DMR-Seg: Endowing Multimodal Large Language Models with Segmentation Capability via Discretized Mask Representations

Jiru Deng, Wenhao Wu, Zhiheng Li*
Shenzhen International Graduate School, Tsinghua University, Shenzhen, China
*zhhli@tsinghua.edu.cn

***Abstract*—Referring Image Segmentation (RIS) aims to segment objects based on natural language expressions. Existing methods leverage Multimodal Large Language Models (MLLMs) to interpret text descriptions and localize objects, while mask generation follows two paradigms. The first approach represents segmentation masks using polygon coordinates, which inherently suffers from precision errors. The alternative approach integrates MLLMs with Segmentation Foundation Models (SFMs). Combining two models with differing output mechanisms can lead to optimization inefficiencies and hinder the effective propagation of information. In this work, we propose DMR-Seg, an image segmentation method that integrates mask generation within MLLM. We use the modified VQVAE to discretize binary segmentation masks into sequences of tokens. By training the MLLM to predict these sequences autoregressively, the model is endowed with the capability to achieve image segmentation. To reduce decoding dependency caused by autoregressive generation, we further designed a training strategy that proactively injects token noise. Experimental results on multiple datasets demonstrate that our model outperforms existing segmentation methods. Furthermore, ablation studies confirm the effectiveness of the proposed training strategy.**

***Index Terms*—Referring Image Segmentation, Multimodal Large Language Model, Autoregressive Generation, VQVAE**

I. Introduction

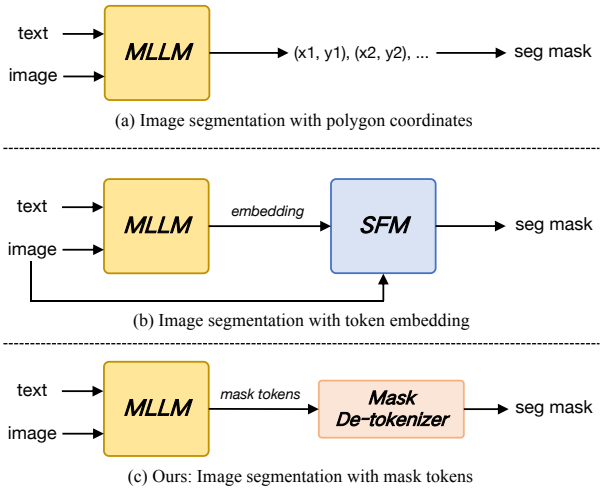
Referring Image Segmentation (RIS) represents an integral intersection of natural language processing and computer vision. Unlike traditional segmentation which is often constrained by predefined semantic categories, RIS requires models to localize and generate binary masks for specific instances based on arbitrary natural language descriptions (e.g., "the person wearing yellow clothes"). Consequently, RIS demands a sophisticated cross-modal understanding to bridge the semantic gap between textual expressions and visual pixels, offering transformative potential for human-computer interaction and intelligent image editing.

Recent advancements in multimodal foundation models [1], [2] have significantly enhanced text-image alignment, driving progress in RIS. Early methods [3]–[5] primarily utilized CLIP [1] for cross-modal feature alignment to facilitate text-guided segmentation. Currently, Multimodal Large Language Models (MLLMs) [6]–[8] represent the dominant paradigm, leveraging their inherent knowledge to interpret complex textual descriptions. Existing MLLM-based segmentation approaches generally follow two distinct paradigms. The first paradigm [9], [10] involves feeding both the image and the referring expression into MLLMs, which subsequently generate a sequence of polygon coordinates. The interior region of the predicted polygon is then extracted to derive the final segmentation mask. This approach inherently necessitates a trade-off between accuracy and efficiency, as generating finer-grained segmentation masks demands a larger number of coordinate points. The alternative paradigm [11]–[13] focuses on leveraging MLLMs to generate embeddings enriched with segmentation information, which are subsequently employed as visual prompts to guide segmentation foundation models (SFMs) [14], [15] in producing the final segmentation masks. A key characteristic of these methods is their high dependence on SFMs. Due to output modality gaps between MLLMs and SFMs, custom modules are required for information bridging. Moreover, the direct coupling of MLLMs with SFMS necessitates joint optimization, incurring high training costs, complexity, and limited scalability.

