

# LEOPARD<sup>🐆</sup>: A VISION LANGUAGE MODEL FOR TEXT-RICH MULTI-IMAGE TASKS

**Anonymous authors**

Paper under double-blind review

## ABSTRACT

Text-rich images, where text serves as the central visual element guiding the overall understanding, are prevalent in real-world applications, such as presentation slides, scanned documents, and webpage snapshots. Tasks involving multiple text-rich images are especially challenging, as they require not only understanding the content of individual images but reasoning about inter-relationships and logical flows across multiple visual inputs. Despite the importance of these scenarios, current multimodal large language models (MLLMs) struggle to handle such tasks due to two key challenges: (1) the scarcity of high-quality instruction tuning datasets for text-rich multi-image scenarios, and (2) the difficulty in balancing image resolution with visual feature sequence length. Low-resolution encoding impairs the recognition of embedded text, while high-resolution encoding quickly exceeds the model’s maximum sequence length under multi-image settings. To address these challenges, we propose LEOPARD, a MLLM designed specifically for handling vision-language tasks involving multiple text-rich images. First, we curated about one million high-quality multimodal instruction-tuning data, tailored to text-rich, multi-image scenarios. Second, we developed an adaptive high-resolution multi-image encoding module to dynamically optimize the allocation of visual sequence length based on the original aspect ratios and resolutions of the input images. Experiments across a wide range of benchmarks demonstrate our model’s superior capabilities in text-rich, multi-image evaluations and competitive performance in general domain evaluations. We are committed to open-source models and will release all collected data, code, and checkpoints to the community<sup>1</sup>.

## 1 INTRODUCTION

Multimodal large language models (MLLMs) have revolutionized vision-language tasks, driving advancements in a variety of areas such as image captioning and object detection (Wang et al., 2023b; Zhang et al., 2024; Zang et al., 2024). These improvements extend to applications involving *text-rich images* where text serves as the primary visual element guiding image comprehension, such as visual document understanding (Mathew et al., 2021) and scene text recognition (Singh et al., 2019b). Traditional OCR-based pipelines in these text-rich visual scenarios are being replaced by end-to-end approaches that directly encode intertwined multimodal inputs (Wu et al., 2023b; Zhang et al., 2023; Tang et al., 2024), leading to improved efficiency and accuracy in handling text-rich images.

Despite these advancements, the majority of existing open-source MLLMs, like LLaVAR (Zhang et al., 2023) and mPlug-DocOwl-1.5 (Hu et al., 2024a), have primarily focused on optimizing performance for text-rich *single-image tasks*. This focus inherently limits their applicability in many real-world scenarios, where tasks often involve *multiple inter-connected images*. For instance, multi-page visual document understanding requires integrating information spread across different pages to capture the logical flow across the whole document (Tito et al., 2022; Landeghem et al., 2023). To understand presentation slides, grasping the overarching narrative necessitates understanding multiple slides with unique but interrelated content (Tanaka et al., 2023). These vision-language tasks on multiple text-rich images require advanced capabilities that go beyond merely recognizing text and visuals within a single image; they involve understanding and reasoning about relationships and

<sup>1</sup><https://anonymous.4open.science/r/Leopard-8E26/>.

054  
055  
056  
057  
058  
059  
060  
061  
062  
063  
064  
065  
066  
067  
068  
069  
070  
071  
072  
073  
074  
075  
076  
077  
078  
079  
080  
081  
082  
083  
084  
085  
086  
087  
088  
089  
090  
091  
092  
093  
094  
095  
096  
097  
098  
099  
100  
101  
102  
103  
104  
105  
106  
107

Figure 1: Left: A demonstration of a text-rich multi-image task. Models need to reason about the textual content across multiple images to answer the question correctly. LEOPARD successfully generates the right answer while baselines fail. Right: Evaluation results of LEOPARD and three baselines. Our model surpasses its counterparts across text-rich multi-image benchmarks by a large margin, maintaining comparable performance on single and general evaluations.

logical flows across multiple visual inputs. While some models – such as OpenFlamingo (Awadalla et al., 2023), VILA (Lin et al., 2023), Ide cs2 (Laureçon et al., 2024b) – have made strides toward supporting multi-image inputs, they mainly focus on scenarios with natural images but fall short in understanding sequences of text-rich images with interrelated textual and visual information. We plot the performance of representatives of the aforementioned models in Figure 1. Upon examining their training data and model architecture, we identified two primary limitations within these models.

First, there is a scarcity of high-quality instruction tuning datasets on text-rich multi-image scenarios. Existing visual instruction tuning datasets for text-rich images are predominantly based on single-image inputs (Kae et al., 2018; Singh et al., 2019b; Masry et al., 2022; Tang et al., 2024), which limits the model ability to generalize and reason across multiple images. Second, in text-rich multi-image scenarios, there is a challenge of balancing image resolution and sequence length limitations. Many general-domain MLLMs adopt the low-resolution settings of pre-trained visual encoders (Lin et al., 2023; Jiang et al., 2024). However, for text-rich images, such as scientific reports, recognizing text content becomes difficult at low resolutions. While some approaches overcome this in single-image settings by splitting the original image to preserve high-resolution details (Liu et al., 2024a; Hu et al., 2024a), this approach is less effective when applied to multiple images, as it quickly exceeds model’s maximum sequence length. Moreover, compressing such long-sequence representations into shorter ones leads to significant information loss, thereby degrading model performance (Awadalla et al., 2023; Laureçon et al., 2023). Thus, a critical balance must be struck between maintaining sufficient visual detail and keeping sequence lengths manageable.

In this paper, we introduce a novel multimodal large language model, named LEOPARD<sup>2</sup>. LEOPARD is specifically designed to handle complex, text-rich, multi-image tasks. To train LEOPARD, we first curated about one million high-quality multimodal instruction-tuning data, tailored to the text-rich, multi-image scenarios. This dataset spans three key domains that are commonly encountered in real-world scenarios: (1) multi-page documents, (2) multi-charts and multi-tables, (3) webpage trajectories. These scenarios capture the increasing complexity and multimodal nature of modern digital information. In addition, to enable high-resolution encoding in multi-image inputs, we equipped LEOPARD with an adaptive high-resolution multi-image encoding module. Specifically, it dynamically optimizes the allocation of visual sequence length based on the original aspect ratios and resolutions of the input images. We then apply pixel shuffling to losslessly compress (Chen

<sup>2</sup>Leopards have remarkable visual adaptations that allow them to track prey both from afar and up close, making them highly efficient hunters.

et al., 2024a) long visual feature sequences into shorter ones. This approach allows the model to accommodate multiple high-resolution images without compromising detail or clarity.

We conducted experiments on 13 vision-language benchmark datasets, evaluating LEOARD from multiple perspectives. Consistent improvements were observed when training LEOARD with two distinct base model architectures: LLaVA and Ide cs2. Our results demonstrate LEOARD’s superior performance on 5 text-rich, multi-image benchmarks, outperforming the best open-source MLLM by an average of 9.61 points. Moreover, LEOARD remains highly competitive in text-rich single-image tasks and general-domain vision-language benchmarks, achieving comparable results to state-of-the-art MLLMs without extensive re-tuning. Further ablation studies confirm the effectiveness of our instruction-tuning dataset and the adaptive high-resolution encoding module. These findings highlight LEOARD’s strong performance and versatility across various multimodal applications.

## 2 RELATED WORK

Multimodal Large Language Models (MLLMs). Many approaches have been proposed for building MLLMs, leveraging different architectural designs. A widely adopted approach is the decoder-only architecture, exemplified by LLaVA (Liu et al., 2023b), Emu2 (Sun et al., 2023), and Intern-VL (Chen et al., 2024b). These models typically incorporated a visual encoder to encode images, a vision-language connector to project visual features into the language feature space, and a language model that processes both visual and textual information jointly. Another line of work employed cross-attention architectures where encoded image features are integrated with textual tokens via cross-attention layers, as seen in Flamingo (Alayrac et al., 2022), OpenFlamingo (Awadalla et al., 2023) and CogVLM (Wang et al., 2023a). Such a design allows models to retain the benefits of a fully intact language model but introduces new parameters to manage the visual-textual interplay.

Text-rich MLLMs. Text-rich images are traditionally processed in pipelines (Singh et al., 2019a; Hu et al., 2020), where an OCR module first recognized text from the image, followed by processing through a language model. To improve efficiency and avoid error propagation, with the advent of MLLMs, end-to-end approaches become more popular recently. For instance, LLaVAR (Zhang et al., 2023) utilized a dataset of 400K instances with OCR-enhanced text to outperform LLaVA on various text-rich VQA tasks. Subsequent models such as UReader (Ye et al., 2023), TextMonkey (Liu et al., 2024d), and Mplug-DocOwl-1.5 (Hu et al., 2024a) recognized the importance of high-resolution encoding for accurate text comprehension, so they adopted strategies that cropped single images into multiple sub-images to preserve the original resolution during visual encoding. However, these approaches are primarily trained on single-image data, and struggle to generalize effectively to multi-image scenarios. Furthermore, the straightforward partitioning technique encounters challenges with multi-image inputs, as the sequence length rapidly increases with the number of images.

