DigitalLLaVA: Incorporating Digital Cognition Capability for Physical World Comprehension in Multimodal LLMs

Shiyu Li[1,2], Pengxu Wei[2,3], Pengchong Qiao[1,2,4], Chang Liu[5*], Jie Chen[1,2,4*]
[1]School of Electronic and Computer Engineering, Peking University, Shenzhen, China
[2]Pengcheng Laboratory, Shenzhen, China
[3]Sun Yat-Sen University, Guangzhou, China
[4]AI for Science (AI4S)-Preferred Program, Peking University Shenzhen Graduate School, China
[5]Department of Automation and BNRist, Tsinghua University, Beijing, China
{shiyuli, pcqiao}@stu.pku.edu.cn, weipx3@mail.sysu.edu.cn
liuchang2022@tsinghua.edu.cn, jiechen2019@pku.edu.cn

Abstract

Multimodal Large Language Models (MLLMs) have shown remarkable cognitive capabilities in various cross-modal tasks. However, existing MLLMs struggle with tasks that require physical digital cognition, such as accurately reading an electric meter or pressure gauge. This limitation significantly reduces their effectiveness in practical applications like industrial monitoring and home energy management, where digital sensors are not feasible. For humans, physical digits are artificially defined quantities presented on specific carriers, which require training to recognize. As existing MLLMs are only pre-trained in the manner of object recognition, they fail to comprehend the relationship between digital carriers and their reading. To this end, referring to human behavior, we propose a novel **DigitalLLaVA** method to explicitly inject digital cognitive abilities into MLLMs in a two-step manner. In the first step, to improve the MLLM's understanding of physical digit carriers, we propose a ***digit carrier mapping*** method. This step utilizes object-level text-image pairs to enhance the model's comprehension of objects containing physical digits. For the second step, unlike previous methods that rely on sequential digital prediction or digit regression, we propose a ***32 bit floating point simulation*** approach that treats digit prediction as a whole. Using digit-level text-image pairs, we train three float heads to predict 32-bit floating-point numbers using 0/1 binary classification. This step significantly reduces the search space, making the prediction process more robust and straightforward. Being simple but effective, our method can identify very precise metrics (i.e., accurate to ± 0.001) and provide floating-point results, showing its applicability in digital carrier domains.

Introduction

The emergence of Multimodal Large Language Models (MLLMs) (OpenAI 2023; Team et al. 2023; Liu et al. 2023a; Li et al. 2023b; Liu et al. 2024a) has shown significant promise in handling complex tasks, such as image captioning and visual question answering. Despite these advancements, MLLMs face substantial challenges in specialized domains. Physical digital cognition, a critical aspect of Artificial Intelligence for Science (AI4Science), addresses some of these challenges. It finds broad applications in fields such as medical diagnostics, smart manufacturing, industrial automation, smart homes, the Internet of Things (IoT), augmented reality (AR) and virtual reality (VR).

For humans, the physical digit is an artificially defined quantity presented on a specific carrier that requires postnatal training to learn to recognize it. It is a fundamental component of human intelligence that allows humans to interact with and make sense of the world around us. In contrast, existing MLLMs struggle due to their inability to grasp numerical concepts and relationships. As shown in Fig. 1, the answers from GPT4V, Gemini Pro, and LLaVA are all incorrect. In the vernier-caliper case, Gemini Pro fails to provide an answer, while both GPT-4V and LLaVA show significant deviations. This is mainly because these models are pre-trained for object recognition but lack cognitive learning for physical digits. As a result, MLLMs lack understanding of physical quantities and misinterpret the physical information presented in images. In the thermometer case, the results from GPT-4V, Gemini Pro, and LLaVA all approximate 36 degrees, but diverge from the ground truth of 36.6 degrees. This discrepancy arises because they encode each digit individually, making it difficult to predict the correct sequence of numbers consistently. Although some studies on LLM have explored methods for encoding digits (Charton 2021; Anil et al. 2022; Grosse et al. 2023; Zhu et al. 2024), they have not performed well in terms of precision, efficiency, and scope.

To bridge this gap, we introduce an innovative method, **DigitalLLaVA**, to explicitly inject digital cognitive abilities into MLLMs in a two-step manner. In the first step, humans typically confirm the carrier before recognizing digits, as different carriers have various physical rules. However, MLLMs are pre-trained with a small proportion of carrier data. Therefore, we propose a ***digit carrier mapping*** step, where we finetune the MLLM's image encoder with object-level text-image pairs to enhance its recognition ability of physical carriers. In the second step, as predicting continuous numbers accurately is challenging for MLLMs due to the large search space, we propose a ***32-bit floating-point simulation*** step. This step approximates continuous numbers using discrete 32-bit floating-point representations. Specifically, we use digit-level text-image pairs to train three

*Corresponding author.

Question: What is the reading from the thermometer?

GPT4V: The reading from the thermometer is 36.5 degrees Celsius.

Gemini: The time is 36 degrees.

LLaVA: The thermometer reading is just above 36.8 degrees.

Ours: The reading is 36.6 degrees.

GT: The thermometer reads 36.6 °C.

Question: What is the reading from the speedometer?

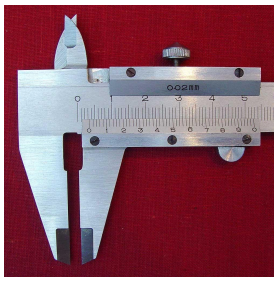
GPT4V: The reading is 100 km/h.

Gemini: The reading on the speedometer in the image is 0 kilometers per hour (km/h).

LLaVA: The speedometer in the image shows a speed of 108 km/h.