Inspired by the implementations of autoregressive image generation [16], we propose a novel approach that redefines image segmentation as a sequence generation task. Our method consists of two stages. First, we modified the encoder and decoder architectures of VQVAE [17] to enable it to serve as a mask tokenizer. The encoder transforms segmentation masks into discrete sequences, which serve as supervision for MLLMs. The decoder reconstructs masks from these sequences. This approach bridges pixel-wise masks and the discrete token sequences required for MLLMs. Second, we train a MLLM to generating mask codes (i.e., discrete sequences) in an autoregressive manner. During inference, the discrete sequences output by the MLLM are directly converted into segmentation masks via the decoder. This paradigm enables MLLMs to predict mask sequences at the token level, thereby endowing them with image segmentation capabilities. Figure 1 illustrates the differences between previous approaches and our proposed framework. Notably, our method eliminates the dependency on external segmentation models, streamlining the implementation.

The autoregressive output paradigm of MLLMs implies that an erroneous mask code at any generation step can mislead subsequent predictions, leading to error accumulation and

*Corresponding Author



(a) Image segmentation with polygon coordinates

(b) Image segmentation with token embedding

(c) Ours: Image segmentation with mask tokens

Fig. 1: Different implementation of RIS.

consequently steering segmentation results toward incorrect objects. To address this issue, we introduce a noise injection strategy that actively substitutes a portion of mask codes with random alternatives during training, thereby enabling the MLLM to predict correct codes in the presence of noise. This approach enhances the MLLM's mask prediction capability. Results on mainstream RIS benchmarks demonstrate the effectiveness of our method.

Our contribution can be summarized as follows:

1) We propose a novel RIS method that represents segmentation masks as discrete tokens, enabling MLLMs to directly output segmentation results without relying on SFMs.
2) We propose a training strategy that injects code noise into the sequence to reduce the model's dependence on previously predicted tokens during autoregressive generation.
3) Our method outperforms previous approaches on multiple RIS benchmarks, achieving superior segmentation performance. Ablation studies further confirm the effectiveness of the proposed training strategy.

II. Related Work

A. Referring Image Segmentation

The core of RIS task is cross-modal understanding. Early RIS models [18], [19] incorporate linguistic information into the segmentation pipeline. As CLIP [1] can extract cross-modal semantics, CRIS [5] integrates it with a segmentation model based on the MaskFormer [15] architecture, spurring numerous studies [3], [4], [20] to explore this paradigm for RIS. Recently, the research landscape has shifted toward MLLMs. For instance, LISA [12] synergizes SAM and LLaVA [6] by utilizing token embeddings to steer the segmentation process. While subsequent iterations [13], [21], [22] have further refined this framework, these methods generally rely on the combination of models and pipeline design, which inherently increases computational complexity.

B. Multimodal Large Language Model

Research on MLLMs fundamentally builds upon cross-modal representation learning, pioneered by CLIP [1] through contrastive alignment of visual-textual spaces. Following the rapid evolution of LLMs, contemporary efforts focus on effectively injecting visual features into the language embedding space. While early works like the LLaVA series [6], [23], [24] established the efficacy of direct projection layers, more recent state-of-the-art models, such as Qwen2-VL [7] and InternLM-XComposer [8], have shifted toward dynamic high-resolution and native aspect-ratio scaling to capture fine-grained spatial details. Such architectural advancements enable MLLMs to transcend global scene description, achieving the precise visual-linguistic parsing necessary for RIS tasks. Within this context, MLLMs must leverage their enhanced perception to bridge the gap between abstract referring expressions and dense, pixel-level semantic representations.

III. Method

A. Overall Architecture

The overall framework of DMR-Seg is illustrated in Fig. 2. We decouple the training of the mask tokenizer from that of the MLLM. In the first stage, we modify the architecture and training loss of VQVAE [17] to enable it to function as a mask tokenizer. We jointly optimize the parameters of the encoder, decoder, and codebook through masked reconstruction, thereby allowing masks to be represented as compositions of discrete codes. In the second stage, the frozen mask encoder is used to convert masks into discrete sequences, represented as mask codes. The MLLM is then trained to generate these mask codes conditioned on the image and textual description. During inference, the predicted mask codes from the MLLM are sent to the mask decoder to produce the final segmentation results.

