, cars, and airplanes. An image contains 1.5 classes on average. Following previous works [17], [42], we evaluate on the validation set. **Pascal Context** [25] extends PascalVOC to 59 classes, supplemented by a background class. Compared to PascalVOC, it provides dense annotations for the whole scene. We evaluate on the test set, comprising 5 104 images with an average of 4.8 classes per image. **ADE20K** [41] is a scene parsing dataset with 150 fine-grained classes. We use its validation set comprising 2 000 images with an average of 9.9 classes per image. **OpenImages-V7** [1] provides annotations for a large set of images with a widely diverse spectrum of objects and real-world scenarios. For the following evaluation, we leverage the point-wise annotations of the validation set, with 36 702 images featuring 5 827 distinct class labels. For each object, a set of positive and negative point annotations is provided. For this evaluation, for each image, we consider only classes present in the image.

Implementation. For all experiments, we use the original pretrained weights of the respective vision-language models as provided by their authors, namely CLIP [29], OpenCLIP [6], an open-source replication of CLIP, and BLIP [15] and MetaCLIP [35]. We apply the GEM architecture with the proposed normalization and adaptive temperature and one iteration for all datasets and models. We compute a dense semantic segmentation prediction for each image as follows: For each patch we compute the cosine similarity between the patch tokens of the vision encoder and the text embedding of each dataset class name. We use the following prompt as input for the text encoder: *"a photo of a {class name}"*. Finally, we upsample the segmentation predictions to the input image size via bilinear interpolation. If the input image is larger than the one used during the model training, we adapt the learned positional embeddings via bicubic interpolation. Note that ***we do not perform any retraining nor fine-tuning*** of the vision-language model, showing the possibility to localize queries with models trained on image-level only and without the need for any localization information during training or fine-tuning.

Method	Encoder	Model	Training Dataset		Loc.	Loc.	mIoU		
			Pretraining	Annotation	anno.	FT	VOC	Context	ADE
SPNet [34]	ResNet101	scratch	COCO, VOC, Context	SM	✓	✓	15.6†	4.0†	-
ZS3Net [3]	ResNet101	scratch	VOC, Context	SM	✓	✓	17.7†	7.7†	-
MaskCLIP(3) [7]	ViT-B/16	CLIP	COCO	SM	✓	✓	-	45.9	23.7
OpenSeg [10]	ENet-B7+FPN	ALIGN	COCO, Loc. Narr	IT, UM	✓	✓	72.2	48.2	24.8
CLIP-ES [19]	ResNet101	CLIP	COCO-Stuff-171	IC	✓	✓	75.0	-	-
OVSeg [18]	ViT-B/16	CLIP	COCO-Stuff-171	UM	✓	✓	94.5	55.7	29.6
ViL-Seg [20]	ViT-B/16	scratch	GCC	IT	✗	✓	34.4†	16.3†	-
GroupViT [36]	ViT-S/16	scratch	GCC+YFCC	IT	✗	✓	52.3	22.4	9.2
SegCLIP [24]	ViT-B/16	CLIP	CC, COCOcap	IT, ICap	✗	✓	52.6	24.7	8.7
OVSegmentor [37]	ViT-B/16	DINO	GCC	IT	✗	✓	53.8	20.4	5.6
PACL [26]	ViT-B/16	CLIP	WIT-400M +CC12M, YFCC	IT	✗	✓	72.3	50.1	31.4
CLIP [29]	ViT-B/16	CLIP	WIT-400M	IT	✗	✗	10.4	7.7	1.7
MaskCLIP(2) [8]	ViT-B/16	scratch	YFCC	IT	✗	✗	-	17.2	10.2
MaskCLIP [42]	ViT-B/16	CLIP	WIT-400M	IT	✗	✗	-	25.5	-
MaskCLIP* [42]	ViT-B/16	CLIP	WIT-400M	IT	✗	✗	28.6	23.8	10.2
CLIP Surgery [17]	ViT-B/16	CLIP	WIT-400M	IT	✗	✗	-	29.3	-
CLIP Surgery* [17]	ViT-B/16	CLIP	WIT-400M	IT	✗	✗	41.2	30.5	12.9
GEM (our)	ViT-B/16	CLIP	WIT-400M	IT	✗	✗	**46.2**	**32.6**	**15.7**
GEM (our)	ViT-B/16	MetaCLIP	metaclip-400M	IT	✗	✗	**46.8**	**34.5**	**17.1**