Ours: The reading is 218 km/h.

GT: The reading is 218 km/h.



Question: What is the reading from the vernier caliper?

GPT4V: The reading is 1.5cm.

Gemini: I can't determine the reading from the vernier caliper in the image.

LLaVA: The total reading on the vernier caliper is 1.60 cm.

Ours: The reading is 0.362cm.

GT: The reading is 0.362cm.

Figure 1: The answers from three MLLMs: GPT-4v, Gemini Pro, and LLaVA were evaluated against our method. Across three distinct physical quantities, these MLLMs failed to provide correct answers. In contrast, our approach yielded accurate responses for all evaluated physical entities.

float heads to predict a 1-bit signal, an 8-bit exponent, and a 23-bit fraction, which are then combined to obtain the precise digit. By employing float32, we only need to predict 32-bit 0/1 binary classification, significantly reducing the search space. During sentence inference, traditional numeric tokens are replaced with a specialized float token. If the MLLM head outputs a float token, it triggers the float head to predict the digital values. Our method substantially enhances the model's capability to recognize physical digits with high precision, i.e., accurate to 0.001, marking a significant advancement in precisely understanding the world.

Our contributions are threefold:

- First, we identify and analyze the limitations of current MLLMs in physical digital cognition.
- Second, we propose DigitalLLaVA, which incorporates a digit carrier mapping step and a 32-bit floating-point simulation step, explicitly designed to inject digital cognitive abilities into MLLMs.
- Third, we demonstrate through extensive experiments that our approach significantly enhances the ability of MLLMs to recognize physical digits with high precision, specifically accurate to 0.001. This highlights its potential for applicability across various domains.

Related Works

Multimodal LLMs. Recently, due to the rapid development of LLM (Devlin et al. 2018; Brown et al. 2020; Wei et al. 2022; Touvron et al. 2023; OpenAI 2023), multimodal large language models have also got a lot of attention (Hao et al. 2022; Alayrac et al. 2022; Li et al. 2023a; Huang et al. 2024; Lin et al. 2023). MLLMs are transformed into a format compatible with LLMs through the use of CLIP vision encoder (Radford et al. 2021; Sun et al. 2023). For instance, LLaVA-v1.5 (Liu et al. 2024b) is developed by building upon the model Vicuna v1.5 (Chiang et al. 2023). There are several strategies to improve the performance of MLLMs. A common method during pre-training involves maintaining the pre-trained components, such as visual encoders and LLMs, in a fixed state while training a learnable interface (Chen et al. 2023; Pi et al. 2023; Li et al. 2024). Additionally, some approaches adopt parameter-efficient fine-tuning by integrating lightweight trainable adapters into the models (Gao et al. 2023; Luo et al. 2024). Furthermore, some works (Li et al. 2022; Driess et al. 2023) utilize learnable queries to extract visual information and harnesses LLMs to generate language informed by the visual features. In contrast to prior works, our approach originates from the issues inherent in MLLMs. We specifically address the problems that MLLMs face in the physical world and improve the LLaVA-v1.5 model from two perspectives: the mapping of digital carriers and the application of float32.

Digital Information In Multimodal LLMs. Recent studies have identified prevalent issues with digital inaccuracies within LLMs (Jiang et al. 2020; Charton 2021; Zhu et al. 2024; Anil et al. 2022; Grosse et al. 2023). Similarly, in MLLMs, the problem of imprecise digital recognition frequently arises (Lu et al. 2023; Driess et al. 2023; Zhang et al. 2024). For instance, direct number recognition is often inaccurate, a challenge that can be partially mitigated by OCR technologies (Ye et al. 2023; Liu et al. 2024c,a; Hu et al. 2024). Our work takes a step further by introducing the issue of physical quantification. This is not merely about reading numbers directly, but rather it involves interpreting physical information within images and translating meaningful real-world results into digital values. This demands not only the model's precise generation of digits but also a understanding of the real physical world.

Methodology

In this section, we present DigitalLLaVA, an advanced model for the comprehensive and precise processing of physical digital cognition. For a given multimodal input sequence $s = (I_1, I_2, ..., I_m, t_1, t_2, ..., t_n)$, with image tokens $(I_1, I_2, ..., I_m)$ and word tokens $(t_1, t_2, ..., t_n)$, the MLLM models the conditional probability of the word sequence given the image context as $P(s) = \prod_{i=1}^{n} P(t_i|t_{<i}, I_{\leq m})$. The next word token's probability, conditioned on both the image features and the preceding text, is computed using $P(t_i|t_{<i}, I_{\leq m}) = softmax(W_\nu \cdot h_{i-1})$ where h_{i-1} represents the combined contextual information from both modalities up to the previous token, and W_ν is the model's word embedding matrix. This approach allows MLLMs to generate text that is coherent and contextually aligned with the provided image content.

For our method, during the inference stage, the input is an image and a corresponding question, and the output is the answer. For questions related to the physical world, after generating the hidden states, the model sends this feature to two different heads. The LLM head is the inherent head of LLaVA, which in our model is responsible for generating tokens for the non-digital part and routing the digital part to simulate float32. Specifically, as illustrated in Fig 2, for the answer "the reading is 0.362 cm," the LLM head generates s' = ("the", "reading", "is", "[Float]", "cm"). The [Float] token is then received by all three float heads, which uses hidden states to generate the corresponding float32 value, converting it to the final number: 0.362.

Digital Carrier Mapping

MLLMs often struggle to comprehend physical digits and accurately interpret physical information in images. This deficiency is primarily due to their limited exposure to images of digital carriers and the scarcity of descriptive data about physical quantities within those images.

