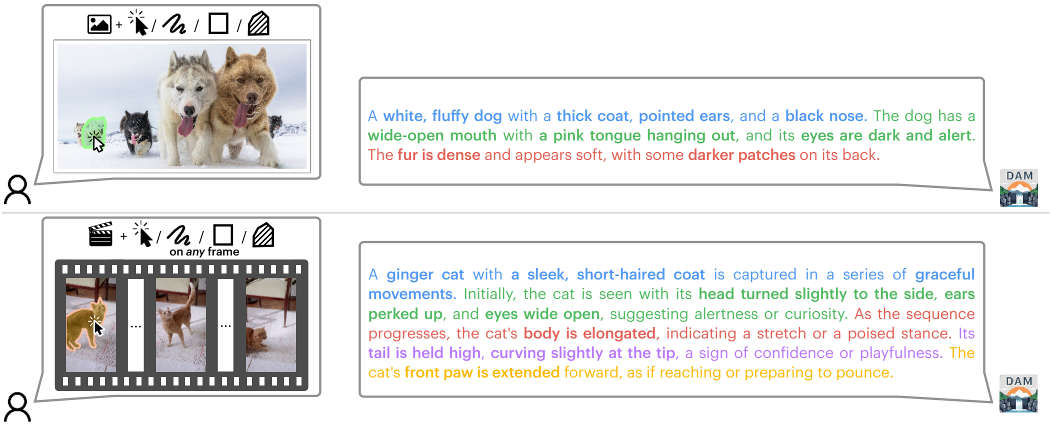
Describe Anything: Detailed Localized Image and Video Captioning

Long Lian[1,2], Yifan Ding[1], Yunhao Ge[1], Sifei Liu[1], Hanzi Mao[1], Boyi Li[1,2],
Marco Pavone[1], Ming-Yu Liu[1], Trevor Darrell[2], Adam Yala[2,3], Yin Cui[1]
[1]NVIDIA [2]UC Berkeley [3]UCSF



A **white, fluffy dog** with a **thick coat**, **pointed ears**, and a **black nose**. The dog has a **wide-open mouth** with **a pink tongue hanging out**, and its **eyes are dark and alert**. The **fur is dense** and appears soft, with some **darker patches** on its back.

A **ginger cat** with **a sleek, short-haired coat** is captured in a series of **graceful movements**. Initially, the cat is seen with its **head turned slightly to the side, ears perked up**, and **eyes wide open**, suggesting alertness or curiosity. As the sequence progresses, the cat's **body is elongated**, indicating a stretch or a poised stance. Its **tail is held high**, **curving slightly at the tip**, a sign of confidence or playfulness. The cat's **front paw is extended** forward, as if reaching or preparing to pounce.

Figure 1. **Describe Anything Model (DAM)** generates **detailed localized captions** for user-specified regions within **images** (top) and **videos** (bottom). DAM accepts various region specifications, including clicks, scribbles, boxes, and masks. For videos, specifying the region in *any frame* suffices.

Abstract

*We present the **Describe Anything Model (DAM)**, a novel vision-language model for detailed localized captioning (DLC) in images and videos. DAM uses a focal prompt to encode high-resolution details of user-specified regions and a localized vision backbone to merge these with global context. To address the limited availability of DLC data, we introduce a **Semi-supervised Learning (SSL) Data Pipeline (DLC-SDP)** that augments segmentation datasets with unlabeled web images. We also propose **DLC-Bench**, an attribute-based evaluation benchmark that bypasses the need for comprehensive reference captions. DAM achieves state-of-the-art results on 10 benchmarks spanning keyword, phrase, and detailed captioning tasks.*

1. Introduction

Image captioning has been a longstanding challenge in computer vision and NLP [18], as it requires understanding visual content and expressing it in natural language. While recent Vision-Language Models (VLMs) excel in image-level captioning, generating detailed captions for specific regions—especially in videos capturing dynamic content—remains open.

Most existing VLMs (*e.g.*, GPT-4o [49]) lack precise localization. Methods using 2D cues like bounding boxes [29, 69, 72, 85] tend to yield short phrases, while longer captions often introduce irrelevant details (Fig. 4). We identify three key challenges and propose targeted solutions:

1. **Loss of Region Details:** Local features extracted from global representations often miss fine details, especially for small objects. Cropping improves detail but loses context. We address this with **DAM**, which uses a focal prompt and localized vision backbone to capture both detail and context.
2. **Limited Datasets:** Datasets like RefCOCOs [30, 45] and Visual Genome [33] offer only short phrases, insufficient for rich captioning. We propose **DLC-SDP**, a self-supervised pipeline combining segmentation labels with unlabeled web images.
3. **Limitations in Benchmarks.** Current benchmarks rely heavily on reference captions and language similarity metrics. Since these references often lack comprehensive regional detail, models are penalized for generating correct but unlisted information. To mitigate this, we propose **DLC-Bench**, a new benchmark that evaluates captions based on attribute accuracy instead of exact reference matches, better capturing the quality of localized descriptions.

Our focus on *localized image and video captioning* across multiple granularities enables DAM to achieve state-of-the-art results on 7 benchmarks.

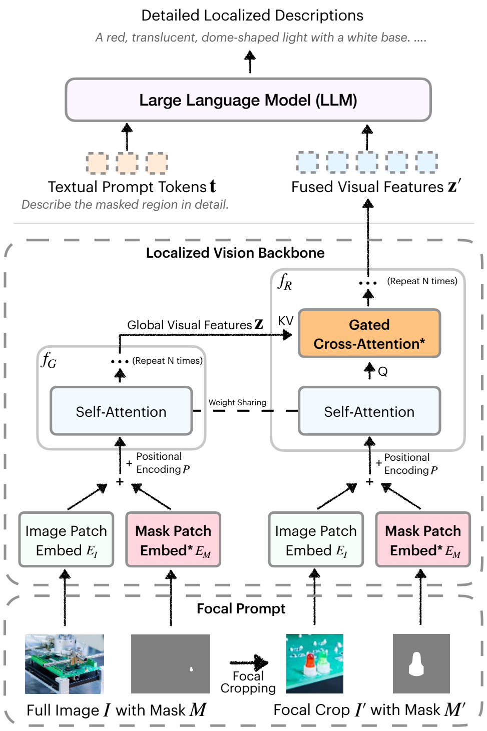

Detailed Localized Descriptions
A red, translucent, dome-shaped light with a white base. ....
Large Language Model (LLM)
Textual Prompt Tokens $\mathbf{t}$
Describe the masked region in detail.
Fused Visual Features $\mathbf{z}'$
Localized Vision Backbone
f_R
(Repeat N times)
Global Visual Features $\mathbf{z}$
KV
Gated Cross-Attention*
Q
f_G
(Repeat N times)
Self-Attention
Weight Sharing
Self-Attention
+ Positional Encoding P
+ Positional Encoding P
+
+
Image Patch Embed E_I
Mask Patch Embed* E_M
Image Patch Embed E_I
Mask Patch Embed* E_M
Focal Prompt
Focal Cropping
Full Image I with Mask M
Focal Crop I' with Mask M'

Figure 2. **Architecture of the Describe Anything Model (DAM).** DAM uses a *focal prompt* to encode user-specified regions with high token density while preserving context. A *localized vision backbone* processes both the full image and a focal crop, aligning visual embeddings via gated cross-attention. The extracted features and prompt tokens are then fed into a large language model for context-aware descriptions. * indicates zero-initialization.

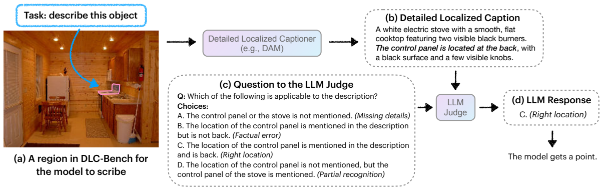
2. Related Work

Vision-Language Models (VLMs). VLMs integrate visual and textual inputs and are typically divided into BLIP-style [2, 4, 22, 24, 35–37] and LLaVA-style [5, 7, 10, 19, 20, 40, 42–44, 68, 73, 86] models, yet most lack precise localization.

Localized Image Captioning. Generating region-specific captions is challenging. While methods like SoM [76, 77] use visual markers, region-aware VLMs [15, 27, 29, 41, 51, 55, 60, 69, 70, 80, 83–85] introduce referring inputs, they often miss fine details. Our focal prompt and localized vision backbone address this, while our SSL pipeline mitigates the limited data issue.

Benchmarking Localized Captioning. Traditional benchmarks rely on reference captions and textual matching metrics [3, 6, 39, 50, 66], which may not capture factual accuracy. Recent works [80, 81] use Sentence-BERT or LLMs, yet these can unfairly penalize models due to missing details in the reference captions. DLC-Bench avoids such limitations.

Vision Models with Focus. Prior works enhance attention on salient regions using techniques like focal self-attention [75] and related methods. In contrast, our focal prompt explicitly prioritizes user-specified regions for accurate, detailed captions.


Task: describe this object
Detailed Localized Captioner (e.g., DAM)
(b) Detailed Localized Caption
(c) Question to the LLM Judge
LLM Judge
(d) LLM Response
C. *(Right location)*
(a) A region in DLC-Bench for the model to scribe

A white electric stove with a smooth, flat cooktop featuring two visible black burners. ***The control panel is located at the back***, with a black surface and a few visible knobs.

Q: Which of the following is applicable to the description?
Choices:
A. The control panel or the stove is not mentioned. *(Missing details)*
B. The location of the control panel is mentioned in the description but is not back. *(Factual error)*
C. The location of the control panel is mentioned in the description and is back. *(Right location)*
D. The location of the control panel is not mentioned, but the control panel of the stove is mentioned. *(Partial recognition)*

The model gets a point.