Table 2. **Comparison on zero-shot semantic segmentation:** Models marked with † are evaluated under relaxed constraints, specifically on a subset of unseen classes. * indicates our evaluation. We use the following abbreviations: COCO: COCO2017, GCC: Google Conceptual Captions 12M, YFCC: YFCC15M, CC: Conceptual Captions, COCOCap: COCO Captions. SM: segmentation mask, IT: image-text, ICap: image caption, UM: unlabeled mask, IC: image classes.

Evaluation. Zero-shot segmentation entails the ability of a model to segment objects in an image without prior training on the evaluated classes. Following common practice [24, 36, 37], we evaluate **zero-shot semantic segmentation** by the mean Intersection over Union (mIoU) for PascalVOC, PascalContext and ADE20K. Following [36], we resize each input image to have a shorter-side length of 448. For PascalVOC we predict only the foreground classes and get the background by thresholding the softmax-normalized-similarity between the patch tokens and the text embedding of each class name (using a fixed threshold of 0.85). For Pascal Context, we follow common practice and evaluate only on the 59 foreground classes. ADE20K provides a dense annotation and therefore does not consider background. For **zero-shot point prediction**, we leverage the OpenImages-V7 dataset. For each positive class in the image, we scale the prediction between 0 and 1 and use a fixed threshold of 0.5 to obtain the predicted mask. We follow the dataset guidelines [1] and compute the IoU over the sets of positive and negative ground-truth points for all classes in the respective image and average across the dataset, denoted p-mIoU.

4.2. Comparison to State-of-the-Art

Zero-Shot Semantic Segmentation. We first compare the proposed approach for the task of zero-shot semantic segmentation. We consider three groups of state-of-the-art methods in open-vocabulary segmentation: First, we consider methods trained or fine-tuned with some form of labeling information, e.g., hand-annotated segmentation masks, such as OpenSeg [10], CLIP-RIS [39], MaskCLIP(3) [7], and OVSeg [18]. Note that most of those methods are trained on similar domains and vocabulary as the test datasets. Second, we report the performance of models trained explicitly for segmentation on image-caption pair annotations, i.e., GroupViT [36], OVSegmentor [37], SegCLIP [24], and ViL-Seg [20]. While those methods do not use location annotation, they still fine-tune existing backbones for the task of localization. We also consider PACL [26] in this group, which trains a decoder on top of CLIP using a loss designed for patch grouping. Finally, we directly compare against methods that perform training-free zero-shot segmentation, namely MaskCLIP, MaskCLIP(2), and CLIPSurgery. We report the mIoU in Table 2. The proposed method consistently outperforms all training-free approaches. Further, training-free methods are able to outperform vision-language models fine-tuned specifically for localization on the more complex dataset PascalContext and ADE20K surpassing all other models except PACL.

Zero-Shot Point Prediction. To evaluate the true open-vocabulary qualities of the proposed method, we compare our method on the OpenImageV7 dataset with a vocabulary of almost 6k label classes to the strongest available trained or fine-tuned semantic segmentation models from Table 2, namely OVSeg, SegCLIP, and GroupViT, as well as to all training-free methods. Table 3 reports the p-mIoU