To address this issue, we propose a digital carrier mapping mechanism that merges visual and word embeddings into a shared vector space. This mapping leverages the model's linguistic proficiency to interpret visual data, enabling a more natural and comprehensive understanding of digital carriers. The alignment process involves the following steps:

Let $E_v \in R^{d_v \times V}$ represent the visual embedding matrix and $E_w \in R^{d_w \times W}$ denote the word embedding matrix, where d_v and d_w are the dimensions of the visual and word embeddings, respectively. V and W are the sizes of the visual and word vocabularies. Instead of a direct matrix transformation, the alignment between visual and word embeddings is achieved through a contrastive learning process. Specifically, images are paired with their corresponding textual descriptions, which are then encoded into text embeddings using the CLIP text encoder. The CLIP image encoder is fine-tuned with a contrastive loss function to align the image embeddings with these text embeddings. The contrastive loss function is essential for the embedding alignment process. Let E_{v_i} and E_{w_j} denote the i-th visual embedding and the j-th word embedding, respectively. The contrastive loss function can be defined as follows:

$$sim_{i,j} = \frac{E_{v_i} \cdot E_{w_j}}{|E_{v_i}| \cdot |E_{w_j}|} \tag{1}$$

$$L_{con} = -\frac{1}{2}\left(\sum_{i=1}^{N} \log \frac{e^{sim_{i,i}}}{\sum_{j=1}^{N} e^{sim_{i,j}}} + \sum_{j=1}^{N} \log \frac{e^{sim_{j,j}}}{\sum_{i=1}^{N} e^{sim_{i,j}}}\right) \tag{2}$$

where $sim_{i,j}$ denote the cosine similarity between the i-th visual embedding and the j-th word embedding. These similarity values are converted into a probability distribution using the softmax function, which computes the loss function across both image and text dimensions. This embeddings mapping process ensures that the visual and word representations are coherently aligned, thereby enhancing the model's digital cognition capability.

32-bit Float-point Simulation

Float32 Operation Conventional MLLMs rely on a sequential digital prediction approach, which can lead to errors in predicting a single digit and consequently make the entire number incorrect. While some methods transform numbers into specific tokens, direct regression for predicting numbers results in a very large search space. This issue is particularly problematic in tasks involving physical digits, where precise cognition and accuracy is crucial.

To address this, we enhance digital cognitive ability by simulating Float32, also known as single-precision floating-point format, which uses 32 bits (4 bytes) of memory. This format includes one sign bit, an 8-bit exponent, and a 23-bit fraction. By using this approach, we only need to predict a 32-bit binary classification, significantly reducing the search space. Float32 has a representational range of approximately -1.18×10^{-38} to 3.4×10^{38}, with a precision of about 7 decimal digits. Moreover, for physical measurement tasks, we chose not to use float16 or bfloat16 due to their limitations. Float16 has a limited range (maximum 65504), while bfloat16 offers lower precision (2 decimal digits). By opting for Float32, we ensure a balance between range and precision, making it more suitable for digital cognition.

The calculation of a float32 number follows the equation (IEEE-Std 2019):

$$F_{num} = (-1)^{sign} \times 2^{exponent-127} \times \left(1 + \frac{fraction}{2^{23}}\right) \tag{3}$$

For instance, in Fig.2 on the right, the number 0.362 can be split into three parts: the signal, which is 0 indicating a positive sign, the exponent, which is 125 corresponding to 2^2, and the fraction part, which consists of 23 bits of binary representing 0.477 in decimal.

To simulate the operation of float32, Our model incorporates three specialized float heads, each designed to simulate distinct components of floating-point numbers, enhancing the granularity of digital representation. The signal head, which operates as a binary classifier, is pivotal in determining the sign of a floating-point number. It is implemented through a two-layer multilayer perceptron (MLP), chosen for its simplicity and effectiveness in predicting the single binary outcome. The exponent head and fraction head, on