Multi-image MLLMs. Efforts have been made in training MLLMs with multi-image inputs due to the prevalence of multi-image scenarios in real-world applications. Mantis (Jiang et al., 2024) introduced a multi-image instruction tuning dataset on a variety of natural image scenarios. Besides, both VILA (Lin et al., 2023) and Ide cs-2 (Laureçon et al., 2024b) incorporated image-text interleaved data during their pre-training. LLaVA-Next-Interleave (Li et al., 2024c) further extended this by incorporating videos and multi-view 3D data into the training pipeline. However, these works primarily target natural images and general visual understanding, leaving a gap in handling text-rich, multi-image scenarios. Natural images typically follow a different distribution from text-rich images and often do not demand high-resolution processing. As a result, many existing multi-image MLLMs struggle to generalize to text-rich scenarios. Our work aims to address this gap by specifically focusing on multi-image settings where text-rich images are the primary input.

Concurrent Works Released in 08/2024 and 09/2024. Very recently, multi-image training for MLLMs has attracted intense attention from researchers. Several concurrent efforts have included multi-image interleaved data to train their models, such as LLaVA-OneVision 08/2024 (Li et al., 2024b), Ide cs3 08/2024, Laureçon et al., 2024a), NVLM (09/2024, Dai et al., 2024), mPlug-DocOwl-2 (09/2024, Hu et al., 2024b), Molmo (09/2024, Deitke et al., 2024) and Qwen2-VL (09/2024, Wang et al., 2024). This trending paradigm highlights the significant practical value of multi-image MLLMs by enhancing their ability to tackle a wide range of real-world applications. The incorporation of multi-image instruction tuning data is therefore of paramount importance.

162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215

Figure 2: The overall model pipeline. Given raw image inputs, we first compute the optimal allocation of sub-image numbers and splitting strategy for all images based on their resolution and aspect ratio. The images undergo padding, resizing, and splitting operations. Both sub-images and resized original images are then encoded into a sequence of visual features. These sequences subsequently undergo a pixel shuffle operation that concatenates every four features. The visual features are projected into the language embedding space via a vision-language connector. Finally, the large language model then integrates these visual and language embeddings to generate responses.

### 3 METHOD

LEOPARD follows the typical design of decoder-only vision language models (Liu et al., 2023b; 2024a; Li et al., 2024c), including a visual encoder, a vision language connector, and a language model (LM), as shown in Figure 2. Specially, the input images are first passed through the visual encoder, which extracts high-level visual features and captures essential semantic information. These visual features are then projected into the language representation space via the vision-language connector. After this transformation, the visual tokens are interleaved with the textual tokens, resulting in a sequence of interleaved text-visual tokens. This interleaved sequence is then fed into the LM, which processes these inputs in a causal manner, leveraging the contextual dependencies between text and visual information to generate coherent outputs that align with both modalities.

#### 3.1 MULTI-IMAGE TEXT-RICH INSTRUCTION TUNING DATASET

To train LEOPARD, we construct a large instruction-tuning dataset named LEOPARD-INSTRUCT, comprising 925K instances, with 739K specifically designed for text-rich, multi-image scenarios. While we extensively surveyed existing open-source datasets, we only identified 154K usable text-rich, multi-image samples, which is far from sufficient for effective instruction tuning, as shown in prior MLLM studies (Jiang et al., 2024; Laurion et al., 2024b; Li et al., 2024c). To address this data scarcity, we developed several data collection pipelines to collect high-quality text-rich, multi-image data, resulting in additional 585K instances. Each instance consists of a set of images along with corresponding task instructions and responses. The dataset details are presented in Table 1, and a detailed breakdown of its composition can be found in Appendix A.1.

Documents and Slides are common sources of multi-image data that primarily contain text and require cross-page context integration to fully understand the information.

These data is collected in three ways. First, we include 69K public multi-page document and slide datasets (Tito et al., 2022; Landeghem et al., 2023; Zhu et al., 2022; Tanaka et al., 2023), covering a variety of document types such as scanned handwriting, printed documents, and digital PDFs. Second, we adapt two single-page document datasets, DocVQA (Mathew et al., 2021) and ArxivQA (Li et al., 2024d), for multi-image settings. Following Jiang et al. (2024), we randomly merge 2 to 4 single-page instances by concatenating their respective images and Q-A pairs. Prompts like “in the second image” are added to direct the model’s focus to the appropriate image. These merged samples help the model learn how natural language references align with corresponding image features. Third, we collect raw slides from Se d et al. (2021) and SlideShare<sup>3</sup> and use GPT-4o to generate Q-A pairs and reasoning steps. We show the prompt to GPT in Figure 5. Upon manually reviewing 100 instances annotated by GPT-4o, we found an accuracy rate over 90%, indicating high annotation quality.

Tables and Charts provide highly organized, structured quantitative information, often involving complex data patterns and relationships, requiring the integration of both visual and textual elements for accurate interpretation.

To address the lack of instruction tuning data involving multiple tables or charts, we use the following strategies. First, we include 21K open-source multi-chart and multi-table datasets (Zhao et al., 2022; Pal et al., 2023), originally stored in JSON or DataFrame formats. We programmatically render these tables as images, converting them into multimodal data. Details of rendering can be found in Appendix A.3. Second, we utilize the TableGPT (Li et al., 2024e) dataset and split each table into multiple sub-tables, then convert them into figures, thereby creating multi-modal, multi-table instruction data. Third, we apply the same merging strategy used for combining single-page documents to synthesize multi-image datasets. This approach integrates several single-chart datasets, including ChartGemma (Masry et al., 2024), ChartQA (Masry et al., 2022), DVQA (Kae et al., 2018), and FigureQA (Kahou et al., 2018). Besides, we generate new multi-chart data from social reports of the Pew Research Center<sup>4</sup> that feature multiple interrelated charts within the articles under the same topic. We download charts from the website and use GPT-4o to create 20K Q-A pairs that require multi-chart understanding.

Webpage Snapshots consist of sequential images representing web pages, providing visual context for user interactions and tasks. Understanding webpage is a critical skill for MLLMs to evolve into fully autonomous web agents (Deng et al., 2023; He et al., 2024). To collect and standardize relevant data, we format several web-related multimodal datasets into a Q-A structure as follows:

1. Web action prediction data: We include Mind2Web (Deng et al., 2023) and OmniACT (Kapoor et al., 2024), where we divide long web snapshots into multiple sub-figures, and plot bounding boxes based on the coordinates of web elements. Then GPT-4o is used to convert the original action data into a Q-A format, where the task is to identify the correct element to interact with.
2. Web-based classification data: We incorporate WebScreenshots (Aydos, 2020), WebVision (Li et al., 2017), and WebUI (Wu et al., 2023a). We utilize the web snapshots in these datasets and employ GPT-4o to generate Q-A pairs on webpage understanding, including chain-of-thought reasoning steps. The prompting details are provided in Figure 6.

Augmenting with Rationales. In contrast to single-image tasks, multi-image scenarios typically require MLLMs to integrate information across multiple images, making cross-image reasoning difficult to train when only the final answer is provided (Zheng et al., 2023; Hu et al., 2023). To address this, we employ GPT-4o to generate chain-of-thought (CoT) rationales for inherently multi-image datasets (excluding those formed by merging single-image data) that lack CoT annotations. This results in 250K instances with GPT-annotated reasoning, with the prompt detailed in Figure 7.

<sup>3</sup><https://www.slideshare.net>

<sup>4</sup><https://www.pewresearch.org>

Other Domains. We also include datasets from various other domains such as maps (MapQA, Chang et al., 2022), infographics (InfographicVQA, Mathew et al., 2022), mathematical diagrams (MathV360K, Shi et al., 2024), and abstractive diagrams (IconQA, Lu et al., 2021). We also incorporate mixed-domain datasets for text-rich images, including LLaVAR (Zhang et al., 2023), Monkey, Li et al., 2024f, and mPlugDocReason (Hu et al., 2024a). We remove duplicate subsets from these mixed-domain datasets. Among these datasets, 64K samples consist of multi-image data, while the remaining are single-image samples. To preserve natural image understanding ability, we add 313K samples from ShareGPT4V (Chen et al., 2023), an instruction dataset for natural images.