B. Mask Tokenizer

We regard segmentation masks as binary images comprising solely black and white pixels. And we adopt a modified VQVAE as the tokenizer to transform the masks into discrete tokens. The codebook $Z \in \mathbb{R}^{K \times d_{vq}}$ of the VQVAE contains K vectors. The encoder and decoder of the modified VQVAE are shown in Fig. 3. The encoder has a downsampling rate of 16. This rate is consistent with the patch size used in the vision encoders of some MLLMs, ensuring token-level alignment. During training, the model optimizes its parameters by reconstructing segmentation masks. The encoder transforms the normalized mask $M \in \mathbb{R}^{H \times W \times 1}$ into features $f_m \in \mathbb{R}^{h \times w \times d_{vq}}$, and the quantizer converts these features into discrete codes $q \in [K]^{h \times w}$. The quantizer matches each vector in f_m with the closest vector in Z in terms of Euclidean distance and finds the corresponding code index:

$$q^{(i,j)} = \arg\min_{z_k \in Z} ||z_k - f_m^{(i,j)}|| \in [K], \tag{1}$$

where z_k is the vector within Z. Replace the vectors in f_m with the corresponding selected vectors from the codebook Z

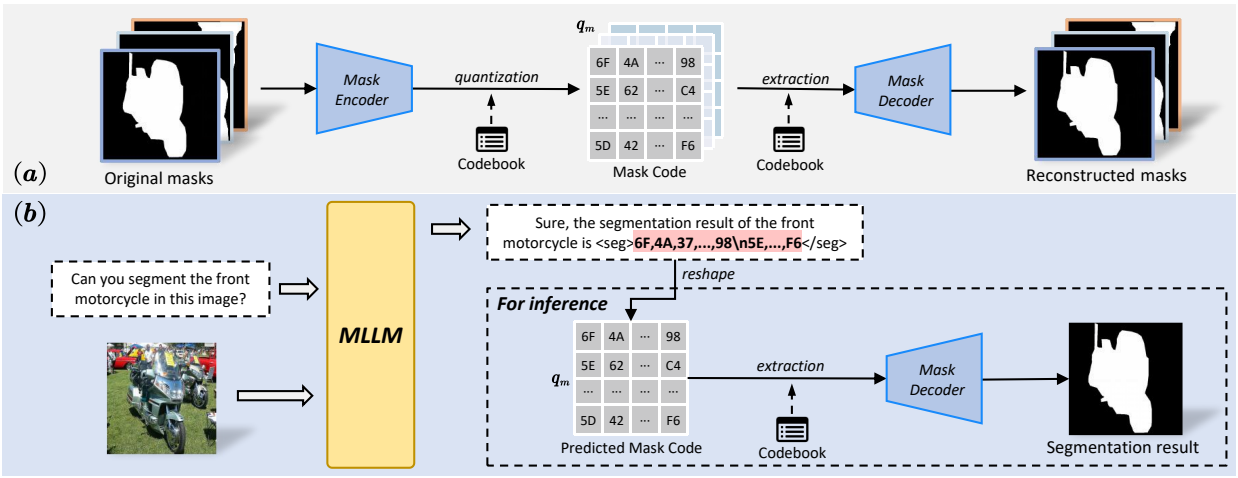


Fig. 2: Overall framework of DMR-Seg. Our method proceeds in two stages: (a) We train a mask tokenizer to reconstruct segmentation masks. Then we extract the mask codes as the training materials for the next stage. (b) We train the MLLM to generate these mask codes conditioned on images and textual descriptions. At inference time, the mask codes predicted by the MLLM are sent to the mask decoder to produce the final segmentation results.