Figure 3. **We propose DLC-Bench, a benchmark tailored to detailed localized captioning.** In DLC-Bench, a captioning model is prompted to describe a specified image region (a). The generated description (b) is then evaluated by querying an LLM Judge (c). Points are assigned or deducted based on the LLM's response (d). The question we show in (c) is an example of positive questions.

3. DAM: Describe Anything Model

As shown in Fig. 2, DAM generates detailed localized descriptions for specified regions in images and videos, balancing local detail and global context through a *focal prompt* and a *localized vision backbone*.

3.1. Task Formulation

Given N frames $I^{(i)} \in \mathbb{R}^{H \times W \times 3}$ and binary masks $M^{(i)} \in \{0,1\}^{H \times W}$, the goal is to produce caption T:

$$T = \text{CaptioningModel}\left(\{I^{(i)}, M^{(i)}\}_{i=1}^{N}\right) \qquad (1)$$

Binary masks are used since other localization forms (points, scribbles, boxes) can be converted via segmentation models like SAM [31] for each image and SAM 2 [56] for each frame within a video.

3.2. Model Architecture

As shown in Fig. 2, DAM comprises two main components: the *focal prompt* and the *localized vision backbone*.

3.2.1 Focal Prompt

To balance detailed regional representation with context retention, we introduce the *focal prompt*, which includes the full image and a focal crop centered on the region of interest, alongside their masks.

The focal crop is obtained by expanding the bounding box of the mask by a factor of 3 in both dimensions, covering a region up to 9 times larger. This ensures the inclusion of surrounding context while preserving fine-grained details. The focal prompt consists of (1) the full image and mask, and (2) the focal crop and its corresponding mask.

3.2.2 Localized Vision Backbone

Similar to ViTs [23], we use patch embeddings for the image and an additional mask embedding layer. The full image I and mask M pass through a global encoder $f_{\text{G}}(\cdot)$,

Method	LVIS (%)		PACO (%)	
	Sem. Sim. (↑)	Sem. IoU (↑)	Sem. Sim. (↑)	Sem. IoU (↑)
LLaVA-7B [44]	49.0	19.8	42.2	14.6
Shikra-7B [15]	49.7	19.8	43.6	11.4
GPT4RoI-7B [83]	51.3	12.0	48.0	12.1
Osprey-7B [80]	65.2	38.2	73.1	52.7
Ferret-13B [78]	65.0	37.8	-	-
VP-SPHINX-7B [41]	86.0	61.2	74.2	49.9
VP-LLAVA-8B [41]	86.7	61.5	75.7	50.0
DAM-8B (Ours)	**89.0**	**77.7**	**84.2**	**73.2**

Table 1. LVIS [28] and PACO [54] open-class **keyword-level** captioning benchmarks. DAM excels particularly in the challenging PACO benchmark that requires distinguishing between objects and parts.

Method	BLEU	METEOR	ROUGE-L	CIDEr	SPICE
Shikra-7B [15]	18.2	15.3	25.2	49.8	22.0
GPT4RoI-7B [83]	19.7	17.7	29.9	61.7	24.0
Ferret-7B [78]	11.1	8.8	22.7	38.1	17.5
VP-SPHINX-13B [41]	15.2	15.6	27.2	67.4	24.0
RegionGPT-7B [27]	16.1	16.7	27.4	54.6	20.5
DAM-8B (Ours)	**22.6**	**17.8**	**31.2**	**74.7**	**25.5**

Table 2. Zero-shot evaluation on **phrase-level** dataset Flickr30k Entities [52]. Our model achieves 7.34% relative improvement against previous best.

	Short Captioning Metrics					Long Cap. Metrics
Method	BLEU	METEOR	ROUGE-L	CIDEr	SPICE	CLAIR
Shikra-7B [15]	29.5	11.1	23.9	42.7	9.0	34.5
GPT4RoI-7B [83]	27.1	11.6	26.8	59.9	11.1	43.9
Ferret-7B [78]	24.6	10.7	22.3	39.7	8.2	45.2
GLaMM-7B [55]	23.2	10.1	23.8	51.1	8.7	43.8
VP-SPHINX-13B [41]	22.6	10.7	22.6	32.4	7.6	51.2
RegionGPT-7B [27]	25.4	12.2	25.3	42.0	8.1	37.2
DAM-8B (Ours)	**38.7**	**19.4**	**37.1**	**70.0**	**16.9**	**57.9**

Table 3. Zero-shot evaluation on the **detailed captioning** dataset Ref-L4 [14]. Our method achieves 39.5% and 13.1% average relative improvement on the short/long language-based captioning metrics, respectively.

while the focal crop I' and mask M' are processed by a regional encoder $f_{\mathrm{R}}(\cdot)$, incorporating global features $\mathbf{z}$:

$$\mathbf{x} = E_{\mathrm{I}}(I) + E_{\mathrm{M}}(M) + P, \quad \mathbf{z} = f_{\mathrm{G}}(\mathbf{x}), \tag{2}$$

$$\mathbf{x}' = E_{\mathrm{I}}(I') + E_{\mathrm{M}}(M') + P, \quad \mathbf{z}' = f_{\mathrm{R}}(\mathbf{x}', \mathbf{z}), \tag{3}$$

where $E_{\mathrm{I}}(\cdot)$ and $E_{\mathrm{M}}(\cdot)$ are image and mask embeddings, $\mathbf{x}$ and $\mathbf{x}'$ encode both image and mask information, and P is positional encoding. The mask embedding layer E_{M} is initialized to zero to maintain the VLM's pre-trained behavior before fine-tuning.

To incorporate global context, we introduce gated cross-attention adapters [2, 36] into each transformer block of f_{R}. After self-attention and feed-forward layers, we apply gated cross-attention:

$$\mathbf{h}^{(l)'} = \mathbf{h}^{(l)} + \tanh(\gamma^{(l)}) \cdot \mathrm{CrossAttn}(\mathbf{h}^{(l)}, \mathbf{z}), \tag{4}$$

$$\mathbf{h}^{(l)}_{\text{Adapter}} = \mathbf{h}^{(l)'} + \tanh(\beta^{(l)}) \cdot \mathrm{FFN}(\mathbf{h}^{(l)'}), \tag{5}$$

where $\mathbf{h}^{(l)}$ is the l-th transformer block output, $\gamma^{(l)}$ and $\beta^{(l)}$ are learnable scaling parameters (initialized to zero), and CrossAttn attends to global features $\mathbf{z}$. The adapter output replaces $\mathbf{h}^{(l)}$ in subsequent layers. To reduce parameters, f_{R} shares self-attention weights with f_{G}.

The fused features from both encoders are input to the LLM, along with textual tokens, for detailed, context-aware descriptions.

4. DLC-SDP: SSL-based Data Pipeline

High-quality training data is critical for DAM. We propose DLC-SDP, a two-stage SSL pipeline:

4.1. Stage 1: Leveraging Existing Annotations

We reframe the task as vision-grounded description expansion. Using high-quality masks and keywords from segmentation datasets, we prompt the VLM to expand each keyword into a detailed caption. The model is trained to predict these descriptions without the initial keywords.

Method	#Params	Pos (%)	Neg (%)	Avg (%)
General VLMs:				
GPT-4o [49]	-	43.4	79.6	61.5
Claude 3.5 Sonnet [62]	-	13.3	47.6	30.4
Gemini 1.5 Pro [63, 64]	-	24.6	63.4	44.0
Llama-3.2 Vision [24]	11B	30.7	63.8	47.3
VILA1.5-Llama-3 [40]	8B	22.5	61.0	41.8
InternVL2.5 [20, 21, 71]	8B	15.9	42.0	28.9
LLaVA v1.6 [42–44]	7B	15.4	55.0	35.2
Qwen2.5-VL [65, 68]	7B	20.3	62.2	41.2
VILA1.5 [40]	3B	16.0	50.0	33.0
Region-specific VLMs (full / cropped input):				
GPT4RoI [83]	7B	6.5/3.5	46.2/52.0	26.3/27.7
Shikra [15]	7B	2.7/8.0	41.8/51.4	22.2/29.7
Ferret [78]	7B	6.4/14.2	38.4/46.8	22.4/30.5
RegionGPT [27]	7B	13.0/10.6	41.4/46.4	27.2/28.5
ControlCap [85]	0.3B	18.3/ 3.6	75.6/53.6	47.0/28.6
SCA [29]	3B	3.4/ 0.1	44.6/18.4	24.0/ 9.3
OMG-LLaVA [84]	7B	0.9/ 5.6	16.0/32.6	8.5/19.1
VP-SPHINX [41]	13B	11.7/26.3	33.2/71.6	22.5/49.0
DAM (Ours)	3B	**52.3**	**82.2**	**67.3**

Table 4. **Accuracies on detailed localized captioning in our proposed DLC-Bench.** DAM outperforms previous API-only models, open-source models, and region-specific VLMs on detailed localized captioning. Underlined: the second-best method.

4.2. Stage 2: SSL with Unlabeled Data

Inspired by self-training in image classification [8, 9, 59, 74], we:

1. **Mask Generation:** Use open-vocabulary segmentation models [29, 40] on unlabeled web images.
2. **Description Generation:** DAM generates detailed captions for these regions.
3. **Confidence Filtering:** Apply CLIP-based filtering to retain high-quality samples.
4. **Data Expansion:** Add new (image, mask, description) triplets to the training set.

An LLM also summarizes detailed captions into shorter forms for flexible generation.