### 3.2 ADAPTIVE HIGH-RESOLUTION MULTI-IMAGE ENCODING

Image resolution significantly influences the visual perception and understanding capabilities of MLLMs, particularly when processing text-rich images. Low-resolution images often cause printed text to become blurred or unreadable, resulting in misinterpretations, perception errors, and visual hallucinations. The visual resolution of most existing MLLMs is determined by their pre-trained visual encoders, which are typically limited to low resolutions such as 224 or 336 pixels (Liu et al., 2023a; Lin et al., 2023; Jiang et al., 2024). These low-resolution constraints can hinder MLLMs to accurately understand textual information embedded within images.

To overcome these limitations, a natural solution is dividing a high-resolution image into multiple smaller sub-images, each of which is independently processed by the model’s visual encoder (Liu et al., 2024a; Dong et al., 2024). This partitioning allows for the extraction of more fine-grained visual details, making it possible to capture small or densely packed textual elements. However, a major drawback of this approach is that it significantly increases the length of visual feature sequence. When applied to scenarios involving multiple image inputs, the feature sequences are easily exceeding the model’s maximum sequence length limit. To address the issue, we follow the image-splitting idea and propose a novel adaptive high-resolution multi-image encoding strategy as follows.

**Image Allocation Computing:** To prevent the number of sub-image visual features from exceeding the LLM’s maximum sequence length, we first set a budget  $M$  for the total number of sub-images. We allocate this budget proportionally to each input image based on their original sizes. For each image  $i$  with dimensions  $h_i \times w_i$ , we calculate the initial number of sub-images  $S_i$  as:

$$S_i = \frac{h_i}{v} \times \frac{w_i}{v} \times k; \tag{1}$$

where  $v$  is the resolution of visual encoder (e.g.,  $v = 364$  pixels). If the total number of patches satisfies  $\sum_i S_i \leq M$ , we proceed with these sub-image counts. Otherwise, we scale down these counts proportionally using a scaling factor  $b = \frac{M}{\sum_i S_i}$ , resulting in adjusted sub-image counts:

$$S_i^0 = b S_i; \tag{2}$$

**Image Partitioning:** For each image, we perform a grid search over possible number of rows and columns  $r, c$  (where  $1 \leq r, c \leq S_i^0$  and  $r \times c = S_i^0$ ) to find the optimal cropping configuration that maximizes the effective resolution within the allocated sub-images (Li et al., 2024a). This configuration results in the original image being padded and resized to a target resolution of  $r \times v; w_i^0 = c \times v$ . We then divide the image into  $c$  sub-images of size  $(v \times v)$ . Additionally, the original image is directly resized to  $(w_i^0 \times v)$ , which provides a global view of the visual content.

**Image Encoding** Most vision encoders transform an image into a sequence of visual features  $v \in \mathbb{R}^{L \times d}$ , where  $L$  represents the sequence length and  $d$  denotes the feature dimension. Typically,  $L$  is in the hundreds, e.g., the SigLIP encoder yields a visual feature sequence in the shape of  $L = 676$  and  $d = 1152$  for the input image. Given that most LLMs have a sequence length of only 8K tokens, this implies that without any text input, the model can encode at most 12 images, which severely limits the image allocation budget. To mitigate this issue, inspired by the pixel shuffling operation (Chen et al., 2024a; Laugel et al., 2024), we apply a similar strategy to the visual features. Specifically, we concatenate adjacent visual features along the feature dimension,

<sup>5</sup> $M$  is a hyperparameter, and we provide experiments on varying different  $M$  in Figure 3.

Table 2: A detailed comparison of the model training details between baseline models and LEOPARD, including image resolution, vision encoder, backbone LLM, number of parameters (Param.), pre-training (PT.) data size, and instruction tuning (IT.) data size of baselines. AnyRes denotes the resolution selecting method proposed by Liu et al. (2024a) and Adapt HR. represents the proposed adaptive high-resolution multi-image encoding strategy.

Models	Visual Encoder	Resolution	Backbone LLM	Param.	PT.	IT.
Otter-9B (Li et al., 2023)	CLIP ViT-L	224 <sup>2</sup>	LLaMA-7B	9B	30M	5.1M
Emu2-Chat (Sun et al., 2023)	EVA-02-CLIP	448 <sup>2</sup>	LLaMA-33B	37B	-	160M
MM1-7B-Chat (McKinzie et al., 2024)	CLIP ViT-H	378 <sup>2</sup>	-	7B	-	1.5M
VILA1.5-8B (Lin et al., 2023)	SigLIP	384 <sup>2</sup>	LLaMA3-8B	8B	50M	1M
mPlug-DocOwl-1.5 (Hu et al., 2024a)	CLIP ViT-L	448 <sup>2</sup> (x9 crops)	LLaMA-7B	8B	4M	1M
Ide cs2-8B (Laurençon et al., 2024b)	SigLIP	980 <sup>2</sup>	Mistral-7B	8B	350M	20M
LLaVA-NeXT-Inter (Li et al., 2024c)	SigLIP	AnyRes	Qwen1.5-7B	7B	1.3M	1.2M
Mantis-LLaVA (Jiang et al., 2024)	SigLIP	384 <sup>2</sup>	LLaMA3-8B	8B	0.5M	1M
Mantis-Ide cs2 (Jiang et al., 2024)	SigLIP	980 <sup>2</sup>	Mistral-7B	8B	350M	1M
LEOPARD-LLaVA (Ours)	SigLIP	Adapt HR.	LLaMA3.1-8B	8B	0.5M	1.2M
LEOPARD-Ide cs2 (Ours)	SigLIP	980 <sup>2</sup>	Mistral-7B	8B	350M	1.2M

effectively reducing the sequence length by a factor of  $\frac{1}{n}$ . This results in a compressed visual feature sequence of length  $\frac{1}{n} \times 2 \times R \times n$ . By decreasing the sequence length in this way, we are able to accommodate more images within the sequence length constraints of the LLM. To incorporate visual features into the LLM, we first project the encoded visual feature sequences into the textual input embedding space using a vision-language connector. Since the partitioned images yield feature sequences of variable length, we introduce special tokens into the textual input to demarcate the image features to help the model distinguish visual features. Specifically, the sequence for  $n$  images is formatted as: `[Image]: <Img> < Visual Feature Sequence = <Img> g`, where `<Img>` and `= <Img>` are special tokens. An illustrative example of this sequence formatting is provided in Figure 2.

## 4 EXPERIMENT

### 4.1 IMPLEMENTATION DETAILS

**Model Architecture.** We train our models on two base architectures: LLaVA (Liu et al., 2023a) and Ide cs2 (Laurençon et al., 2024b). For LEOPARD-LLaVA, we use SigLIP-SO-400M (Zhai et al., 2023) with 364 × 364 image resolutions as the visual encoder since it supports larger resolution than the commonly used 224 × 224 resolution CLIP visual encoder (Radford et al., 2021). Each image is encoded into a sequence of  $26 \times 26 = 676$  visual features under a patch size of 14. With the visual feature pixel shuffling strategy, each image is further processed into a sequence of 169 visual features. We limit the maximum number of images ( $n$ ) in each sample to 50, which produces up to 8,450 visual features in total. Following Liu et al. (2023a), we adopt a two-layer MLPs as the visual-language connector. We use LLaMA-3.1 (Meta et al., 2024) as the LM.

For LEOPARD-Ide cs2, we follow the architecture of Ide cs2-8B which uses SigLIP-SO-400M as the visual encoder but increases its image resolution to 980 × 980 to make the text legible. The features outputted by the visual encoder are compressed with a feature resampler of 5 tokens per image. Ide cs2-8B adopts the Mistral-7B (Jiang et al., 2023) as the LM.

**Training Details.** When training LEOPARD-LLaVA, we first train the visual-language connector using LLaVA’s 558K multimodal pre-training dataset. Subsequently, we fine-tune the model (with both the connector and the LM unfrozen) using LEOPARD-INSTRUCT data. As for LEOPARD-Ide cs2, it is pre-trained on a dataset comprised of over 350M multimodal samples. Given the computational challenges of reproducing such extensive pre-training, and to ensure a fair comparison with baselines that utilize the pre-trained Ide cs2 checkpoint, we directly adopt Ide cs2’s visual feature resampler and fine-tune the model on the LEOPARD-INSTRUCT dataset.