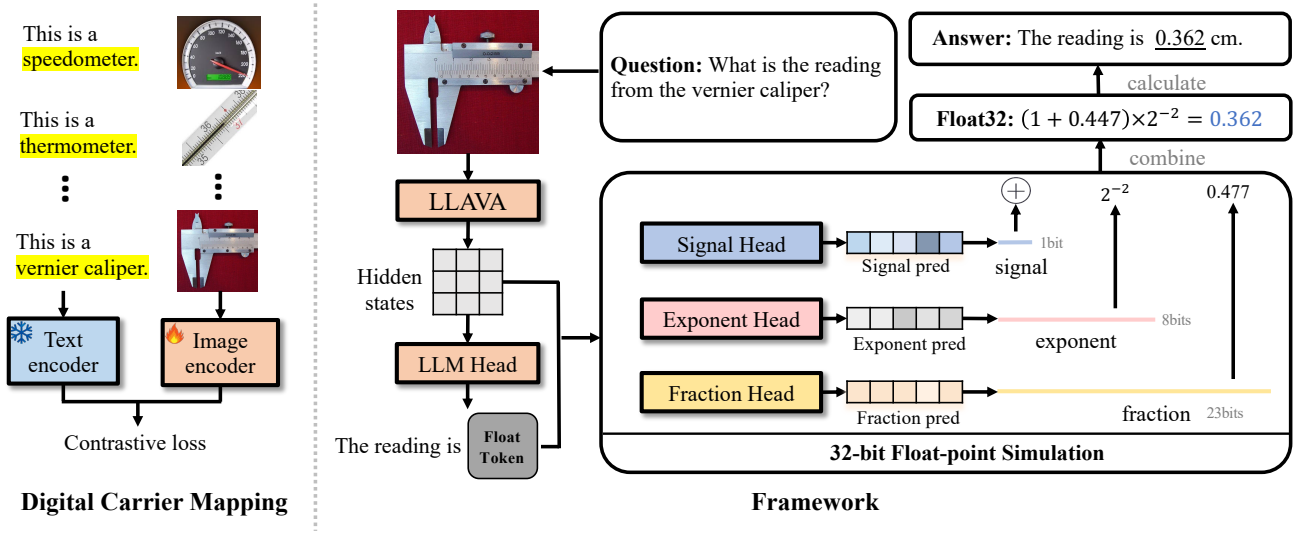


Figure 2: Overview of our DigitalLLaVA. We employ a two-stage coarse-to-fine framework. Initially, in the first stage, we fine-tune the image encoder through contrastive learning, enabling the MLLM to better understand digital carriers. In the second stage, we utilize a 32-bit floating-point simulation to precisely generate digits, i.e., accurate to 0.001.

the other hand, leverage the power of self-attention mechanisms to simulate the exponent and fraction components, capturing the scale of digits with an approach that considers the contextual relationships within the component.

Loss Function As the signal part only requires predicting a single value, different loss functions are employed: cross-entropy loss (CE loss) for the signal part and mean squared error loss (MSE loss) for the exponent and fraction heads. Hence, the float loss for the float simulation is given by:

$$\mathcal{L}_{float} = \begin{cases} -\sum_{i=1}^{N} CE(y_i, \hat{y}_i) \\ \frac{1}{N}\sum_{i=1}^{N} MSE(y_{exp,i}, \hat{y}_{exp,i}) \\ \frac{1}{N}\sum_{i=1}^{N} MSE(y_{frac,i}, \hat{y}_{frac,i}) \end{cases} \tag{4}$$

where $CE(y, \hat{y})$ denotes the cross-entropy loss function. $MSE(y_{exp}, \hat{y}_{exp})$ represents the mean squared error for the exponent part and fraction part. The total loss is:

$$\mathcal{L}_{total} = \sum_{i=1}^{N} -\log P(t_i'|t_{<i}, I_{\leq m}) \cdot 1_{t_i' \neq [F]} + \mathcal{L}_{float} \cdot 1_{t_i'=[F]} \tag{5}$$

The total loss $\mathcal{L}_{total}$ is composed of two components. First, for each token prediction position i, we calculate the negative log-likelihood of the model's prediction t_i' given the input sequence $t_{<i}, I_{\leq m}$. This term is multiplied by an indicator function $1_{t_i' \neq [F]}$ to ensure that only standard tokens contribute to the standard cross-entropy loss. Second, we add the float loss $\mathcal{L}_{float}$, which represents the additional loss specifically for float tokens. This float loss term is multiplied by an indicator function $1_{t_i'=[F]}$ to ensure that only float tokens contribute to this extra loss. In summary, the total loss accounts for both standard token predictions and the special handling of float token to update float heads.

Experiments and Results

In this section, we conducted several experiments to validate the effectiveness of our method: a digital experiment to assess performance in specialized domains, a VQA experiment for general performance, a precision experiment and a carrier recognition experiment to demonstrate the efficacy of 32-bit floating-point simulation and digit carrier remapping. The results demonstrate that our method not only efficiently predicts numbers but also maintains the general capabilities of MLLMs. This proves to be an effective approach for enhancing MLLM performance in specialized domains.

Digital Results

In this section, We conducted experiments on three tasks to thoroughly assess our model's digital cognitative capabilities: a question answering task, a more complex comprehension and reasoning task, and a multi-turn conversation task. These tasks were chosen to evaluate the model's ability to recognize digital details, extract relevant information, and engage in complex interactions.

The first task, question-answering, involves a straightforward format where the model is presented with an image and a direct question about its content, assessing the model's ability to extract and comprehend specific information from the visual input. The second task, comprehension and reasoning, goes a step further by requiring the model not only to recognize elements within an image but also to perform a certain level of reasoning or inference based on the visual information. This task evaluates the model's ability to understand the scene and apply logical reasoning to arrive at a conclusion. The third task involves multi-turn conversations, providing an interactive scenario where the model engages in a dialogue with a human participant. The conversation progresses through a series of exchanges, with each subsequent question building on the model's previous responses.

Methods	Question Answering			Comprehension and Reasoning			Multi-turn Conversations		
	Gauge	Stopwatch	A-V	Gauge	Stopwatch	A-V	Gauge	Stopwatch	A-V
Finetune	3.5	12.2	6.0	19.8	4.1	11.0	8.6	1.0	14.2
Lora	5.0	8.7	11.2	1.0	17.6	12.1	13.6	9.2	3.0
GPT4V (OpenAI 2023)	6.1	3.7	9.0	5.2	19.4	6.8	4.7	8.9	6.3
Gemini Pro (Team et al. 2023)	7.3	2.1	19.8	11.0	18.9	4.3	19.2	5.4	8.6
LLaVA (Liu et al. 2024b)	14.0	4.6	18.8	5.1	9.9	11.2	6.1	8.7	13.4
OpenFlamingo-v2 (Awadalla et al. 2023)	6.0	2.3	8.4	4.1	7.3	4.2	3.2	5.9	13.2
MiniGPT-4-v2 (Chen et al. 2023)	12.0	6.3	13.6	7.2	5.6	14.2	7.5	6.3	16.4
P10 (Charton 2021)	54.7	31.2	26.5	33.8	40.4	49.1	44.0	51.2	46.9
P1000 (Charton 2021)	44.3	46.0	33.5	35.2	37.1	45.5	48.3	46.0	54.3
UReader (Ye et al. 2023)	3.0	10.5	1.7	3.7	15.1	8.3	5.2	11.4	7.7
XVAL (Golkar et al. 2023)	26.8	45.5	38.6	29.0	37.8	34.5	33.1	25.2	36.0
Ours	**83.0**	**81.1**	**80.9**	**81.8**	**86.9**	**80.5**	**83.8**	**80.5**	**80.2**