Method	Loc. anno.	Loc. FT	p-mIoU	fps
OVSeg* [18]	✓	✓	22.5	1.41
SegCLIP* [24]	✓	✗	32.1	21.39
GroupViT* [36]	✓	✗	36.5	24.61
CLIP [29]	✗	✗	27.6	42.10
MaskCLIP* [42]	✗	✗	42.0	42.43
CLIPSurgery* [17]	✗	✗	47.8	38.47
GEM-CLIP (our)	✗	✗	**50.9**	37.24
GEM-MetaCLIP (our)	✗	✗	**51.9**	37.24
GroundingSAM* [14, 22]	✓	✓	53.3	0.59
GEM-SAM-CLIP (our)	✓	✗	**53.4**	0.45
GEM-SAM-MetaCLIP (our)	✓	✗	**55.2**	0.45

Table 3. **Comparison on zero-shot point prediction:** We choose the best performing available approaches for ADE20K from Table 2 and apply them on the OpenImagesV7 dataset. We further report inference speed as fps for each model on one Nvidia A6000.

and the inference speed for all methods. First, we observe that training-free methods, i.e., GEM, CLIPSurgery, and MaskCLIP, provide a substantially better performance than trained or fine-tuned methods supporting the intuition that fine-tuning on a smaller, but cleaner dataset reduces the vocabulary leading to lower performance on datasets with large vocabulary like OpenImagesV7 (see the SM for qualitative comparisons). For completeness, we also report numbers for the recently released GroundingSAM architecture [14, 22], which uses labeled bounding boxes and class-agnostic masks during training. To directly compare, we use the output of GEM to label masks generated by prompting SAM with a grid of points. Even in this case, the our proposed training-free method outperforms the fine-tuned GroundingSAM architecture.

4.3. Ablation

Temperature. To assess the performance of the proposed components, we first regard the impact of normalization and adaptive temperature. To this end, we compute the proposed adaptive temperature following in Section 3, i.e., $\tau = \frac{n \cdot \sqrt{d}}{\sum_i ||x_i||_2}$ and report the segmentation performance for multiples of this temperature for ViT-B/16 on two datasets, PascalVOC and PascalContext in Figure 4. We observe that, the combination of normalization and temperature achieves the highest mIoU consistently across both datasets Moreover, it achieved this performance consistently with the proposed temperature (multiplication factor equal to 1), indicating the effectiveness of our proposed heuristic as well as the robustness and generalizability as it allows to adapt to the specific characteristics of the input vector.

Iterations. Second we consider the impact of the number of iterations on the performance of the system. To this end, we evaluate PascalVOC and PascalContext for $K = \{0, 1, 2, 3\}$ iterations and also compare to the performance of CLIPSurgery in Table 4. Overall, we observe that more iterations, namely two, slightly improve performance for VOC, a dataset with few classes per image, and that fewer iterations work better for Context, a dataset with more classes per image. While the number of iterations can be used as a tunable hyperparameter, we fixed it throughout the paper to one to allow for a real zero-shot scenario.

