

---

# MaVEn: An Effective Multi-granularity Hybrid Visual Encoding Framework for Multimodal Large Language Model

---

Chaoya Jiang<sup>1</sup>†, Hongrui Jia<sup>1</sup>†, Haiyang Xu<sup>2,\*</sup>, Wei Ye<sup>1,\*</sup>, Mengfan Dong<sup>1</sup>,  
Ming Yan<sup>2</sup>, Ji Zhang<sup>2</sup>, Fei Huang<sup>2</sup>, Shikun Zhang<sup>1</sup>

<sup>1</sup> National Engineering Research Center for Software Engineering, Peking University

<sup>2</sup> Alibaba Group

{jiangchaoya, wye, zhangsk}@pku.edu.cn,  
{shuofeng.xhy, fei.huang}@alibaba-inc.com

## Abstract

This paper presents MaVEn, an innovative Multi-granularity Visual Encoding framework designed to enhance the capabilities of Multimodal Large Language Models (MLLMs) in multi-image reasoning. Current MLLMs primarily focus on single-image visual understanding, limiting their ability to interpret and integrate information across multiple images. MaVEn addresses this limitation by combining discrete visual symbol sequences, which abstract coarse-grained semantic concepts, with traditional continuous representation sequences that model fine-grained features. This dual approach bridges the semantic gap between visual and textual data, thereby improving the model’s ability to process and interpret information from multiple images effectively. Additionally, we design a dynamic reduction mechanism by for long-sequence continuous features to enhance multi-image processing efficiency. Experimental results demonstrate that MaVEn significantly enhances MLLMs’ understanding in complex multi-image scenarios, while also improving performance in single-image contexts.

## 1 Introduction

Current multimodal large models (MLLMs) [37] concentrate on understanding single images [24, 43, 36, 22], which significantly restricts their ability to interpret and integrate information across multiple images. As shown in Figure 1, typical scenarios [17] involving multiple images include Knowledge Based VQA, Visual Relation Inference, Multi-image Reasoning and so on. These scenarios present a wide array of practical applications.

Present strategies predominantly adopt a data-centric approach, where methods such as those proposed in [1, 15, 41, 2] aim to strengthen the multi-image capabilities of Multimodal Large Language Models (MLLMs) by introducing interleaved image-text data during the pre-training and fine-tuning phases. Although some efficacy has been achieved, training solely based on interleaved data still falls short in many multi-image scenarios. This is primarily because current MLLMs remain fundamentally designed for single-image scenarios. This raises the question of whether the visual feature encoding and bridging methods of MLLMs, originally designed for single-image input scenarios, are suitable for multi-image inputs.

Current MLLMs encode visual inputs using either discrete symbol encoding [11, 13, 35, 4] or continuous sequence encoding [7, 27, 34]. For the continuous sequence feature encoding category, the

---

\*corresponding authors.

†These authors contributed equally to this work.

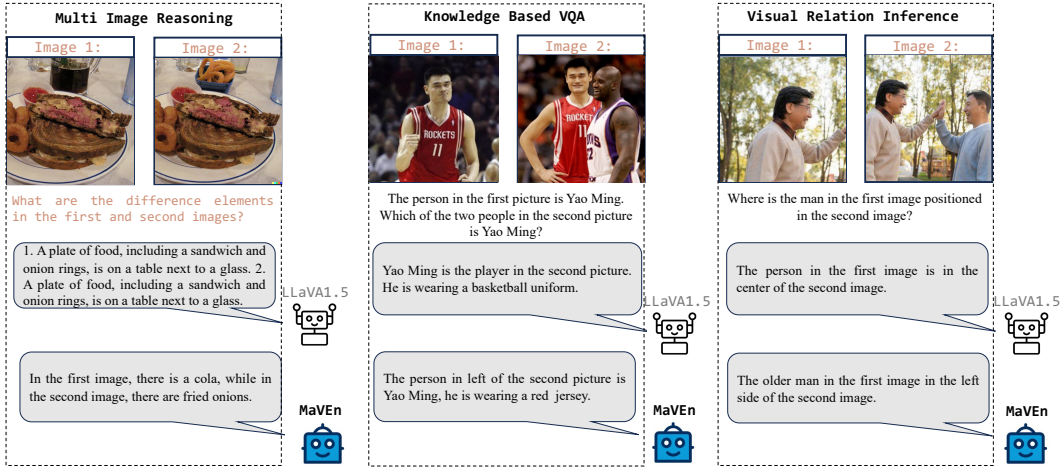


Figure 1: We compared the performance of the classic single-image task trained MLLM LLaVA1.5 [22] and our model in three multi-image scenarios including Multi Image Reasoning, Knowledge Based VQA and Visual Relation Inference. LLaVA1.5 exhibits significant limitations in multi-image scenarios.

following issues are present: (1) Excessively Long Visual Feature Sequences: Most current MLLMs utilize linear layers to bridge the visual sequence outputs from Vision Transformers (ViTs) [7]. Given the lengthy encoding sequences of images and the finite context input length of current MLLMs, the extended feature sequence inputs in multi-image contexts result in complex computational overhead and adversely affect model performance. (2) Imprecise Visual Information Encoding: Some MLLMs employ fixed-length latent queries to encode visual features through architectures like Q-Former [18]. While this approach somewhat reduces the length of visual sequences, recent studies [20] suggest that it still does not encode visual information from images with sufficient accuracy. There remains a misalignment with textual representations, leading to the model’s confusion regarding visual information.

Moreover, recent works [11, 13, 4] have started to explore encoding images as discrete symbol sequences. Compared to complex continuous visual representations, discrete visual representations offer simpler and clearer high-level semantic abstractions, closely aligning with the discrete nature of textual representations. Consequently, discrete visual encoding looks like more conducive to complex multi-image reasoning. However, given that discrete visual representations tend to be coarser in granularity, relying solely on them may overlook fine-grained details within images [40].