Table 1: Results on three tasks using digital datasets. We employ the accuracy metric to measure the model's performance, as physical measurement problems typically have a unique correct answer. Our model was trained on all datasets and tasks in a single training process.

datasets	Time	Speed	Vernier	T-M	Cylinder	Protractor	Spring	Gauge	Stopwatch	A-V
train set	7200	2600	3000	7000	2000	1800	5000	4000	6000	3000
test set	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000

Table 2: The number of text-image pairs in the training and test sets of each dataset.

This task assesses the model's capacity to retain context throughout the conversation and to generate responses that are coherent, contextually appropriate, and precise.

Datasets and Metrics. In our experiments, we evaluate our approach on ten distinct datasets, each representing a physical world task requiring fine-grained cognition and conceptual understanding. These datasets are Time Clock, Speedometer, Vernier Caliper, Thermometer (tm), Graduated Cylinder, Protractor, Spring Scale, Pressure Gauge, Stopwatch, and Ammeter-Voltmeter (A-V). Table 2 displays the number of text-image pairs in the training and test sets for all datasets. all images are of a resolution of 256 by 256 pixels. The question formats are exemplified by queries such as 'What is the reading from the xxx?'. Our model was trained on all datasets and tasks in a single training process. We used accuracy as the metric to measure the model's performance, as physical measurement tasks typically have a unique correct answer.

Comparison methods. We utilized LLaVA-v1.5-7b as our base model and conducted a comparative analysis with several baselines, including Finetune, Lora (Hu et al. 2021). "Finetune" refers to conducting instruction fine-tuning on LLaVA-v1.5-7b directly using the dataset, while "LoRA" involves applying LoRA fine-tuning to LLaVA-v1.5-7b. Additionally, we explored various token encoding schemes P10, P1000 (Charton 2021). We also evaluated against state-of-the-art MLLMs like GPT4V (OpenAI 2023), Gemini Pro (Team et al. 2023), LLaVA (Liu et al. 2024b), OpenFlamingo-v2 (Awadalla et al. 2023), MiniGPT-4-v2 (Chen et al. 2023). UReader (Ye et al. 2023) model is known for its strong performance on OCR digits.

Result Analysis. Table 1 shows the results of the proposed framework and SOTA methods on Gauge, stopwatch and Ammeter-Voltmeter datasets. In all three tasks, our method demonstrated significant improvements. Notably, on the Gauge and Stopwatch datasets, our approach outperformed the closest method, P10, by approximately 36.7% and 31.6% on average across three tasks. This indicates that our method possesses robustness and accuracy in fine-grained recognition tasks. Furthermore, our method has consistently achieved performance levels exceeding 80% across all datasets, which attests to its remarkable efficiency and versatility. These improvements underscore the effectiveness of our method and highlight its ability to deliver precise digits in simple question scenarios.

VQA Results

We also conducted experiments on standard Visual Question Answering (VQA) tasks to validate that DigitalLLaVA is a versatile method for extending capabilities. VQA task requires a model to understand and interpret the visual content of an image, comprehend the semantics of the question, and generate a coherent and contextually appropriate response. VQA challenges combine image recognition, object detection, and language understanding, making it a complex and comprehensive test of a model's ability to integrate and process multimodal information.

Datasets and Metrics. To evaluate the general performance of DigitalLLaVA, we selected six datasets commonly used by most MMLMs, covering capabilities in science (Lu et al. 2022), mathematical reasoning (Lu et al. 2024), real-world cognition (X.AI 2023), Object Hallucination (Yifan Li and Wen 2023) and comprehensive evaluation (Fu et al. 2024; Liu et al. 2023b). All datasets consist of multiple-choice questions or have a single correct answer. Therefore, accuracy is the metric to measure the model's performance.

Methods	MathVista	ScienceQA	POPE	RealWorldQA	MME	MMBench
OpenFlamingo-v2 (Awadalla et al. 2023)	18.6	44.8	52.6	35.2	607.2	7.9
MiniGPT-4-v2 (Chen et al. 2023)	23.1	54.7	60.0	30.7	968.4	3.2
LLaVA (Liu et al. 2024b)	25.6	69.2	**86.1**	54.8	**1808.4**	59.1
DigitalLLaVA	**32.2**	**70.8**	85.8	**58.3**	1757.7	**59.3**

Table 3: Comparison of general performance across different models on various datasets. The bold numbers indicate the best performance in each dataset. MME score is the total score across 14 sub-tasks, with each sub-task having a maximum score of 200, while the other datasets have a maximum score of 100.

Comparison methods. Since our method is fine-tuned on LLaVA-v1.5-7b (Liu et al. 2024b), we primarily compare our results with LLaVA. OpenFlamingo-v2 and MiniGPT-4-v2 serve as baseline models are included for reference.

Result Analysis. As shown in Table 3, our method, DigitalLLaVA, achieved results slightly better than LLaVA-v1.5-7b. It outperformed LLaVA on four datasets and was slightly lower on the other two datasets.This demonstrates that our approach maintains the general capabilities of MLLMs and is an efficient method for enhancing domain-specific abilities. Notably, our method improved performance on MathVista by 6.6%, indicating that 32-bit floating-point simulation is beneficial for MMLM digital reasoning tasks.