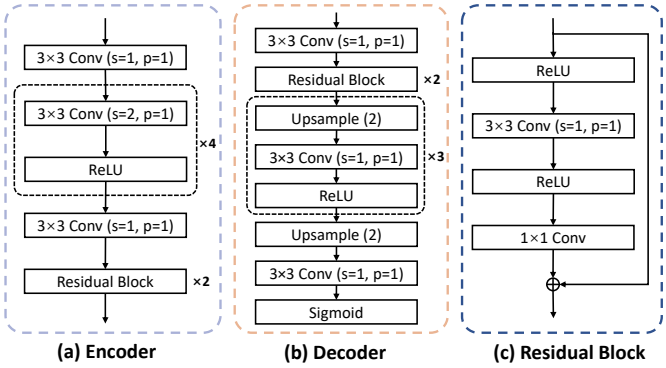


Fig. 3: The encoder and decoder architecture of the mask tokenizer.

to form the new feature $\hat{f}_m \in \mathbb{R}^{h\times w\times d_{vq}}$. Then, f_m is passed to the decoder and upsampled to produce the reconstructed mask $\hat{M} \in \mathbb{R}^{H\times W\times 1}$.

The loss function consists of three parts: reconstruction loss, codebook loss, and commitment loss. For natural images, the reconstruction loss is the mean square error (MSE) between pixel values. However, mask reconstruction involves only the binary classification of pixels. Therefore, we use binary cross-entropy loss as the reconstruction loss, while the other losses remain unchanged. The total training objectives are:

$$\mathcal{L}_{BCE}(M,\hat{M}) = \frac{-\sum_{i=1}^{H}\sum_{j=1}^{W}[x_{i,j}\log p_{i,j}+(1-x_{i,j})\log(1-p_{i,j})]}{H\cdot W} \quad (2)$$

$$\mathcal{L} = \mathcal{L}_{BCE}(M,\hat{M}) + ||\text{sg}[f_m]-z||_2^2 + \beta||f_m-\text{sg}[z]||_2^2 \quad (3)$$

where $p_{i,j}$ represents the probability of the pixel at (i,j) being a valid (white) pixel, β represents the weight of commitment loss, and $\text{sg}[\cdot]$ means stop gradient.

The code indices q form the discretized representation of the segmentation masks. Flattening q to form 1D sequence can serve as training data for the MLLM. To represent the range of indices from 0 to 255 in a uniform format, we convert them into hexadecimal. Each index is represented as two hexadecimal digits, which facilitates processing by the MLLM.

Only the codebook and mask decoder are used during inference, as shown on the bottom right of Fig. 2. The mask codes predicted by the MLLM are first rearranged into a 2D structure. Then, the corresponding vectors are retrieved from the codebook and fed into the decoder. A binarization step with a threshold of 0.5 is applied to the decoded output to obtain the final segmentation masks.

C. Multimodal Large Language Model

The MLLM leverages its visual comprehension capabilities and world knowledge to accurately identify target objects from textual descriptions and generate mask codes as the segmentation results. Before training the MLLM, we use the mask tokenizer to extract mask codes from segmentation data and flatten these codes into 1D sequences. Newline characters "$\backslash n$" are inserted between rows of codes to enhance the MLLM's perception of 2D spatial relationships, due to its inherent use of 1D positional encoding. The start and end of the mask code sequence are demarcated with special tokens `<seg>` and `</seg>`, respectively. These mask codes are then integrated into the response template to form the textual training data, as shown in the upper middle of Fig. 2 (b). Furthermore, we construct textual inputs by combining object descriptions with eight predefined templates, such as "Can you segment the {*object description*} in this image?".

The MLLM takes an input image I and a referring expression T as input and outputs mask codes in an autoregressive manner. For each code c_i, the predicted probability is conditioned on the input content and the previously generated codes $\{c_1, c_2, ..., c_{i-1}\}$. This mechanism aligns seamlessly with the

generative paradigm of MLLMs. For a sequence of mask codes of length L, the probability of the target answer is given by:

$$p(c_1, c_2, ..., c_L|I, T) = \prod_{i=1}^{L} p(c_i|I, T, c_{<i}) \tag{4}$$

Since the MLLM is only required to predict textual mask codes, the cross-entropy loss is employed for parameter optimization. Compared to other approaches that rely on additional modules, our method reduces the complexity of training. Moreover, this design allows the model to focus purely on learning semantic representations without the interference of extra architectural constraints, thereby improving training efficiency. The simplicity of the optimization objective facilitates more stable convergence across different datasets.