In this study, we introduce **MaVEN**: an effective and efficient **Multi-granularity Hybrid Visual Encoding** framework. MaVEN utilizes discrete visual symbol sequences to abstract coarse-grained semantic concepts, aiding in multi-image understanding, while traditional continuous representation sequences model fine-grained features to support detailed understanding. Accordingly, we investigate the synergy of multi-granularity visual features within the novel framework, design a dynamic reduction mechanism for long-sequence continuous features to enhance multi-image processing efficiency, and propose a multi-stage model training methodology aimed at improving multi-image comprehension. Experimental results demonstrate that the proposed method effectively enhances the understanding capabilities of MLLMs in complex multi-image scenarios, while also improving performance in single-image contexts. In summary, the contributions of this study include:

- We introduce a framework that combines discrete and continuous visual representations to enhance multi-image reasoning in MLLMs. This framework improves the model’s ability to process and interpret information from multiple images effectively.
- We design a dynamic reduction mechanism for long-sequence continuous visual features to increase the efficiency of multi-image processing in MLLMs.
- Our approach demonstrates remarkable performance across various multi-image scenarios and also shows advantages in standard single-image benchmarks.

## 2 Related Work

### 2.1 Multimodal Large Language Models

Existing Multimodal Large Language Models (MLLMs) [37] typically consist of a visual encoder, a visual interface, and a large language model (LLM). The visual encoder converts visual data into continuous sequence features. The visual interface then maps these features into the LLM’s semantic space, allowing the LLM to process visual information. Current research focuses on developing effective visual interfaces. There are two main types: Latent-Query based Models: Used in MLLMs like BLIP-2 [18] and MiniGPT-4 [43], this approach uses a fixed number of learnable latent vectors as query vectors in an attention mechanism. These vectors interact with visual sequence representations to summarize and integrate visual information, effectively reducing sequence length but potentially losing some visual details. Linear Mapping Models: Used in MLLMs like LLaVA [24], this method directly maps visual feature sequences into the LLM’s text embedding space via a linear layer. This approach retains complete visual information but results in longer output sequences.

### 2.2 Visual Semantic Encoding Representations in MLLMs

Efficient visual semantic encoding has become a key research area for MLLMs. Researchers have developed various methods to represent visual information, including: Continuous Visual Encoders: Examples include Visual Transformer (ViT) [7] and Swin-Transformer [27]. ViT segments images into patches and processes them sequentially, while Swin-Transformer uses a sliding window mechanism to capture local structures more efficiently. These methods excel in capturing image details but face challenges in aligning with textual encoding [20]. Discrete Visual Encoders: These methods encode images into discrete sequences similar to text tokens, aligning visual and textual information more closely. Examples include VQ-VAE [35], VQ-GAN [13], and SEED [11]. VQ-VAE uses self-supervised learning to create a visual vocabulary from image patches. VQ-GAN combines VQ-VAE with generative adversarial networks to capture semantic information and generate high-quality images. SEED, the latest approach, encodes images into discrete visual sequences with one-dimensional causal dependencies, aiming to extract high-level semantics for visual understanding and generation tasks.

## 3 Method

As illustrated in Figure 2, we propose an MLLM architecture that leverages multi-granularity visual features for enhanced multi-image understanding. Visual images are encoded as both discrete symbol sequences and continuous high-dimensional vector sequences. The discrete visual symbol sequences capture essential coarse-grained visual concepts from the images, while the continuous vector sequences encapsulate fine-grained details. Furthermore, to minimize redundant and irrelevant visual representations in the continuous visual sequences and thereby reduce the input context length in multi-image scenarios, we also introduce a dynamic reduction strategy for visual features, guided by textual semantics.

### 3.1 Multi-Granularity Hybrid Encoding

As shown in Figure 2 (a), assume the input to the MLLM is  $\{S, T\}$ , where  $S = \{I_1, I_2, \dots, I_K\}$  represents a collection of  $K$  images, and  $T$  denotes the corresponding textual content. For each image  $I_k$ ,  $k \in \{1, 2, \dots, K\}$ , we employ both the discrete visual encoder SEED [11] and the continuous visual encoder ViT [7] for encoding.

**Visual Continuous Encoding:** we utilize the Vision Transformer (ViT) model, which is widely adopted by most modern Multimodal Large Language Models (MLLMs). For an RGB image  $I_k$  with dimensions  $W \times H \times 3$ , the image is partitioned into patches of size  $p \times p$ , resulting in  $\frac{W}{p} \times \frac{H}{p}$  patches. These patches are then encoded by the ViT visual encoder into a continuous visual sequence:  $V_c^k = [\bar{v}_1^k, \bar{v}_2^k, \dots, \bar{v}_{n_c}^k]$ . Here,  $n_c = \frac{W}{p} \times \frac{H}{p}$ , and  $\bar{v}_i^k \in \mathbb{R}^z$  represents a continuous vector of  $z$  dimensions. Subsequently, we utilize the text-semantics-aware patch reduction module (details of which will be elaborated in Subsection 3.2) to select patch features relevant to the input textual content  $T$ , thereby reducing the sequence length of  $V_c^k$ , while preserving essential fine-grained information. The reduced feature sequence is denoted as  $V_c^k = [\bar{v}_{p_1}^k, \bar{v}_{p_2}^k, \dots, \bar{v}_{p_{m_c}}^k]$ ,  $m_c \ll n_c$ . Finally, we utilize