We train both LEOPARD-LLaVA and LEOPARD-Ide cs2 on 64 A100-40G GPUs with a global batch size of 128. We use the AdamW optimizer with  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ . Following Jiang et al. (2024), we use a learning rate of  $10^{-5}$  for LEOPARD-LLaVA and  $5 \times 10^{-6}$  for LEOPARD-

Table 3: Experiment results of baseline models and LEOPARD on 8 benchmarks of text-rich images. We use abbreviated benchmark names due to space limits. MVQA: Multi-page DocVQA, MCQA: MultiChartQA, MH: MultiHiertt, VQA<sup>T</sup>: TextVQA, VQA<sup>D</sup>: DocVQA, VWB: VisualWebBench. Following (Tito et al., 2022), for MVQA, DUDE, and VQA<sup>D</sup>, we use average normalized levenshtein similarity (ANLS) as the evaluation metric. For others, accuracy (Acc.) is used as the metric, which measures whether the predicted answer matches exactly with any of the target answers.

Models	Text-Rich Multi-Image						Text-Rich Single-Image			
	MVQA <sup>D</sup>	DUDE	SlideVQA	MCQA	MH	Multi Avg.	VQA <sup>T</sup>	VQA <sup>D</sup>	VWB	Avg.
Otter-9B	0.17	0.15	5.95	1.08	0.14	1.50	23.18	3.53	10.20	12.30
Emu2-Chat	17.58	13.79	0.60	2.40	0.72	7.02	66.60	5.44	18.17	30.07
MM1-7B-Chat	-	-	-	-	-	-	72.80	-	-	-
VILA-LLaMA3-8B	30.75	19.75	24.72	1.87	3.66	16.15	66.30	30.38	23.37	40.02
mPlug-DocOwl-1.5	35.85	16.94	4.54	0.26	0.86	11.69	68.60	82.20	29.80	60.20
Ide cs2-8B	46.67	23.06	25.14	2.59	9.89	21.47	70.40	67.30	23.76	53.82
LLaVA-NeXT-Inter	39.92	24.04	23.46	14.34	3.55	21.06	62.76	75.70	21.36	53.27
Mantis-LLaVA	31.89	17.73	16.81	9.72	3.46	15.92	59.20	39.02	17.88	38.70
Mantis-Ide cs2	51.61	27.74	24.02	12.97	5.48	24.36	63.50	54.03	22.47	46.67
LEOPARD-LLaVA	53.90	35.94	23.83	9.68	10.76	26.82	67.70	68.07	24.91	53.56
LEOPARD-Ide cs2	66.06	40.74	34.93	18.03	10.09	33.97	80.40	74.79	25.60	60.26

Ide cs2 to protect its pretraining knowledge. We use a cosine learning rate scheduler with a linear learning rate warm-up for the first 5% steps. All model variants are trained 1 epoch under the same hyperparameters. It takes around 120 GPU days to train LEOPARD under both settings.

## 4.2 BASELINE MODELS

We compare LEOPARD against a range of existing open-source MLLMs that support multi-image inputs. The baseline models included in our comparison are Otter-9B (Li et al., 2023), Emu2-Chat-34B (Sun et al., 2023), MM1-7B-Chat (McKinzie et al., 2024), Mantis (Jiang et al., 2024), VILA (Lin et al., 2023), Ide cs2-8B (Laurençon et al., 2024b), and LLaVA-NeXT-Interleave (Li et al., 2024c).

Table 4: Experimental results on general domain benchmarks. We abbreviate the image split of ScienceQA as SQA

Models	MIRB	MiBench	MMMU	MathVista	SQA	Avg.
Otter-9B	20.74	43.72	30.89	22.00	60.43	35.55
Emu2-Chat	36.02	58.93	34.10	30.40	65.60	45.03
MM1-7B-Chat	-	-	37.00	35.90	72.60	-
VILA-LLaMA3-8B	40.87	53.70	36.90	35.40	79.90	49.35
mPlug-DocOwl-1.5	25.39	40.80	35.44	29.50	64.40	49.11
Ide cs2-8B	33.02	46.39	42.90	45.00	89.00	51.27
LLaVA-NeXT-Inter	44.38	74.52	38.44	32.10	72.63	52.41
Mantis-LLaVA	40.76	59.96	40.10	34.40	74.90	50.02
Mantis-Ide cs2	41.80	56.80	41.10	40.40	81.30	52.28
LEOPARD-LLaVA	42.00	60.80	43.00	45.50	85.57	55.37
LEOPARD-Ide cs2	41.38	61.74	40.11	44.80	90.38	55.68

## 4.3 EVALUATING BENCHMARKS

We evaluated LEOPARD and baseline methods across three categories of vision-language tasks on (1) single text-rich image evaluation, (2) multiple text-rich images evaluation, and (3) general reasoning evaluation. Benchmarks for (1) include TextVQA (Singh et al., 2019b), DocVQA (Mathew et al., 2021), and VisualWebBench (Liu et al., 2024c). Benchmarks for (2) include Multi-page DocVQA (Tito et al., 2022), DUDE (Landeghem et al., 2023), SlideVQA (Tanaka et al., 2023), Multihiertt (Zhao et al., 2022), and MultiChartQA (Anonymous, 2024), which cover a diverse range of typical multi-image tasks, such as document understanding and slide question answering. Benchmarks for (3) include MMMU (Yue et al., 2024), MathVista (Lu et al., 2024), ScienceQA (Saikh

Table 5: Ablation studies of LEOPARD-LLaVA from four different perspectives: (1) evaluating the impact of Adaptive High-Resolution Encoding, (2) pre-training LLaVA by initializing with checkpoints from either LLaMA-3 or LLaMA-3.1, and (3) examining the impact of using different data domains for instruction tuning, including doc, chart, and web.

Ablation Settings	Text-Rich Multi-Image				Text-Rich Single		General	
	MVQA <sup>D</sup>	DUDE	SlidesVQA	Multi Avg.	TextVQA	DocVQA	MMMU	MathVista
(?) Our Best Setting (as in Table 3): LLaMA-3.1 + Adaptive +								
LEOPARD-LLaVA	53.90	35.94	23.83	37.89	67.70	68.07	43.00	45.50
(1) Effect of Adaptive High-Resolution Encoder: LLaMA-3.1 +								
- w/o Adaptive	40.44	26.16	20.93	29.17#)	60.18	44.69	41.00	42.40
(2) Effect of Backbone LLM: LLaMA-3 + Adaptive +								
- with LLaMA-3.1	48.66	32.64	25.75	35.62#)	67.08	54.92	41.22	42.10
(3) Effect of Data Domains: LLaMA-3.1 + Adaptive								
- with chart web	43.79	29.50	23.10	32.13#)	66.78	56.60	40.67	44.80
- with doc web	54.33	35.65	18.73	36.23#)	66.86	50.78	41.89	39.60
- with doc chart	54.62	35.70	20.79	37.02#)	67.40	67.82	41.78	44.00

et al., 2022), MIRB (Zhao et al., 2024) and MiBench (Liu et al., 2024b), which evaluate MLLMs from different perspectives, including world knowledge, mathematics, and scientific reasoning.

#### 4.4 MAIN EXPERIMENTAL RESULTS

##### Question 1: How does LEOPARD compare to state-of-the-art MLLMs on vision-language tasks?

LEOPARD achieves outstanding performance on text-rich, multi-image benchmarks, as shown in Table 3. Notably, both LEOPARD-LLaVA and LEOPARD-Ide cs2 significantly outperform all baselines. LEOPARD-Ide cs2 becomes the strongest open-source MLLM in this area, achieving an average improvement of 9.61 points over the previous best performance.

In single-image text-rich scenarios, LEOPARD outperforms several recent strong models, including VILA and LLaVA-NeXT. LEOPARD even achieves slightly higher average scores than the state-of-the-art mPlug model, despite mPlug being trained on 4M single-image data while LEOPARD is tuned on < 200K. This demonstrates that training on multi-image data from LEOPARD-INSTRUCT also benefits model performance on single-image tasks.

In addition, we evaluate LEOPARD on general-domain benchmarks which contain both multi-image and single-image instances. As shown in Table 4, LEOPARD outperforms other open-source MLLMs on these benchmarks. Remarkably, LEOPARD surpasses Mantis, its counterpart multi-image model trained on the same foundational architecture and a comparable volume of data. This performance demonstrates the high quality and diversity of the LEOPARD-INSTRUCT dataset, which effectively preserves our model’s general image understanding capabilities.

##### Question 2: Is the one-million text-rich multi-image dataset effective for instruction tuning?

Mantis-Ide cs2 is trained on a combination of natural multi-image data and text-rich single-image data. However, LEOPARD-Ide cs2 outperforms Mantis-Ide cs2 by 12.8 points on text-rich multi-image benchmarks. This disparity indicates that developing strong multi-image text-rich capabilities through cross-domain transfer, such as with Mantis data, presents significant challenges. This finding underscores the importance of optimizing LEOPARD using high-quality, diverse, and well-curated multi-image text-rich datasets that are specifically tailored for complex multi-image scenarios.