Precision and Accuracy Results

The critical importance of high-precision predictions, where accuracy must meet minimal error margins, is widely acknowledged in both industrial and medical applications. Compared to previous methods, we use float32 encoding for digital values, meaning the model only needs to make 32 binary (0/1) predictions, which is much simpler than predicting the actual numbers directly. We demonstrate how our framework effectively covers various digital measurement scenarios. Compared to previous methods, our model shows significant improvements in both precision and accuracy.

Model	MAE ↓	RMSE↓	Acc↑	Acc(0.1)↑
LLaVA	0.58	0.64	39.1	39.3
Carrier	0.67	0.61	50.0	51.6
MSE	0.53	0.52	43.3	43.9
P10	0.36	0.42	49.8	50.3
P1000	0.34	0.40	49.5	50.2
Float32	0.25	0.21	70.6	76.0
Ours	**0.15**	**0.07**	**84.9**	**90.1**

Table 4: Performance across different models. MAE and RMSE are normalized results, and Acc (0.01) considers predictions within a tolerance of 0.01 as correct. The ↓ indicates higher values are better, ↑ indicates lower values are better.

Datasets and Metrics. We still use the previously mentioned ten digital carrier datasets. To better evaluate the accuracy and precision of various models, we employ Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). MAE measures the average absolute difference between the predicted and actual values. Compared to accuracy, if the discrepancy between the results and the ground truth is small, the MAE value will also be low. RMSE assesses the square root of the average of squared differences between the predicted and actual values. Compared to MAE, RMSE is more sensitive, as the errors are squared before averaging. This sensitivity allows RMSE to better indicate whether the model truly understands the physical carrier or is merely guessing. Additionally, we use accuracy within tolerance to measure the accuracy within specified error tolerance ranges, such as ±0.01 or ±0.1. By comparing this with overall accuracy, we can evaluate the model's performance in terms of precision. In this experiment, we utilize normalized MAE and RMSE, both with a maximum value of 1. Accuracy is measured with a maximum value of 100 and within a tolerance range of ±0.1 (Acc0.1).

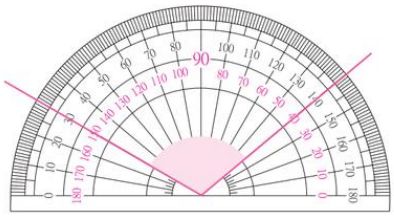
Comparison methods. Since our method is fine-tuned on LLaVA-v1.5-7b, we primarily compare our results with LLaVA. The term "Carrier" indicates the use of the digital carrier mapping method, where digits are predicted sequentially. In contrast, "MSE" adds one extra digital token and employs regression to directly predict the numbers. "Float32" refers to our fine-tuning on LLaVA using only 32-bit float-point simulation method. 10 positional encoding (P10), numbers are represented as sequences of five tokens: one sign token (+ or -), three digits (from 0 to 9) for the fraction, and a symbolic token (ranging from E-100 to E+100) for the exponent, resulting in an additional 210 tokens. For example, the number 3.14 is represented as 314.10^{-2} and encoded as [+, 3, 1, 4, E-2]. 1000 positional encoding (P1000) offers a more compact representation. The fraction is encoded as a single token (ranging from 0 to 999), and a number is represented as a triplet (sign, fraction, exponent), leading to an additional 1100 tokens.

Result Analysis. As shown in Table 4, our method significantly outperforms all other models across all metrics. Specifically, our method achieves the lowest MAE and RMSE values of 0.15 and 0.07, respectively, indicating superior precision and robustness in predictions. During the experiment, the MSE method performed well on the training set but less effectively on the test set, indicating potential overfitting. This may be because MSE directly predicts numbers without understanding their intrinsic meaning, highlighting the importance of numerical encoding. The accuracy metrics also demonstrate the effectiveness of the Float32 method, showing remarkable precision performance. Compared to the standard accuracy, our method improved accuracy by approximately 5.2% at Acc (0.01), while other methods showed only minor improvements about 1%.

Question: What is the reading from the stopwatch?

Ours: Stopwatch reads 60 seconds.

GT: Stopwatch reads 53 seconds.



Question: What is the reading from the protractor?

Ours: The reading is 140 degrees.

GT: The reading is 110 degrees.

Question: What is the reading from the thermometer?

Ours: The reading is 36.5 degrees.

GT: The temperature is 36.6 degrees.

Figure 3: Three examples where our model outputs incorrect readings: stopwatch, protractor, and thermometer reading.

Carrier Recognition Results

Model	Precision	Recall	F1-score
LLaVA	82.6	83.7	83.3
DigitalLLaVA	97.6	96.4	97.0

Table 5: Comparison of carrier recognition performance between LLaVA and DigitalLLaVA based on precision, recall, and F1-score. The scores are averaged across all datasets.

To evaluate the effectiveness of the digital carrier mapping method, we conducted a physical carrier recognition experiment. In a zero-shot setting, we tested the fine-tuned model's ability to recognize physical carriers by asking, "What is this object?" We measured the average precision, recall, and F1 score across all datasets. As shown in Table 5, Our method outperforms the baseline LLaVA, not only demonstrating the necessity of carrier remapping but also showing that 32-bit floating-point simulation does not lead to overfitting and remains effective in object recognition.

Failure Case Analysis

In this failure case analysis, we examine scenarios where our model encounters difficulties in generating accurate answers. By dissecting the reasons contributing to these failures, we aim to better understand the limitations of our current approach and gain insights into areas for model improvement. As shown in Fig. 3, we present three failure cases that shows errors in different aspects.