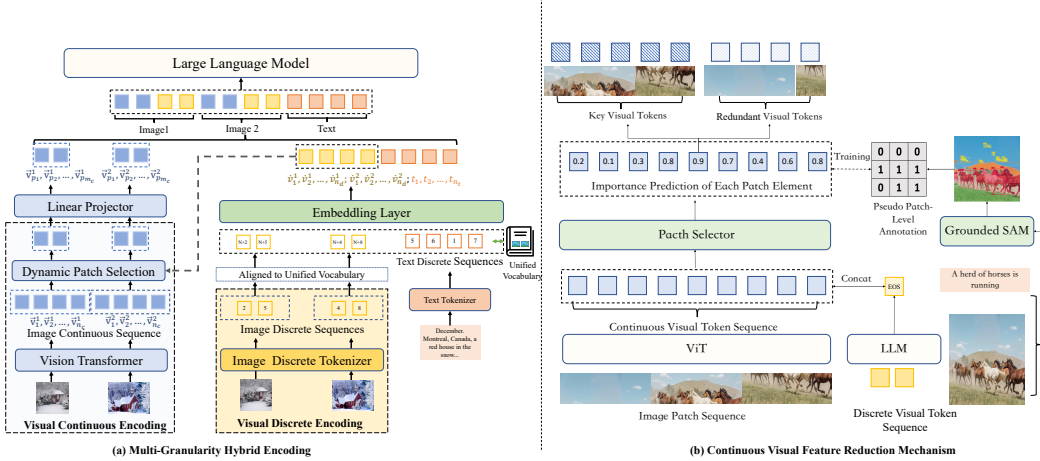


Figure 2: Subfigure (a) illustrates the structural schematic of our proposed Multi-Granularity Hybrid Encoding, while subfigure (b) demonstrates the mechanism for the reduction of continuous visual tokens under the guidance of discrete visual information.

an Multi-Layer Perceptron, akin to that used in LLaVA 1.5, as a bridging projector to project  $V_c$  into the semantic space of the LLM embedding layer.

**Visual Discrete Encoding:** image  $I_k$  is tokenized by the image discrete tokenizer  $D_v$  (for details on the design and training of the discrete tokenizer, please refer to the original work [11]) into a visual discrete symbol sequence  $V_d = [d_1^k, d_2^k, \dots, d_{n_d}^k]$ , where  $d_i^k \in [1, 2, \dots, N_v]$ . Here,  $N_v$  denotes the size of the visual discrete encoding vocabulary.

**Unified Multimodal Vocabulary:** Given that text modalities naturally possess a discrete vocabulary, merging the visual discrete vocabulary with the textual discrete vocabulary forms a unified multimodal vocabulary. The advantage of this approach lies in its ability to achieve a unified representation of both visual and textual modalities, effectively addressing the semantic gap between them. Assume that the vocabulary size of the LLM is  $N$ , and the vocabulary size of the visual discrete tokenizer is  $N_v$ . The expanded multi-modal unified vocabulary size thus becomes  $N_u = N + N_v$ . Concurrently, we align each element in  $V_d$  with the index of the unified vocabulary to obtain the final discrete encoding:  $V_d = [\hat{v}_1^k, \hat{v}_2^k, \dots, \hat{v}_{n_d}^k]$ , where  $\hat{v}_i^k = d_i^k + N$ . Finally, the weight matrix of the LLM’s embedding layer,  $W$ , is also expanded from  $N \times z$  to  $N_u \times z$ . Consequently, the weight matrix of the LLM’s embedding layer,  $W$ , is expanded from  $N \times z$  to  $N_u \times z$ . This adjustment enables the embedding layer of the LLM to concurrently encode features from both visual and textual discrete tokens. The embedded representation of  $V_d$  is denoted as  $[\hat{v}_1^k, \hat{v}_2^k, \dots, \hat{v}_{n_d}^k]$  which is output by the expanded embedding layer. Finally, we sequentially insert the continuous visual tokens before the discrete visual token embeddings outputted by the embedding layer. The final visual representation inputted into the LLM is:  $[\vec{v}_{p_1}^k, \vec{v}_{p_2}^k, \dots, \vec{v}_{p_{m_c}}^k, \hat{v}_1^k, \hat{v}_2^k, \dots, \hat{v}_{n_d}^k]$ .

### 3.2 Continuous Visual Tokens Reduction Mechanism

We aim for the discrete visual tokens to abstract high-level, coarse-grained semantics from the images, while the continuous visual tokens complement this with low-level, fine-grained details. However, we found that the continuous visual tokens output by the Vision Transformer (ViT) encompass a considerable amount of redundancy, with many tokens possessing repetitive or superfluous semantics. Consequently, as shown in Figure 2 (b), we propose a continuous visual token reduction mechanism guided by the coarse-grained semantics of discrete visual tokens, aimed at achieving semantic synergy between coarse-grained and fine-grained representations.

Firstly, after obtaining the sequence of discrete visual tokens  $V_d = [\hat{v}_1^k, \hat{v}_2^k, \dots, \hat{v}_{n_d}^k]$ , we append an  $\langle \text{EOS} \rangle$  token to it. This sequence  $V_d = [\hat{v}_1^k, \hat{v}_2^k, \dots, \hat{v}_{n_d}^k, t_{eos}]$  is then passed through the LLM to obtain the final layer’s output hidden state of the  $\langle \text{EOS} \rangle$  token denoted as  $h_{eos}$ , which represents the global information of the discrete visual tokens:  $h_{eos} = \text{LLM}([\hat{v}_1^k, \hat{v}_2^k, \dots, \hat{v}_{n_d}^k, t_{eos}])$ . we then concatenate the EOS token with each image patch token as  $\vec{v}_i^k = \text{concat}(\vec{v}_i^k, h_{eos})$ , where  $\vec{v}_i^k \in R^z, h_{eos} \in R^z, \vec{v}_i^k \in R^{2z}, i \in \{1, 2, \dots, n_c\}$ . Then the concatenated patch features  $\vec{v}_i^k$  are