Furthermore, LEOPARD-Ide cs2 surpasses its base model, Ide cs2, by 6.4 points across three single-image text-rich benchmarks, though Ide cs2 is trained on over 20M instruction data that includes text-rich tasks like DocVQA and TextVQA. This highlights that the LEOPARD-INSTRUCT provides unique advantages to MLLMs that are not adequately addressed by existing datasets.

##### Question 3: Does Adaptive high-resolution multi-image encoding improve MLLM performance?

To assess the effectiveness of the proposed adaptive high-resolution multi-image encoding, we compared LEOPARD with a variant that excludes this feature (w/o Adaptive in Table 5). We

486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539

Figure 3: Impact of the sub-image budget on the resulting model across four benchmarks.  $\alpha=1$  means original images are not partitioned into sub-images.

notice a significant performance decline across all text-rich benchmarks, particularly on document-related benchmarks like DocVQA (-23.4), Multi-page DocVQA (-13.5), and DUDE (-9.8). This observation supports our hypothesis that high-resolution image encoding is especially beneficial for text-rich images, particularly those with dense text content such as document pages.

#### 4.5 MORE ANALYSIS

Question 4: How does data from different domains contribute to instruction tuning?

LEOPARD-INSTRUCT mainly cover three main domains (i.e., documents & slides (doc), tables & charts (chart), and websites (web)). To assess the impact of data from different domains, we conduct ablation studies on three variants of LEOPARD, with the results presented in Table 5. Removing any part of the training data results in performance degradation. The most significant drop occurs when we exclude document data while removing web data leads to a slight decrease. However, the mixed-domain datasets, such as LLaVAR and mPlugDocReason, also contain data in these domains which are challenging to isolate and ablate. This may contribute to the relatively preserved performance even after the ablation of certain data sources.

Question 5: What is the influence of different image budgets in adaptive multi-image encoding?

In our adaptive multi-image encoding module, we define a budget for the maximum number of sub-images that the model can process. To evaluate the impact of such image partitioning, we train LEOPARD using different values of  $m$ : 25, 50, 75, as well as a baseline setting where no image partitioning is applied and the number of sub-images equals the number of original images. According to the results plotted in Figure 3, model performance peaks or plateaus around 50. Thus, we adopt 50 as the default value for training LEOPARD. These results show that increasing image numbers does not consistently improve performance, as input sequences can become excessively long and even exceed the model's sequence length limit.

Question 6: How does the backbone language model affect the performance?

To ensure a fair comparison with multi-image competitor models, Mantis-LLaVA and VILA1.5, we also evaluate a variant of LEOPARD using LLaMA-3 instead of LLaMA-3.1, aligning its backbone language model architecture with these two baselines. According to Table 5, this substitution results in only a slight drop in average performance on text-rich multi-image tasks (2.2). Nevertheless, comparing with results in Table 5, LEOPARD-LLaMA-3 still substantially outperforms both baselines in all tasks, such as Multi-page DocVQA (+16.8 over Mantis and +17.9 over VILA) and DUDE (+14.9 over Mantis and +12.9 over VILA). These results indicate that LEOPARD's superior performance is not simply a result of the upgraded backbone large language models.

## 5 CONCLUSION

In this paper, we introduce LEOPARD, a novel MLLM specifically designed for text-rich, multi-image tasks. LEOPARD is equipped with two key innovations: (1) LEOPARD-INSTRUCT, a large-scale instruction-tuning dataset that encompasses a wide range of text-rich, multi-image instructions, and (2) an adaptive image encoding module capable of processing multiple high-resolution images efficiently. Our experimental results across diverse benchmarks highlight LEOPARD's superior performance compared to existing open-source MLLMs, particularly in text-rich multi-image scenarios. Further analysis and ablation studies underscore the effectiveness of both the collected dataset and adaptive encoding strategy, solidifying LEOPARD's contribution to multimodal research.

## REFERENCES

- 540  
541  
542 Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel  
543 Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan  
544 Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian  
545 Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo  
546 Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language  
547 model for few-shot learning, 2022.
- 548 Anonymous. A benchmark for multi-chart question answering. *UnderReview*, 2024. URL <https://vlcodes.github.io/MultiChartQA/>  
549
- 550 Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe,  
551 Yonatan Bitton, Samir Yitzhak Gadre, Shiori Sagawa, Jenia Jitsev, Simon Kornblith, Pang Wei  
552 Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. Open amingo: An open-source  
553 framework for training large autoregressive vision-language models. *CoRR*, abs/2308.01390,  
554 2023.
- 555 Fahri Aydos. Webscreenshots, 2020. URL <https://www.kaggle.com/ds/202248>  
556
- 557 Shuaichen Chang, David Palzer, Jialin Li, Eric Fosler-Lussier, and Ningchuan Xiao. Mapqa: A  
558 dataset for question answering on choropleth maps. *CoRR*, abs/2211.08545, 2022.  
559
- 560 Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin.  
561 Sharegpt4v: Improving large multi-modal models with better captions. *CoRR*, abs/2311.12793,  
562 2023.
- 563 Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong,  
564 Kongzhi Hu, Jiapeng Luo, Zheng Ma, Ji Ma, Jiaqi Wang, Xiaoyi Dong, Hang Yan, Hwei Guo,  
565 Conghui He, Botian Shi, Zhenjiang Jin, Chao Xu, Bin Wang, Xingjian Wei, Wei Li, Wenjian  
566 Zhang, Bo Zhang, Pinlong Cai, Licheng Wen, Xiangchao Yan, Min Dou, Lewei Lu, Xizhou Zhu,  
567 Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. How far are we to gpt-4v? closing  
568 the gap to commercial multimodal models with open-source suites, 2024a.
- 569 Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong  
570 Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning  
571 for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer  
572 Vision and Pattern Recognition*, pp. 24185–24198, 2024b.
- 573 Wenliang Dai, Nayeon Lee, Boxin Wang, Zhuoling Yang, Zihan Liu, Jon Barker, Tuomas Rintamaki,  
574 Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nvlm: Open frontier-class multimodal llms.  
575 *arXiv preprint arXiv:2409.11402*, 2024.  
576
- 577 Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Moham-  
578 madreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open  
579 weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*  
580 2024.  
581
- 582 Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samuel Stevens, Boshi Wang, Huan Sun, and Yu Su.  
583 Mind2web: Towards a generalist agent for the web. *Advances in Neural Information Processing  
584 Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS2023,  
585 New Orleans, LA, USA, December 10 - 16, 2023*, 2023.
- 586 Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Songyang Zhang,  
587 Haodong Duan, Wenwei Zhang, Yining Li, Hang Yan, Yang Gao, Zhe Chen, Xinyue Zhang, Wei  
588 Li, Jingwen Li, Wenhai Wang, Kai Chen, Conghui He, Xingcheng Zhang, Jifeng Dai, Yu Qiao,  
589 Dahua Lin, and Jiaqi Wang. Internlm-xcomposer2-4khd: A pioneering large vision-language  
590 model handling resolutions from 336 pixels to 4k. *CoRR*, abs/2404.06512, 2024.  
591
- 592 Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan,  
593 and Dong Yu. Webvoyager: Building an end-to-end web agent with large multimodal models.  
*arXiv preprint arXiv:2401.13919*, 2024.