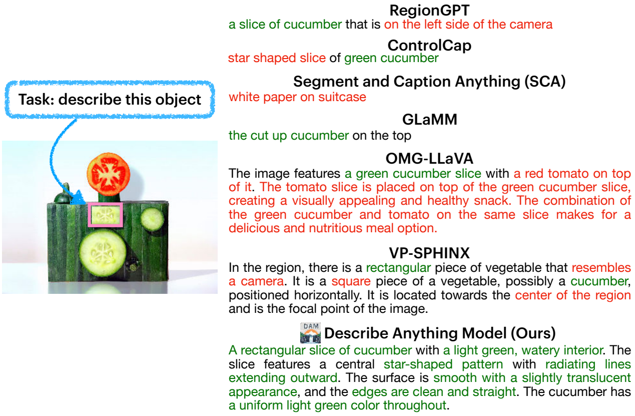
5. DLC-Bench: Benchmark for DLC

We propose DLC-Bench to evaluate localized captioning without full reference captions. The evaluation consists of:

1. Generating a detailed description for each masked re-

Method	SC	AD	TD	HD†	Avg.
Zero-shot:					
Qwen2-VL-7B [68]	3.30	2.54	2.22	2.12	2.55
InternVL2-26B [20]	4.08	**3.35**	3.08	2.28	3.20
GPT-4o-mini [49]	3.89	3.18	2.62	2.50	3.05
GPT-4o [49]	4.15	3.31	**3.11**	2.43	3.25
Osprey-7B [80]	3.30	2.66	2.10	1.58	2.41
Ferret-7B [78]	3.20	2.38	1.97	1.38	2.23
Elysium-7B [67]	2.35	0.30	0.02	**3.59**	1.57
Artemis-7B [53]	3.42	1.34	1.39	2.90	2.26
DAM-8B (Ours)	**4.45**	3.30	3.03	2.58	**3.34**
In-domain:*					
VideoRefer-7B [81]	4.44	3.27	3.10	3.04	3.46
DAM-8B (Ours)	**4.69**	**3.61**	**3.34**	**3.09**	**3.68**

Table 5. **Performance on detailed localized video description on VideoRefer-Bench-D [81].** †: We provide analysis on hallucination scores (HD) in Appendix B.2. *: trained on in-domain VideoRefer-700k with regard to VideoRefer-Bench, both sourcing videos from Panda-70M [17].



RegionGPT
a slice of cucumber that is on the left side of the camera

ControlCap
star shaped slice of green cucumber

Segment and Caption Anything (SCA)
white paper on suitcase

GLaMM
the cut up cucumber on the top

OMG-LLaVA
The image features a green cucumber slice with a red tomato on top of it. The tomato slice is placed on top of the green cucumber slice, creating a visually appealing and healthy snack. The combination of the green cucumber and tomato on the same slice makes for a delicious and nutritious meal option.

VP-SPHINX
In the region, there is a rectangular piece of vegetable that resembles a camera. It is a square piece of a vegetable, possibly a cucumber, positioned horizontally. It is located towards the center of the region and is the focal point of the image.

Describe Anything Model (Ours)
A rectangular slice of cucumber with a light green, watery interior. The slice features a central star-shaped pattern with radiating lines extending outward. The surface is smooth with a slightly translucent appearance, and the edges are clean and straight. The cucumber has a uniform light green color throughout.

Figure 4. DAM generates detailed and localized descriptions, whereas prior works generate descriptions that are less precise. Green: correct description. Red: factual error or mislocalization.

gion.

2. An LLM judges the description using a set of curated positive and negative questions.

Positive questions check for required attributes, while negative questions penalize irrelevant details. With 892 manually verified questions, DLC-Bench offers a flexible and accurate evaluation. Details are in Appendix C.

6. Results

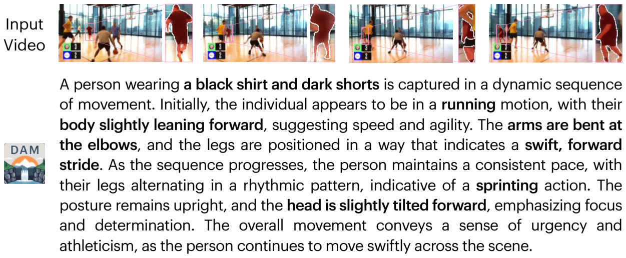
DAM excels at localized captioning across multiple granularities, achieving SOTA on 7 in-domain and zero-shot benchmarks (Tabs. 1 to 6).

6.1. Quantitative Results

Keyword-Level Captioning: On LVIS [28] and PACO [54], DAM greatly outperforms prior work (Tab. 1).
Phrase-Level Captioning: Zero-shot tests on Flickr30k Entities [52] show a 7.34% relative improvement (Tab. 2).
Detailed Captioning: On Ref-L4 [14] (Tab. 3) and DLC-Bench Tab. 4, DAM significantly outperforms existing VLMs, including GPT-4o [49].
Video Captioning: On HC-STVG [61] and VideoRefer-Bench [81], DAM shows a 16.83% improvement on HC-STVG and outperforms prior methods in both zero-shot and in-domain settings on VideoRefer-Bench.

Method	BLEU@4	METEOR	ROUGE-L	CIDEr	SPICE
Osprey-7B [80]	0.7	12.0	18.0	1.2	15.6
Ferret-13B [78]	0.5	10.2	17.0	1.2	11.2
Shikra-7B [15]	1.3	11.5	19.3	3.1	13.6
Merlin-7B [79]	3.3	11.3	26.0	10.5	20.1
Artemis-7B [53]	15.5	18.0	40.8	53.2	25.4
VideoRefer-7B [81]	16.5	18.7	42.4	68.6	28.3
DAM-8B (Ours)	**19.8**	**21.0**	**45.9**	**91.0**	**31.4**

Table 6. **Detailed localized *video* captioning** on HC-STVG [61].



A person wearing **a black shirt and dark shorts** is captured in a dynamic sequence of movement. Initially, the individual appears to be in a **running** motion, with their **body slightly leaning forward**, suggesting speed and agility. The **arms are bent at the elbows**, and the legs are positioned in a way that indicates a **swift, forward stride**. As the sequence progresses, the person maintains a consistent pace, with their legs alternating in a rhythmic pattern, indicative of a **sprinting** action. The posture remains upright, and the **head is slightly tilted forward**, emphasizing focus and determination. The overall movement conveys a sense of urgency and athleticism, as the person continues to move swiftly across the scene.

Figure 5. **DAM reliably describes user-specified objects in videos, even under strong camera and object motion and heavy occlusion.**

6.2. Qualitative Results

Qualitative comparisons in images (Fig. 4) and videos (Fig. 5) show DAM's superior accuracy and detail.

6.3. Ablations

Visual Prompting. Localized inputs and context are key for accurate descriptions. Full images lack focus (48.7% accuracy), while local crops enhance detail but lose context (60.1%). Adding cross-attention and focal crops separately improves results (63.2% and 65.4%). Our best method, the focal prompt, combines both to achieve 67.3% accuracy.
Data Scaling. Expanding supervised datasets enhances performance (53.3% to 63.8%). Adding SSL with 10% unannotated SA-1B images further improves accuracy to 67.3%, demonstrating our data pipeline's scalability.

We present **tables for the above ablations** as well as **more ablations** in Appendix D.

7. Conclusion

We introduced DAM for detailed localized captioning in images and videos, which balances local detail and global context via a focal prompt and localized vision backbone. Our SSL-based DLC-SDP leverages segmentation data and unlabeled web images to produce high-quality captions, and DLC-Bench offers an attribute-based evaluation. DAM achieves SOTA performance on 7 benchmarks.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv:2303.08774*, 2023. 7

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *NeurIPS*, 2022. 2, 3

[3] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *ECCV*, pages 382–398. Springer, 2016. 2, 1

[4] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, et al. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv:2308.01390*, 2023. 2

[5] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv:2309.16609*, 2023. 2

[6] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *ACL Workshops*, 2005. 2, 1

[7] Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Sağnak Taşırlar. Fuyu-8b: A multimodal architecture for ai agents, 2023. 2

[8] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv:1911.09785*, 2019. 3

[9] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *NeurIPS*, 2019. 3

[10] Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. Paligemma: A versatile 3b vlm for transfer. *arXiv:2407.07726*, 2024. 2

[11] H Caesar, J Uijlings, and V Ferrari. Coco-stuff: Thing and stuff classes in context. arxiv. *arXiv:1612.03716*, 2016. 5, 7

[12] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: High quality object detection and instance segmentation. *TPAMI*, 2019. 7

[13] David Chan, Suzanne Petryk, Joseph E Gonzalez, Trevor Darrell, and John Canny. Clair: Evaluating image captions with large language models. *arXiv:2310.12971*, 2023. 1

[14] Jierun Chen, Fangyun Wei, Jinjing Zhao, Sizhe Song, Bohuai Wu, Zhuoxuan Peng, S-H Gary Chan, and Hongyang Zhang. Revisiting referring expression comprehension evaluation in the era of large multimodal models. *arXiv:2406.16866*, 2024. 3, 4, 1

[15] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv:2306.15195*, 2023. 2, 3, 4

[16] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. *arXiv:2311.12793*, 2023. 8

[17] Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, et al. Panda-70m: Captioning 70m videos with multiple cross-modality teachers. In *CVPR*, pages 13320–13331, 2024. 4, 1

[18] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv:1504.00325*, 2015. 1

[19] Xi Chen, Josip Djolonga, Piotr Padlewski, Basil Mustafa, Soravit Changpinyo, Jialin Wu, Carlos Riquelme Ruiz, Sebastian Goodman, Xiao Wang, Yi Tay, et al. On scaling up a multilingual vision and language model. In *CVPR*, 2024. 2

[20] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv:2312.14238*, 2023. 2, 3, 4, 7

[21] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv:2412.05271*, 2024. 3, 7

[22] Wenliang Dai, Junnan Li, D Li, AMH Tiong, J Zhao, W Wang, B Li, P Fung, and S Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. arxiv 2023. *arXiv:2305.06500*, 2023. 2