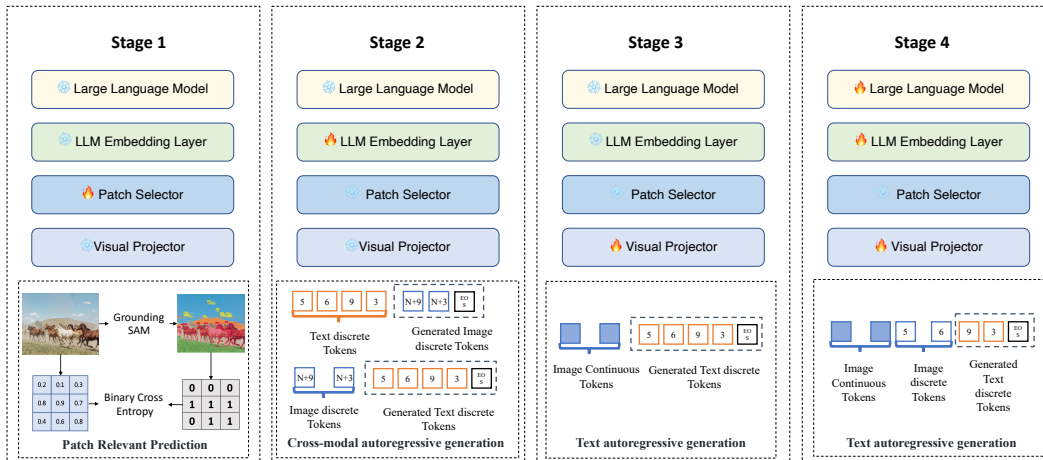


Figure 3: The diagram illustrates the training schematic for MaVen. We divide the training of MaVen into four stages, where the snowflake icon indicates that the model parameters are frozen during training, and the flame icon indicates that the model parameters are updated during training.

fed to the patch selector. The patch selector is an Multilayer Perceptron (MLP) denoted as  $\mathbf{F}$  that contains three linear layers and is used to predict the relevant score between patches and the discrete visual tokens. The output of the last linear layer has only one dimension and will be fed to a Sigmoid activation function to predict the relevant score  $a_i$  with the discrete visual tokens as  $a_i = \mathbf{F}(\vec{v}_i^k, h_{eos})$ . According to the prediction of patch selector  $\mathbf{F}$ , the top- $m$  key image patch tokens are kept and the unselected patch tokens which generally have lower relevant scores will be discarded. Finally, we reconstruct the reduced visual sequence as  $v^k = [\vec{v}_{p_1}^k, \dots, \vec{v}_{p_m}^k]$ , where  $m = n_c \times \alpha$ , and  $\alpha$  is a hyper-parameter and named Keeping Ratio which is used to control the proportion of selected patches to total patches.

**Construction of Patch-level pseudo-label annotation:** To train the patch selector, we constructed patch-level pseudo-label annotations based on Grounding SAM [32] (a recent state-of-the-art open-text vocabulary semantic segmentation model). We observed that the high-dimensional semantics encapsulated within the discrete visual tokens are largely consistent with the semantics of the image captions. Inspired by this observation, as shown in Figure 2 (b), we opted to use image captions as a proxy for the high-dimensional semantic abstraction of the image. We employed Grounding SAM to perform text-guided semantic segmentation of the images. After obtaining the semantic segmentation pixel masks, we computed the overlap between each patch and the mask labels. If there is an overlap, the corresponding patch label is set to 1; otherwise, the label is set to 0.

### 3.3 Training Paradigm of MaVen

The training process of MaVen is divided into four stages. In the first stage, we utilized image-text datasets like COCO [21] and Visual Genome (VG) [14] to annotate 1 million semantic segmentation masks with textual annotations, based on Grounding SAM [32]. These masks were subsequently converted into patch-level pseudo-labels. Utilizing this dataset, we trained the Patch Selector while keeping other model parameters frozen.

In the second stage, we exclusively trained the embedding layer of the LLM (Large Language Model) to adapt to our expanded vocabulary for the LLM. Consequently, we utilized the LLaVA 558k single-image pretraining dataset [24] and the MMC4 interleaved image-text dataset [45] for training. At this stage, we employed only the visual discrete encoding, eschewing the visual continuous encoding, with the aim of adapting the LLM embedding layer to the expanded unified vocabulary. We trained using a cross-modal autoregressive generation task; given an input that might contain images, we obtained tokenized discrete sequences through the text and image tokenizers. This enabled us to generate discrete image token sequences from text discrete token sequences and vice versa.

In the third stage, our objective is to optimize the visual projector so that the continuous visual tokens, after being processed by the visual projector, align with the semantic space distribution of the unified multimodal vocabulary embeddings. Therefore, during this phase, we train solely the visual projector.

Method	Multi Modal Dialogue	Visual Story Telling List	Visual Relation Inference	Multi Modal Cloze	Knowledge Grounded QA	Text Rich Images QA	Multi Image Reasoning
BLIP-2 [19]	11.96	20.10	3.67	18.25	39.73	30.53	39.53
mPLUG-Owl [36]	12.67	19.33	5.40	16.25	33.27	32.47	42.50
InstructBLIP [6]	33.58	24.41	11.49	21.20	47.40	44.40	48.55
LLaMA-Adapter-v2 [10]	14.22	17.57	13.51	18.00	44.80	32.00	44.03
LLaVA [25]	7.79	10.70	8.27	15.85	36.20	28.33	41.53
MiniGPT-4 [44]	13.70	17.07	7.95	16.60	30.27	26.40	43.50
LLaVA-1.5 [23]	27.17	14.32	11.62	31.65	46.4	38.87	44.58
Otter [15]	15.37	15.57	11.39	16.00	41.67	27.73	43.85
OpenFlamingo [2]	16.88	24.22	13.85	21.65	32.00	30.60	41.63
VP-G-C [17]	<b>37.50</b>	<b>25.20</b>	25.90	22.15	48.60	44.93	50.28
MaVEn	34.63	21.53	<b>30.24</b>	<b>33.35</b>	<b>51.53</b>	<b>47.33</b>	<b>54.38</b>

Table 1: Average results of **zero-shot evaluation** on each task of **DEMON Benchmark** [17].

Method	Vision Encoder	Language Model	Avg. All	Avg. Img	Avg. Video
BLIP-2 [19]	ViT-g (1.3B)	Vicuna (7B)	46.4	49.7	36.7
mPLUG-Owl [36]	ViT-L (0.3B)	LLaMA (7B)	34	37.9	23
InstructBLIP [6]	ViT-g (1.3B)	Vicuna (7B)	53.4	58.8	38.1
LLaMA-Adapter-v2 [10]	ViT-L (0.3B)	LLaMA (7B)	32.7	35.2	25.8
Otter [15]	ViT-L (0.3B)	LLaMA (7B)	33.9	35.2	30.4
LLaVA [25]	ViT-L (0.3B)	Vicuna (7B)	33.5	37.0	23.8
MiniGPT-4 [44]	ViT-g (1.3B)	Vicuna (7B)	42.8	47.4	29.9
LLaVA-1.5 [23]	ViT-L	Vicuna (7B)	58.6	66.1	37.3
MaVEn	ViT-L + SEED (1.3B)	Vicuna (7B)	<b>60.89</b>	<b>65.85</b>	<b>42.11</b>

Table 2: Average results of **zero-shot evaluation** on each task category of **SEED Benchmark** [17].

