# Visual Instruction Tuning for Multimodal Large Language Models: A Survey

Yingquan Chen[1]
[1] Shenzhen University,Shenzhen, China
2022280297@email.szu.edu.cn

***Abstract*—Multimodal large language models (MLLMs) integrate visual and textual data, enabling a range of applications, from visual question answering to tampering detection. However, traditional task-specific models lack the flexibility needed for complex, user-driven tasks. Visual Instruction Tuning (VIT) addresses this gap by training MLLMs to follow diverse visual instructions in a unified framework. This survey reviews the foundations and recent advancements in VIT, covering dataset creation, training methodologies, model architectures, and evaluation methods. We categorize prominent models by their application domains, including tampering detection, image generation, and interactive dialogue, discussing each model's unique contributions. Finally, we outline future directions for improving dataset diversity, training efficiency, and model interpretability to advance VIT research.**

***Index Terms*—Visual Instruction Tuning, Multimodal Large Language Models, Instruction-following Datasets, General-purpose Vision-language Models**

## 1. Introduction

THE rapid development of multimodal large language models (MLLMs) has transformed artificial intelligence, allowing these models to integrate visual and textual data to achieve complex cross-modal understanding and generation capabilities. MLLMs now play essential roles in tasks ranging from visual question answering and image description to content moderation and tampering detection. However, traditional models are constrained by task-specific limitations, with rigid architectures designed for narrowly defined functions. This inflexibility hinders their ability to handle dynamic, multi-task scenarios that require user-driven, instruction-following behavior.

Visual Instruction Tuning (VIT) has emerged as an approach to address these limitations. Originally applied within natural language processing (NLP), instruction tuning trains models to interpret and follow natural language commands. VIT extends this concept to MLLMs, creating models that can dynamically adapt to a variety of visual tasks based on language instructions. This paradigm shift from single-task specialization to general-purpose functionality enables MLLMs to become more interactive, versatile, and adaptable to real-world, instruction-based applications.As shown in Fig. 1, there is a growing focus on visual instruction fine-tuning

This survey systematically examines the latest developments in visual instruction tuning for MLLMs, with a focus on the unique challenges and innovations introduced by this new paradigm. Specifically, we organize the survey as follows:

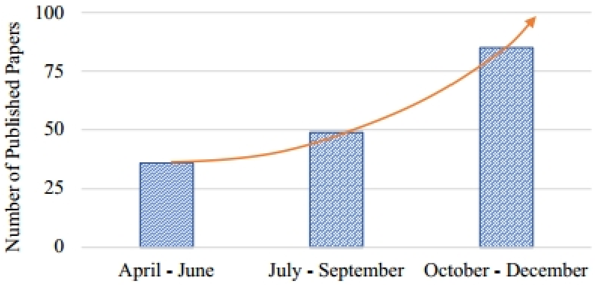


Fig. 1: The number of publications on visual instruction tuning in 2024.

1) **Background and Need for Instruction Tuning:** We discuss the limitations of traditional computer vision models and the evolution toward instruction-based paradigms that enable adaptable, instruction-following capabilities.
2) **Data Creation for Instruction Tuning:** We explore methods for constructing high-quality instruction-following datasets, from fully human-annotated to AI-assisted generation approaches.
3) **Training Architectures for Visual Instruction Tuning:** We introduce common architectures, including vision encoders, language models, and multimodal adapters, and explain how these components integrate to process instruction-following data effectively.
4) **Evaluation Methodologies:** We examine the primary evaluation methods for instruction-tuned MLLMs, including human assessments, automated LLM-based evaluations, and quantitative metrics.
5) **Representative Models by Application Domains:** We categorize key models based on their application areas, such as tampering detection, image generation, and multimodal dialogue, highlighting their unique architectures and capabilities.
6) **Future Directions:** We conclude by outlining key challenges and potential directions for advancing visual instruction tuning, including improvements in data diversity, training efficiency, and model interpretability.

This survey provides a structured and comprehensive review of visual instruction tuning (VIT) in multimodal large language models (MLLMs), offering researchers a valuable resource to understand current methodologies and explore future research directions in developing adaptable, instruction-following MLLMs. Specifically, we cover key aspects of VIT, including data creation, training techniques, model architectures, and evaluation strategies. By systematically organizing recent advancements, this survey aims to support the continued development and application of multimodal instruction tuning across diverse, real-world tasks.

## 2. Background

The development of computer vision task paradigms has evolved significantly, transitioning from single-task models with rigid architectures to adaptable, instruction-based frameworks that support diverse, user-defined tasks. This section reviews this evolution by first examining the limitations of traditional task paradigms, then introducing the instruction-based paradigm and the methodology of Visual Instruction Tuning (VIT), and concluding with a comparative analysis of these two paradigms to highlight the necessity of VIT in modern multimodal large language models (MLLMs)[4].

### 2.1 *Traditional Task Paradigm in Computer Vision*

The traditional task paradigm in computer vision is characterized by specialized models designed for fixed, single-purpose tasks. As illustrated in Fig. 2(a), these models include segmentation networks, object detection models, and other task-specific architectures that are tailored to address individual tasks. For example, segmentation models are equipped with specialized heads for mask prediction, while object detection models rely on bounding box modules. Although effective in performing their intended tasks within controlled environments, these models lack flexibility. Each new task often requires significant architectural adjustments, making it challenging to adapt to complex, multi-task applications where various user-defined instructions are needed. This single-purpose approach restricts the application of traditional models, limiting their scalability and versatility in dynamic environments.

### 2.2 *Instruction-Based Task Paradigm*

In response to the limitations of traditional models, the instruction-based task paradigm introduces an adaptive and flexible approach where models are guided by natural language instructions to complete tasks[9]. As shown in Fig. 2(b), this paradigm enables multimodal models to interpret and respond to a variety of tasks using a unified, language-based interface. Originally developed in the natural language processing (NLP) field, instruction tuning allows large language models (LLMs) to process diverse commands and adapt their outputs based on user-provided instructions. Visual Instruction Tuning (VIT) extends this concept to multimodal applications, training MLLMs to handle both visual and textual instructions seamlessly. This shift from single-task specialization to general-purpose functionality allows models to generalize across diverse tasks without needing architectural modifications, thereby bridging the gap between single-purpose vision models and flexible, general-purpose AI.

### 2.3 *Visual Instruction Tuning (VIT) Methodology*

Visual Instruction Tuning (VIT) aims to create general-purpose, multimodal models capable of following user-defined instructions across a wide range of visual tasks. The VIT process typically consists of two key stages:

1) **Instruction-Following Dataset Creation**: High-quality datasets are essential for effective instruction tuning. These datasets, which may be generated manually or with AI assistance, are formatted as (instruction, input, output) triplets to enable models to learn from diverse instructions and visual contexts.
2) **Supervised Fine-Tuning**: Using these datasets, MLLMs undergo supervised fine-tuning to develop accurate instruction-following capabilities. This tuning process enables models to dynamically interpret instructions within a unified framework, making them more adaptable and responsive to user-defined tasks[17].

The VIT methodology creates a standardized task interface that leverages natural language instructions, allowing MLLMs to function as versatile, interactive models suited to complex, real-world applications.

### 2.4 *Comparing Traditional and Instruction-Based Paradigms*

A comparison of the traditional and instruction-based paradigms reveals the advantages and trade-offs associated with each. Traditional models are precise and efficient for specific, well-defined tasks but lack the flexibility needed in environments that demand adaptability. In contrast, instruction-based models, particularly those fine-tuned through VIT, provide a flexible, user-directed interface that allows them to generalize across a variety of tasks based on natural language instructions. However, this flexibility requires extensive computational resources and diverse, high-quality datasets to ensure robust performance across tasks, making instruction-tuned models resource-intensive to develop and deploy.