- 594 Yu-Chung Hsiao, Fedir Zubach, Maria Wang, and Jindong Chen. Screenqa: Large-scale question-  
595 answer pairs over mobile app screenshots, 2024.
- 596
- 597 Anwen Hu, Haiyang Xu, Jiabo Ye, Ming Yan, Liang Zhang, Bo Zhang, Chen Li, Ji Zhang, Qin Jin,  
598 Fei Huang, and Jingren Zhou. mplug-docowl 1.5: Uni ed structure learning for ocr-free document  
599 understanding [CoRR](#), abs/2403.12895, 2024a.
- 600 Anwen Hu, Haiyang Xu, Liang Zhang, Jiabo Ye, Ming Yan, Ji Zhang, Qin Jin, Fei Huang, and  
601 Jingren Zhou. mplug-docowl2: High-resolution compressing for ocr-free multi-page document  
602 understanding [arXiv preprint arXiv:2409.03420](#), 2024b.
- 603
- 604 Ronghang Hu, Amanpreet Singh, Trevor Darrell, and Marcus Rohrbach. Iterative answer prediction  
605 with pointer-augmented multimodal transformers for textvqa [2020 IEEE/CVF Conference on  
606 Computer Vision and Pattern Recognition, CVPR 2020](#), 2020.
- 607 Yushi Hu, Otilia Stretcu, Chun-Ta Lu, Krishnamurthy Viswanathan, Kenji Hata, Enming Luo, Ranjay  
608 Krishna, and Ariel Fuxman. Visual program distillation: Distilling tools and programmatic  
609 reasoning into vision-language models [CoRR](#), abs/2312.03052, 2023.
- 610
- 611 Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot,  
612 Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier,  
613 L  lio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas  
614 Wang, Timothee Lacroix, and William El Sayed. Mistral 7B [CoRR](#), abs/2310.06825, 2023.
- 615 Dongfu Jiang, Xuan He, Huaye Zeng, Cong Wei, Max Ku, Qian Liu, and Wenhui Chen. MANTIS:  
616 interleaved multi-image instruction tuning [CoRR](#), abs/2405.01483, 2024.
- 617
- 618 Kushal K  e, Brian L. Price, Scott Cohen, and Christopher Kanan. DVQA: understanding data  
619 visualizations via question answering. [2018 IEEE Conference on Computer Vision and Pattern  
620 Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018](#) pp. 5648–5656. Computer  
621 Vision Foundation / IEEE Computer Society, 2018.
- 622
- 623 Samira Ebrahimi Kahou, Vincent Michalski, Adam Atkinson, Akos K  c  r, Adam Trischler, and  
624 Yoshua Bengio. Figureqa: An annotated figure dataset for visual reasoning [6th International  
625 Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3,  
2018, Workshop Track Proceedings](#). OpenReview.net, 2018.
- 626
- 627 Raghav Kapoor, Yash Parag Butala, Melisa Russak, Jing Yu Koh, Kiran Kamble, Waseem AlShikh,  
628 and Ruslan Salakhutdinov. Omniact: A dataset and benchmark for enabling multimodal generalist  
629 autonomous agents for desktop and web [CoRR](#), abs/2402.17553, 2024.
- 630
- 631 Jordy Van Landeghem, Rafal Powalski, Renit Tito, Dawid Jurkiewicz, Matthew B. Blaschko, Lukasz  
632 Borchmann, Mick  l Coustaty, Sien Moens, Michal Pietruszka, Bertrand Anckaert, Tomasz  
633 Stanislawek, Pawe  l Ziak, and Ernest Valveny. Document understanding dataset and evaluation  
634 (DUDE). In [IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France,  
October 1-6, 2023](#), pp. 19471–19483. IEEE, 2023.
- 635
- 636 Hugo Lauren  on, Lucile Saulnier, L  o Tronchon, Stas Bekman, Amanpreet Singh, Anton Lozhkov,  
637 Thomas Wang, Siddharth Karamcheti, Alexander M. Rush, Douwe Kiela, Matthieu Cord, and  
638 Victor Sanh. OBELICS: an open web-scale iterated dataset of interleaved image-text docu-  
639 ments. In [Advances in Neural Information Processing Systems 36: Annual Conference on Neural  
640 Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 -  
16, 2023](#), 2023.
- 641
- 642 Hugo Lauren  on, Andr  s Mara oti, Victor Sanh, and L  o Tronchon. Building and better under-  
643 standing vision-language models: insights and future directions [arXiv preprint arXiv:2408.12637](#)  
2024a.
- 644
- 645 Hugo Lauren  on, L  o Tronchon, Matthieu Cord, and Victor Sanh. What matters when building  
646 vision-language models [CoRR](#), abs/2405.02246, 2024b.
- 647
- 648 Hugo Lauren  on, Andr  s Mara oti, Victor Sanh, and L  o Tronchon. Building and better understand-  
ing vision-language models: insights and future directions, 2024.

- 648 Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. Otter: A  
649 multi-modal model with in-context instruction tuning. CoRR, abs/2305.03726, 2023.
- 650  
651 Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li,  
652 Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer, 2024a.
- 653  
654 Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei  
655 Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. arXiv preprint  
656 arXiv:2408.03326, 2024b.
- 657  
658 Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li.  
659 Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. CoRR,  
660 abs/2407.07895, 2024c.
- 661  
662 Lei Li, Yuqi Wang, Runxin Xu, Peiyi Wang, Xiachong Feng, Lingpeng Kong, and Qi Liu. Multi-  
663 modal arxiv: A dataset for improving scientific comprehension of large vision-language models.  
664 In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics  
665 (Volume 1: Long Papers) ACL 2024, Bangkok, Thailand, August 11-16, 2024, pp. 14369–14387.  
666 Association for Computational Linguistics, 2024d.
- 667  
668 Peng Li, Yeye He, Dror Yashar, Weiwei Cui, Song Ge, Haidong Zhang, Danielle Rinski Fainman,  
669 Dongmei Zhang, and Surajit Chaudhuri. Table-gpt: Table fine-tuned GPT for diverse table tasks.  
670 Proc. ACM Manag. Data, 2(3):176, 2024e.
- 671  
672 Wen Li, Limin Wang, Wei Li, Eirikur Agustsson, and Luc Van Gool. Webvision database: Visual  
673 learning and understanding from web data. CoRR, abs/1708.02862, 2017.
- 674  
675 Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu,  
676 and Xiang Bai. Monkey: Image resolution and text label are important things for large multi-  
677 modal models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern  
678 Recognition, pp. 26763–26773, 2024f.
- 679  
680 Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz,  
681 Mohammad Shoeybi, and Song Han. VILA: on pre-training for visual language models. CoRR,  
682 abs/2312.07533, 2023.
- 683  
684 Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction  
685 tuning. CoRR, abs/2310.03744, 2023a.
- 686  
687 Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023b.
- 688  
689 Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee.  
690 Llava-next: Improved reasoning, ocr, and world knowledge, January 2024a. https://  
691 llava-vl.github.io/blog/2024-01-30-llava-next/
- 692  
693 Haowei Liu, Xi Zhang, Haiyang Xu, Yaya Shi, Chaoya Jiang, Ming Yan, Ji Zhang, Fei Huang,  
694 Chunfeng Yuan, Bing Li, and Weiming Hu. Mibench: Evaluating multimodal large language  
695 models over multiple images. CoRR, abs/2407.15272, 2024b.
- 696  
697 Junpeng Liu, Yifan Song, Bill Yuchen Lin, Wai Lam, Graham Neubig, Yuanzhi Li, and Xiang  
698 Yue. Visualwebbench: How far have multimodal llms evolved in web page understanding and  
699 grounding? CoRR, abs/2404.05955, 2024c.
- 700  
701 Yuliang Liu, Biao Yang, Qiang Liu, Zhang Li, Zhiyin Ma, Shuo Zhang, and Xiang Bai. Textmonkey:  
An ocr-free large multimodal model for understanding documents. arXiv preprint, 2403.04473,  
2024d.
- Pan Lu, Liang Qiu, Jiaqi Chen, Tanglin Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and  
Song-Chun Zhu. Iconqa: A new benchmark for abstract diagram understanding and visual language  
reasoning. In Proceedings of the Neural Information Processing Systems Track on Datasets and  
Benchmarks, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual, 2021.

- 702 Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter  
703 Clark, and Ashwin Kalyan. Dynamic prompt learning via policy gradient for semi-structured  
704 mathematical reasoning. The Eleventh International Conference on Learning Representations,  
705 ICLR 2023, Kigali, Rwanda, May 1-5, 2023. OpenReview.net, 2023.
- 706 Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng,  
707 Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning  
708 of foundation models in visual contexts. The Twelfth International Conference on Learning  
709 Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024. OpenReview.net, 2024.
- 710 Ahmed Masry, Xuan Long Do, Jia Qing Tan, Sha q Joty, and Enamul Hoque. Chartqa: A bench-  
711 mark for question answering about charts with visual and logical reasoning. Findings of the  
712 Association for Computational Linguistics: ACL 2022, pp. 2263–2279, 2022.
- 713 Ahmed Masry, Megh Thakkar, Aayush Bajaj, Aaryaman Kartha, Enamul Hoque, and Sha q  
714 Joty. Chartgemma: Visual instruction-tuning for chart reasoning in the vaixiv preprint  
715 arXiv:2407.04172, 2024.
- 716 Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document  
717 images. In Proceedings of the IEEE/CVF winter conference on applications of computer vision,  
718 pp. 2200–2209, 2021.
- 719 Minesh Mathew, Viraj Bagal, Ruben Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. In-  
720 fographicvqa. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer  
721 Vision, pp. 1697–1706, 2022.
- 722 Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter,  
723 Dhruti Shah, Xianzhi Du, Futang Peng, Floris Weers, Anton Belyi, Haotian Zhang, Karanjeet  
724 Singh, Doug Kang, Ankur Jain, Hongyu et, Max Schwarzer, Tom Gunter, Xiang Kong, Aonan  
725 Zhang, Jianyu Wang, Chong Wang, Nan Du, Tao Lei, Sam Wiseman, Guoli Yin, Mark Lee, Zirui  
726 Wang, Ruoming Pang, Peter Grasch, Alexander Toshev, and Yinfei Yang. MM1: methods, analysis  
727 & insights from multimodal LLM pre-training CoRR, abs/2403.09611, 2024.
- 728 Meta, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle,  
729 Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony  
730 Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark,  
731 Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozi  
732 Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris  
733 Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong,  
734 Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny  
735 Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino,  
736 Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael  
737 Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson,  
738 Graeme Nail, Grgoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar,  
739 Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra,  
740 Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet  
741 Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi,  
742 Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph  
743 Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani,  
744 Kate Plawiak, Ke Li, Kenneth Heald, Kevin Stone, and et al. The llama 3 herd of models CoRR,  
745 abs/2407.21783, 2024.
- 746 Vaishali Pal, Andrew Yates, Evangelos Kanoulas, and Maarten de Rijke. MultiTabQA: Generating  
747 tabular answers for multi-table question answering. Proceedings of the 61st Annual Meeting of  
748 the Association for Computational Linguistics (Volume 1: Long Papers) Toronto, Canada, 2023.  
749 Association for Computational Linguistics.
- 750 Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal,  
751 Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever.  
752 Learning transferable visual models from natural language supervision. Proceedings of the  
753 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event  
754 volume 139 of Proceedings of Machine Learning Research, pp. 8748–8763. PMLR, 2021.