[23] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv:2010.11929*, 2020. 2

[24] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv:2407.21783*, 2024. 2, 3, 4, 7

[25] Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva-02: A visual representation for neon genesis. *Image and Vision Computing*, 2024. 7

[26] Xiuye Gu, Yin Cui, Jonathan Huang, Abdullah Rashwan, Xuan Yang, Xingyi Zhou, Golnaz Ghiasi, Weicheng Kuo, Huizhong Chen, Liang-Chieh Chen, et al. Dataseg: Taming a universal multi-dataset multi-task segmentation model. *NeurIPS*, 2024. 3

[27] Qiushan Guo, Shalini De Mello, Hongxu Yin, Wonmin Byeon, Ka Chun Cheung, Yizhou Yu, Ping Luo, and Sifei Liu. Regiongpt: Towards region understanding vision language model. In *CVPR*, 2024. 2, 3

[28] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 3, 4, 1, 5, 7

[29] Xiaoke Huang, Jianfeng Wang, Yansong Tang, Zheng Zhang, Han Hu, Jiwen Lu, Lijuan Wang, and Zicheng Liu. Segment and caption anything. In *CVPR*, 2024. 1, 2, 3

[30] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *EMNLP*, 2014. 1

[31] 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. 2

[32] Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Shahab Kamali, Matteo Malloci, Jordi Pont-Tuset, Andreas Veit, Serge Belongie, Victor Gomes, Abhinav Gupta, Chen Sun, Gal Chechik, David Cai, Zheyun Feng, Dhyanesh Narayanan, and Kevin Murphy. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://storage.googleapis.com/openimages/web/index.html*, 2017. 5, 7

[33] 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*, 2017. 1

[34] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020. 5, 7

[35] Hugo Laurençon, Lucile Saulnier, Léo Tronchon, Stas Bekman, Amanpreet Singh, Anton Lozhkov, Thomas Wang, Siddharth Karamcheti, Alexander Rush, Douwe Kiela, et al. Obelics: An open web-scale filtered dataset of interleaved image-text documents. *NeurIPS*, 2024. 2

[36] 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. 3

[37] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023. 2

[38] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *ECCV*, 2022. 7

[39] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004. 2, 1

[40] Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. In *CVPR*, 2024. 2, 3, 6, 7

[41] Weifeng Lin, Xinyu Wei, Ruichuan An, Peng Gao, Bocheng Zou, Yulin Luo, Siyuan Huang, Shanghang Zhang, and Hongsheng Li. Draw-and-understand: Leveraging visual prompts to enable MLLMs to comprehend what you want. In *ICLR*, 2025. 2, 3, 6

[42] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023. 2, 3, 7

[43] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024.

[44] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *NeurIPS*, 2024. 2, 3, 7

[45] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, 2016. 1

[46] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection. In *ECCV*, 2022. 7

[47] Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. Scaling open-vocabulary object detection. *NeurIPS*, 2024. 7

[48] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *ICCV*, 2017. 5, 7

[49] OpenAI. Gpt-4o system card, 2024. 1, 3, 4, 7

[50] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002. 2, 1

[51] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv:2306.14824*, 2023. 2

[52] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, pages 2641–2649, 2015. 3, 4, 1

[53] Jihao Qiu, Yuan Zhang, Xi Tang, Lingxi Xie, Tianren Ma, Pengyu Yan, David Doermann, Qixiang Ye, and Yunjie Tian. Artemis: Towards referential understanding in complex videos. *NeurIPS*, 37:114321–114347, 2024. 4, 1

[54] Vignesh Ramanathan, Anmol Kalia, Vladan Petrovic, Yi Wen, Baixue Zheng, Baishan Guo, Rui Wang, Aaron Marquez, Rama Kovvuri, Abhishek Kadian, et al. Paco: Parts and attributes of common objects. In *CVPR*, pages 7141–7151, 2023. 3, 4, 1, 5

[55] Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S Khan. Glamm: Pixel grounding large multimodal model. In *CVPR*, 2024. 2, 3

[56] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv:2408.00714*, 2024. 2, 5, 7

[57] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *ICCV*, 2021. 9

[58] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *ICCV*, 2019. 1, 3

[59] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *NeurIPS*, 2020. 3

[60] Zeyi Sun, Ye Fang, Tong Wu, Pan Zhang, Yuhang Zang, Shu Kong, Yuanjun Xiong, Dahua Lin, and Jiaqi Wang. Alpha-clip: A clip model focusing on wherever you want. In *CVPR*, 2024. 2

[61] Zongheng Tang, Yue Liao, Si Liu, Guanbin Li, Xiaojie Jin, Hongxu Jiang, Qian Yu, and Dong Xu. Human-centric spatio-temporal video grounding with visual transformers. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(12):8238–8249, 2021. 4, 1

[62] Claude Team. Claude 3.5 sonnet, 2024. 3, 4, 7

[63] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv:2312.11805*, 2023. 3, 4, 7

[64] Gemini Team, M Reid, N Savinov, D Teplyashin, Lepikhin Dmitry, T Lillicrap, JB Alayrac, R Soricut, A Lazaridou, O Firat, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv:2403.05530*, 2024. 3, 4, 7

[65] Qwen Team. Qwen2.5-vl, 2025. 3, 7

[66] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, 2015. 2, 1

[67] Han Wang, Yongjie Ye, Yanjie Wang, Yuxiang Nie, and Can Huang. Elysium: Exploring object-level perception in videos via mllm. In *ECCV*, pages 166–185. Springer, 2024. 4

[68] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv:2409.12191*, 2024. 2, 3, 4, 7

[69] Teng Wang, Jinrui Zhang, Junjie Fei, Hao Zheng, Yunlong Tang, Zhe Li, Mingqi Gao, and Shanshan Zhao. Caption anything: Interactive image description with diverse multimodal controls. *arXiv:2305.02677*, 2023. 1, 2

[70] Weiyun Wang, Min Shi, Qingyun Li, Wenhai Wang, Zhenhang Huang, Linjie Xing, Zhe Chen, Hao Li, Xizhou Zhu, Zhiguo Cao, et al. The all-seeing project: Towards panoptic visual recognition and understanding of the open world. *arXiv:2308.01907*, 2023. 2

[71] Weiyun Wang, Zhe Chen, Wenhai Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Jinguo Zhu, Xizhou Zhu, Lewei Lu, Yu Qiao, and Jifeng Dai. Enhancing the reasoning ability of multimodal large language models via mixed preference optimization. *arXiv:2411.10442*, 2024. 3, 7

[72] Jialian Wu, Jianfeng Wang, Zhengyuan Yang, Zhe Gan, Zicheng Liu, Junsong Yuan, and Lijuan Wang. Grit: A generative region-to-text transformer for object understanding. *arXiv:2212.00280*, 2022. 1

[73] Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, et al. Vila-u: a unified foundation model integrating visual understanding and generation. *arXiv:2409.04429*, 2024. 2

[74] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *CVPR*, pages 10687–10698, 2020. 3

[75] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, and Jianfeng Gao. Focal self-attention for local-global interactions in vision transformers. *arXiv:2107.00641*, 2021. 2

[76] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv:2310.11441*, 2023. 2, 6, 7

[77] Zhengyuan Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. The dawn of lmms: Preliminary explorations with gpt-4v (ision). *arXiv:2309.17421*, 2023. 2

[78] Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity. In *ICLR*, 2024. 3, 4

[79] En Yu, Liang Zhao, Yana Wei, Jinrong Yang, Dongming Wu, Lingyu Kong, Haoran Wei, Tiancai Wang, Zheng Ge, Xiangyu Zhang, et al. Merlin: Empowering multimodal llms with foresight minds. In *ECCV*, pages 425–443. Springer, 2024. 4

[80] Yuqian Yuan, Wentong Li, Jian Liu, Dongqi Tang, Xinjie Luo, Chi Qin, Lei Zhang, and Jianke Zhu. Osprey: Pixel understanding with visual instruction tuning. In *CVPR*, 2024. 2, 3, 4, 1

[81] Yuqian Yuan, Hang Zhang, Wentong Li, Zesen Cheng, Boqiang Zhang, Long Li, Xin Li, Deli Zhao, Wenqiao Zhang, Yueting Zhuang, et al. Videorefer suite: Advancing spatial-temporal object understanding with video llm. *arXiv:2501.00599*, 2024. 2, 4, 1

[82] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, 2023. 2, 7

[83] Shilong Zhang, Peize Sun, Shoufa Chen, Min Xiao, Wenqi Shao, Wenwei Zhang, Yu Liu, Kai Chen, and Ping Luo. Gpt4roi: Instruction tuning large language model on region-of-interest. *arXiv:2307.03601*, 2023. 2, 3

[84] Tao Zhang, Xiangtai Li, Hao Fei, Haobo Yuan, Shengqiong Wu, Shunping Ji, Chen Change Loy, and Shuicheng Yan. Omg-llava: Bridging image-level, object-level, pixel-level reasoning and understanding. *arXiv:2406.19389*, 2024. 3

[85] Yuzhong Zhao, Yue Liu, Zonghao Guo, Weijia Wu, Chen Gong, Qixiang Ye, and Fang Wan. Controlcap: Controllable region-level captioning. pages 21–38, 2024. 1, 2, 3

[86] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv:2304.10592*, 2023. 2

Describe Anything: Detailed Localized Image and Video Captioning

Supplementary Material

A. Evaluation Benchmarks

Our DAM is designed to perform well at *localized image and video captioning* across *multiple granularities*, including keyword, phrase, and detailed captions. Therefore, we evaluate and achieve SOTA in 10 in-domain and zero-shot benchmarks:

1. The LVIS open-class keyword-level benchmark in Tab. 1.
2. PACO open-class keyword-level benchmark (including object and parts as regions) in Tab. 1.
3. Flickr30k Entities phrase-level benchmark Tab. 2.
4. Ref-L4 detailed captioning benchmark Tab. 3.
5. Our proposed DLC-Bench detailed localized captioning benchmark Tab. 4
6. VideoRefer detailed video captioning benchmark Tab. 5.
7. HC-STVG detailed video captioning benchmark Tab. 6

We offer an explanation for each setup.