### 2.5 *The Need for Instruction Tuning in MLLMs*

Instruction tuning represents a pivotal advancement in model design, enabling MLLMs to overcome the limitations of traditional task-specific approaches by adopting a flexible, instruction-following framework. This evolution empowers a single model to perform multiple tasks based on natural language commands, enhancing usability and scalability in real-world, multi-task settings. The capacity to follow diverse instructions without requiring architectural modifications is fundamental for building adaptable, general-purpose AI, positioning VIT as a foundational method in the continued development of multimodal large language models.

## 3. Visual Instruction-following data

In this section, we present a categorization of instruction tuning datasets based on their creation methods. We classify them into three main types: (1) Fully Human-Annotated Data, (2) Hybrid Human-AI Generated Data, and (3) Fully AI-Generated Data.

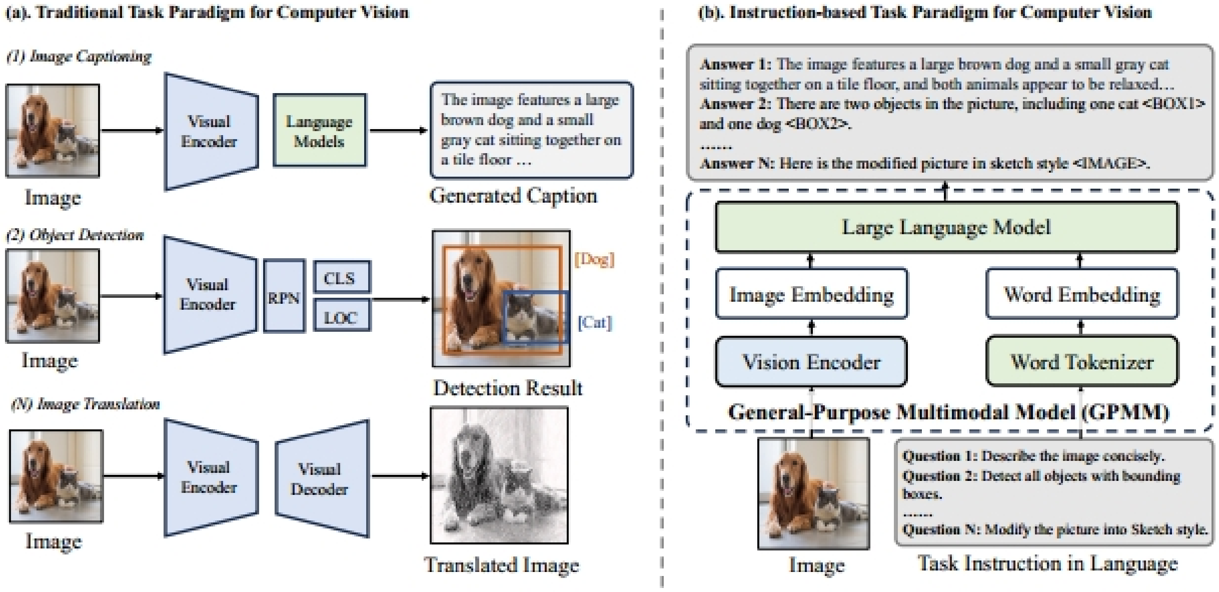

(a). Traditional Task Paradigm for Computer Vision
(1) Image Captioning
Visual Encoder
Language Models
The image features a large brown dog and a small gray cat sitting together on a tile floor ...
Image
Generated Caption
(2) Object Detection
Visual Encoder
RPN
CLS
LOC
[Dog]
[Cat]
Image
Detection Result
(N) Image Translation
Visual Encoder
Visual Decoder
Image
Translated Image
(b). Instruction-based Task Paradigm for Computer Vision
Answer 1: The image features a large brown dog and a small gray cat sitting together on a tile floor, and both animals appear to be relaxed...
Answer 2: There are two objects in the picture, including one cat <BOX1> and one dog <BOX2>.
......
Answer N: Here is the modified picture in sketch style <IMAGE>.
Large Language Model
Image Embedding
Word Embedding
Vision Encoder
Word Tokenizer
General-Purpose Multimodal Model (GPMM)
Question 1: Describe the image concisely.
Question 2: Detect all objects with bounding boxes.
......
Question N: Modify the picture into Sketch style.
Image
Task Instruction in Language

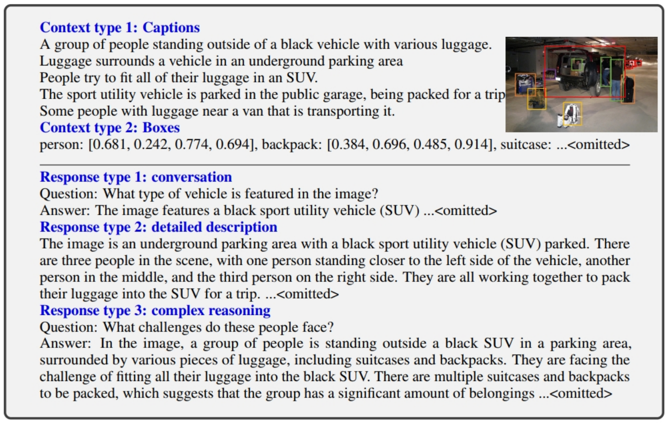
Fig. 2: Illustrations of traditional task paradigm for computer vision in (a) and instruction-based task paradigm for computer vision in (b). Compared with the paradigm in (a) that solves each single task independently by a dedicated model with task instruction implicitly designed in the model architecture, the new task paradigm with visual instruction tuning enables a general-purpose multimodal model that can follow arbitrary instructions and thus solve arbitrary tasks specified by the user.Figure is from [4].

### 3.1 Fully Human-Annotated Data

Fully human-annotated datasets are crafted exclusively by human annotators without any AI assistance. These datasets are often curated for high accuracy and quality, as human annotators can provide precise instructions and outputs for specific visual tasks. Examples of such datasets are typically used in tasks requiring detailed understanding, such as object detection, segmentation, and image captioning. However, the creation of fully human-annotated data is resource-intensive and less scalable, limiting its ability to cover a wide range of instruction types and visual contexts.

### 3.2 Hybrid Human-AI Generated Data

Hybrid human-AI generated datasets combine human expertise with AI assistance to create instruction-following data. In this approach, human annotators might generate initial examples or provide seed data, which AI models, such as large language models, then expand upon to generate additional instruction-output pairs. This hybrid method allows for scalability while maintaining a certain level of quality and diversity. By blending human oversight with AI-generated data, these datasets achieve a balance between precision and variety, making them suitable for training models to follow a wide array of instructions across different visual tasks.Step1, as shown in Fig. 4, shows us how to combine the now popular artificial intelligence to generate data.Just give the original images and assign some prompt to GPT and let GPT generate some fine-tuned datasets for specific areas.In the work of Q-instruct[23], authors used this combination of manual and AI to generate instruction data.



**Context type 1: Captions**
A group of people standing outside of a black vehicle with various luggage.
Luggage surrounds a vehicle in an underground parking area
People try to fit all of their luggage in an SUV.
The sport utility vehicle is parked in the public garage, being packed for a trip
Some people with luggage near a van that is transporting it.
**Context type 2: Boxes**
person: [0.681, 0.242, 0.774, 0.694], backpack: [0.384, 0.696, 0.485, 0.914], suitcase: ...<omitted>

**Response type 1: conversation**
Question: What type of vehicle is featured in the image?
Answer: The image features a black sport utility vehicle (SUV) ...<omitted>
**Response type 2: detailed description**
The image is an underground parking area with a black sport utility vehicle (SUV) parked. There are three people in the scene, with one person standing closer to the left side of the vehicle, another person in the middle, and the third person on the right side. They are all working together to pack their luggage into the SUV for a trip. ...<omitted>
**Response type 3: complex reasoning**
Question: What challenges do these people face?
Answer: In the image, a group of people is standing outside a black SUV in a parking area, surrounded by various pieces of luggage, including suitcases and backpacks. They are facing the challenge of fitting all their luggage into the black SUV. There are multiple suitcases and backpacks to be packed, which suggests that the group has a significant amount of belongings ...<omitted>

Fig. 3: One example to illustrate the instruction-following data. The top block shows the contexts such as captions and boxes used to prompt GPT, and the bottom block shows the three types of responses. Note that the visual image is not used to prompt GPT, we only show it here as a reference.Figure is from [11].