In the first case, the model misinterpreted a stopwatch reading. It mistakenly considered the 0 seconds displayed on the inner circle as the only relevant information. This led to an erroneous output of 60 seconds instead of the correct 53 seconds. The second case involves a protractor misreading. The model failed to consider the angle formed by both the left and right lines. This led to an incorrect output of 140 degrees instead of the ground truth of 110 degrees. The third case involves a thermometer reading scenario. The model's output of 36.5 degrees deviated slightly from the ground truth of 36.6 degrees. This discrepancy may be due to model inaccuracies exacerbated by image compression, resulting in an unclear image and scale. These failure cases indicate that our model still has some deficiencies for physical comprehension. It may require more precise fine-tuning of img encoder or the use of other vision pre-trained models to improve fine-grained recognition.

Conclusion

In this paper, we enhance Multimodal Large Language Models (MLLMs) with the capability for physical digital cognition. Specifically, since humans must identify the carriers before reading digits, and noting that previous MLLMs did not have a rich representation of the corresponding carriers, our DigitalLLaVA introduce a **digit carrier mapping** to improve the model's understanding of these carriers. Furthermore, to enhance the model's precision in reading digits from the carriers, we propose a **32-bit floating-point simulation** to represent digits. Our DigitalLLaVA demonstrates robust performance across multiple datasets, effectively addressing the challenge of interpreting physical objects. Codes are accessible in supplementary material.

Broader Impact. MLLMs have gained significant popularity in recent years, often seen as a key pathway towards achieving AGI (Artificial General Intelligence). This paper identifies a critical gap in MLLMs' capabilities for physical digital cognition, a crucial aspect due to the physical world's integral role in human experience. By addressing this gap, we aim to contribute to the advancement of MLLMs within specialized domains and hope to inspire further development in these areas.

Acknowledgments

This work was supported in part by the National Key R&D Program of China (No. 2022ZD0118201), the Shenzhen Medical Research Funds in China (No. B2302037), the Natural Science Foundation of China (No. 61972217, 32071459, 62176249, 62006133, 62271465), the AI for Science (AI4S)-Preferred Program, Peking University Shenzhen Graduate School, China, the National Natural Science Foundation of China (NSFC) under Grant No. 62376292, No. 6240071660, U24B600013, U21A20470, and Guangdong Provincial General Fund No. 2024A1515010208.

References

Alayrac, J.-B.; Donahue, J.; Luc, P.; Miech, A.; Barr, I.; Hasson, Y.; Lenc, K.; Mensch, A.; Millican, K.; Reynolds, M.; et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35: 23716–23736.

Anil, C.; Wu, Y.; Andreassen, A.; Lewkowycz, A.; Misra, V.; Ramasesh, V.; Slone, A.; Gur-Ari, G.; Dyer, E.; and Neyshabur, B. 2022. Exploring length generalization in large language models. *Advances in Neural Information Processing Systems*, 35: 38546–38556.

Awadalla, A.; Gao, I.; Gardner, J.; Hessel, J.; Hanafy, Y.; Zhu, W.; Marathe, K.; Bitton, Y.; Gadre, S.; Sagawa, S.; et al. 2023. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Charton, F. 2021. Linear algebra with transformers. *arXiv preprint arXiv:2112.01898*.

Chen, J.; Zhu, D.; Shen, X.; Li, X.; Liu, Z.; Zhang, P.; Krishnamoorthi, R.; Chandra, V.; Xiong, Y.; and Elhoseiny, M. 2023. Minigpt-v2: large language model as a unified interface for vision-language multi-task learning. *arXiv preprint arXiv:2310.09478*.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; Stoica, I.; and Xing, E. P. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Driess, D.; Xia, F.; Sajjadi, M. S.; Lynch, C.; Chowdhery, A.; Ichter, B.; Wahid, A.; Tompson, J.; Vuong, Q.; Yu, T.; et al. 2023. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*.

Fu, C.; Chen, P.; Shen, Y.; Qin, Y.; Zhang, M.; Lin, X.; Yang, J.; Zheng, X.; Li, K.; Sun, X.; Wu, Y.; and Ji, R. 2024. MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models. arXiv:2306.13394.

Gao, P.; Han, J.; Zhang, R.; Lin, Z.; Geng, S.; Zhou, A.; Zhang, W.; Lu, P.; He, C.; Yue, X.; et al. 2023. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*.

Golkar, S.; Pettee, M.; Eickenberg, M.; Bietti, A.; Cranmer, M.; Krawezik, G.; Lanusse, F.; McCabe, M.; Ohana, R.; Parker, L.; et al. 2023. xval: A continuous number encoding for large language models. *arXiv preprint arXiv:2310.02989*.

Grosse, R.; Bae, J.; Anil, C.; Elhage, N.; Tamkin, A.; Tajdini, A.; Steiner, B.; Li, D.; Durmus, E.; Perez, E.; et al. 2023. Studying large language model generalization with influence functions. *arXiv preprint arXiv:2308.03296*.

Hao, Y.; Song, H.; Dong, L.; Huang, S.; Chi, Z.; Wang, W.; Ma, S.; and Wei, F. 2022. Language models are general-purpose interfaces. *arXiv preprint arXiv:2206.06336*.