D. Noise Injection

The MLLM is required to generate mask codes progressively in an autoregressive manner. This implies that during inference, the currently predicted code is influenced by previously generated results. Decoding errors accumulate progressively as the code length increases. The presence of erroneous tokens within the mask sequence induces a spatial misalignment between the mask and ground-truth boundaries, which may further cause the model to misidentify the target object. To solve this issue, we intentionally introduce erroneous codes into the sequences during training. Specifically, for each training sequence, we randomly select 10% of the codes in the sequence and replace them with other hexadecimal digits. The code predicted at the current step can attend to all previous codes, which are not entirely produced by the ground truth (GT). This encourages the model to reduce its reliance on previous codes and increase its attention to visual tokens from the input image.

IV. Experiments

A. Experimental Settings

Datasets. Our training data is composed of three types. For semantic segmentation data, we employ ADE20K [25], COCO-Stuff [26], PACO-LVIS [27], PartImageNet [28] and PASCAL-Part [29]. The data for each category is extracted into binary masks. Semantic segmentation data contain a substantial number of complex boundaries, which helps the model enhance its edge representation capability. For the RIS data, we use refCLEF, refCOCO, refCOCO+, and refCOCOg [30], [31]. To mitigate catastrophic forgetting and maintain MLLM's visual comprehension capabilities, we use MMInstruct [32] as the visual question answering (VQA) data.

Evaluation metrics. We follow prior work and report cumulative Intersection-over-Union (cIoU), which is defined as the ratio of the total intersection area to the total union area between predicted and GT masks over the entire dataset.

Implementation details. For semantic segmentation data, we employ data augmentation techniques including rotation, scaling, and random cropping. For RIS data, we only resize the images. The codebook size of the mask tokenizer is set to 256, which is sufficient for mask representation. And the tokenizer is trained with a learning rate of 1e-4. The commitment loss weight β is set to 0.25. We use DeepSeek-VL-7B [33] as the MLLM, whose vision encoder has a downsampling rate identical to that of the mask tokenizer. The MLLM is trained for 10 epochs on segmentation data and VQA data. We apply LoRA [34] for efficient fine-tuning, with a learning rate of 3e-4 and a cosine decay strategy including 1% warm-up. We use AdamW as the optimizer. All training is conducted on 4 NVIDIA H20 GPUs. At inference time, mask codes are generated via greedy search.

B. Main Results

We compare DMR-Seg with several of the most competitive RIS methods. Table I presents the performance of various methods across three widely adopted RIS benchmarks. Existing methods can be classified into three paradigms. Polygon-based methods represent segmentation masks as polygons by outputting coordinate points, thereby eliminating the need for SFMs. Token-embedding-based methods use hidden state from MLLMs as prompts to guide SFMs in generating masks. Other methods integrate MLLMs and SFMs by introducing additional bridging modules. Compared to polygon-based approaches, our method achieves an average improvement of 5.6% across all datasets. Moreover, it obtains the best performance on multiple validation and test sets, even surpassing methods that rely on SFMs. On the validation sets of RefCOCO, RefCOCO+, and RefCOCOg, our approach outperforms previous methods by 0.7%, 0.2%, and 0.3%, respectively. On the RefCOCO testB set, it exceeds GroundHog-7B [36] by a margin of 1.1%. On the two test sets of RefCOCO+, our method shows improvements of 0.9% and 0.4% compared with SAM4MLLM.

While SAM4MLLM [37] leverages SAM [14] to achieve high segmentation accuracy, it incurs greater computational overhead. Although our method exhibits slightly lower performance than SAM4MLLM [37] on two test sets, the gap is minimal. Furthermore, our method offers strong extensibility. As an optional step, SAM [14] can be employed to further refine the segmentation masks based on our results, thereby continuously improving performance.

C. Ablation Study

In this section, we conduct a comprehensive ablation study to investigate the contribution of each component to the final performance. We report the results on the validation set using cIoU.

Contribution of different types of training data. Table II shows the contribution of different data types to the model's performance. When using only RIS data, the model did not show satisfactory segmentation capability. The introduction of semantic segmentation data leads to a significant improvement in performance. A notable reason is that each ground truth in the semantic segmentation dataset can be decomposed into multiple binary masks. These data contain diverse labels and complex boundary contours, which help the model learn

TABLE I: Results on referring segmentation datasets. TestA and testB are two different subsets. We use cIoU as the evaluation metric. SFM denotes Segmentation Foundation Model. Our method achieves the best performance across multiple datasets without employing an additional SFM.