### 3.3 Fully AI-Generated Data

Fully AI-generated datasets are created entirely by large language models or other AI systems without human intervention[10]. In this approach, AI models generate both the instructions and corresponding outputs for various visual tasks, often using seed prompts or predefined templates to guide generation. This method is highly scalable and can produce extensive datasets with rich and diverse instructions. However, fully AI-generated data may sometimes lack the accuracy

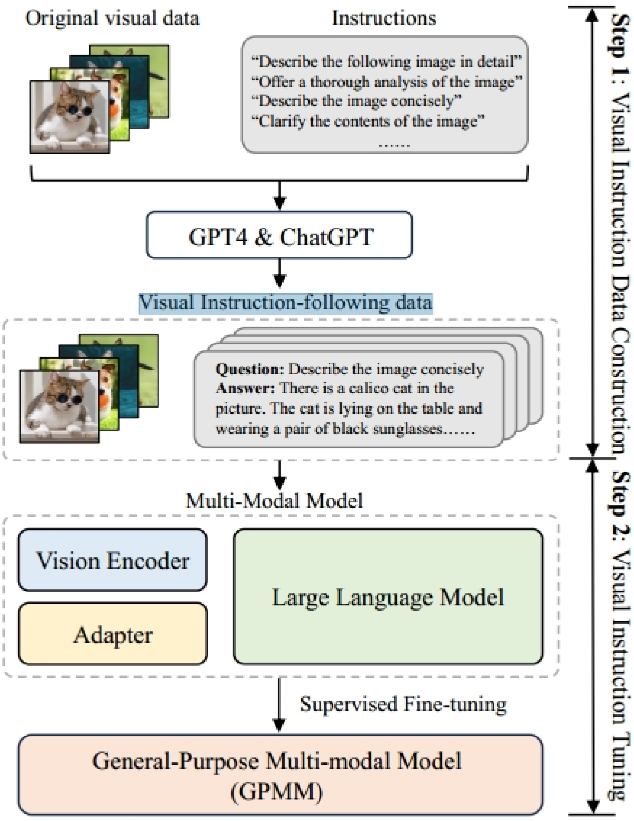

Original visual data
Instructions
"Describe the following image in detail"
"Offer a thorough analysis of the image"
"Describe the image concisely"
"Clarify the contents of the image"
......
GPT4 & ChatGPT
Visual Instruction-following data
Question: Describe the image concisely
Answer: There is a calico cat in the picture. The cat is lying on the table and wearing a pair of black sunglasses......
Multi-Modal Model
Vision Encoder
Adapter
Large Language Model
Supervised Fine-tuning
General-Purpose Multi-modal Model (GPMM)
Step 1: Visual Instruction Data Construction
Step 2: Visual Instruction Tuning

Fig. 4: Pipeline of visual instruction tuning.

and specificity of human-generated content, necessitating additional quality control steps. Fully AI-generated datasets are particularly useful for training general-purpose multimodal models that need to handle a broad spectrum of instruction types efficiently.

## 4. Training Architectures for Visual Instruction Tuning

Visual instruction tuning utilizes a multimodal model architecture to extract and process features from both image and text components within instruction-following data,as shown in Fig. 5. This architecture generally comprises a vision encoder and a large language model (LLM) as core components, which together facilitate the model's ability to interpret and respond to natural language instructions. In this section, we introduce the primary network architectures employed in visual instruction tuning, covering components for vision learning, language processing, and the mechanisms connecting these elements to form a unified multimodal framework[11].

### 4.1 Vision Encoder Architecture

Vision encoders play a crucial role in extracting rich visual features from input images, enabling the model to interpret visual data effectively. Transformer-based vision encoders, such as the Vision Transformer (ViT), are widely adopted for their versatility and effectiveness[3]. Vision Transformers use multi-head self-attention layers and feed-forward networks to extract image features in a sequential manner, making them highly suitable for complex visual tasks.

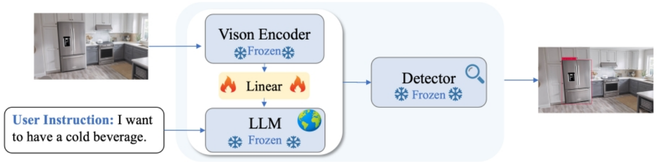


Fig. 5: An example of a basic framework for fine-tuning the linear mapping layer of a large model

Different pre-trained vision encoder variants are selected based on task specificity. For instance, CLIP-pretrained ViTs are commonly used for broad image understanding tasks due to their generalization capabilities, whereas more detailed and fine-grained visual analyses often utilize SAM-pretrained ViTs[27]. In applications requiring temporal data processing, such as video tasks, ViTs can be extended with temporal encoding layers to capture time-related information and model dynamic visual content.

For 3D image feature learning, specialized models like Point-BERT[26] and PointNet[18] are employed to process PointCloud data, facilitating a deeper understanding of spatial structures within 3D visual inputs. These models are designed to effectively handle the complexity of 3D spaces, enriching the model's capability to manage a broader range of visual modalities.

### 4.2 Language Model Architecture

Language models, typically large transformer-based models, are integral to interpreting the natural language instructions provided to the multimodal model. Standard Transformer architectures, such as those comprising encoder-decoder layers, are popular choices. Each encoder block in these models typically includes a multi-head self-attention layer and a multi-layer perceptron (MLP), while decoder blocks contain both multi-head attention and masked multi-head attention layers to handle sequential dependencies in text data.

Prominent LLMs used in visual instruction tuning include models like LLaMA, which are known for their adaptability across a wide array of language tasks. Based on the LLaMA architecture, several instruction-tuned variants, such as Vicuna[16] and Guanaco, have been developed to further optimize the model's text comprehension and generation abilities for instruction-following tasks. These language models provide the foundational text-processing capabilities required to interpret and generate contextually relevant responses based on the visual inputs and user instructions.

### 4.3 Multimodal Connection Mechanisms

To achieve effective multimodal instruction tuning, the visual encoder and language model must be seamlessly integrated, enabling coherent data flow between visual and textual domains. Commonly, an adapter module is employed to align the features from the vision encoder with the language model's embedding space. This adapter acts as a bridge, translating visual embeddings into a compatible format for the LLM, thereby ensuring smooth cross-modal interaction.

For example, LLaVA (Large Language and Vision Assistant)[19] and other similar frameworks use lightweight adapters, such as linear transformation layers, to map the output of the vision encoder into the word embedding space of the language model. This alignment ensures that the LLM can effectively interpret the visual context provided by the encoder, integrating it with the natural language instructions to generate coherent outputs.

In some cases, the adapter is designed with more advanced capabilities, such as dynamic routing, which selectively adjusts data flow based on the task or input modality. This approach enhances the flexibility of the model, allowing it to efficiently manage diverse instruction types across different visual tasks.

### 4.4 Overview of Data Flow in Training Architectures

The training data flow for visual instruction tuning generally follows a sequence of stages: the vision encoder first processes visual inputs to generate feature embeddings, which are then passed through an adapter module to align with the language model's input space. The language model, equipped with both the aligned visual embeddings and natural language instructions, then produces the model's response.

This integrated data flow structure enables instruction-tuned MLLMs to learn complex, multimodal instructions, combining visual understanding with responsive text generation. By leveraging both vision and language architectures, along with an efficient connection mechanism, visual instruction tuning frameworks can be trained to handle diverse instruction-following tasks with increased adaptability and interactivity.

