DRAGONFLY: MULTI-RESOLUTION ZOOM-IN ENCODING ENHANCES VISION-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advancements in vision-language models (VLMs) have highlighted the benefits of processing images at higher resolutions and leveraging multi-crop features to retain native resolution details. However, current vision transformers (ViTs) often struggle to capture fine-grained details from non-dominant objects, charts, and embedded text, limiting their effectiveness in certain tasks. In this paper, we push beyond the conventional high-resolution and multi-crop techniques by not only preserving but also zooming in past the native resolution of images and extracting features from a large number of image sub-crops. This enhancement allows our model to better extract fine-grained details, overcoming the limitations of current ViTs. To manage the increased token count and computational complexity, we show that a simple mean-pooling aggregation over tokens is effective. Our model, Dragonfly[1], achieves competitive performance on general tasks such as ScienceQA and AI2D, and excels in tasks requiring fine-grained image understanding, including TextVQA and ChartQA. On average, across ten general-domain benchmarks, Dragonfly ranks at the top, outperforming models that are significantly larger or trained on much larger datasets. Notably, Dragonfly sets new benchmarks on several biomedical tasks, achieving 91.6% accuracy on the SLAKE (compared to 84.8% for Med-Gemini) and a 67.1% token F1 score on Path-VQA (compared to 62.7% for Med-PaLM M). On biomedical image captioning tasks, Dragonfly attains state-of-the-art results majority of the performance metrics. Overall, our work highlights the persistent challenge of engineering visual representations with fixed-resolution ViTs, and proposes a simple yet effective solution to address this issue and boost performance in both general and specialized domains.

1 INTRODUCTION

Recent advances in Vision-Language Models (VLMs) have highlighted the critical role of effectively integrating visual data into Large Language Models (LLMs). These models, especially those emphasizing visual instruction alignment, map rich, real-world visual data into the latent space of LLMs using sophisticated image encoding techniques. This process typically involves dividing images into patch-level tokens through powerful image encoders, which are then aligned with the LLM during visual instruction-tuning (Liu et al., 2023b;a; Yang et al., 2023; Li et al., 2023b; Xu et al., 2023; McKinzie et al., 2024a; Laurençon et al., 2024; You et al., 2023; Zhang et al., 2024).

Early VLMs processed images at fixed, low resolutions, requiring high-resolution images to be downsampled to fit model input dimensions. This downsampling often causes shape distortion, loss of fine details, and reduced overall visual richness—especially for tasks that demand fine-grained visual understanding. However, recent works have demonstrated the benefits of using higher-resolution encoders, where leveraging high-resolution inputs improves performance across various tasks (Bai et al., 2023b; Zhang et al., 2024; Chen et al., 2023c; Laurençon et al., 2024; McKinzie et al., 2024a). Moreover, approaches like Llava-1.5 (Liu et al., 2023a) and Llava-UHD (Xu et al., 2023) incorporate multi-crop techniques, allowing models to handle images at or close to their native resolution. This aligns with the conventional wisdom in computer vision that preserving images near their original resolution retains crucial information, which is vital for tasks requiring fine-grained visual understanding, such as text recognition in charts or other dense visual content.

[1]Upon acceptance, we will open-source our instruction-tuning dataset, model, and codebase.

In this paper, we extend this high-resolution encoding approach by introducing a novel strategy: featurizing images with multi-crops that exceed their native resolution. By zooming in at this level, we aim to mitigate limitations in existing Vision Transformers (ViTs), particularly their difficulty in extracting fine-grained details from non-dominant objects, charts, and embedded text (Li et al., 2023a; Bai et al., 2023b; Hong et al., 2024; Ye et al., 2023). While one might expect that zooming beyond native resolution adds no additional information and should not help if ViTs are functioning perfectly, in practice, they often miss subtle image details. As a result, zooming in helps capture information that ViTs currently struggle to extract. However, this high-resolution zoom-in and multi-crop method introduces a new challenge: the number of image tokens increases drastically with higher resolutions and more crops, significantly raising context length and computational demands. For instance, an image with a resolution of 336x336 is converted into 576 visual tokens using a CLIP-ViT-L/14 architecture (Radford et al., 2021). With five such image crops, this number already exceeds 2,800 tokens (Liu et al., 2023a). To manage this token complexity, we adopt a simple mean-pooling strategy for each high-resolution zoomed-in crop. Empirically, we find that this straightforward method—compressing visual tokens via mean pooling—strikes the best balance between computational efficiency and feature preservation. Although more advanced token-reduction methods (e.g., learnable approaches) may perform better with larger datasets, our experiments in the supervised fine-tuning setting show that mean pooling consistently delivers strong results across both general and biomedical benchmarks.