- 756 Tanik Saikh, Tirthankar Ghosal, Amish Mittal, Asif Ekbal, and Pushpak Bhattacharyya. Scienceqa:  
757 a novel resource for question answering on scholarly articles. *ISJ. Digit. Libr.*, 23(3):289–301,  
758 2022.
- 759  
760 Athar Se d, Prasenjit Mitra, Jian Wu, and C Lee Giles. Extractive research slide generation using  
761 windowed labeling ranking. In *Proceedings of the Second Workshop on Scholarly Document  
762 Processing*. Association for Computational Linguistics, 2021.
- 763  
764 Wenhao Shi, Zhiqiang Hu, Yi Bin, Junhua Liu, Yang Yang, See-Kiong Ng, Lidong Bing, and Roy Ka-  
765 Wei Lee. Math-llava: Bootstrapping mathematical reasoning for multimodal large language models.  
766 *CoRR*, abs/2406.17294, 2024.
- 767  
768 Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh,  
769 and Marcus Rohrbach. Towards VQA models that can read. In *IEEE Conference on Computer  
770 Vision and Pattern Recognition, CVPR 2019*, 2019a.
- 771  
772 Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh,  
773 and Marcus Rohrbach. Towards vqa models that can read. *Proceedings of the IEEE/CVF  
774 conference on computer vision and pattern recognition*, pp. 8317–8326, 2019b.
- 775  
776 Quan Sun, Yufeng Cui, Xiaosong Zhang, Fan Zhang, Qiyang Yu, Zhengxiong Luo, Yueze Wang,  
777 Yongming Rao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative multimodal models  
778 are in-context learners. *CoRR*, abs/2312.13286, 2023.
- 779  
780 Ryota Tanaka, Kyosuke Nishida, Kosuke Nishida, Taku Hasegawa, Itsumi Saito, and Kuniko Saito.  
781 Slidevqa: A dataset for document visual question answering on multiple images. *Proceedings  
782 of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 13636–13645, 2023.
- 783  
784 Jingqun Tang, Chunhui Lin, Zhen Zhao, Shu Wei, Binghong Wu, Qi Liu, Hao Feng, Yang Li, Siqi  
785 Wang, Lei Liao, Wei Shi, Yuliang Liu, Hao Liu, Yuan Xie, Xiang Bai, and Can Huang. Textsquare:  
786 Scaling up text-centric visual instruction tuning. *CoRR*, abs/2404.12803, 2024.
- 787  
788 Ruben Tito, Dimosthenis Karatzas, and Ernest Valveny. Hierarchical multimodal transformers for  
789 multi-page docvqa. *CoRR*, abs/2212.05935, 2022.
- 790  
791 Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu,  
792 Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the  
793 world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.
- 794  
795 Weihang Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang,  
796 Lei Zhao, Xixuan Song, Jiazheng Xu, Bin Xu, Juanzi Li, Yuxiao Dong, Ming Ding, and Jie Tang.  
797 CogVlm: Visual expert for pretrained language models, 2023a.
- 798  
799 Xiao Wang, Guangyao Chen, Guangwu Qian, Pengcheng Gao, Xiao-Yong Wei, Yaowei Wang,  
800 Yonghong Tian, and Wen Gao. Large-scale multi-modal pre-trained models: A comprehensive  
801 survey. *Machine Intelligence Research*, 20(4):447–482, 2023b.
- 802  
803 Jason Wu, Siyan Wang, Siman Shen, Yi-Hao Peng, Jeffrey Nichols, and Jeffrey P. Bigham. Webui: A  
804 dataset for enhancing visual UI understanding with web semantics. *Proceedings of the 2023  
805 CHI Conference on Human Factors in Computing Systems CHI 2023, Hamburg, Germany April  
806 23-28, 2023*, pp. 286:1–286:14. ACM, 2023a.
- 807  
808 Yang Wu, Shilong Wang, Hao Yang, Tian Zheng, Hongbo Zhang, Yanyan Zhao, and Bing Qin. An  
809 early evaluation of gpt-4v (ision). *arXiv preprint arXiv:2310.16534*, 2023b.
- 810  
811 Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Guohai Xu, Chenliang Li, Junfeng Tian,  
812 Qi Qian, Ji Zhang, Qin Jin, Liang He, Xin Lin, and Fei Huang. UReader: Universal OCR-free  
813 visually-situated language understanding with multimodal large language models. *Findings  
814 of the Association for Computational Linguistics: EMNLP 2023* Singapore, December 2023.  
815 Association for Computational Linguistics.

810 Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu  
811 Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal under-  
812 standing and reasoning benchmark for expert agpt. Proceedings of the IEEE/CVF Conference  
813 on Computer Vision and Pattern Recognition, pp. 9556–9567, 2024.

814 Yuhang Zang, Wei Li, Jun Han, Kaiyang Zhou, and Chen Change Loy. Contextual object detection  
815 with multimodal large language models. International Journal of Computer Vision, pp. 1–19,  
816 2024.

817 Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language  
818 image pre-training. In IEEE/CVF International Conference on Computer Vision, ICCV 2023,  
819 Paris, France, October 1-6, 2023, pp. 11941–11952. IEEE, 2023.

820 Jingyi Zhang, Jiaying Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A  
821 survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2024.

822 Yanzhe Zhang, Ruiyi Zhang, Jiuxiang Gu, Yufan Zhou, Nedim Lipka, Diyi Yang, and Tong  
823 Sun. Llavir: Enhanced visual instruction tuning for text-rich image understanding. CoRR,  
824 abs/2306.17107, 2023.

825 Bingchen Zhao, Yongshuo Zong, Letian Zhang, and Timothy M. Hospedales. Benchmarking multi-  
826 image understanding in vision and language models: Perception, knowledge, reasoning, and  
827 multi-hop reasoning. CoRR, abs/2406.12742, 2024.

828 Yilun Zhao, Yunxiang Li, Chenying Li, and Rui Zhang. Multihiertt: Numerical reasoning over multi  
829 hierarchical tabular and textual data. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio  
830 (eds.), Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics  
831 (Volume 1: Long Papers) ACL 2022, Dublin, Ireland, May 22-27, 2022, pp. 6588–6600. Associ-  
832 ation for Computational Linguistics, 2022.

833 Ge Zheng, Bin Yang, Jiajin Tang, Hong-Yu Zhou, and Sibeil Yang. Ddcot: Duty-distinct chain-  
834 of-thought prompting for multimodal reasoning in language models. Advances in Neural  
835 Information Processing Systems 36: Annual Conference on Neural Information Processing  
836 Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023, 2023.

837 Fengbin Zhu, Wenqiang Lei, Fuli Feng, Chao Wang, Haozhou Zhang, and Tat-Seng Chua. Towards  
838 complex document understanding by discrete reasoning. In MM'22: The 30th ACM International  
839 Conference on Multimedia, Lisboa, Portugal, October 10 - 14, 2022, pp. 4857–4866. ACM, 2022.

840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863

## A APPENDIX

### A.1 LEOPARD-INSTRUCT

To train LEOPARD, we created a large instruction-tuning dataset LEOPARD-INSTRUCT, with 925K instances, including 739K designed for text-rich, multi-image scenarios. Despite surveying existing datasets, we found only 154K suitable text-rich, multi-image samples – insufficient for effective instruction tuning, which is far from sufficient for effective instruction tuning, as shown in prior MLLM studies (Jiang et al., 2024; Laurin et al., 2024b; Li et al., 2024c). To overcome this limitation, we developed several data collection pipelines to collect high-quality text-rich, multi-image data, resulting in additional 185K instances.