## 5. Evaluation of Visual Instruction Tuning Models

Evaluating the performance of instruction-tuned multimodal large language models (MLLMs) is a critical step to ensure that these models can effectively interpret and execute a wide range of visual instructions. In this section, we discuss the main evaluation methodologies used in visual instruction tuning research, including human evaluation, automated evaluation with advanced LLMs (e.g., GPT-4), and traditional quantitative metrics. Each evaluation method serves different aspects of model performance, providing a comprehensive understanding of the model's capabilities across various tasks and instruction types.

### 5.1 Human Evaluation

Human evaluation is essential for assessing the nuanced capabilities of instruction-tuned MLLMs, particularly for tasks that require complex understanding and interaction. This evaluation approach involves human annotators who assess the model's outputs on criteria such as relevance, coherence, fluency, and appropriateness with respect to the given instructions. Since human judgment captures subtle aspects of language and context, human evaluation is often considered the gold standard for evaluating tasks that are inherently subjective or complex.

Common tasks assessed through human evaluation include visual question answering (VQA), image description, and content generation, where precise, contextually appropriate responses are crucial. While human evaluation offers valuable insights, it is time-consuming and resource-intensive, which limits its scalability for large-scale benchmarking.

### 5.2 Automated Evaluation with Advanced LLMs

Due to the resource limitations of human evaluation, recent studies have explored automated evaluation techniques using advanced LLMs, such as GPT-4[1], to approximate human judgments. In this setup, the model-generated outputs are assessed by GPT-4, which can evaluate aspects such as helpfulness, relevance, accuracy, and detail. GPT-4 assigns a score to each response, usually on a predefined scale (e.g., 1 to 10), with higher scores indicating better performance[12].

GPT-4-based evaluation has the advantage of being scalable, consistent, and less resource-intensive than human evaluation, while still capturing complex linguistic and contextual nuances. However, this method may introduce biases inherent in the evaluating LLM, and its effectiveness is often dependent on the quality and relevance of the evaluation prompts provided. Automated evaluation with LLMs is especially valuable for high-level performance assessment across large instruction-following datasets.

### 5.3 Quantitative Metric Evaluation

Traditional quantitative metrics remain a fundamental component of evaluating instruction-tuned MLLMs. These metrics offer objective and repeatable measurements for tasks that have clear, well-defined evaluation standards. For example, in discriminative tasks such as image classification, object detection, and visual grounding, commonly used metrics include accuracy, precision, recall, and F1 score. These metrics enable straightforward comparisons of model performance and are essential for tasks with established benchmarks.In the multimodal large language model, IOU[29] and other indicators can also be introduced to calculate the predicted mask results at the pixel level and compare with the actual GT mask[8].

For generative tasks, metrics such as BLEU, ROUGE, and CIDEr scores are frequently employed to evaluate the relevance and quality of model-generated descriptions. Complex image reasoning tasks, including visual question answering, are often evaluated using metrics such as accuracy or exact match scores. Recently developed benchmarks, such as MMBench[13] and SeedBench[6], further enhance the scope of quantitative evaluation by offering comprehensive test sets across multiple task categories and evaluation criteria, covering abilities such as fine-grained perception and logical reasoning[15].

### 5.4 Evaluation Summary and Insights

The combination of human, automated, and quantitative evaluations provides a holistic assessment framework for instruction-tuned MLLMs. Human evaluation captures nuanced language understanding, automated evaluation offers scalable assessment through LLM-based grading, and quantitative metrics deliver objective performance measures across standard benchmarks. Together, these methods offer a comprehensive understanding of model strengths and limitations, ensuring that instruction-tuned MLLMs are evaluated for both accuracy and contextual adaptability in responding to visual instructions.

## 6. Representative Models in Visual Instruction Tuning

In this section, we categorize prominent models in visual instruction tuning based on their application domains, highlighting representative models in each area. These models showcase varied approaches and architectures tailored for specific visual instruction-following tasks, such as visual question answering, image generation, multimodal dialogue, tampering detection, and interactive multimodal tasks. This organization provides an overview of key advancements across subfields, offering insights into each model's strengths and contributions.

### *6.1 Models for Visual Question Answering and Image Description*

Visual question answering (VQA) and image description tasks require models to interpret images and provide coherent textual responses to specific questions or prompts. Representative models in this domain prioritize robust vision-text alignment to handle complex image content effectively.

*1) LLaVA (Large Language and Vision Assistant).* LLaVA [11] is a foundational model for VQA and image description tasks, integrating a Vision Transformer (ViT) as the visual encoder and a large language model (LLM) to process textual instructions. An adapter module bridges the vision and language components, allowing LLaVA to accurately align visual features with natural language instructions. This architecture enables LLaVA to handle complex visual prompts and generate detailed responses, making it effective for VQA and descriptive tasks where precise visual-text alignment is crucial.

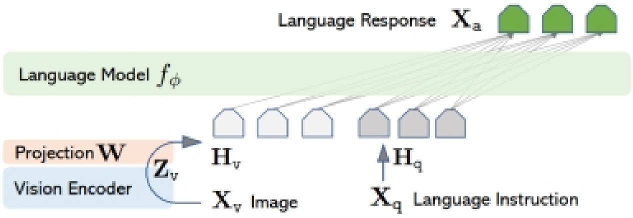


Fig. 6: Illustration of the LLaVA[11].Figure is from[11].

*2) InstructBLIP.* InstructBLIP [2] is a visual instruction tuning pipeline, which help construct a general-purpose multimodal model that can handle a broad range of vision tasks via a universal task interface with languages as task instructions. As shown in Fig. 7, InstructBLIP consists of a Query Transformer (Q-Former) that extracts instruction-aware visual features from the output embeddings of a frozen image encoder. These visual features are then fed as soft prompt input to a frozen Language Model (LLM). During instruction tuning, the Q-Former is fine-tuned while the image encoder and LLM remain frozen. This architecture allows for the extraction of task-relevant visual features based on the given instructions, enhancing the model's ability to follow instructions and generate responses. With a comprehensive study on vision-language instruction tuning, it demonstrates the effectiveness of InstructBLIP on zero-shot generalization to unseen tasks. The framework achieves state-of-the-art performance on a diverse set of vision-language tasks and provides novel techniques for instruction-aware visual feature extraction and balanced dataset sampling.

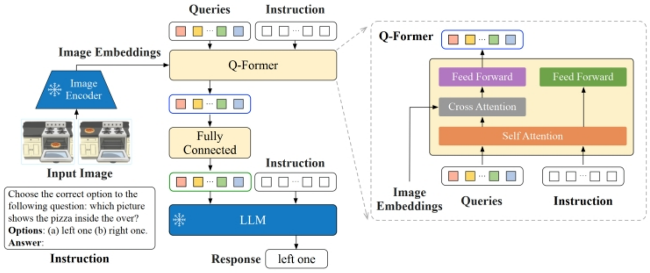
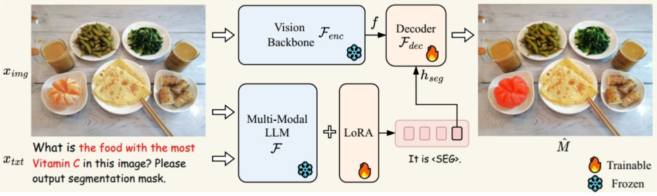


Fig. 7: Illustration of the InstructBLIP[2].Figure is from[2].

### *6.2 Models for Image Reasoning Segmentation*

Instruction-following in image generation requires models to not only understand textual inputs but also generate visual outputs that align with the instructed criteria.

*1) Lisa.* Lisa[5] stands out for its ability to generate images that precisely adhere to detailed textual prompts. Its strength lies in its robust integration of natural language understanding and generative image modeling. The model leverages a transformer-based architecture to process and encode textual descriptions, capturing their semantic meaning. This encoded text representation is then used to guide a generative model, such as a Generative Adversarial Network (GAN), to create images that match the input instructions. Lisa has demonstrated significant success in translating complex textual instructions into high-quality, contextually accurate images, offering precise control over visual details. Lisa's innovative approach has made it a powerful tool for creative applications, such as digital art, design, and interactive content generation.