A.1. Keyword-level Localized Benchmarks

Open-class keyword-level localized captioning benchmarks, proposed in [80], require the model to output keywords containing the object and part entities to describe the region. In contrast to closed-class keyword-level localized captioning, which constraints the model output to several choices provided, open-class keyword-level localized captioning takes free-form text outputs from the model. The evaluation results are in Tab. 1.

1. For LVIS [28], this involves predicting the class name as a keyword, given the segmentation mask of an object. A typical class name ranges from one word to four words.
2. For PACO [54], this involves predicting the class name of an object in the mask if the mask contains a full object, or the object name and the part name if the mask contains an object part. This is especially challenging because it would require the model to understand nuances between full objects and object parts.

A.2. Phrase-level Localized Captioning Benchmarks

Phrase-level localized captioning task requires the model to output a phrase containing a brief description for each the region that includes object identification and attributes typically within a few words. The metrics typically used in phrase-level benchmarks are CIDER, METEOR, BLEU, ROUGE_L, and SPICE [3, 6, 39, 50, 66]. We refer these metrics as short captioning metrics, as opposed to metrics from LLM-based evaluations that support evaluating detailed captions.

We also perform zero-shot evaluation on the grounded phrases in Flickr30k Entities [52], where our model is not trained on the entities annotated in the training split of Flickr30k Entities. Results are in Tab. 2.

A.3. Detailed Localized Captioning Benchmarks

Detailed localized captioning task requires the model to output a detailed description for each the region with the length spanning from a long sentence to multiple sentences.

1. We perform zero-shot evaluation on detailed captions in the Objects365 [58] split of Ref-L4 [14] since we do not train on Objects365 dataset. We evaluate the prediction quality by computing short captioning metrics and CLAIR [13] score against the reference captions in the dataset. We use CLAIR to evaluate raw detailed outputs, while we summarize both the prediction and ground truth with GPT-4o-mini [49] before evaluation with short captioning metrics. No ground truth or reference captions are provided to GPT-4o-mini, with the LLM setting exactly the same for all models for fairness. Results are in Tab. 3.
2. We evaluate our model with DLC-Bench, our proposed benchmark for fine-grained region-based captioning. This evaluation is also zero-shot. We present details about our benchmark in Appendix C. Results are in Tab. 4.

A.4. Detailed Localized Video Captioning Benchmarks

1. We conduct evaluation on HC-STVG [61], a spatial-temporal video grounding dataset with detailed captions used in prior and concurrent work [53, 81]. Following prior work [53], we evaluate the quality of localized captions with CIDER, METEOR, BLEU, ROUGE_L, and SPICE [3, 6, 39, 50, 66]. Results are in Tab. 6.
2. We also perform evaluation on the detailed localized video description benchmark in VideoRefer-Bench proposed by concurrent work [81]. GPT-4o is used to provide four dimensions of scores on a scale of 1 to 5. The four dimensions are Subject Correspondence (SC), Appearance Description (AD), Temporal Description (TD), and Hallucination Detection (HD). Zero-shot setting indicates that our model is not trained on Panda-70M [17], the dataset that VideoRefer-Bench sources the videos from. In-domain setting indicates mixing the detailed caption subset of VideoRefer-700k, which is also curated from Panda-70M [17], into our training data. Results are in Tab. 5.

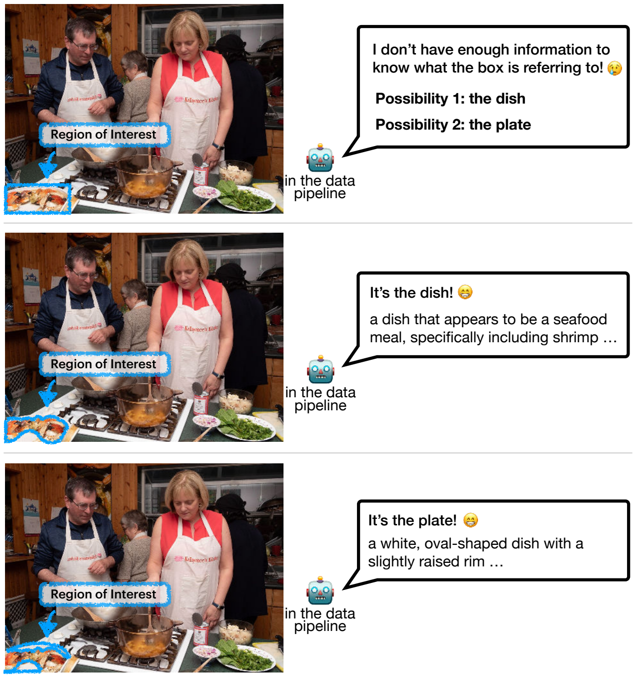

Region of Interest
in the data pipeline
I don't have enough information to know what the box is referring to! 😢
Possibility 1: the dish
Possibility 2: the plate
Region of Interest
in the data pipeline
It's the dish! 😊
a dish that appears to be a seafood meal, specifically including shrimp ...
Region of Interest
in the data pipeline
It's the plate! 😊
a white, oval-shaped dish with a slightly raised rim ...

Figure A.1. **Caveats for using boxes to indicate region of interests.** Top: Using a box to indicate the region of interest leads to ambiguity. Middle and Bottom: Switching to a mask representation leads to more specific referring and correct descriptions.

B. Discussions

B.1. The Caveats of Using Referring Boxes in Data Pipeline

Caveats exist when boxes are used to refer to regions in the data pipeline. As shown in Fig. A.1, boxes can be ambiguous in terms of what they are referring to, causing uncertainty for the VLM that we use in our data pipeline. In contrast, masks are much more specific in terms of the region that it is referring to. This motivates us to use manually annotated masks in existing segmentation datasets rather than bounding boxes in order to curate high-quality data for DLC with little referring ambiguity. We additionally take in manually annotated keywords (*e.g.*, class names, part names, entities) in the datasets for regions we are annotating in our data pipeline to further reduce the ambiguity and potential confusion for our VLM in the data pipeline.

B.2. The Pitfall of Using Reference Captions in Benchmarks

As discussed in Sec. 5, caveats exist for using a "ground truth" reference caption for benchmarking localized descriptions. Specifically, since such a reference caption is hardly comprehensive and may not contain all the details about the region of interest, the metrics from the benchmark will treat the correct details in the caption prediction about the region of interest that are not mentioned in the ground truth reference caption as hallucinations. This discourages the model from generating detailed captions.

We analyzed the performance of our method in HD (hallucination detection) sub-task in VideoRefer-Bench [81] and found that our model often predicts correct details that are not present in the reference caption. Specifically, the example in Fig. A.2 shows this phenomenon. While our model's prediction includes appearance and motion details about the change of the person's gesture and expression, such details are not mentioned in the reference caption in the dataset. Since the GPT evaluator does not see the video and uses the ground truth caption as the only source of information, it incorrectly believes that the gestures and expressions are hallucinations and gives our caption a very low score for the hallucination detection dimension. However, the evaluation is not valid, as our model is correct in the descriptions about the gestures and expressions.

This indicates that the lower score on this sub-task is *not due to the hallucination of our model*, but rather due to the missing details in the reference caption and the fact that our model, evaluated in a zero-shot setting, does not have awareness for what types of details are preferred by or included in the reference caption.

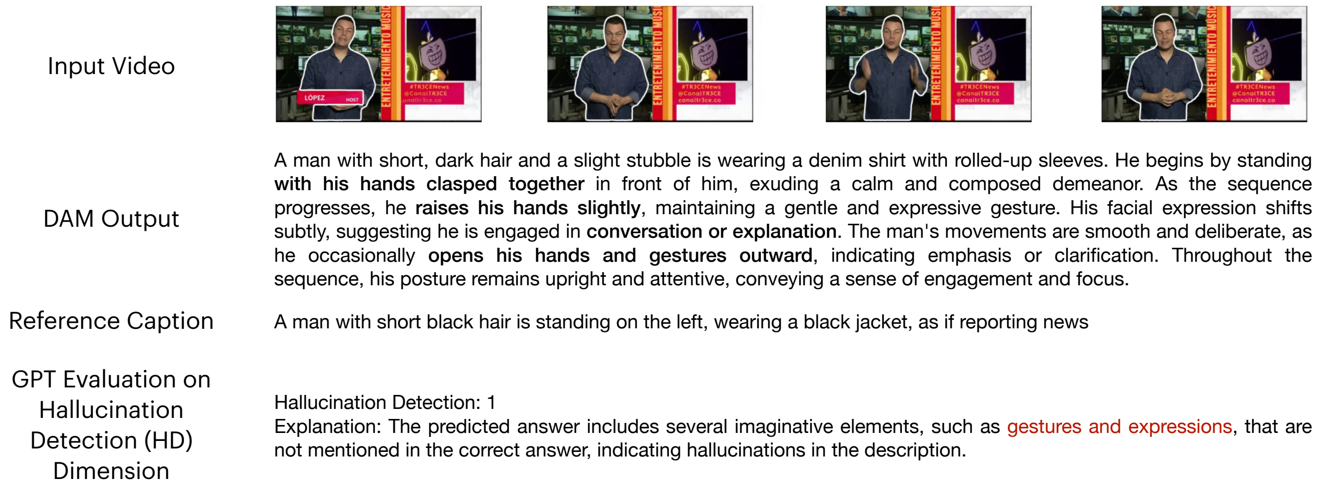
B.3. Potential Limitations

DAM is only trained for multi-granular localized captioning, especially for detailed localized captioning (DLC) and is not specifically optimized for other general vision-language tasks. However, DAM is designed for in-depth analysis for the task of multi-granular image and video localized descriptions rather than for breadth for general vision-language understanding, which justifies the design choice.