Paradigm	Method	w/ SFM	refCOCO			refCOCO+			refCOCOg	
			val	testA	testB	val	testA	testB	val	test
Polygon	VistaLLM-7B [10]	×	74.5	76.0	72.7	69.1	73.7	64.0	69.0	70.9
Token Embedding (Hidden State)	LISA-7B [12]	✓	74.9	79.1	72.3	65.1	70.8	58.1	67.9	70.6
	PixelLM-7B [21]	✓	73.0	76.5	68.2	66.3	71.7	58.3	69.3	70.5
	GSVA-7B [13]	✓	77.2	78.9	73.5	65.9	69.6	59.8	72.7	73.3
	LaSagnA-7B [22]	✓	76.8	78.7	73.8	66.4	70.6	60.1	70.6	71.9
	VisionLLM v2 [11]	✓	76.6	79.3	74.3	64.5	69.8	61.5	70.7	71.2
Others	Text4Seg-7B [35]	✓	78.8	81.5	74.9	72.5	77.4	65.9	74.3	74.4
	GroundHog-7B [36]	✓	78.5	79.9	75.7	70.5	75.0	64.9	74.1	74.6
	SAM4MLLM-8B [37]	✓	79.8	**82.7**	74.7	74.6	80.0	67.2	75.5	**76.4**
Mask Token (ours)	**DMR-Seg-7B**	×	**80.5**	82.2	**76.8**	**74.8**	**80.9**	**67.6**	**75.8**	76.2

TABLE II: Ablation study on training data.

Semantic Seg	Referring Seg	VQA	refCOCO/+/g val		
	✓		74.6	70.4	71.8
	✓	✓	75.3	71.8	72.9
✓	✓		78.8	73.5	74.7
✓	✓	✓	**80.5**	**74.8**	**75.8**

TABLE III: Ablation study on the ratio of code noise.

Noise Injection	refCOCO/+/g val		
0%	74.8	70.3	71.6
5%	77.3	72.1	73.4
10%	**80.5**	**74.8**	**75.8**
15%	79.7	74.4	75.2

the representation from segmentation masks to mask codes. On this basis, incorporating VQA data further improves the model's performance. The VQA data aims to maintain the model's visual understanding and reasoning abilities, while also helping it identify the correct objects based on referring descriptions.

Noise injection ratio. We conducted ablation studies on various ratios of noise injection strategies, as illustrated in Table III. When no code noise was introduced (i.e., training solely using real mask codes), the model's performance significantly deteriorated. This indicates that the model heavily relies on previously predicted mask codes during the inference phase. By introducing a certain amount of code noise into the sequences, the model's sequence dependency was significantly reduced. It achieved optimal performance when the noise ratio was set at 10%. However, further increasing the noise beyond this point led to a decline in performance. This suggests that, given the current volume of data, excessive noise makes it difficult for the model to effectively learn the representation of the masks. Fig 4 illustrates the impact of noise injection on segmentation performance.

Components of mask tokenizer. We evaluated the downsampling and upsampling modules of the VQVAE to assess the contribution of key components to mask representation.

Fig. 4: The impact of noise injection on segmentation performance. The closest correspondence between segmentation results and GT is observed at a noise ratio of 10%.

We used four downsampling modules with a downsampling rate of 16. This design ensures that the number of mask codes matches the number of input image tokens, which facilitates learning the correspondence between them, though this is not an absolute requirement. We conducted an experiment using only three downsampling modules with a downsampling rate of 8, a setup consistent with some conventional VQVAE architectures. This configuration requires the model to produce four times as many mask codes as image tokens. Results on the refCOCO validation set showed a performance decrease of 2.2%. Sequential dependencies and spatial relationships were identified as the main factors behind this drop. Additionally, we tried to replace the combination of upsampling and convolutional layers in the decoder with a single transposed convolutional layer. Experiments on the refCOCO validation set showed a performance decline of 1.6%. This indicates that using an interpolation-based upsampling module helps restore fine-grained details. Furthermore, ablation studies on the codebook size indicate that larger sizes lead to underutilization of codebook entries. We replaced the custom reconstruction loss used during training with the original mean-square-error (MSE) loss, which led to a performance degradation of 0.8%. This suggests that the specialized loss function was better suited to the model's optimization objective.