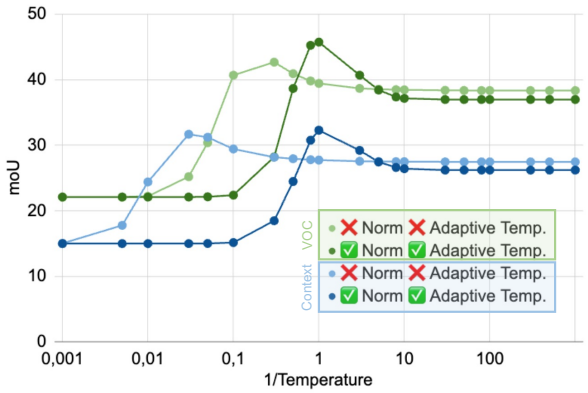


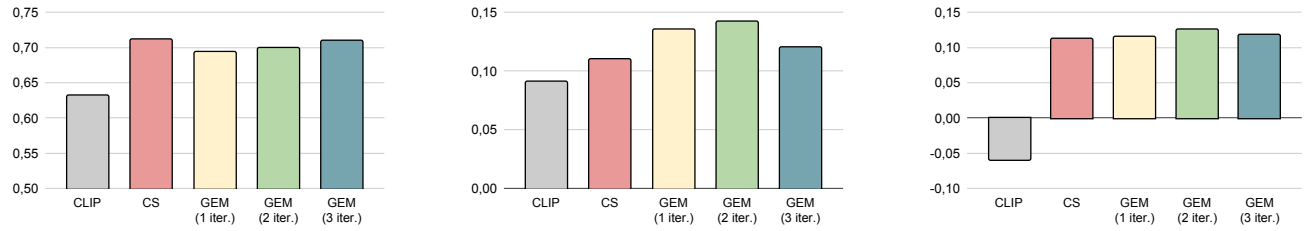
Figure 4. Evaluation of localization performance for CLIP ViT-B/16 (left) for the PascalVOC and PascalContext dataset with and without normalization and adaptive temperature. The proposed temperature provides best results in both settings.

	CS	GEM			
iter	0	0	1	2	3
VOC	41.2	45.1	45.5	**46.2**	45.6
Context	30.5	31.5	**32.6**	31.9	31.1

Table 4. Influence of iterations for the self-self-attention in the GEM architecture. More iterations are better for fewer classes per image, less iterations work better for more classes.

4.4. Architecture and Model Size

To explore the generalization abilities of GEM, we extend our analysis beyond the ViT-B/16 model to ViT-B/32 and ViT-L/14 as well as to other vision-language backbones, namely OpenCLIP [30], an open-source replication of CLIP. We further investigate the generality on architectures similar to CLIP, vi7., BLIP [15], which is trained with a multi-task objective, and MetaCLIP [35], the currently best-performing zero-shot classification model. Table 5 shows the results for the different models and backbones. It shows that the GEM method consistently improves localization performance across all model sizes. As expected, for a fixed ViT-B size, increasing the patch size from 16 to 32 reduces the performance slightly. We further observe that larger ViT-L encoders do not yield better localization performance. Specifically, GEM-ViT-B/16 consistently outperforms its larger counterparts GEM-ViT-L/14. Finally, BLIP, as the only model trained with multi-objectives, tends to perform worse in localization than models trained solely with an image-text contrastive loss.



(a) Patch-patch similarity (b) Object-Background Contrast (c) Text-Object Contrast

Figure 5. Analysis of localization properties of CLIP, CLIPSurgery, and GEM. Each metric is computed on the training set of PascalVOC.

Backbone	Model	VOC	Context	V7
ViT-B/16	CLIP	46.2	32.6	50.9
	OpenCLIP	43.1	31.7	49.9
	BLIP	42.8	23.5	45.2
	MetaCLIP	**46.8**	**34.5**	**51.9**
ViT-B/32	CLIP	**40.5**	27.0	46.6
	OpenCLIP	39.3	23.9	45.5
	MetaCLIP	38.2	**28.2**	**46.7**
ViT-L/14	CLIP	44.6	**28.6**	**46.3**
	OpenCLIP	40.0	27.5	42.4
	BLIP	32.1	21.4	44.9
	MetaCLIP	**45.7**	26.9	40.9

Table 5. Evaluation of the GEM architecture on various pretrained vision-language backbones showing better performance for smaller patch size (ViT-B/16 compared to ViT-B/32) and architecture (ViT-B compared to ViT-L).

4.5. Analysis of Localization Properties

In Figure 5, we assess the factors contributing to the localization performance of the proposed method. We assume that for good localization in vision-language models, two essential properties must be fulfilled: visual distinctiveness as the meaningful grouping of visual feature representations, and vision-language alignment as the alignment of these groups with the textual descriptions encoded by the language model. To capture the visual distinctiveness, we consider two metrics: first, (a) patch-patch similarity, the similarity among patches within each layer, as well as, second, (b) object-background contrast, the contrast between foreground and background patch tokens. For this metric, we leverage the segmentation masks of the training set of the PascalVOC dataset [9]. For vision-language alignment, (c), we measure the contrast between the similarity of the text embedding, the text-[EOS] token, and the foreground patch embeddings, and the similarity of the text-[EOS] token and the background patches.