In summary, **our contributions** are as follows:

- We introduce Dragonfly, a new large VLM that processes images using multiple image crops that zoom beyond native resolution. By employing simple mean-pooling aggregation on high-resolution crops, Dragonfly efficiently reduces visual token counts while preserving fine-grained image details, all without the need for extensive pretraining. Dragonfly excels performance on general-domain benchmarks such as ScienceQA and AI2D, and performs especially well in tasks requiring fine-grained image understanding, like ChartQA and TextVQA. Among models in the 7-8B parameter range, Dragonfly ranks highest on average across ten evaluated benchmarks, outperforming even larger models or those trained on significantly more data.
- We highlight the model's strong performance on biomedical tasks, where detailed comprehension of high-resolution images is critical. Fine-tuned on a biomedical instruction-tuning dataset, Dragonfly achieves state-of-the-art or competitive results across benchmarks such as VQA, image captioning, and radiology report generation. Notable outcomes include 91.6% accuracy on SLAKE, a 67.1 token F1 score on Path-VQA, and a 50.9 CIDEr score on MIMIC-CXR captioning—these are the highest reported numbers to the best of our knowledge.
- We curate a dataset of 2.4 million supervised finetuning samples for the general domain and 1.4 million for the biomedical domain. While most of the data are publicly available, we carefully balanced and deduplicated the dataset across multiple tasks and image modalities (for the biomedical domain), which we believe will be beneficial to the community. Upon acceptance, we will release both instruction-tuning datasets, along with our training and evaluation code, and the fine-tuned models for both general and biomedical domains.

2 Related Work

Large Multimodal Models (LMMs) The advancement of large multimodal models (LMMs) has greatly impacted vision-language research by enabling the integration of visual information into large language models (LLMs). Methods such as visual feature alignment have become essential for merging vision and language through visual instruction-tuning (Liu et al., 2023b;a; Dai et al., 2023; Yang et al., 2023; Li et al., 2023b; Xu et al., 2023; McKinzie et al., 2024a; Laurençon et al., 2024; You et al., 2023; Awadalla et al., 2023). For instance, Liu et al. (2023b) employs a fully connected layer to project image embeddings, generated by a pretrained CLIP encoder (Radford et al., 2021), into the embedding space of a large language model. Despite these successes, many models downscale input images to fixed, low resolutions, which sacrifices fine visual details—particularly problematic in domains like biomedicine, where high-resolution image inputs are crucial for understanding intricate visual details (McKinzie et al., 2024a; Laurençon et al., 2024).

Handling High-Resolution Inputs and Capturing Fine-Grained Details Handling high-resolution inputs in vision-language models presents significant challenges, particularly due to the exponential growth in image tokens that increases computational demands. For instance, a 336x336 resolution image produces 576 visual tokens in a CLIP-ViT-L/14 architecture, and with multiple crops, this number can exceed 2,800 tokens (Liu et al., 2023a). Several approaches, such as Xu et al. (2023), have attempted to mitigate this by segmenting native-resolution images into smaller slices to retain detailed visual information while maintaining computational feasibility. Similarly, curriculum learning approaches like Qwen-VL (Bai et al., 2023b), PaLI-3 (Chen et al., 2023c), and PaLI-X (Chen et al., 2023c) have been explored to gradually scale input resolution, however, these methods still struggle with very large image sizes and require significant resources. Additionally, capturing fine-grained, local details—essential for tasks such as segmentation—remains a challenge for models like CLIP, which are trained on global image-level representations and often miss important regional semantics (Wu et al., 2023; Xu et al., 2022; Zhong et al., 2022). Although fine-tuning methods such as Rao et al. (2022) and Wang et al. (2022) have shown improvements in dense prediction tasks, these models still require substantial modifications to fully overcome limitations in locality and fine-grained detail capture. One potential way to overcome these limitations is to zoom in beyond the native resolution of an image, which enables models to extract even finer details that may not be fully captured at standard resolutions. By focusing on smaller regions of the image at higher magnification, this approach helps to compensate for the shortcomings of current ViTs in capturing localized and intricate features. To the best of our knowledge, no prior work has systematically explored the benefits of zooming in beyond an image's native resolution.