V. Conclusion

In this work, we introduce a novel approach for referring image segmentation. We improve the VQVAE to achieve a discrete representation of segmentation masks. This enables the multimodal large language model to predict segmentation

results by generating discrete mask codes without relying on additional segmentation models, thereby reducing model complexity and training difficulty. To address the decoding dependency caused by autoregressive generation, we intentionally inject code noise into the sequences during training, enabling the model to develop robust prediction capabilities even when exposed to erroneous historical information. Experimental results demonstrate the effectiveness of our method. We hope this work will provide new insights into integrating image segmentation within the autoregressive framework.

REFERENCES

[1] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.

[2] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer, "Sigmoid loss for language image pre-training," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 11 975–11 986.

[3] C. Liu, H. Ding, and X. Jiang, "Gres: Generalized referring expression segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 23 592–23 601.

[4] Z. Yang, J. Wang, Y. Tang, K. Chen, H. Zhao, and P. H. S. Torr, "Lavt: Language-aware vision transformer for referring image segmentation," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18 134–18 144, 2022.

[5] Z. Wang, Y. Lu, Q. Li, X. Tao, Y. Guo, M. Gong, and T. Liu, "Cris: Clip-driven referring image segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 11 686–11 695.

[6] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," *Advances in neural information processing systems*, vol. 36, pp. 34 892–34 916, 2023.

[7] P. Wang, S. Bai, S. Tan, S. Wang, Z. Fan, J. Bai, K. Chen, X. Liu, J. Wang, and W. Ge, "Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution," 2024.

[8] P. Zhang, X. Dong, B. Wang, Y. Cao, C. Xu, L. Ouyang, Z. Zhao, H. Duan, S. Zhang, and S. Ding, "Internlm-xcomposer: A vision-language large model for advanced text-image comprehension and composition," 2023.

[9] W. Wang, Z. Chen, X. Chen, J. Wu, X. Zhu, G. Zeng, P. Luo, T. Lu, J. Zhou, Y. Qiao *et al.*, "Visionllm: Large language model is also an open-ended decoder for vision-centric tasks," *Advances in Neural Information Processing Systems*, vol. 36, pp. 61 501–61 513, 2023.

[10] P. Shraman, H. Guangxing, H. Rui, N. Sayan, L. Ser-Nam, B. Nicolas, W. Qifan, C. Rama, and A. Amjad, "Jack of all tasks, master of many: Designing general-purpose coarse-to-fine vision-language model," *arXiv preprint arXiv:2312.12423*, 2023.

[11] J. Wu, M. Zhong, S. Xing, Z. Lai, Z. Liu, Z. Chen, W. Wang, X. Zhu, L. Lu, T. Lu *et al.*, "Visionllm v2: An end-to-end generalist multimodal large language model for hundreds of vision-language tasks," *Advances in Neural Information Processing Systems*, vol. 37, pp. 69 925–69 975, 2024.

[12] X. Lai, Z. Tian, Y. Chen, Y. Li, Y. Yuan, S. Liu, and J. Jia, "Lisa: Reasoning segmentation via large language model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 9579–9589.

[13] Z. Xia, D. Han, Y. Han, X. Pan, S. Song, and G. Huang, "Gsva: Generalized segmentation via multimodal large language models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 3858–3869.

[14] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4015–4026.

[15] B. Cheng, A. Schwing, and A. Kirillov, "Per-pixel classification is not all you need for semantic segmentation," *Advances in neural information processing systems*, vol. 34, pp. 17 864–17 875, 2021.

[16] P. Sun, Y. Jiang, S. Chen, S. Zhang, B. Peng, P. Luo, and Z. Yuan, "Autoregressive model beats diffusion: Llama for scalable image generation," *arXiv preprint arXiv:2406.06525*, 2024.

[17] V. D. O. Aaron, O. Vinyals *et al.*, "Neural discrete representation learning," *Advances in neural information processing systems*, vol. 30, 2017.

[18] G. Luo, Y. Zhou, X. Sun, L. Cao, C. Wu, C. Deng, and R. Ji, "Multi-task collaborative network for joint referring expression comprehension and segmentation," *IEEE*, 2020.