Hu, A.; Xu, H.; Ye, J.; Yan, M.; Zhang, L.; Zhang, B.; Li, C.; Zhang, J.; Jin, Q.; Huang, F.; et al. 2024. mplug-docowl 1.5: Unified structure learning for ocr-free document understanding. *arXiv preprint arXiv:2403.12895*.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Huang, S.; Dong, L.; Wang, W.; Hao, Y.; Singhal, S.; Ma, S.; Lv, T.; Cui, L.; Mohammed, O. K.; Patra, B.; et al. 2024. Language is not all you need: Aligning perception with language models. *Advances in Neural Information Processing Systems*, 36.

IEEE-Std. 2019. IEEE Standard for Floating-Point Arithmetic. *IEEE Std 754-2019 (Revision of IEEE 754-2008)*, 1–84.

Jiang, C.; Nian, Z.; Guo, K.; Chu, S.; Zhao, Y.; Shen, L.; and Tu, K. 2020. Learning numeral embedding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2586–2599.

Li, C.; Wong, C.; Zhang, S.; Usuyama, N.; Liu, H.; Yang, J.; Naumann, T.; Poon, H.; and Gao, J. 2024. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *Advances in Neural Information Processing Systems*, 36.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023a. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, 19730–19742. PMLR.

Li, J.; Li, D.; Xiong, C.; and Hoi, S. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, 12888–12900. PMLR.

Li, Y.; Zhang, Y.; Wang, C.; Zhong, Z.; Chen, Y.; Chu, R.; Liu, S.; and Jia, J. 2023b. Mini-Gemini: Mining the Potential of Multi-modality Vision Language Models. *arXiv:2403.18814*.

Lin, B.; Zhu, B.; Ye, Y.; Ning, M.; Jin, P.; and Yuan, L. 2023. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*.

Liu, H.; Li, C.; Li, Y.; and Lee, Y. J. 2023a. Improved Baselines with Visual Instruction Tuning.

Liu, H.; Li, C.; Li, Y.; Li, B.; Zhang, Y.; Shen, S.; and Lee, Y. J. 2024a. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2024b. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Liu, Y.; Duan, H.; Zhang, Y.; Li, B.; Zhang, S.; Zhao, W.; Yuan, Y.; Wang, J.; He, C.; Liu, Z.; et al. 2023b. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*.

Liu, Y.; Yang, B.; Liu, Q.; Li, Z.; Ma, Z.; Zhang, S.; and Bai, X. 2024c. Textmonkey: An ocr-free large multimodal model for understanding document. *arXiv preprint arXiv:2403.04473*.

Lu, P.; Bansal, H.; Xia, T.; Liu, J.; Li, C.; Hajishirzi, H.; Cheng, H.; Chang, K.-W.; Galley, M.; and Gao, J. 2023. Mathvista: Evaluating math reasoning in visual contexts with gpt-4v, bard, and other large multimodal models. *arXiv e-prints*, arXiv–2310.

Lu, P.; Bansal, H.; Xia, T.; Liu, J.; Li, C.; Hajishirzi, H.; Cheng, H.; Chang, K.-W.; Galley, M.; and Gao, J. 2024. MathVista: Evaluating Mathematical Reasoning of Foundation Models in Visual Contexts. In *International Conference on Learning Representations (ICLR)*.

Lu, P.; Mishra, S.; Xia, T.; Qiu, L.; Chang, K.-W.; Zhu, S.-C.; Tafjord, O.; Clark, P.; and Kalyan, A. 2022. Learn to Explain: Multimodal Reasoning via Thought Chains for Science Question Answering. In *The 36th Conference on Neural Information Processing Systems (NeurIPS)*.

Luo, G.; Zhou, Y.; Ren, T.; Chen, S.; Sun, X.; and Ji, R. 2024. Cheap and quick: Efficient vision-language instruction tuning for large language models. *Advances in Neural Information Processing Systems*, 36.

OpenAI, O. 2023. GPT-4 Technical Report.

Pi, R.; Gao, J.; Diao, S.; Pan, R.; Dong, H.; Zhang, J.; Yao, L.; Han, J.; Xu, H.; and Zhang, L. K. T. 2023. Detgpt: Detect what you need via reasoning. *arXiv preprint arXiv:2305.14167*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Sun, Q.; Fang, Y.; Wu, L.; Wang, X.; and Cao, Y. 2023. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*.

Team, G.; Anil, R.; Borgeaud, S.; Wu, Y.; Alayrac, J.-B.; Yu, J.; Soricut, R.; Schalkwyk, J.; Dai, A. M.; Hauth, A.; et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. Llama: Open and efficient foundation language models (2023). *arXiv preprint arXiv:2302.13971*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

X.AI. 2023. Grok 1.5v. https://x.ai/blog/grok-1.5v.

Ye, J.; Hu, A.; Xu, H.; Ye, Q.; Yan, M.; Xu, G.; Li, C.; Tian, J.; Qian, Q.; Zhang, J.; et al. 2023. Ureader: Universal ocr-free visually-situated language understanding with multimodal large language model. *arXiv preprint arXiv:2310.05126*.

Yifan Li, K. Z. J. W. W. X. Z., Yifan Du; and Wen, J.-R. 2023. Evaluating Object Hallucination in Large Vision-Language Models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Zhang, R.; Jiang, D.; Zhang, Y.; Lin, H.; Guo, Z.; Qiu, P.; Zhou, A.; Lu, P.; Chang, K.-W.; Gao, P.; et al. 2024. MathVerse: Does Your Multi-modal LLM Truly See the Diagrams in Visual Math Problems? *arXiv preprint arXiv:2403.14624*.

Zhu, X.; Li, J.; Liu, Y.; Ma, C.; and Wang, W. 2024. Improving Small Language Models' Mathematical Reasoning via Mix Thoughts Distillation. *arXiv preprint arXiv:2401.11864*.