Biomedical Applications of LMMs LMMs have shown considerable promise in biomedical applications, where detailed comprehension of high-resolution image regions is critical. Models such as BiomedGPT (Zhang et al., 2023a) and LLaVA-Med (Li et al., 2024a) integrate medical imaging and literature to address specialized tasks in the biomedical domain. General-purpose models like Med-PaLM (Tu et al., 2024), Med-Flamingo (Moor et al., 2023), and Med-Gemini (Saab et al., 2024) have also been adapted for medical applications, showcasing the potential of LMMs to tackle complex vision-language tasks. Our work builds on studies such as McKinzie et al. (2024a) and Laurençon et al. (2024), focusing on visual instruction-tuning and efficient high-resolution image processing.

3 Dragonfly Architecture

We introduce our multi-resolution visual encoding approach and the strategies employed to manage the large number of visual tokens resulting from it. The workflow of our architecture is illustrated in Figure 1.

3.1 Multi-resolution Visual Encoding

We employ a multi-resolution visual encoding strategy using a shared image encoder trained on a fixed resolution of $R \times R$. Following techniques from previous works (Liu et al., 2023a; Xu et al., 2023), our framework processes larger images by dividing them into multiple sub-images, each matching the encoder's native resolution. Specifically, given an image I, we resize it into three distinct resolutions: a low-resolution image I^l of size $R \times R$, a medium-resolution image I^m of size $x^m R \times y^m R$, and a high-resolution image I^h of size $x^h R \times y^h R$. The medium- and high-resolution images are then divided into sub-images, resulting in two sets of sub-images, $\{I_i^m\}_{i=1}^{x^m \times y^m}$ and $\{I_j^h\}_{j=1}^{x^h \times y^h}$, with each sub-image aligned to the encoder's training resolution $R \times R$. We adopt the any-resolution segmentation method from Xu et al. (2023) to divide images into sub-images. This method selects a resolution grid from a predefined set of grids that closely match the original image's aspect ratio. For medium resolution, the possible grids are $\{(2, 2), (1, 4), (4, 1)\}$, resulting in four sub-images. For high resolution, we use the grids $\{(6, 6), (3, 12), (12, 3)\}$, producing 36 sub-images in total.

The image encoder encodes each sub-image into a sequence of visual tokens $\{v_1, \ldots, v_n\}$. These tokens, extracted from the various sub-images, are projected into the latent space of the language model via a projection layer P, generating a corresponding sequence of projected tokens $\{t_1, \ldots, t_n\}$. The projected tokens from different sub-images are concatenated to form a comprehensive representation of the image, which is then used for understanding by the LLM. However, due to the large number of