We train using the LLaVA 558K image-text caption dataset, where images are encoded solely as sequences of continuous visual tokens:  $V_c^k = [v_{p_1}^k, v_{p_2}^k, \dots, v_{p_{m_c}}^k]$  without employing visual discrete coding. The model is required to generate captions for the images based on the visual input.

In the fourth stage, we introduce instruction fine-tuning data with the aim of enhancing the MLLM’s capability to follow human instructions. During this phase, the MLLM undergoes comprehensive fine-tuning with the LLaVA 665k instruction fine-tuning datasets, unfreezing all model parameters except for those of the visual encoder and patch selector for training and optimization.

## 4 Experiments

### 4.1 Experiment Setting

**Dataset:** We initially generated 1 million pseudo-labels for patch-level text semantic relevance by utilizing the COCO[21], Visual Genome (VG)[14], and RefCOCO datasets [38], following the methodology delineated in Subsection 3.2 and leveraging Ground SAM. These pseudo-labels were subsequently employed to train the patch selector within MaVEn. During the second phase of model training, the embedding layer of MaVEn was refined using the MMC4-core dataset [45] and the LLaVA 558K single-image pre-training dataset [22]. In the third phase, the visual projector component of MaVEn was further trained using the LLaVA 558K single-image dataset. Finally, in the final phase, we fine-tuned the model with the LLaVA 665K instruction fine-tuning dataset.

**Training Settings:** MaVEn utilizes the ViT-L model [31] with a patch size of  $14 \times 14$  and is pre-trained at a resolution of  $336 \times 336$ , resulting in a continuous token length of 567 for the encoded image. For image discrete tokenization, SEED [11] is employed to tokenize the image into 32 discrete tokens. For the continuous visual tokens, during patch reduction, we set the Keeping Ratio to **0.25**, meaning that only 25% of the continuous tokens are retained. Consequently, the length of the final continuous visual token sequence decreases from 576 to 144, while the length of the discrete token sequence is 32. Ultimately, the entire visual hybrid encoding sequence has a length of 176. The large language model Vicuna [42], with 7 billion parameters, is used to handle multi-modal features. The AdamW optimizer [28] is used for optimization. During the instruction tuning stage, the entire model is trained for 1 epoch with a learning rate of  $2e-5$  and a batch size of 256. All experiments was performed using 8 NVIDIA A100 GPUs, each with 80GB of memory.



Method	#Params	General VQA		General VQA (Zero-shot)			Zero-shot Multi-modal Benchmarks		
		VQAv2	GQA	VizWizQA	TextVQA	SciQA	MME	MMBench	MM-Vet
BLIP-2 [19]	8.2B	65.0	41.0	19.6	42.5	61.0	1293.84	-	22.4
InstructBLIP [6]	8.2B	-	49.2	34.5	50.1 <sup>†</sup>	60.5	1212.82	36.0	26.2
Unified-IO <sub>XL</sub> [29]	2.9B	77.9	-	57.4 <sup>‡</sup>	-	-	-	-	-
PaLM-E-12B [8]	12B	76.2	-	-	-	-	-	-	-
Shikra [5]	7.2B	77.4	-	-	-	-	-	58.8	-
Qwen-VL-Chat [3]	9.6B	78.2	57.5	38.9	61.5 <sup>‡</sup>	<b>68.2</b>	1487.58	60.6	-
LLaVA [23]	7.2B	71.3	41.3	36.7	50.2 <sup>†</sup>	61.5	502.82	36.2	28.1
MiniGPT-4 [23]	7.2B	65.2	30.8	30.2	52.3 <sup>†</sup>	58.4	581.67	23.0	22.1
LLaVA1.5 [23]	7.2B	78.5	62.0	50.0	58.2 <sup>†</sup>	66.8	1510.70	64.3	<b>30.5</b>
MaVEn	7.2B	<b>79.1</b>	<b>62.5</b>	<b>50.5</b>	59.8 <sup>†</sup>	67.3	<b>1530.10</b>	<b>65.2</b>	30.4

Table 3: **Performance comparison on visual question answering and zero-shot multi-modal benchmarks.** For VQA, accuracy is reported. Note that specialists are fine-tuned on each individual dataset. † denotes OCR inputs are utilized. ‡ indicates the model has trained on the dataset.

## 4.2 Main Results

To validate the effectiveness of MaVEn in multi-image scenarios, we evaluated MaVEn’s performance multi-image visual understanding and reasoning. Within the multi-image visual understanding context, we assessed the model using DemonBench[17] and SEED-Bench[16]. DemonBench comprises seven scenarios involving multi-image reasoning and understanding, including tasks such as Multi-Modal Dialogue, Visual Relation Inference, Knowledge Grounded QA, and Multi-Image Reasoning. SEED-Bench, on the other hand, encompasses questions related to video comprehension. Additionally, we also tested the performance of MaVEn in single-image scenarios.

### 4.2.1 Effectiveness of MaVEn on Multi-image Visual Comprehension

**Results on DemonBench:** As shown in the Table 1, we evaluated our model on DemonBench, comparing it with several multi-image data-trained MLLM models such as Openflamingo, Otter, VPG-C, as well as single-image scenario MLLM models. Our model attained the highest scores in tasks such as Visual Relation Inference, Multi-Modal Cloze, Text-Rich Images QA, Knowledge-Grounded QA, and Multi-Image Reasoning, underscoring MaVEn ’s significant superiority in multi-image understanding and reasoning tasks. It also achieved comparable performance in Visual Storytelling and Multi-Modal Dialogue.