Table 6 provides a detailed breakdown of the composition of the LEOPARD-INSTRUCT dataset. This table includes the name, domain, and sample size of sub-datasets. Additionally, it specifies how we construct multi-image samples, the number of images per sample, and the presence of rationales.

Table 6: Details of the constructed LEOPARD-INSTRUCT dataset. Images denotes the image number of one sample in each dataset.

Dataset	Domain	Multi-image	Images	Rationales	#Samples (K)
ArxivQA (Li et al., 2024d)	Doc	Reformed	1-3	Existing	81
DUDE (Landeghem et al., 2023)	Doc	Public	1-50	Augmented	23
MP-DocVQA (Tito et al., 2022)	Doc	Public	1-20	Augmented	36
DocVQA (Mathew et al., 2021)	Doc	No	1	None	39
TAT-DQA (Zhu et al., 2022)	Doc	Reformed	2-5	Augmented	13
SlidesGeneration (Seid et al., 2021)	Slides	Repurposed	1-20	Augmented	3
SlidesVQA (Tanaka et al., 2023)	Slides	Public	20	Augmented	10
Slideshare	Slides	Collected	2-8	Augmented	3
MultihierT (Zhao et al., 2022)	Table	Public	3-7	Existing/Augmented	15
MultiTabQA (Pal et al., 2023)	Table	Public	1-2	Augmented	6
TableGPT (Li et al., 2024e)	Table	Split	2	Existing	4
TabMWP (Lu et al., 2023)	Table	No	1	Existing	23
ChartGemma (Masry et al., 2024)	Chart	Reformed	1-4	Existing	65
DVQA (Kae et al., 2018)	Chart	Reformed	1-3	None	200
FigureQA (Kahou et al., 2018)	Chart	Reformed	1-2	None	36
ChartQA (Masry et al., 2022)	Chart	Reformed	2	Augmented	32
PewMultiChart	Chart	Collected	2	Augmented	20
Mind2Web (Deng et al., 2023)	Web	Split	1-5	None	7
WebsiteScreenshots (Aydos, 2020)	Web	No	1	Augmented	2
Omniaact (Kapoor et al., 2024)	Web	No	1	None	1
RICO (Hsiao et al., 2024)	Web	Reformed	1-4	None	25
WebVision (Li et al., 2017)	Web	No	1	Existing	1
WebUI (Wu et al., 2023a)	Web	No	1	None	19
LLaVAR (Zhang et al., 2023)	Mix	No	1	Existing	15
MathV360k (Shi et al., 2024)	Mix	No	1	None	38
Monkey (Li et al., 2024f)	Mix	Reformed	1-3	None	92
MPlugDocReason (Hu et al., 2024a)	Mix	No	1	Existing	25
IconQA (Lu et al., 2021)	Other	Public	1-6	Augmented	64
InfographicVQA (Mathew et al., 2022)	Other	No	1	Augmented	23
MapQA (Chang et al., 2022)	Other	Reformed	1-2	None	4
Total	-	-	-	-	925

We draw a chart to illustrate the data composition of LEOPARD-INSTRUCT dataset 4.

918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971

Figure 4: An illustration of the proportion of sub-datasets and domains in the proposed dataset.

## A.2 PROMPTS

We specify the prompt used during the data construction process as follows:

Figure 5: The prompt used for generating Q-A pairs with rationales for slide decks data.

972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016  
1017  
1018  
1019  
1020  
1021  
1022  
1023  
1024  
1025

Figure 6: The prompt used for generating Q-A pairs with rationales for webpage data.

Figure 7: We use this prompt for the generation of chain-of-thought rationales given original question, answer, and images.

### A.3 DETAILS OF TABLE RENDERING

To convert the textual table dataset into a multimodal dataset, the JSON or DataFrame format data is transformed into tabular images using Python. We utilize three Python packages, `dataframeimage`<sup>6</sup>, `pandas`<sup>7</sup>, and `matplotlib`<sup>8</sup> with various styling to enhance the diversity of the rendered images. To ensure the clarity and legibility of the plotted images, the original data is filtered by excluding any tables that contain more than 20 rows. This threshold was set to maintain the recognizability of the resulting images.

### A.4 QUALITATIVE RESULTS

We show two examples to give an illustrative demonstration of the model's performance. As can be seen from Figure 8, LEOPARD can not only capture detailed data in multiple tables precisely but also perform cross-table calculations, therefore it can answer the complex question correctly. Another example is demonstrated in Figure 9, LEOPARD can accurately perceive the prominent information under a high-resolution four-page document, demonstrating effective text-rich abilities under multi-image scenarios.

<sup>6</sup>[https://github.com/dexplo/dataframe\\_image](https://github.com/dexplo/dataframe_image)

<sup>7</sup><https://pandas.pydata.org/>

<sup>8</sup><https://matplotlib.org/>

1026  
1027  
1028  
1029  
1030  
1031  
1032  
1033  
1034  
1035  
1036  
1037  
1038  
1039  
1040  
1041  
1042  
1043  
1044  
1045  
1046  
1047  
1048  
1049  
1050  
1051  
1052  
1053  
1054  
1055  
1056  
1057  
1058  
1059  
1060  
1061  
1062  
1063  
1064  
1065  
1066  
1067  
1068  
1069  
1070  
1071  
1072  
1073  
1074  
1075  
1076  
1077  
1078  
1079

**Image 1**

	For the years ended December 31, 2013	For the years ended December 31, 2012	For the years ended December 31, 2011
0			
1	Balance, beginning of period	\$325	\$434
2	Sales inducements deferred	—	7
3	Amortization — Unlock charge [1]	-72	-82
4	Amortization charged to income	-33	-34
5	Amortization charged to business dispositions [2]	-71	—
6	Balance, end of period	\$149	\$325

**Image 2**

**Image 3**

	Non-vested Units	Restricted Units (in thousands)	Weighted-Average Grant-Date Fair Value
0	Non-vested at beginning of year	309	25.08
1	Granted	—	—
2	Vested	-306	25.04
3	Forfeited	-3	28.99
4	Non-vested at end of year	—	\$—

**Image 4**

	December 31, 2013	December 31, 2013_1	December 31, 2012	December 31, 2012_1	
0	Amount	Percent	Amount	Percent	
1	Fixed maturities, available-for-sale ("AFS"), at fair value	\$62,357	79.2%	\$85,922	81.6%
2	Fixed maturities, at fair value using the fair value option ("FVO")	844	1.1%	1,087	1.0%
3	Equity securities, AFS, at fair value	868	1.1%	890	0.8%
4	Mortgage loans	5,598	7.1%	6,711	6.4%
5	Policy loans, at outstanding balance	1,420	1.8%	1,997	1.9%
6	Limited partnerships and other alternative investments	3,040	3.9%	3,015	2.9%
7	Other investments [1]	521	0.7%	1,114	1.1%
8	Short-term investments	4,008	5.1%	4,581	4.3%
9	Total investments excluding equity securities, trading	78,656	100%	105,317	100%
10	Equity securities, trading, at fair value [2]	19,745		28,933	
11	Total investments	\$98,401		\$134,250	

Question What is the average value of Fixed maturities, available-for-sale ("AFS"), at fair value for Amount and Balance, beginning of period in 2013 ?

LEOPARD: To find the average value of Fixed maturities, available-for-sale ("AFS"), at fair value and Balance, beginning of period in 2013, we need to follow these steps:

- Identify the values from the images:
  - Fixed maturities, available-for-sale ("AFS"), at fair value for Amount in 2013: \$62,357 (from the fourth image).
  - Balance, beginning of period in 2013: \$325 (from the first image).
- Add these two values together:
 
$$[62,357 + 325 = 62,682]$$
- Divide the sum by 2 to find the average:
 
$$[\frac{62,682}{2} = 31,341]$$

Thus, the average value of Fixed maturities, available-for-sale ("AFS"), at fair value and Balance, beginning of period in 2013 is \$31,341.

Answer: 31341.0.

Figure 8: An example of multi-table reasoning of LEOPARD.

1080  
1081  
1082  
1083  
1084  
1085  
1086  
1087  
1088  
1089  
1090  
1091  
1092  
1093  
1094  
1095  
1096  
1097  
1098  
1099  
1100  
1101  
1102  
1103  
1104  
1105  
1106  
1107  
1108  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1120  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1130  
1131  
1132  
1133

Multi-Page Document

**Question:** Who is the author of the book Design And Engineering Of Production Systems?

**LEOPARD:** The author of the book 'Design and Engineering of Production Systems' is listed on the first page under the title 'Professors' Examination Problems and Solutions'. The name 'Farhad Azadian' is mentioned as the author. The answer is Farhad Azadian

Figure 9: An example of multi-page document question answering of LEOPARD.