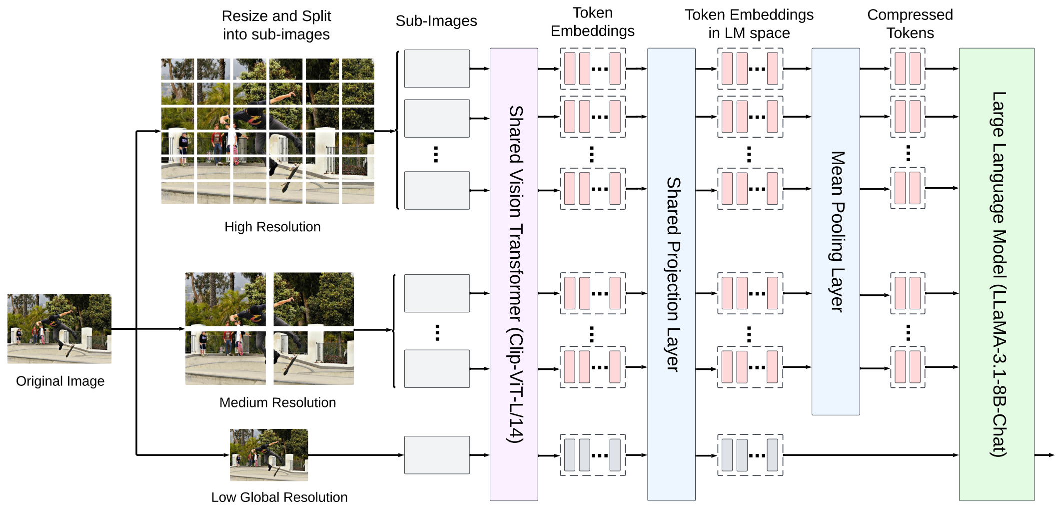

Figure 1: Overview of our proposed Dragonfly framework. The original image is resized into low, medium, and high resolutions. The medium- and high-resolution images are divided into sub-images, matching the encoder's training resolution. All sub-images pass through a shared vision encoder to produce visual tokens. The projection layer then maps the visual tokens to the language space. Afterward, the mean-pooling layer reduces the embeddings from each sub-image into 36 tokens.

sub-images, especially from the high-resolution set, incorporating all these sub-images can result in longer context lengths and introduce noise during training. In the following sections, we discuss strategies to mitigate these challenges.

3.2 Token Aggregation

We adopt a simple mean pooling strategy to reduce the number of visual tokens while still leveraging high-resolution images. All images are resized to 336×336 and processed using the CLIP-ViT-L/14 model, which outputs 576 tokens. For the low-resolution image, we retain all 576 tokens. For the medium- and high-resolution images, the image is divided into 40 sub-images (4 for medium resolution and 36 for high resolution). Each sub-image passes through the image encoder, producing 576 tokens, which are reshaped into a 24×24 token grid. We then apply mean pooling to this grid, reducing it to a 6×6 grid using a sliding window of size 4 with a stride of 4, resulting in 36 tokens per sub-image. All 40 sub-images are then concatenated, with separator tokens placed between them, forming the complete image representation. This results in 576 tokens from the low-resolution image, 4×36 tokens from the medium resolution, and 36×36 tokens from the high resolution, yielding a total of 2,016 image tokens.

4 Experiments

In this section, we first introduce our implementation and experimental setup. We then present ablations and baseline comparisons to validate our design choices. Next, we evaluate Dragonfly against other models of similar scale across multiple general-domain benchmarks. Finally, we continue training Dragonfly on a biomedical dataset, resulting in Dragonfly-Med, and assess its performance on biomedical tasks.

4.1 Implementation

Dragonfly uses Llama3.1-8B-chat (Meta AI, 2024) as the backbone and CLIP-ViT-L/14 (Radford et al., 2021) as the image encoder. CLIP-ViT-L/14 accepts images with a resolution of 336×336, and our highest resolution is either 2016×2016 or 1008×4032, depending on the native aspect ratio of the image. An analysis of the resolutions across our training data revealed that these high resolutions cover approximately 99.5% of images at their native resolution. Additionally, after applying the

Dragonfly multi-crop zoom-in method, 95% of the images are zoomed in by at least 2x, and 65% are zoomed in by at least 4x. A cumulative density plot of the ratio between our high-resolution images and their native resolution is provided in Supplementary Figure 4.

For training Dragonfly, we adopt the two-stage visual instruction-tuning framework introduced by Liu et al. (2023b). In the first stage, the LLM and vision encoder are frozen, with only the projection layer being trained. This stage allows the projection layer to effectively learn how to map visual tokens into the language space while preserving the pre-established alignment of the LLM. The model is trained for one epoch on the LLaVA-Pretrain dataset (Liu et al., 2023b), which consists of 558K image-text pairs, using a global batch size of 64 and a learning rate of 2e-5.

In the second stage, the entire model undergoes fine-tuning on a high-quality visual instruction-tuning dataset. This step is crucial for further aligning visual features with the language space, thereby optimizing the model's performance in vision-language tasks. For this supervised fine-tuning, we curated a dataset comprising 2.4M image-instruction samples from various sources, which include detailed image descriptions, complex reasoning tasks, and question-answering tasks. Further details about this instruction-tuning dataset are provided in Appendix Sections B and D. The model is trained for one epoch with a global batch size of 16, with a learning rate of 2e-6.