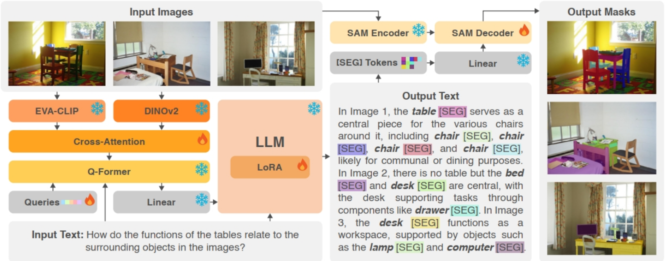


$x_{img}$
$x_{txt}$
What is the food with the most Vitamin C in this image? Please output segmentation mask.
Vision Backbone $\mathcal{F}_{enc}$
$f$
Decoder $\mathcal{F}_{dec}$
$h_{seg}$
Multi-Modal LLM $\mathcal{F}$
LoRA
It is <SEG>.
$\hat{M}$
Trainable
Frozen

Fig. 8: Illustration of the Lisa[5].Figure is from[5].

In terms of implementation, as shown in Fig. 8. Lisa's process begins with the extraction of textual features using a transformer model, which allows the system to understand complex and nuanced instructions. These features are then mapped to visual elements, ensuring that the generated image aligns with the semantic content of the input text. The generative model then synthesizes an image based on these visual cues. Additionally, Lisa supports iterative refinement, allowing users to fine-tune the generated images by adjusting various visual aspects until the desired output is achieved. This combination of advanced text-to-image alignment and image generation capabilities enables Lisa to perform both creative image synthesis and detailed manipulation, making it highly effective for applications in digital art, product design, and content creation.

*2) Prima.* PRIMA[20] is a novel Vision-Language Model designed for multi-image pixel-grounded reasoning segmentation. It addresses the gap in the field by integrating pixel-level grounding with robust multi-image reasoning capabilities. PRIMA is trained on M4SEG, a benchmark consisting of approximately 224K question-answer pairs that necessitate fine-grained visual understanding across multiple images. The model is optimized for computational efficiency, using an efficient vision module that queries fine-grained visual representations across multiple images, reducing TFLOPS by 25.3%.



In Image 1, the ***table*** [SEG] serves as a central piece for the various chairs around it, including ***chair*** [SEG], ***chair*** [SEG], ***chair*** [SEG], and ***chair*** [SEG], likely for communal or dining purposes. In Image 2, there is no table but the ***bed*** [SEG] and ***desk*** [SEG] are central, with the desk supporting tasks through components like ***drawer*** [SEG]. In Image 3, the ***desk*** [SEG] functions as a workspace, supported by objects such as the ***lamp*** [SEG] and ***computer*** [SEG].

**Input Text:** How do the functions of the tables relate to the surrounding objects in the images?

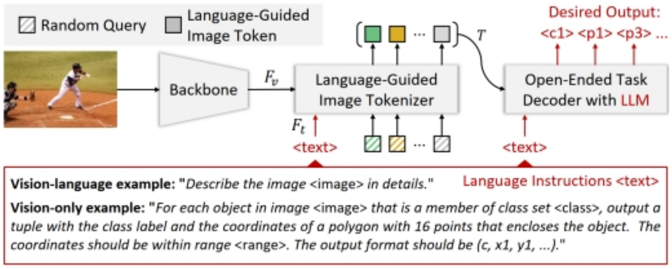
Fig. 9: Illustration of the PRIMA[20]. Figure is from[20].

As depicted in Fig. 9, PRIMA's architecture combines a large Vision-Language Model backbone with a vision module that utilizes self-supervised semantic features and a segmentation module for pixel grounding. The model is designed to efficiently identify and compare object functionalities and contextual relationships across scenes. It employs an instruction-guided multi-image adaptation module to reason across multiple images with fine-grained grounding. PRIMA's strong performance and computational efficiency, as demonstrated in Figure 2 of the original paper, showcase its capability to produce contextually rich, pixel-grounded explanations while maintaining high accuracy in pixel-level reasoning. This makes PRIMA a significant advancement for applications requiring detailed comparative analysis across images, such as in medical imaging or e-commerce product comparison.

### 6.3 Models for Object Detection

Object detection aims to identify and locate the objects in a given image or video frame. In general-purpose multimodal models with visual instruction tuning, object detection involves using visual instructions to guide the model in identifying and localizing objects within an image.

*1) Vision LLM.* In VisionLLM[22], object detection is one of the visioncentric tasks that the framework is designed to address. It leverages LLMs to handle object detection in an instructionbased way which is open-ended and customizable, allowing for the flexible definition and management of object detection tasks using language instructions. As shown in Fig. 10, VisionLMM consists of 3 core designs. The first is the language instructions that unify a diverse range of vision tasks and enable flexible task configuration. The second is the Instruction-Aware Image Tokenizer that extracts the required visual information according to the provided language instructions for effective comprehension and parsing of the visual input. The third one is the LLM-based opentask decoder. It takes inputs the extracted visual embeddings and language instruction embeddings and generates the expected results for various vision tasks. VisionLLM enables instruction-based task configuration, such as finegrained object detection and coarse-grained object detection, and achieves an mAP of over 60 % on the COCO dataset, which places it on par with detection-specific models.


▨ Random Query
■ Language-Guided Image Token
Desired Output:
<c1> <p1> <p3> ...
Backbone
$F_v$
Language-Guided Image Tokenizer
$T$
Open-Ended Task Decoder with LLM
$F_t$
<text>
<text>
Language Instructions <text>

**Vision-language example:** "*Describe the image <image> in details.*"

**Vision-only example:** "*For each object in image <image> that is a member of class set <class>, output a tuple with the class label and the coordinates of a polygon with 16 points that encloses the object. The coordinates should be within range <range>. The output format should be (c, x1, y1, ...).*"

Fig. 10: Illustration of the VisionLLM[22].Figure is from[22].

*2) ChatSpot.* ChatSpot[28] propose precise referring instruction tuning, which aims to enable multimodal large language models (MLLMs) to support fine-grained interaction. It focuses on utilizing a diverse range of prompts, like points and bounding boxes, as the location prompts to indicate the specific regions of interest (RoIs) in images. Precise referring instruction tuning improves the flexibility and userfriendliness of the interaction with MLLMs, particularly in the context of vision-language tasks. As illustrated in Fig. 11, the proposed unified end-to-end multimodal large language model, ChatSpot, comprises 3 main designs: an image encoder, a decoder-only large language model (LLM), and a modality alignment block. The image encoder processes visual inputs, while the LLM handles language understanding and generation. The modality alignment block aligns visual tokens with the language semantic space, enabling seamless integration of vision and language modalities for diverse forms of interaction, including mouse-clicking, drawing boxes, and native language input. ChatSpot exhibits promising performance on a series of designed evaluation tasks.