B.4. Computational Efficiency

DAM incorporates our proposed localized vision encoder, which differs from the SigLIP [82] vision encoder used in [40] by adding two key components: *patch embedding layers* for encoding the mask and *cross-attention blocks*. Importantly, these components do not alter the dimensions or sequence length of the vision features passed to the large language model, ensuring that the parameter count and computational efficiency of the large language model are unaffected. Since the vision encoder represents only a small fraction of the total parameters and computational operations, the overall increase in FLOPs and parameter count remains marginal, maintaining the model's efficiency.

To be more specific, unlike prior works that derive regional features from *image features* for each region, the regional feature used in our approach comes directly from *a global and a focal view of the input image*, with cross-attention enhancing the focal representation. This design



DAM Output

A man with short, dark hair and a slight stubble is wearing a denim shirt with rolled-up sleeves. He begins by standing **with his hands clasped together** in front of him, exuding a calm and composed demeanor. As the sequence progresses, he **raises his hands slightly**, maintaining a gentle and expressive gesture. His facial expression shifts subtly, suggesting he is engaged in **conversation or explanation**. The man's movements are smooth and deliberate, as he occasionally **opens his hands and gestures outward**, indicating emphasis or clarification. Throughout the sequence, his posture remains upright and attentive, conveying a sense of engagement and focus.

Reference Caption

A man with short black hair is standing on the left, wearing a black jacket, as if reporting news

GPT Evaluation on Hallucination Detection (HD) Dimension

Hallucination Detection: 1
Explanation: The predicted answer includes several imaginative elements, such as gestures and expressions, that are not mentioned in the correct answer, indicating hallucinations in the description.

Figure A.2. **The pitfall of using reference captions for caption evaluation.** Evaluation benchmarks based on reference captions may incorrectly treat correct details in the predicted caption as hallucination. Since the GPT evaluator relies solely on the ground truth caption without viewing the video, it mistakenly flags gestures and expressions as hallucinations, resulting in a low score. However, the evaluation is invalid since the predicted details are correct.

is justified as the vision encoder is much smaller than the LLM (400M vs. 3B/8B parameters), with minimal latency impact (*0.06s compared to 1.49s for 3B LLM* as measured in our pipeline). This overhead is outweighed by the benefits of preserving fine details that global image features miss as indicated in Tab. A.3), especially for small regions. Finally, DAM 3B outperforms much larger models in challenging (Tab. 4), showing our efficiency.

B.5. Training Data

In addition to the details in data annotation presented in Appendix F.1, we discuss the training data of our work in this section and present a comparison with recent works. Compared with recent work Ferret [78] which used 1.1M *unreleased* samples and RegionGPT [27] which used 1.5M *unreleased* samples, we train our model on a comparable amount of data (1.5M samples). However, we obtain much better performance (Tab. 4), which shows the effectiveness of DAM. Furthermore, we will release both our datasets and our data pipeline for reproducibility and ease of direct comparisons in future research.

B.6. Performances of Baseline Models on DLC-Bench

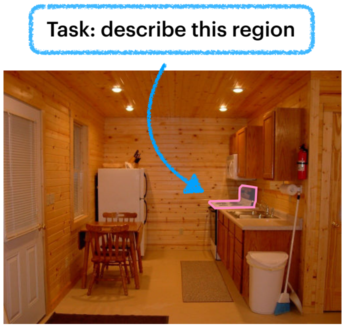
Interestingly, region-specific VLMs often perform on par or worse than generic VLMs. This is likely because many are trained on datasets with short regional captions, leading them to produce brief, phrase-level descriptions. Even when prompted for longer descriptions [27, 84], these models tend to include irrelevant details about the background, speculations, or hallucinations, due to insufficient regional information. Providing crops instead of full images leads to mixed results for different region-specific VLMs since these models are not designed to describe regions in crops.

C. Details for DLC-Bench

Image and Instance Selection. We leveraged a subset of the Objects365 v2 [58] validation set, which was manually annotated with segmentation masks in [26], for image and instance selection. We collected a set of 892 challenging questions from this subset, each containing one object of interest. Each question is manually inspected, and questions with ambiguous or unclear answers are filtered out. To maintain the integrity of our benchmark, we conducted deduplication to ensure that no images used in the benchmark were present in our training dataset for detailed localized captioning.

Positive Question Generation. For each masked region, we prompted an off-the-shelf Visual Language Model (VLM) to generate a list of parts. Subsequently, for the whole object and each part, we asked the VLM to generate a list of properties covering aspects such as color, shape, texture, materials, and size. Each property is stored in the form `([object name], [part name], [property name], [property value])`. For example, if the masked region is a corgi, the VLM could describe the brown fur of the corgi as `(corgi, fur, color, brown)`.

We used this list of properties as a starting point for manual curation. We then manually added significant properties that the VLM missed, revised inaccurate properties, and removed hallucinated or ambiguous properties from the VLM outputs. Finally, we turned these properties into questions



(a) An example region in the benchmark for the model to describe

A white electric stove with a smooth, flat cooktop featuring two visible black burners. The control panel is located at the back, with a black surface and a few visible knobs. The stove has a slightly raised backguard.

(b) An example description to be evaluated

Q: Which of the following is applicable to the description?
Choices:
A. The control panel or the stove is not mentioned. (score: 0)
B. The location of the control panel is mentioned in the description but is not back. (score: -1)
C. The location of the control panel is mentioned in the description and is back. (score: 1)
D. The location of the control panel is not mentioned, but the control panel of the stove is mentioned. (score: 0.5)

(c) An example positive question in DLC-Bench

Q: Which of the following is applicable to the description?
Choices:
A. The induction cooktop surface of the stove is mentioned in the description. (score: -1)
B. The induction cooktop surface of the stove is not mentioned in the description. (score: 1)
C. The stove is not mentioned in the description. (score: 0)

(d) An example negative question in DLC-Bench

Figure A.3. **An example from DLC-Bench for detailed localized captioning.** **(a)** The process begins by prompting a model to describe a specified region within the image. The resulting description is then evaluated using a text-only LLM as a judge that rates each response by answering positive and negative questions. **(b)** shows an example description to be evaluated. **(c)** Positive questions are designed to test whether the model correctly identifies specific details within the described region. The model receives points for accurate details and is penalized for factual errors. The bold option (option C) indicates that the LLM judge believes that option C is applicable, allowing the model to get a point for this example positive question. **(d)** Negative questions ensure the model refrains from mentioning irrelevant or nonexistent details. Mislocalization or hallucination results in penalties to prevent false positives. The bold option (option B) indicates that the LLM judge believes that option B is applicable, allowing the model to get a point for this negative question.

that test whether a description accurately covers the property.

Negative Question Generation. We targeted mislocalization and hallucination, which are two types of negatives (*i.e*., cases in which a property or an object should not be included in the descriptions). Specifically, for mislocalization errors, we prompted the VLMs to generate a list of objects in the image that are not in the masked region. We also prompted the VLMs to generate a list of parts that are commonly associated with the object type of the masked region but are not present or visible in the masked object in the image (*e.g*., the head of a corgi if it is occluded and thus not included in the masked region).

To avoid biasing towards one specific off-the-shelf VLM, we leveraged multiple VLMs for different instances to generate initial positives and negatives. Specifically, we annotated 34 regions using GPT-4o [49], 35 using Gemini 1.5 Pro [63, 64], and 31 using Anthropic Claude 3.5 Sonnet [62] for the initial property generation. We used the same image prompting method for all VLMs as we did when prompting the VLMs in the first stage of our data pipeline.

Note that the choices for each question are mutually exclusive, which ensures one option is always valid and leaves no room for two options to be true at the same time.

Scoring Mechanism. Our evaluation methodology involves scoring the models based on their ability to include correct details and exclude incorrect or irrelevant information.

To evaluate a model like DAM for its ability to output detailed localized captions, we first prompt the model to generate descriptions for each of the masked instances. Then, instead of directly asking our model to provide answers to these questions, we prompt a text-only LLM, Llama 3.1 8B [24], to serve as a judge to rate the localized descriptions according to the positive and negative questions.

For each model-generated description, we apply the following scoring rules:

- **Positive Scoring:** For each positive question, if the description correctly includes the specified detail, the model receives a point. To prevent models from artificially inflating their scores by generating excessively long descriptions and guessing details, we penalize incorrect details and discourage models from including uncertain or erroneous content. If the detail is mentioned but incorrectly (*e.g*., wrong color), a penalty of one point is applied. No point is awarded if the description does not mention the detail. Partial points (0.5 points) are awarded for answers that are partially correct but insufficiently detailed. Note that the model gets positive points only when the object recognition is correct, as the correctness of the details depends on the correctness of the overall region recognition. We present a positive example in Fig. A.3(c).

- **Negative Scoring:** For each negative question, if the description appropriately excludes the incorrect or irrelevant detail, the model gets a point. If the description includes the detail, indicating mislocalization or hallucination, a penalty is applied. The model gets zero or negative points when the object recognition is incorrect, since otherwise a caption that is random and completely off could get high scores on the negative questions. We present a negative example in Fig. A.3(d).

The positive (negative) score for a model is the sum of points for positive (negative) questions, normalized by the maximum possible score to yield a percentage for comparison. We also average the positive and negative scores to obtain an overall score, which represents the model's overall capability in detailed localized captioning.