**Results on SEED-Bench:** Furthermore, we assessed our model on SEED-Bench [16], particularly focusing on video understanding tasks like action prediction, action Recognition and procedure understanding. As shown in Table 2 The experiments revealed that our approach significantly outperformed existing models like LLaVA 1.5, Otter and so on. For instance, MaVEn exhibited a 12-point improvement over Otter [15] in video understanding (30.4 -> 42.11). These findings underscore MaVEn ’s effectiveness in multi-image understanding scenarios.

### 4.2.2 Effectiveness of MaVEn on Single-image Visual Comprehension

We intend to explore the influence of MaVEn on the model’s abilities of single image visual comprehension and generation. To achieve this objective, we carried out assessments on common benchmarks, such as Visual Question Answering (VQA) [12, 30, 33] and recently designed MLLM-focused Multi-modal Benchmarks including MME [9], MMBench [26], MM-Vet [39].

**Results on Benchmark Tasks** As summarized in Table 3. We compared performance of MaVEn to other SOTA MLLMs such as BLIP2[19], InstructBLIP [6], Shikra [5], and Qwen-VL-Chat [3]. Our experimental results show that our approach can also successfully enhances the performance across a range of single image understanding task. Notably, MaVEn outperforms LLaVA-1.5 [25] in terms of consistency and accuracy across all VQA datasets.

**MLLM-oriented Multi-modal Benchmarks.** We also evaluate MaVEn on four recently popular single-image multi-modal benchmarks in a zero-shot manner. The results of our evaluation are listed in Table 3. We discovered that after implementing MaVEn, all three models exhibited improvements

Discrete	Continuous	SEED-Bench	DEMONBench	VQA	MMBench
✓	✗	43.2	24.19	56.07	34.5
✗	✓	58.6	30.66	78.5	64.3
✓	✓	60.89	39.51	79.1	65.21

Table 4: **Ablation evaluation on multi-modal benchmarks** We evaluated the performance of various ablation targets on both multi-image (SEED-Bench, DEMONBench ) and single-image (VQA, MMBench) benchmarks. MMBench [26].

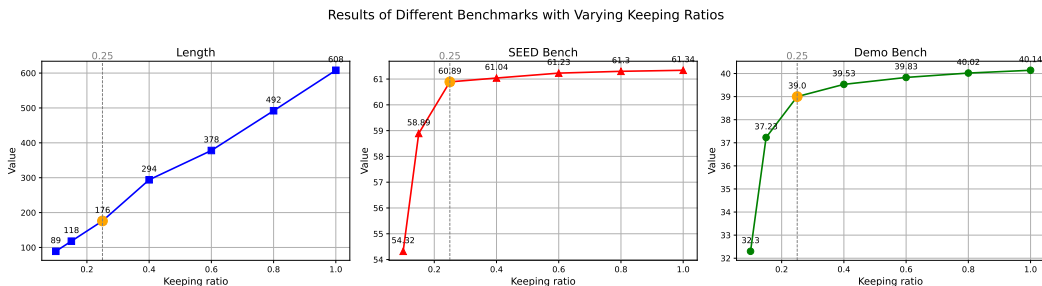


Figure 4: Evaluation Results of MaVen on different benchmarks with varying Keeping Ratios.

across multiple benchmarks. Notably, for LLaVA and MiniGPT-4, the enhancement was particularly evident on the MME [9] benchmark. For instance, after implementing MaVen, LLaVA’s MME score improved from 581.67 to 653.94. These results highlight MaVen role in advancing the state-of-the-art in both single-image and multi-image visual comprehension tasks.

### 4.3 Ablation Study

#### 4.3.1 Effectiveness of Multi-Granularity Hybrid Encoding

To ascertain the efficacy of multi-granularity hybrid encoding, we conducted training using solely visual discrete encoding and visual continuous encoding, respectively. We then evaluated and compared the outcomes on both multi-image and single-image evaluation benchmarks. The results are detailed in the Table 4 below. We observed that, compared with utilizing only visual continuous encoding or employing a hybrid of visual discrete and continuous encoding, the model solely on visual discrete encoding exhibits subpar performance in both multi-image and single-image contexts. This underperformance is likely due to the nature of discrete visual feature encoding, which, while capturing the high-dimensional information of the image, forfeits a significant amount of low-dimensional, fine-grained details, resulting in a lossy encoding process. As a result, it fares poorly in tasks like image reasoning and understanding, which demand meticulously detailed information. Moreover, the model with only continuous encoding also do not deliver optimal performance, particularly in multi-image tasks. This further indicates that models based solely on visual continuous encoding are unsuitable for multi-image scenarios.

#### 4.3.2 Efficient of Continuous Visual Token Reduction Mechanism

To verify the effectiveness of patch reduction, we compared the length of visual tokens at different Keeping Ratios and analyzed the performance across various benchmarks. We experimented with the Keeping Ratios from 0.1 to 1.0. The experimental results are shown in the Figure 4. We observed that when the Keeping Ratio was 0.1, the number of visual tokens decreased to 89, a significant reduction from the initial number of patches. However, the model’s performance across multiple benchmarks also significantly declined. Therefore, despite the reduction in the number of visual tokens, the performance loss was too substantial, making it an unsuitable final choice. When the Keeping Ratio was 0.25, the number of visual tokens remained relatively low, but the model’s performance was more stable. In this case, the model exhibited balanced performance across various benchmarks, effectively reducing the number of visual tokens while maintaining a high-performance level. Therefore, we ultimately chose a Keeping Ratio of 0.25.