[19] H. Ding, C. Liu, S. Wang, and X. Jiang, "Vision-language transformer and query generation for referring segmentation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 16 321–16 330.

[20] N. Kim, D. Kim, C. Lan, W. Zeng, and S. Kwak, "Restr: Convolution-free referring image segmentation using transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 18 145–18 154.

[21] Z. Ren, Z. Huang, Y. Wei, Y. Zhao, D. Fu, J. Feng, and X. Jin, "Pixellm: Pixel reasoning with large multimodal model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 26 374–26 383.

[22] C. Wei, H. Tan, Y. Zhong, Y. Yang, and L. Ma, "Lasagna: Language-based segmentation assistant for complex queries," *arXiv preprint arXiv:2404.08506*, 2024.

[23] H. Liu, C. Li, Y. Li, B. Li, Y. Zhang, S. Shen, and Y. J. Lee, "Llava-next: Improved reasoning, ocr, and world knowledge," 2024.

[24] B. Li, Y. Zhang, D. Guo, R. Zhang, F. Li, H. Zhang, K. Zhang, P. Zhang, Y. Li, Z. Liu *et al.*, "Llava-onevision: Easy visual task transfer," *arXiv preprint arXiv:2408.03326*, 2024.

[25] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, and A. Torralba, "Semantic understanding of scenes through the ade20k dataset," *International Journal of Computer Vision*, vol. 127, pp. 302–321, 2019.

[26] H. Caesar, J. Uijlings, and V. Ferrari, "Coco-stuff: Thing and stuff classes in context," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1209–1218.

[27] V. Ramanathan, A. Kalia, V. Petrovic, Y. Wen, B. Zheng, B. Guo, R. Wang, A. Marquez, R. Kovvuri, A. Kadian, A. Mousavi, Y. Song, A. Dubey, and D. Mahajan, "PACO: Parts and attributes of common objects," in *arXiv preprint arXiv:2301.01795*, 2023.

[28] J. He, S. Yang, S. Yang, A. Kortylewski, X. Yuan, J.-N. Chen, S. Liu, C. Yang, and A. Yuille, "Partimagenet: A large, high-quality dataset of parts," *arXiv preprint arXiv:2112.00933*, 2021.

[29] D. Ivan and S. Luciano, "Integration of numeric and symbolic information for semantic image interpretation," *Intelligenza Artificiale*, vol. 10, no. 1, pp. 33–47, 2016.

[30] L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg, "Modeling context in referring expressions," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*. Springer, 2016, pp. 69–85.

[31] S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg, "Referitgame: Referring to objects in photographs of natural scenes," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 787–798.

[32] Y. Liu, Y. Cao, Z. Gao, W. Wang, Z. Chen, W. Wang, H. Tian, L. Lu, X. Zhu, T. Lu *et al.*, "Mminstruct: A high-quality multi-modal instruction tuning dataset with extensive diversity," *arXiv preprint arXiv:2407.15838*, 2024.

[33] L. Haoyu, L. Wen, Z. Bo, W. Bingxuan, D. Kai, L. Bo, S. Jingxiang, R. Tongzheng, L. Zhuoshu, Y. Hao, S. Yaofeng, D. Chengqi, X. Hanwei, X. Zhenda, and R. Chong, "Deepseek-vl: Towards real-world vision-language understanding," 2024.

[34] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen *et al.*, "Lora: Low-rank adaptation of large language models." *ICLR*, vol. 1, no. 2, p. 3, 2022.

[35] M. Lan, C. Chen, Y. Zhou, J. Xu, Y. Ke, X. Wang, L. Feng, and W. Zhang, "Text4seg: Reimagining image segmentation as text generation," *arXiv preprint arXiv:2410.09855*, 2024.

[36] Y. Zhang, Z. Ma, X. Gao, S. Shakiah, Q. Gao, and J. Chai, "Groundhog: Grounding large language models to holistic segmentation," in *Conference on Computer Vision and Pattern Recognition 2024*, 2024.

[37] Y.-C. Chen, W.-H. Li, C. Sun, Y.-C. F. Wang, and C.-S. Chen, "Sam4mllm: Enhance multi-modal large language model for referring expression segmentation," in *European Conference on Computer Vision*. Springer, 2024, pp. 323–340.