We present an example from DLC-Bench in Fig. A.3. The example region in Fig. A.3(a) features a stove with coil burners. An example description of the region is presented in Fig. A.3(b). For the example positive question in Fig. A.3(c), the LLM judge selects option C, as the caption correctly mentions that the control panel is at the back, allowing the model to get a point for this positive question. For the negative question in Fig. A.3(d), the LLM judge selects option B, as the caption correctly indicates that it is not an induction cooktop, allowing the model to get a point for this negative question.

Evaluation Setting. For our models, we follow our inference setting described in Appendix F.

D. Detailed Ablation Studies

D.1. Ablations

Visual Prompting. We analyze different prompting strategies and find that both localized inputs and contextual information are crucial for accurate descriptions. Using only the full image limits focus on specific regions (48.7%), while local crops improve detail but lose context (60.1%). Simply concatenating both performs poorly (42.4%). Adding cross-attention significantly improves performance (63.2%), and using focal crops further enhances results (65.4%). Our best approach, the **focal prompt**, integrates focal crops with cross-attention, achieving **67.3% accuracy** without increasing sequence length for the LLM.

Data Scaling. Expanding supervised datasets boosts performance, demonstrating the value of diverse regional data. Incorporating **semi-supervised learning (SSL)** with 10% of unannotated SA-1B images further improves accuracy to **67.3%**, showcasing our data pipeline's scalability.

More ablations, such as training prior architecture on our data, comparing image-only vs image-video joint training, and ablating prompt augmentations, are in Appendix D.

Dataset	**# Images**	**# Regions**
Stage 1:		
LVIS [28]	90,613	373,551
Mapillary Vistas v2.0 [48]	17,762	100,538
COCO Stuff [11]	28,365	32,474
OpenImages v7 [32, 34]	64,874	96,006
PACO [54]	24599	81325
Stage 2:		
SA-1B (10%)	592,822	774,309
Total	**819,035**	**1,472,884**

Table A.1. **Dataset statistics across stages with total images and regions for training detailed localized image captioning.** In stage 1, we annotated 603k regions across 202k images from existing instance and semantic segmentation datasets. In stage 2, we perform SSL on 10% of SA-1B images without using the masks provided by the datasets, resulting in 774k regions across 593k images. In total, we annotated 1.47M regions across 819k images with detailed localized descriptions. This diverse and high-quality dataset is the key to our model's performance. Note that due to filtering the number of instances and images are lower than the number of instances and images in the original dataset. **We plan to release our annotated dataset to facilitate future research.**

Dataset	**# Videos**	**# Regions**
SA-V [56]	36,922	93,969

Table A.2. **Dataset statistics across stages with total videos and regions for training detailed localized *video* captioning.** We label 94k regions across 37k videos from SA-V dataset [56] for detailed localized video captioning. Note that each region indicates an instance across multiple frames in the video.

Prompting	**XAttn**	**#IT**	**Pos (%)**	**Neg (%)**	**Avg (%)**
Full Image Only	No	196	32.1	65.4	48.7
Local Crop Only	No	196	43.5	76.6	60.1 **(+11.4)**
Full + Local Crop	No*	392	26.3	58.6	42.4 **(-6.3)**
Full + Local Crop	Yes	196	45.7	80.6	63.2 **(+14.5)**
Focal Crop Only	No	196	47.3	**83.6**	65.4 **(+16.7)**
Full + Focal Crop	Yes	196	**52.3**	82.2	**67.3** **(+18.6)**

Table A.3. **Ablation studies across different visual prompts, cross-attention settings, and the number of image tokens.** Local crop denotes cropping without surrounding context. XAttn: cross-attention. #IT: number of image tokens. * indicates the full image and the crop are concatenated on the sequence dimension. **Bold**: Our proposed **focal prompt**.

Model Architecture with the Same Training Data. A model's performance is largely due to two factors: model architecture design and training data. Since both factors differ for different models, it is hard to compare the effectiveness of different model architectures head-to-head.

To this end, we compare our model architecture against

Data	# regions	Pos (%)	Neg (%)	Avg (%)
LVIS	373k	34.0	72.6	53.3
+ additional datasets	602k	47.5	80.0	63.8 (+10.5)
+ SSL on 10% of SA-1B	1.38M	**52.3**	**82.2**	**67.3** (+14.0)

Table A.4. **Our model benefits from diverse datasets generated by DLC-SDP.** Scaling the dataset in size and diversity significantly improves model performance, and SSL further enhances performance using widely available unannotated images.

	VP-SPHINX Arch	Our Arch
Avg (%)	50.2	**63.8**

Table A.5. **Ablations on architecture design compared to our strongest baseline VP-SPHINX [41].** We trained a model with VP-SPHINX [41] architecture on our curated DLC data from various segmentation datasets. The results on DLC-Bench indicate the advantages of our model architecture that allows detailed localized features to be presented to the LLM for DLC.

Prompt Augmentation	Pos (%)	Neg (%)	Avg (%)
No	**52.3**	**82.2**	**67.3**
Yes	51.3	**82.2**	66.7

Table A.6. **Comparison of performance of DAM with and without prompt augmentation.** Prompt augmentation has minimal effect on DAM's performance on DLC-Bench. While descriptions generated by the model may occasionally be less detailed, leading to a slight decrease in the performance on positive questions, we observed that prompt augmentation enhances instruction following when prompts include specific guidelines, such as length constraints. We use the model without prompt augmentation with our benchmark, including ablations, by default.

Setting	Pos (%)	Neg (%)	Avg (%)
Image-only Training	52.3	82.2	67.3
Image+Video Joint Training	**52.4**	**85.4**	**68.9**

Table A.7. **Comparison of performance of our image-only DAM and DAM trained with both localized image description task and localized video description task.** Joint training benefits generating high-quality localized image descriptions compared to image-only training.

VP-SPHINX [41], the strongest prior baseline in most benchmarks that we tested on. By continuously training a VP-SPHINX model [41] on our data after pre-training on the originally proposed datasets. This is a fair comparison since our method is also fine-tuned from a pretrained VLM, VILA-1.5 [40], with two stages of training prior to training on our region-specific dataset.

As shown in Tab. A.5, our model architecture achieves much better performance on detailed localized captioning benchmark DLC-Bench with trained on the same dataset from our proposed data pipeline. This justifies that our proposed focal prompt and localized visual backbone are able to provide more detailed features compared to just the global image features extracted by a vision encoder on the full image with a regional referring feature, as employed in [41].

Prompt Augmentation. We compared variants of our model with and without prompt augmentation. As shown in Tab. A.6, incorporating prompt augmentation slightly degrades our model's performance on the positive questions in our benchmark. We hypothesize that despite introducing variations in the prompts and enhancing the model's instruction-following capabilities, prompt augmentation creates a mismatch between the prompts used during training and those used during evaluation (as we always use the same prompt for evaluation, which is detailed in Appendix F.3). Since the prompt used during evaluation might not be the same as the prompt used in training, the model may also occasionally reference other tasks from our mixing dataset ShareGPT-4V for the length of outputs. This may cause the model to produce outputs that are not as detailed as when it is trained exclusively with the original prompt. Importantly, the model's performance on the negative questions remains unchanged, indicating that prompt augmentation does not lead to hallucinations or mislocalization.

Despite the slight degradation of the performance in the benchmark (0.6% in the overall accuracy), we observed that prompt augmentation improves instruction-following capabilities when prompts include additional instructions, particularly those specifying requirements on the length of the outputs. Therefore, we default to using the model without prompt augmentation in our benchmark evaluations, including ablations. In contrast, we employ the model with prompt augmentation in the qualitative evaluations.

Image-only Training vs Image+Video Joint Training. We also compared our image-only DAM with DAM with image + video joint training in Tab. A.7. We show that our model with image-video joint training slightly outperforms our model with image-only training on detailed localized image captioning. Note that for this ablation study, we keep the model size the same and use the 3B model for both image-only training and image-video joint training. We use image-only training as the default option for results in our benchmark for simplicity.

E. Additional Quantitative Results

Set-of-Marks Prompting. We present a comparison with baseline VLMs that use Set-of-Marks (SoM) prompting [76] in Tab. A.8. SoM leads to degraded results compared to the prompt engineering method used in stage one

Method	#Params	Pos (%)	Neg (%)	Avg (%)
API-only General VLMs:				
GPT-4o (SOM) [49]	-	5.0	29.2	17.1
Claude 3.5 Sonnet (SOM) [62]	-	3.3	33.2	18.2
Gemini 1.5 Pro (SOM) [63, 64]	-	8.2	53.4	30.8
Open-source General VLMs:				
Llama-3.2 Vision (SOM) [24]	11B	16.8	40.4	28.6
Llama-3 VILA1.5 (SOM) [40]	8B	0.6	0.6	0.6
InternVL2.5 (SOM) [20, 21, 71]	8B	8.6	28.6	18.6
LLaVA v1.6 (SOM) [42–44]	7B	2.2	3.8	3.0
Qwen2.5-VL (SOM) [65, 68]	7B	8.5	27.2	17.8
VILA1.5 (SOM) [40]	3B	-0.4	15.4	7.5
DAM (Ours)	3B	**52.3**	**82.2**	**67.3**

Table A.8. **Additional results with existing general VLMs using Set-of-Mark (SoM) prompting [76].** The results are accuracies on detailed localized captioning in DLC-Bench. Compared with results in Tab. 4 which are obtained with the same prompt engineering as we use in the stage 1 of our data pipeline, SoM leads to degraded quality. In this comparison, the advantages of our method, compared with prior baselines, are much larger. Negative numbers are due to penalties from factual errors. Note that region-specific VLMs, including our proposed DAM, have predefined ways of inputting regions, and thus SoM prompting is not applicable to these models.

of our data annotation pipeline. This is mostly because the marks proposed by SoM blend in with the object or the background in complex scenes. They might also mask out some part of the object, which interferes with the model's understanding capabilities. Therefore, for fair comparisons, we use the same prompt engineering method as we use in stage one of our data annotation pipeline in our main result in Tab. 4. Importantly, region-specific VLMs, including DAM, have predefined ways of encoding regional inputs, making SoM inapplicable to these models.