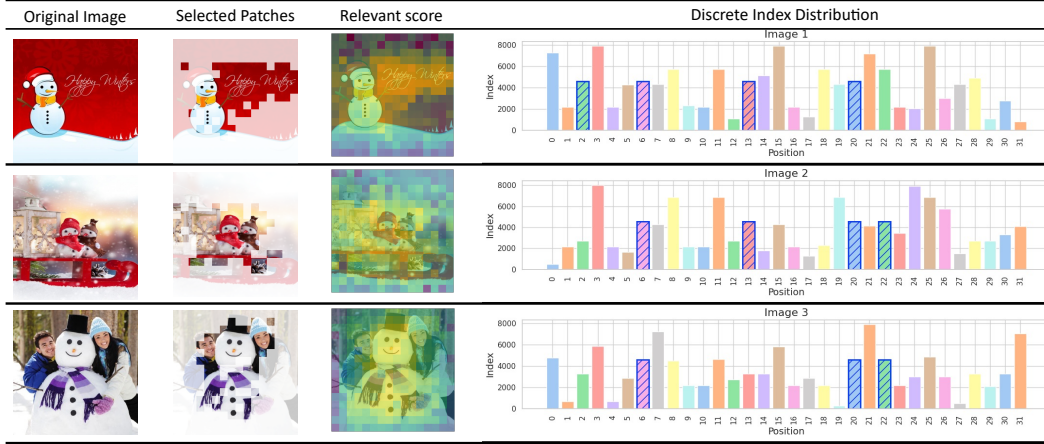


Figure 5: This figure visualizes the distribution of discrete tokens in an image containing index 4568 discrete tokens, along with the relevant score computed based on the Patch Selector and the patches chosen according to the relevant score that are most semantically related to the discrete visual tokens.

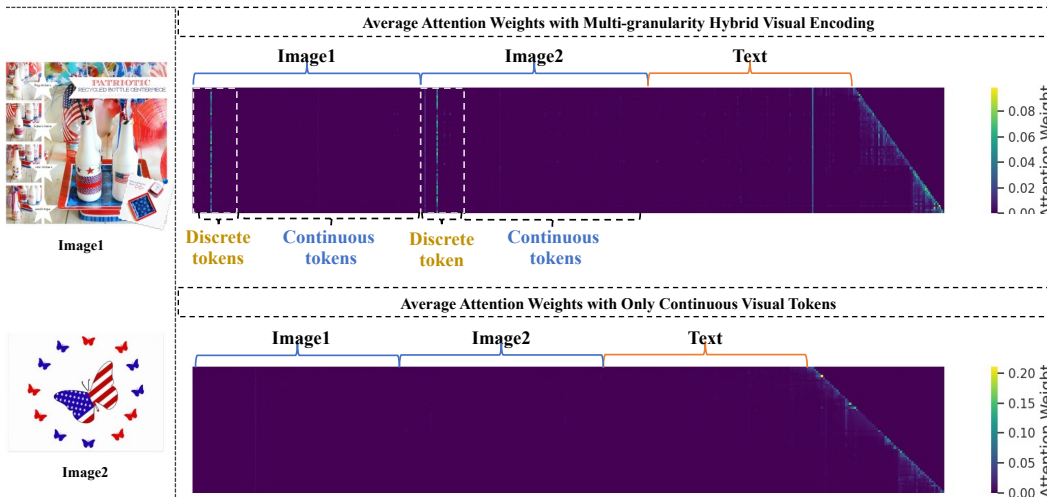


Figure 6: Visualization of the attention maps with and without the visual discrete tokens . We demonstrate the attention maps for the 31-st layers, where the range of visual tokens is indicated by orange and the range of text tokens is indicated by blue.

#### 4.4 Qualitative Analysis

**Semantic Granularity of discrete and continuous visual tokens:** To investigate the semantics of discrete tokens, we randomly selected an index from the visual discrete dictionary and searched the CC/SUB dataset for images containing this index in their encoding. As illustrated in Figure 5, we randomly chose three images with index = 4568. We discovered that all three images featured the depiction of a snowman, suggesting that index = 4568 can represent high-level semantics such as a snowman or white snow. We provided the distribution of discrete tokens for these three images and observed that the position of index = 4568 in the discrete sequences was also notably consistent.

**Semantic synergy between visual discrete and continuous representations:** Furthermore, as shown in Figure 5, we also visualize the relevant score between the patches and the semantics of discrete visual tokens predicted by the patch selector, along with the patch tokens selected based on a Keeping ratio of 0.25. We found that the patch selector tends to choose patches related to the semantics of discrete visual tokens, thereby supplementing the missing low-level fine-grained information of the discrete tokens. *The aforementioned findings further validate that multi-granularity hybrid encoding facilitates mutual assistance and synergy between discrete and continuous representations, thereby achieving efficient multi-granularity semantic encoding.*

**The impact of discrete visual tokens on multi-image reasoning:** To further validate the role of discrete visual tokens in the inference process of multi-image instructions, we visualize the attention weights of the last layer of the LLM. As illustrated in Figure 6, we have MaVEn compare the commonalities between two images. For inputs using only continuous visual tokens, we observe that the model’s attention during inference is primarily focused on text tokens, disregarding visual tokens. This may still be due to the lower semantic granularity of continuous visual tokens, making it difficult to establish direct semantic associations. However, with inputs encoded using multi-granularity visual hybrid encoding, we notice that the model establishes attention associations with discrete visual tokens when answering questions. This indicates that, *in multi-image scenarios, discrete visual tokens guide the LLM to focus on visual information during decoding.*

## 5 Conclusion

In conclusion, we introduces MaVEn, a novel Multi-granularity Hybrid Visual Encoding framework designed to enhance multi-image reasoning in MLLMs. By combining discrete visual symbols for semantic abstraction with continuous sequences for detailed features, MaVEn improves both understanding and processing efficiency. Our dynamic reduction mechanism and multi-stage training strategy further enhance performance. Experimental results confirm that MaVEn significantly boosts MLLM capabilities in both multi-image and single-image contexts.

## References

- [1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, et al. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, volume 35, 2022.
- [2] 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 preprint arXiv:2308.01390*, 2023.
- [3] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *ArXiv*, abs/2308.12966, 2023.
- [4] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.
- [5] Ke Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm’s referential dialogue magic. *ArXiv*, abs/2306.15195, 2023.
- [6] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Albert Li, Pascale Fung, and Steven C. H. Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *ArXiv*, abs/2305.06500, 2023.
- [7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.
- [8] Danny Driess, F. Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzan Wahid, Jonathan Tompson, Quan Ho Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Peter R. Florence. Palm-e: An embodied multimodal language model. In *International Conference on Machine Learning*, 2023.
- [9] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.
- [10] Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, W. Zhang, Pan Lu, Conghui He, Xiangyu Yue, Hongsheng Li, and Yu Jiao Qiao. Llama-adapter v2: Parameter-efficient visual instruction model. *ArXiv*, abs/2304.15010, 2023.