*3) All-Seeing.* All-Seeing (AS) Project [21], which contributes a largescale dataset, named AS-1B, for open-world panoptic visual perception as well as the All-Seeing Model, a universal vision-language model capable of recognizing and understanding context in arbitrary regions. As shown in Fig. 12, The All-Seeing Model (ASM) consists of two modules including a position-aware image tokenizer and an LLMbased decoder. The first module encodes image conditioned the

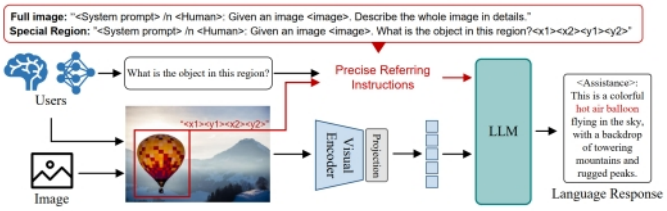

Full image: "<System prompt> /n <Human>: Given an image <image>. Describe the whole image in details."
Special Region: "<System prompt> /n <Human>: Given an image <image>. What is the object in this region?<x1><x2><y1><y2>"
Users
What is the object in this region?
Precise Referring Instructions
"<x1><y1><x2><y2>"
Image
Visual Encoder
Projection
LLM
<Assistance>: This is a colorful hot air balloon flying in the sky, with a backdrop of towering mountains and rugged peaks.
Language Response

Fig. 11: Illustration of the ChatSpot[28].Figure is from[28].

location information represented as bounding boxes, masks, and points, which empowers ASM with the location ability. As the second module inherits world knowledge and reasoning ability from the pre-trained LLMs, it can provide a robust foundation for visual perception. Additionally, ASM designs a special prompt to enable the model to switch to and handle generative or discriminative vision tasks accordingly. The ASM model demonstrates remarkable zero-shot performance in various vision and language tasks, including regional retrieval, recognition, captioning, and questionanswering, and is evaluated on representative vision and vision-language tasks.

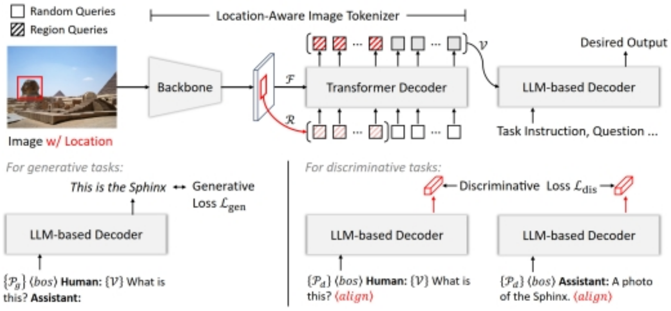

Random Queries
Region Queries
Location-Aware Image Tokenizer
Image w/ Location
Backbone
$\mathcal{F}$
$\mathcal{R}$
Transformer Decoder
$\mathcal{V}$
Desired Output
LLM-based Decoder
Task Instruction, Question ...
For generative tasks:
This is the Sphinx
Generative Loss $\mathcal{L}_{gen}$
LLM-based Decoder
$\{\mathcal{P}_g\}$ $\langle bos \rangle$ Human: $\{\mathcal{V}\}$ What is this? Assistant:
For discriminative tasks:
Discriminative Loss $\mathcal{L}_{dis}$
LLM-based Decoder
LLM-based Decoder
$\{\mathcal{P}_d\}$ $\langle bos \rangle$ Human: $\{\mathcal{V}\}$ What is this? $\langle align \rangle$
$\{\mathcal{P}_d\}$ $\langle bos \rangle$ Assistant: A photo of the Sphinx. $\langle align \rangle$

Fig. 12: Illustration of the All-Seeing[21].Figure is from[21].

### 6.4 Models for Tampering Detection and Localization

Instruction-following in tampering detection requires models that can accurately identify, localize, and explain potential manipulations within images. This domain relies heavily on the model's ability to discern visual inconsistencies and generate precise explanations that align with user instructions.

*1) FakeShield.* FakeShield, developed by Peking University, represents a specialized application of instruction-tuned models in the tampering detection domain. The model uses AI-assisted dataset generation, where large language models (LLMs) help create diverse instruction-output pairs focused on identifying manipulated visual content. FakeShield's architecture includes a visual encoder and an LLM, connected through an adapter module, that work together to highlight tampered regions and generate detailed explanations for detected manipulations[24].

FakeShield stands out for its multi-modal approach to image forgery detection and localization (IFDL). It not only decouples the detection and localization process but also provides a reasonable judgment basis, which alleviates the black-box property and unexplainable issue of existing IFDL methods. By leveraging the capabilities of GPT-4o, FakeShield generates comprehensive triplets consisting of a tampered image, a modified area mask, and a detailed description of the edited region through a meticulously crafted prompt.

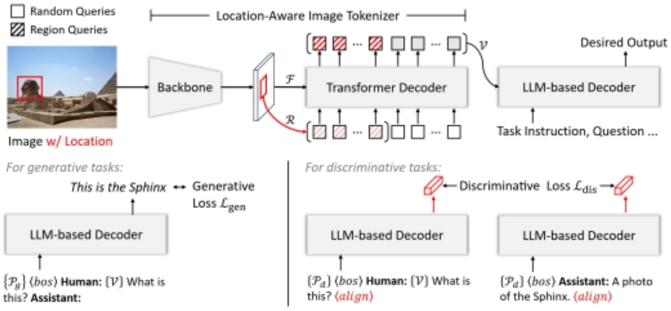

Random Queries
Region Queries
Location-Aware Image Tokenizer
Image w/ Location
Backbone
$\mathcal{F}$
$\mathcal{R}$
Transformer Decoder
$\mathcal{V}$
Desired Output
LLM-based Decoder
Task Instruction, Question ...
For generative tasks:
This is the Sphinx
Generative Loss $\mathcal{L}_{gen}$
LLM-based Decoder
$\{\mathcal{P}_g\}$ $\langle bos \rangle$ Human: $\{\mathcal{V}\}$ What is this? Assistant:
For discriminative tasks:
Discriminative Loss $\mathcal{L}_{dis}$
LLM-based Decoder
LLM-based Decoder
$\{\mathcal{P}_d\}$ $\langle bos \rangle$ Human: $\{\mathcal{V}\}$ What is this? $\langle align \rangle$
$\{\mathcal{P}_d\}$ $\langle bos \rangle$ Assistant: A photo of the Sphinx. $\langle align \rangle$

Fig. 13: Illustration of the FakeShield[24].Figure is from[24].

In terms of implementation, as shown in Fig. 13, FakeShield's process begins with the extraction of textual features using a transformer model, which allows the system to understand complex and nuanced instructions. These features are then mapped to visual elements, ensuring that the generated image aligns with the semantic content of the input text. The generative model then synthesizes an image based on these visual cues. Additionally, FakeShield supports iterative refinement, allowing users to fine-tune the generated images by adjusting various visual aspects until the desired output is achieved. This combination of advanced text-to-image alignment and image generation capabilities enables FakeShield to perform both creative image synthesis and detailed manipulation, making it highly effective for applications in digital art, product design, and content creation.

*2) EditScout.* EditScout[14], introduced by VinAI Research in collaboration with the University of Maryland and Vanderbilt University, is a pioneering framework designed to locate forged regions in images edited using diffusion-based techniques. This method addresses the challenges posed by the latest advancements in image editing technologies, which can produce near-indistinguishable forged regions that blend seamlessly with the original imagery.

EditScout leverages the contextual and semantic strengths of Multimodal Large Language Models (MLLMs) to generate segmentation masks that highlight tampered areas. The framework consists of two core modules: an MLLM-based reasoning query generation module and a segmentation network. The MLLM utilizes visual features extracted from a pretrained visual encoder, such as CLIP, along with a prompt for localizing edited regions to generate the reasoning query. This query is then fed into the SAM to produce the segmentation mask.