F. Implementation Details

F.1. Data Annotation Pipeline

Stage 1. We annotate four existing instance and semantic segmentation datasets for detailed localized descriptions. We use GPT-4V [1] for LVIS [28] annotation and the more advanced GPT-4o [49] for the annotation on Mapillary Vistas v2.0 [48], COCO Stuff [11], and OpenImages v7 [32, 34]. We annotated 603k regions across 202k images with detailed localized descriptions in total in stage 1, as detailed in Tab. A.1. To prompt GPT-4o to output detailed localized descriptions, we input a cropped image and a masked image. While the cropped image allows coarse localization and provides high token density per pixel for clear descriptions, the masked image helps localize the object of interest when there are multiple instances with the same category. The category name is also provided in the text prompt, relieving the model from having to identify the object without the context from the image. We present the prompt for data annotation in Tab. A.9.

Stage 2. We annotate 10% of SA-1B through self-labeling, resulting in 774k annotations across 593k images, as detailed in Tab. A.1. Due to filtering, the final number of instances and images is lower than the original 10% subset of SA-1B. We do not use the masks provided with SA-1B, as they contain a large number of masks for parts. Instead, we employ the open-vocabulary detector OWL-ViT v2 [46, 47] to detect objects in the images, and then use SAM [56] to generate masks for the detected instances. Finally, we use SigLIP [82] to evaluate the image-text similarity, taking the region as an image.

To ensure data quality, we apply extensive filtering (*i.e.*, rejection sampling) based on confidence scores from OWL-ViT v2, SAM, and SigLIP image-text similarity. We also ensure we have at most two instances per image, and for images with two instances, these two instances have to be from different classes. The object category names produced by OWL-ViT v2 are then put into a variant of our Describe Anything model, which is trained on data from stage 1 and optimized for self-labeling. This variant generates descriptions with a 50% probability of incorporating class names during training, as during self-labeling we have a class name as a part of each input. The object category proposals used by OWL-ViT v2 are generated by VILA 1.5 [40].

Detailed localized video captioning. We annotated 94k regions across 37k videos from SA-V dataset [56] for detailed localized video captioning, as detailed in Tab. A.2. Note that each region, also called masklet, indicates an instance across multiple frames in the video. In contrast to the use of SA-1B, where we did not use the masks that come with the dataset, we use the high-quality masklets that come with the videos. We found that many masklets cover parts of an instance, which is not necessarily helpful in describing the whole object as a common use case of our model. Therefore, we performed instance segmentation on the videos with ViTDet [38] + Cascade Mask R-CNN [12] trained by EVA-02 [25] and used voting to match the segmentation masks with the masklets. In this way, we filter out most of the masklets that are parts, since they likely do not correspond to instance masks. The matched masklets carry the class name from the matched instance segmentation mask, which is used in the annotation process to obtain a detailed localized caption for each masklet.

F.2. Model Training

We start from off-the-shelf VILA 1.5 [40] models that are publicly available on HuggingFace. For image-only training, we fine-tune VILA 1.5 3B model. For joint image-video training, we use VILA 1.5 8B model. We use SigLIP [82] vision encoder, following VILA 1.5. To pre-

```
You are responsible to write a very descriptive caption to describe the {{category}} in the provided SEGMENTED image. You may leverage the surrounding context of the SEGMENTED image provided in the CROPPED image.
You must not mention any background in the caption and only describe the {{category}} in the SEGMENTED image! The caption must ONLY contain sufficient details to reconstruct the same {{category}} in the SEGMENTED image but nothing else!
Here are some additional rules you need to follow when describing the {{category}} in the SEGMENTED image:
1. If there are multiple {{category}} in the CROPPED image, focus on the {{category}} in the SEGMENTED image.
2. If the {{category}} in the SEGMENTED image is occluded by other objects, only describe the visible part. DO NOT mention anything that is not directly related to the visible part of {{category}}, such as "A segment of", which part is invisible, etc. For objects with text written on it, describe the object instead of just outputting the text written on it.
Here is the SEGMENTED image that needs caption:
```

Table A.9. Our prompt for data annotation in stage 1.

vent catastrophic forgetting and to maintain instruction following capabilities, we mix in ShareGPT-4V [16] with our localized image/video captioning dataset collected with our proposed data pipeline. Following the VILA 1.5 training and inference recipe, we treat videos as 8 images concatenated in the sequence.

We closely follow VILA 1.5's recipe of the supervised fine-tuning stage and train all modules, including the vision backbone, the projector, and the LLM. We fine-tune the model for 1 epoch. For the 3B model, we use a batch size of 2048 with a learning rate of 1e-4 on 8 Nvidia A100 GPUs. For the 8B model, we use a batch size of 2048 with a learning rate of 1e-5 on 32 Nvidia A100 GPUs. Both models take less than a day to train. We use a cosine scheduler with a warmup ratio of 0.03. No weight decay is used. For training our model that takes in a class name for self-labeling, we randomly put the class name in the prompt with 50% probability. For models without prompt augmentation, which is detailed below, we simply use the prompt "`Describe the masked region in detail.`" Following VILA, we always put image tokens in front of the textual tokens. As for the setting for the focal crop, we extend the crop by 1× the width towards left and right, and 1× the height towards top and bottom, unless we hit the boundaries of the image, in which case we take the boundaries, *i.e.* $\alpha = 3$ and the total area of the crop is enlarged up to 9×. If either the height or width is less than 48 pixels, we take 48 pixels for that direction to encode more context for very small regions, since the small regions themselves do not have much useful information.

Prompt Augmentation. We trained a variant of our model with prompt augmentation to enhance generalization capabilities beyond detailed localized captioning. For these models, during training, we randomly select one of 15 prompts from a predefined set. These prompts may or may not include a `{prompt_suffix}`. The default prompt suffix is *in detail*. However, we introduce variability by conditioning the prompt on the number of words or sentences in the target caption.

Specifically, with a 20% probability, we condition the prompt on the number of sentences, using suffixes like *in one sentence* or *in [number of sentences] sentences* (e.g., *in 2 sentences*). If the caption contains only one sentence, we use phrases like *in a sentence* or *in one sentence*.

With another 20% probability, we condition the prompt on the number of words in the target caption. For captions with a small word count, we use exact numbers (e.g., *in 3 words*). For longer captions (up to 200 words), we may round the word count to the nearest ten and use phrases like *in about 50 words* or *in around 50 words*. If the caption exceeds 200 words, we use the suffix *in more than 200 words*.

The list of prompts that include a `{prompt_suffix}` is as follows:

1. Describe the masked region `{prompt_suffix}`.
2. Describe the masked area `{prompt_suffix}`.
3. What can you describe about the masked region `{prompt_suffix}`?
4. Can you describe the masked region `{prompt_suffix}`?
5. Provide an explanation of the masked region `{prompt_suffix}`.
6. Depict the masked area `{prompt_suffix}`.
7. Portray the masked area `{prompt_suffix}`.
8. Describe what the masked region looks like `{prompt_suffix}`.
9. Illustrate the masked region `{prompt_suffix}`.
10. How would you explain the masked area `{prompt_suffix}`?
11. What details can you provide about the masked region `{prompt_suffix}`?
12. What does the masked region entail `{prompt_suffix}`?
13. How would you illustrate the masked region `{prompt_suffix}`?

14. How would you depict the masked area {prompt_suffix}?
15. How would you portray the masked area {prompt_suffix}?

Additionally, we have prompts that inherently request detailed descriptions without requiring a suffix:

1. Give a detailed description of the masked region.
2. Provide a thorough description of the masked region.
3. Can you explain the details of the masked area?
4. Give a detailed account of the masked region.
5. Describe the masked area comprehensively.
6. Provide an in-depth description of the masked region.
7. Explain the specifics of the masked area.
8. Can you provide a thorough explanation of the masked region?
9. What are the details of the masked area?
10. Provide a comprehensive description of the masked area.
11. What specific details can you provide about the masked region?
12. Can you give an in-depth account of the masked section?
13. What are the main characteristics of the masked region?
14. Give a thorough description of the masked area's details.
15. Provide detailed information about the masked area.

For prompts without a suffix, we do not condition the generation on the number of words or sentences.

During training, we select prompts based on the prompt_suffix:

- If the prompt_suffix is *in detail* (the default option), we may choose from either set of prompts.
- If the prompt_suffix specifies word or sentence counts, we select only from prompts that include {prompt_suffix}.

This approach introduces variability in the prompts, encouraging the model to generate responses with controls from the prompts in mind, thereby enhancing its generalization and instruction-following capabilities.

F.3. Inference Setting

Unless otherwise mentioned, our prompt for obtaining detailed localized image descriptions at inference time is the following:

```
Describe the masked region in detail.
```

Our prompt for obtaining detailed localized video descriptions at inference time is the following:

```
Given the video in the form of a
sequence of frames above, describe
the object in the masked region in the
video in detail.  Focus on appearance,
motion, and actions.  If the motion
involves multiple stages or steps,
break down each stage and describe
the movements or changes sequentially.
Ensure each phase of motion is
described clearly, highlighting
transitions between actions.
```

For Co3Dv2 [57] sequences that we treat as videos, we use the following prompt:

```
Describe the masked region in the
video in detail.  The video consists
of multiple views of a stationary
object.  Focus on the appearance of the
object without mentioning any motion or
actions.
```