- [11] Yuying Ge, Yixiao Ge, Ziyun Zeng, Xintao Wang, and Ying Shan. Planting a seed of vision in large language model. [arXiv preprint arXiv:2307.08041](#), 2023.
- [12] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In [Conference on Computer Vision and Pattern Recognition \(CVPR\)](#), 2017.
- [13] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In [Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition](#), pages 10696–10706, 2022.
- [14] 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. [International journal of computer vision](#), 123:32–73, 2017.
- [15] Bo Li, Yuanhan Zhang, Liangyu Chen, et al. Otter: A multi-modal model with in-context instruction tuning. [arXiv preprint arXiv:2305.03726](#), 2023.
- [16] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. [arXiv preprint arXiv:2307.16125](#), 2023.
- [17] Juncheng Li, Kaihang Pan, Zhiqi Ge, Minghe Gao, Wei Ji, Wenqiao Zhang, Tat-Seng Chua, Siliang Tang, Hanwang Zhang, and Yueting Zhuang. Fine-tuning multimodal llms to follow zero-shot demonstrative instructions. In [The Twelfth International Conference on Learning Representations](#), 2023.
- [18] Junnan Li, Dongxu Li, Silvio Savarese, et al. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In [ICML](#), 2023.
- [19] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. [ArXiv, abs/2301.12597](#), 2023.
- [20] Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. [Advances in Neural Information Processing Systems](#), 35:17612–17625, 2022.
- [21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In [Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13](#), pages 740–755. Springer, 2014.
- [22] Haotian Liu, Chunyuan Li, Yuheng Li, et al. Improved baselines with visual instruction tuning. [arXiv preprint arXiv:2310.03744](#), 2023.
- [23] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. [ArXiv, abs/2310.03744](#), 2023.
- [24] Haotian Liu, Chunyuan Li, Qingyang Wu, et al. Visual instruction tuning. In [NeurIPS](#), 2023.
- [25] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. [ArXiv, abs/2304.08485](#), 2023.
- [26] Yuan Liu, Haodong Duan, Yuanhan Zhang, et al. Mmbench: Is your multi-modal model an all-around player? [arXiv preprint arXiv:2307.06281](#), 2023.
- [27] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In [Proceedings of the IEEE/CVF international conference on computer vision](#), pages 10012–10022, 2021.
- [28] Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. 2018.

- [29] Jiasen Lu, Christopher Clark, Rowan Zellers, Roozbeh Mottaghi, and Aniruddha Kembhavi. Unified-io: A unified model for vision, language, and multi-modal tasks. ArXiv, abs/2206.08916, 2022.
- [30] Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. Ocr-vqa: Visual question answering by reading text in images. In 2019 international conference on document analysis and recognition (ICDAR), pages 947–952. IEEE, 2019.
- [31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In International conference on machine learning, pages 8748–8763. PMLR, 2021.
- [32] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. arXiv preprint arXiv:2401.14159, 2024.
- [33] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 8317–8326, 2019.
- [34] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. arXiv preprint arXiv:2303.15389, 2023.
- [35] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. Advances in neural information processing systems, 30, 2017.
- [36] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. arXiv preprint arXiv:2304.14178, 2023.
- [37] Shukang Yin, Chaoyou Fu, Sirui Zhao, et al. A survey on multimodal large language models. arXiv preprint arXiv:2306.13549, 2023.
- [38] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara 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, pages 69–85. Springer, 2016.
- [39] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. arXiv preprint arXiv:2308.02490, 2023.
- [40] Zheng Zhang, Yong Xu, Jian Yang, Xuelong Li, and David Dian Zhang. A survey of sparse representation: Algorithms and applications. IEEE Access, 3:490–530, 2015.
- [41] Haozhe Zhao, Zefan Cai, Shuzheng Si, Xiaojuan Ma, Kaikai An, Liang Chen, Zixuan Liu, Sheng Wang, Wenjuan Han, and Baobao Chang. Mmicl: Empowering vision-language model with multi-modal in-context learning. arXiv preprint arXiv:2309.07915, 2023.
- [42] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023.
- [43] Deyao Zhu, Jun Chen, Xiaoqian Shen, et al. Minigt-4: Enhancing vision-language understanding with advanced large language models. arXiv preprint arXiv:2304.10592, 2023.
- [44] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigt-4: Enhancing vision-language understanding with advanced large language models. ArXiv, abs/2304.10592, 2023.
- [45] Wanrong Zhu, Jack Hessel, Anas Awadalla, Samir Yitzhak Gadre, Jesse Dodge, Alex Fang, Youngjae Yu, Ludwig Schmidt, William Yang Wang, and Yejin Choi. Multimodal c4: An open, billion-scale corpus of images interleaved with text. Advances in Neural Information Processing Systems, 36, 2024.

## NeurIPS Paper Checklist

### 1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: NA

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

### 2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: NA

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

### 3. Theory Assumptions and Proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: NA

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

#### 4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: NA

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

#### 5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?



Answer: [No]

Justification: NA

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

## 6. Experimental Setting/Details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: NA

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

## 7. Experiment Statistical Significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: NA

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

#### 8. Experiments Compute Resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: NA

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

#### 9. Code Of Ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics [https://neurips.cc/public/EthicsGuidelines?](https://neurips.cc/public/EthicsGuidelines)

Answer: [Yes]

Justification: NA

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

#### 10. Broader Impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: NA

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

#### 11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: NA

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

#### 12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: NA

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, [paperswithcode.com/datasets](https://paperswithcode.com/datasets) has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

#### 13. New Assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: NA

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

#### 14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: NA

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

#### 15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: NA

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.