We see an increase in patch-patch similarity (a) from CLIP to CLIPSurgery most likely due to the clustering induced by the self-self attention and the slight decrease from CLIPSurgery to GEM due to the added normalization and temperature. This is recovered by the higher object-background contrast (b) of GEM over CLIPSurgery and CLIP, pointing to the effective clustering of visual tokens and their ability to distinguish between distinct objects. Finally, the analysis of text-object similarity demonstrates improved alignment between visual tokens and text embeddings, enhancing vision-language integration.

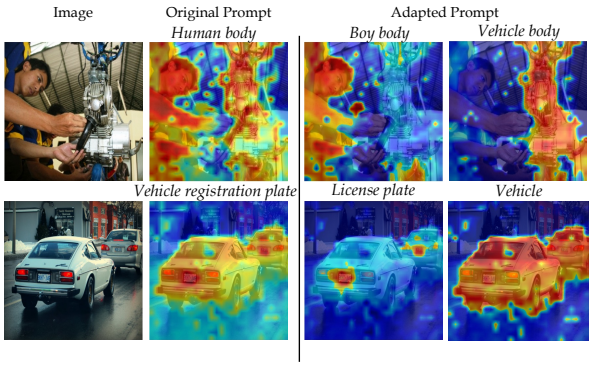


Figure 6. Failure cases and adapted prompts from [1].

4.6. Analysis of Failure Cases

Finally, we review some failure cases in Figure 6 (see SM for more examples). For the first image, when prompted with *"Human body"*, the model segments both the human and the car body. For the second image, prompted with *"Vehicle registration plate"*, the model focuses on both the car and registration plate. This effect can be mitigated by decoupling the emphasized word, as shown for the adapted prompts in Figure 6. We attribute this failure to the text encoder, paving the way for future research.

5. Conclusion

In this work, we proposed GEM, the Grounding Everything Module, which leverages latent localization ability of pretrained vision-language models, enabling SOTA open-vocabulary segmentation without the need for re-training or fine-tuning. GEM utilizes a novel self-self attention pipeline to extract localization information from VLMs. Despite being zero-shot, i.e., without training and hyperparameter tuning, GEM improves the SOTA across all evaluated datasets for training-free methods, and improves the SOTA on OpenImagesV7 even across training methods.

Acknowledgments

Walid Bousselham is supported by the German Federal Ministry of Education and Research (BMBF) project STCL-01IS22067.

References

[1] Rodrigo Benenson and Vittorio Ferrari. From colouring-in to pointillism: revisiting semantic segmentation supervision. *arXiv preprint arXiv:2210.14142*, 2022. 2, 5, 6, 8

[2] Mathieu Blondel and Vincent Roulet. The elements of differentiable programming. *arXiv preprint arXiv:2403.14606*, 2024. 5

[3] Maxime Bucher, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. *NeurIPS*, 2019. 6

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 3

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 3

[6] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *CVPR*, 2023. 1, 5

[7] Zheng Ding, Jieke Wang, and Zhuowen Tu. Open-vocabulary panoptic segmentation with maskclip. In *ICML*, 2022. 3, 6

[8] Xiaoyi Dong, Jianmin Bao, Yinglin Zheng, Ting Zhang, Dongdong Chen, Hao Yang, Ming Zeng, Weiming Zhang, Lu Yuan, Dong Chen, Fang Wen, and Nenghai Yu. Maskclip: Masked self-distillation advances contrastive language-image pretraining. In *CVPR*, 2023. 6, 1

[9] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *ICCV*, 2010. 2, 5, 8

[10] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *ECCV*, 2022. 2, 3, 6

[11] Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting nerf on a diet: Semantically consistent few-shot view synthesis. In *ICCV*, 2021. 1