As depicted in Fig. 14, the EditScout framework takes a user's prompt and image as input, producing a sequence of text tokens that include a special [SEG] token representing the reasoning query and the edit instruction. The second module then uses this [SEG] token as a query to generate a binary mask indicating the edited regions. Notably, only the mask decoder and a part of the MLLM are fine-tuned, while the other components remain frozen. EditScout demonstrates promising results on datasets like MagicBrush, AutoSplice, and PerfBrush, outperforming previous approaches in mIoU

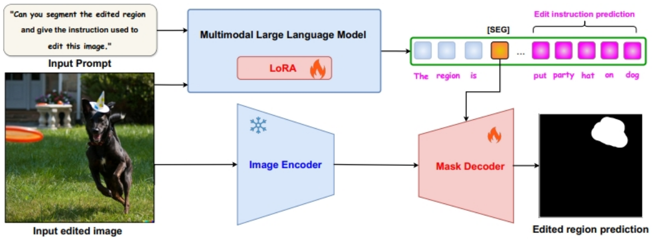

"Can you segment the edited region and give the instruction used to edit this image."
Input Prompt
Multimodal Large Language Model
LoRA
Edit instruction prediction
[SEG]
The region is
put party hat on dog
Image Encoder
Mask Decoder
Input edited image
Edited region prediction

Fig. 14: Illustration of the EditScout[14].Figure is from[14].

and F1-score metrics, especially excelling on the PerfBrush dataset featuring previously unseen types of edits.

### 6.5 Models for Image Generation

Image generation models have revolutionized the way we create and manipulate visual content. These models enable the synthesis of new images from textual descriptions, the transformation of existing images, and the generation of entirely new scenes that are coherent and aesthetically pleasing. The advent of large language models (LLMs) has further expanded the capabilities of image generation by providing a bridge between natural language and visual data. This section will explore the latest models that leverage the power of LLMs for image generation, focusing on their ability to understand and execute complex visual tasks as instructed by textual prompts.

*1) GPT4Tools.* GPT4Tools, proposed by Rui Yang et al. from Tsinghua Shenzhen International Graduate School and Tencent AI Lab, is an innovative framework designed to teach Large Language Models (LLMs) to use multimodal tools effectively[25]. This approach addresses the critical need for LLMs to interact with the visual domain, enhancing their capabilities beyond text-based tasks.

The core innovation of GPT4Tools lies in its ability to transform advanced, proprietary LLMs, such as ChatGPT and GPT-4, into open-source models like LLaMA and OPT, which can utilize tools for various visual tasks. This is achieved through self-instruction, where an advanced teacher model generates an instruction-following dataset by prompting with multi-modal contexts. The dataset is then used to fine-tune the open-source LLMs using Low-Rank Adaptation (LoRA) optimization, enabling them to solve a range of visual problems, including visual comprehension and image generation.

GPT4Tools stands out for its effectiveness in improving the accuracy of tool invocation and its ability to enable zero-shot capacity for unseen tools. As shown in Fig. 15, the framework involves prompting an advanced teacher model, such as GPT-3.5, with image content and tool definitions to generate a dataset rich in tool-related instructions. This dataset is then used to train an open-source LLM, equipping it with the ability to understand and execute visual tasks as instructed by textual prompts.

*2) TEXTBIND.* TEXTBIND, introduced by Huayang Li et al. from Tencent AI Lab, Nara Institute of Science and Technology, and Tsinghua University, is a groundbreaking framework that empowers Large Language Models (LLMs) with multi-turn interleaved multimodal instruction-following capabilities in the wild[7]. This approach addresses the challenge of obtaining high-quality exemplar data by requiring only image-caption pairs to generate multi-turn multimodal instruction-response conversations from a language model.

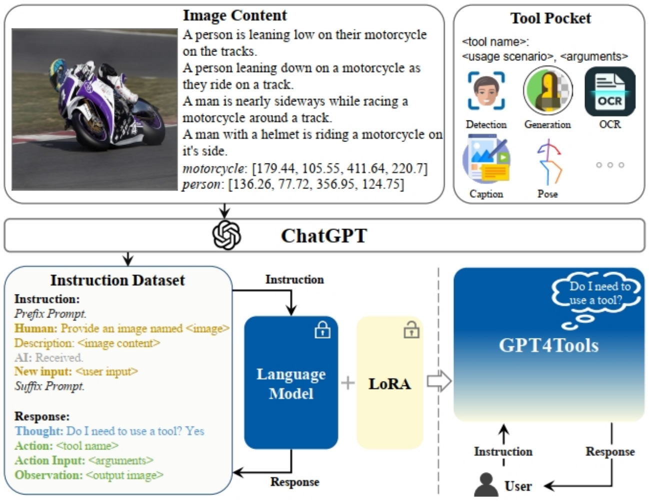


Image Content
A person is leaning low on their motorcycle on the tracks.
A person leaning down on a motorcycle as they ride on a track.
A man is nearly sideways while racing a motorcycle around a track.
A man with a helmet is riding a motorcycle on it's side.
motorcycle: [179.44, 105.55, 411.64, 220.7]
person: [136.26, 77.72, 356.95, 124.75]
Tool Pocket
<tool name>:
<usage scenario>, <arguments>
Detection
Generation
OCR
Caption
Pose
ChatGPT
Instruction Dataset
Instruction:
Prefix Prompt.
Human: Provide an image named <image>
Description: <image content>
AI: Received.
New input: <user input>
Suffix Prompt.
Response:
Thought: Do I need to use a tool? Yes
Action: <tool name>
Action Input: <arguments>
Observation: <output image>
Instruction
Language Model
LoRA
Response
Do I need to use a tool?
GPT4Tools
Instruction
Response
User

Fig. 15: GPT4Tools Framework. The model is trained to understand when to use tools and how to execute visual tasks based on textual instructions.

The core innovation of TEXTBIND lies in its ability to represent images through their textual descriptions and utilize an LLM to generate multi-turn instructions and responses. To ensure the coherence and meaningfulness of the constructed multi-turn conversations, the authors propose strategies such as topic-aware image sampling and human-in-the-loop refinement of in-context demonstrations. TEXTBIND can harvest large-scale datasets given the abundance of public image-caption pairs and provides examples of processing and generating arbitrarily interleaved image-text content.

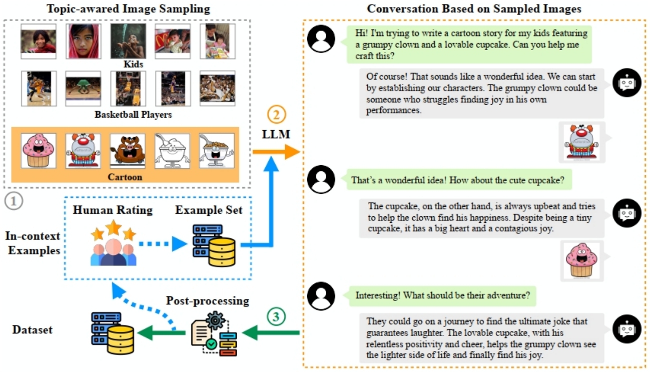
To accommodate interleaved image-text inputs and outputs,as shown in Fig. 16, the authors devise MIM, a multimodal model that emphasizes the reasoning abilities of LLMs and seamlessly integrates image encoder and decoder models. Extensive quantitative and qualitative experiments demonstrate that MIM trained on TEXTBIND achieves remarkable generation capability in multimodal conversations compared to recent baselines.

## 7. Challenges and Future Directions in Visual Instruction Tuning

The field of Visual Instruction Tuning (VIT) has made significant strides, enabling multimodal large language models (MLLMs) to interpret and execute visual tasks based on natural language instructions. However, several challenges remain that need to be addressed to enhance the performance and applicability of these models. This section discusses the current limitations of VIT and proposes future research directions to advance the field.

### 7.1 Diversity and Quality of Instruction-Following Datasets

One of the primary challenges in VIT is the lack of diverse and high-quality datasets. Current datasets often lack the



Hi! I'm trying to write a cartoon story for my kids featuring a grumpy clown and a lovable cupcake. Can you help me craft this?

Of course! That sounds like a wonderful idea. We can start by establishing our characters. The grumpy clown could be someone who struggles finding joy in his own performances.

That's a wonderful idea! How about the cute cupcake?

The cupcake, on the other hand, is always upbeat and tries to help the clown find his happiness. Despite being a tiny cupcake, it has a big heart and a contagious joy.

Interesting! What should be their adventure?