[12] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 1, 3

[13] Aisha Urooj Khan, Hilde Kuehne, Chuang Gan, Niels Da Vitoria Lobo, and Mubarak Shah. Weakly supervised grounding for vqa in vision-language transformers. In *ECCV*, 2022. 1

[14] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *ICCV*, 2023. 3, 7

[15] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 1, 5, 7

[16] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 1

[17] Yi Li, Hualiang Wang, Yiqun Duan, and Xiaomeng Li. Clip surgery for better explainability with enhancement in open-vocabulary tasks. *arXiv preprint arXiv:2304.05653*, 2023. 1, 2, 3, 5, 6, 7

[18] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *CVPR*, 2023. 2, 3, 6, 7

[19] Yuqi Lin, Minghao Chen, Wenxiao Wang, Boxi Wu, Ke Li, Binbin Lin, Haifeng Liu, and Xiaofei He. Clip is also an efficient segmenter: A text-driven approach for weakly supervised semantic segmentation. In *CVPR*, 2023. 6

[20] Quande Liu, Youpeng Wen, Jianhua Han, Chunjing Xu, Hang Xu, and Xiaodan Liang. Open-world semantic segmentation via contrasting and clustering vision-language embedding. In *ECCV*, 2022. 3, 6

[21] Shikun Liu, Shuaifeng Zhi, Edward Johns, and Andrew Davison. Bootstrapping semantic segmentation with regional contrast. In *ICLR*, 2021. 3

[22] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 3, 7

[23] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *NeurIPS*, 2020. 3

[24] Huaishao Luo, Junwei Bao, Youzheng Wu, Xiaodong He, and Tianrui Li. Segclip: Patch aggregation with learnable centers for open-vocabulary semantic segmentation. In *ICML*, 2023. 2, 3, 6, 7

[25] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *CVPR*, 2014. 2, 5

[26] Jishnu Mukhoti, Tsung-Yu Lin, Omid Poursaeed, Rui Wang, Ashish Shah, Philip HS Torr, and Ser-Nam Lim. Open vocabulary semantic segmentation with patch aligned contrastive learning. In *CVPR*, 2023. 2, 3, 6

[27] Felix Petersen. Learning with differentiable algorithms. *arXiv preprint arXiv:2209.00616*, 2022. 5

[28] Felix Petersen, Hilde Kuehne, Christian Borgelt, and Oliver Deussen. Differentiable top-k classification learning. In *ICML*, 2022. 5

[29] 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. In *ICML*, 2021. 1, 5, 6, 7

[30] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *NeurIPS*, 2022. 1, 7

[31] Nina Shvetsova, Felix Petersen, Anna Kukleva, Bernt Schiele, and Hilde Kuehne. Learning by sorting: Self-supervised learning with group ordering constraints. In *ICCV*, 2023. 3

[32] Lawrence Stewart, Francis S Bach, Felipe Llinares López, and Quentin Berthet. Differentiable clustering with perturbed spanning forests. *NeurIPS*, 2023. 5

[33] Feng Wang, Jieru Mei, and Alan Yuille. Sclip: Rethinking self-attention for dense vision-language inference. *arXiv preprint arXiv:2312.01597*, 2023. 2

[34] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic projection network for zero-and few-label semantic segmentation. In *CVPR*, 2019. 6

[35] Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Demystifying clip data. *arXiv preprint arXiv:2309.16671*, 2023. 1, 5, 7

[36] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *CVPR*, 2022. 2, 3, 6, 7

[37] Jilan Xu, Junlin Hou, Yuejie Zhang, Rui Feng, Yi Wang, Yu Qiao, and Weidi Xie. Learning open-vocabulary semantic segmentation models from natural language supervision. In *CVPR*, 2023. 2, 3, 6

[38] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 1

[39] Seonghoon Yu, Paul Hongsuck Seo, and Jeany Son. Zero-shot referring image segmentation with global-local context features. In *CVPR*, 2023. 6

[40] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021. 1

[41] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 2019. 2, 5

[42] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *ECCV*, 2022. 1, 2, 3, 5, 6, 7