They could go on a journey to find the ultimate joke that guarantees laughter. The lovable cupcake, with his relentless positivity and cheer, helps the grumpy clown see the lighter side of life and finally find his joy.

Fig. 16: TEXTBIND Framework. The model is capable of generating multi-turn conversations with interleaved images and text based on textual descriptions of images.

breadth of scenarios and instructions needed to train models effectively for real-world applications. Developing datasets that cover a wider range of visual tasks and instructions is crucial for improving model generalization and robustness. Future work should focus on creating more comprehensive datasets that reflect the complexity and variability of real-world visual tasks.

### 7.2 Model Interpretability and Explainability

As MLLMs become more complex, understanding their decision-making processes becomes increasingly difficult. Enhancing the interpretability of VIT models is essential for building user trust and facilitating better integration into practical applications. Future research should explore techniques to provide insights into how models interpret instructions and make predictions, making the models more transparent and explainable.

### 7.3 Training Efficiency and Resource Management

Many state-of-the-art VIT models require substantial computational resources, which can limit their accessibility and scalability. Improving training efficiency and reducing resource consumption without compromising performance is a critical area for future research. Innovations in training methodologies, such as more efficient optimization algorithms or architectures tailored for VIT, can help address this challenge.

### 7.4 Generalization and Adaptability

While VIT models have shown promise in controlled environments, their ability to generalize to new tasks and scenarios outside of their training data is still limited. Future research should investigate approaches to improve the adaptability of VIT models, allowing them to learn from fewer examples and adapt more effectively to novel visual tasks.

### 7.5 Ethical Considerations and Bias Mitigation

The development and deployment of VIT models raise ethical concerns, particularly regarding the potential for generating misleading or biased content. Future research must consider the ethical implications of VIT applications and develop strategies to mitigate biases in model training and output.

### 7.6 Integration with Emerging Technologies

As new technologies such as augmented reality (AR) and virtual reality (VR) continue to evolve, VIT models should be designed to integrate seamlessly with these platforms. This integration can open up new applications and use cases, such as interactive training simulations and immersive content creation.

In conclusion, the future of VIT lies in addressing these challenges and exploring new research directions. By focusing on dataset diversity, model interpretability, training efficiency, generalization, ethical considerations, and technology integration, VIT can continue to advance towards more capable, adaptable, and responsible multimodal AI systems.

## 8. Conclusion

This survey provides an in-depth examination of Visual Instruction Tuning (VIT), highlighting its significant role in advancing the capabilities of multimodal Large Language Models (MLLMs). VIT has enabled MLLMs to interpret and perform a diverse array of visual tasks based on natural language instructions, surpassing the limitations of traditional, task-specific models. The review covers dataset creation, training methodologies, model architectures, and evaluation approaches, categorizing prominent models by their application domains. While VIT has shown promise, challenges such as dataset diversity, model interpretability, and training efficiency remain. Future research must address these to enhance MLLMs' scalability and applicability. Advancing VIT will bring us closer to achieving truly versatile multimodal AI systems that can interact seamlessly with human instructions across various domains.

## References

[1] Josh Achiam et al. "Gpt-4 technical report". In: *arXiv preprint arXiv:2303.08774* (2023).

[2] Wenliang Dai et al. "InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning". In: *Advances in Neural Information Processing Systems*. Ed. by A. Oh et al. Curran Associates, Inc., 2023, pp. 49250–49267.

[3] Kai Han et al. "A survey on vision transformer". In: *IEEE transactions on pattern analysis and machine intelligence* 45.1 (2022), pp. 87–110.

[4] Jiaxing Huang et al. "Visual instruction tuning towards general-purpose multimodal model: A survey". In: *arXiv preprint arXiv:2312.16602* (2023).

[5] Xin Lai et al. "Lisa: Reasoning segmentation via large language model". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024, pp. 9579–9589.

[6] Bohao Li et al. "Seed-bench: Benchmarking multimodal llms with generative comprehension". In: *arXiv preprint arXiv:2307.16125* (2023).

[7] Huayang Li et al. "Textbind: Multi-turn interleaved multimodal instruction-following in the wild". In: *Findings of the Association for Computational Linguistics ACL 2024*. 2024, pp. 9053–9076.

[8] Shiyang Li et al. "Instruction-following evaluation through verbalizer manipulation". In: *arXiv preprint arXiv:2307.10558* (2023).

[9] Baptist Liefooghe, Dorit Wenke, and Jan De Houwer. "Instruction-based task-rule congruency effects." In: *Journal of Experimental Psychology: Learning, Memory, and Cognition* 38.5 (2012), p. 1325.

[10] Ryan Lingo. "Exploring the Potential of AI-Generated Synthetic Datasets: A Case Study on Telematics Data with ChatGPT". In: *arXiv preprint arXiv:2306.13700* (2023).

[11] Haotian Liu et al. "Visual instruction tuning". In: *Advances in neural information processing systems* 36 (2024).

[12] Lei Liu et al. "Towards Automatic Evaluation for LLMs' Clinical Capabilities: Metric, Data, and Algorithm". In: *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2024, pp. 5466–5475.

[13] Yuan Liu et al. "Mmbench: Is your multi-modal model an all-around player?" In: *European Conference on Computer Vision*. Springer. 2025, pp. 216–233.

[14] Quang Nguyen et al. "EditScout: Locating Forged Regions from Diffusion-based Edited Images with Multimodal LLM". In: *arXiv preprint arXiv:2412.03809* (2024).

[15] Hanseok Oh et al. "INSTRUCTIR: A Benchmark for Instruction Following of Information Retrieval Models". In: *arXiv preprint arXiv:2402.14334* (2024).

[16] Baolin Peng et al. "Instruction tuning with gpt-4". In: *arXiv preprint arXiv:2304.03277* (2023).

[17] Nusrat Jahan Prottasha et al. "Transfer learning for sentiment analysis using BERT based supervised fine-tuning". In: *Sensors* 22.11 (2022), p. 4157.

[18] Charles R Qi et al. "Pointnet: Deep learning on point sets for 3d classification and segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 652–660.

[19] Hugo Touvron et al. "Llama: Open and efficient foundation language models". In: *arXiv preprint arXiv:2302.13971* (2023).

[20] Muntasir Wahed et al. "PRIMA: Multi-Image Vision-Language Models for Reasoning Segmentation". In: *arXiv preprint arXiv:2412.15209* (2024).

[21] Weiyun Wang et al. "The all-seeing project: Towards panoptic visual recognition and understanding of the open world". In: *arXiv preprint arXiv:2308.01907* (2023).

[22] Wenhai Wang et al. "Visionllm: Large language model is also an open-ended decoder for vision-centric tasks". In: *Advances in Neural Information Processing Systems* 36 (2024).

[23] Haoning Wu et al. "Q-instruct: Improving low-level visual abilities for multi-modality foundation models". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024, pp. 25490–25500.

[24] Zhipei Xu et al. "Fakeshield: Explainable image forgery detection and localization via multi-modal large language models". In: *arXiv preprint arXiv:2410.02761* (2024).

[25] Rui Yang et al. "GPT4Tools: Teaching Large Language Model to Use Tools via Self-instruction". In: *Advances in Neural Information Processing Systems*. Ed. by A. Oh et al. Vol. 36. Curran Associates, Inc., 2023, pp. 71995–72007.

[26] Xumin Yu et al. "Point-bert: Pre-training 3d point cloud transformers with masked point modeling". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 19313–19322.

[27] Renrui Zhang et al. "Pointclip: Point cloud understanding by clip". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 8552–8562.

[28] Liang Zhao et al. "Chatspot: Bootstrapping multimodal llms via precise referring instruction tuning". In: *arXiv preprint arXiv:2307.09474* (2023).

[29] Zhaohui Zheng et al. "Distance-IoU loss: Faster and better learning for bounding box regression". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 34. 07. 2020, pp. 12993–13000.