Stage 1 training lasted approximately 4 hours, and Stage 2 training lasted 32 hours on 3 nodes of 8 NVIDIA H100 GPUs, utilizing DeepSpeed ZeRO for distributed training. More details about our training hyperparameters are presented in Supplementary Table 10.

Before presenting our main results, we first validate our design choices by conducting multiple ablations, comparing them against baselines and alternative token reduction strategies.

4.2 Ablation 1: Is Mean-Pooling an Effective Token Reduction Strategy?

Training all baseline models on the full 2.4M instruction-tuning dataset is too time-prohibitive. Therefore, we randomly sample 700K samples from our supervised finetuning mixture and use this reduced dataset to fine-tune all baseline models. All hyperparameters are the same as in the main experiments, as discussed in Section 4.

Table 1: Performance comparison of multiple token reduction strategies for encoding high-resolution images against Dragonfly. The first model is our implementation of LLaVA-1.5-HD, which uses CLIP-ViT-L/14 for both low and medium resolutions, producing 2,880 image tokens. The second model, LLaVA-UHD, results in a variable number of image crops based on the original image size, with each crop producing 64 tokens. The total number of tokens for LLaVA-UHD is therefore variable, with a maximum of 6 crops allowed, resulting in a maximum of 384 image tokens. The third model uses CLIP-ViT-L/14 for low resolution and CLIP-ViT-B/32 for medium and high resolutions, generating 2,576 image tokens. The fourth model is similar to Dragonfly but uses the IDEFICS Perceiver Resampler to reduce the number of tokens to match ours (2,016). All models share the same LLM backbone, LLaMA-3.1-8B-chat, and are trained on the same dataset.

Benchmark	LLaVA-1.5-HD	LLaVA-UHD	Dual Encoder	Perceiver Resampler	Dragonfly
AI2D	63.8	59.9	61.7	60.4	**64.2**
ScienceQA	79.3	76.3	79.5	70.0	**79.7**
ChartQA	54.0	37.2	36.6	48.0	**56.4**
POPE-f1	85.7	85.3	86.2	84.4	**87.7**
GQA	54.1	51.0	51.8	53.4	**55.7**
TextVQA	64.0	51.5	48.5	52.6	**66.5**
VizWiz	56.1	51.8	60.4	56.8	**61.7**
MME	1414.0	1302.1	1314.9	1385.3	**1438.9**

We experimented with multiple alternative token reduction strategies to compare against our mean pooling approach. The first model, **Dual-Encoder**, processes the low-resolution image using the CLIP-ViT-Large model, while the medium- and high-resolution sub-images are handled by the CLIP-ViT-Base model, each resized to 224×224 and generating 49 tokens per sub-image. Both encoders use their own single-layer modality projection. This configuration produces a total of 2,536 image tokens.

Table 2: Ablation study results evaluating the impact of different image resolutions on model performance across multiple benchmarks. The table compares the performance of Dragonfly using low (L), medium (M), and high (H) resolutions individually, as well as in various combinations.

Metric	L	M	H	L + M	L + H	L + M + H
AI2D	60.6	61.8	60.4	**64.5**	63.6	64.2
ScienceQA	76.0	76.2	76.0	79.2	79.0	**79.7**
ChartQA	21.6	48.4	54.1	52.9	**56.2**	**56.2**
Pope-f1	82.2	87.1	86.0	87.5	**87.7**	**87.7**
GQA	49.5	53.1	52.9	54.6	55.2	**55.7**
TextVQA	40.0	55.0	56.4	60.9	65.2	**66.5**
VizWiz	57.4	59.9	56.0	58.7	59.7	**61.7**
MME	1205.3	1311.6	1364.0	1227.4	1397.8	**1438.9**

The second model, **Perceiver Resampler**, follows a similar structure to Dragonfly, but replaces the mean pooling layer with the IDEFICS implementation of the Perceiver Resampler (Alayrac et al., 2022). This resampler uses a depth of 3 and 36 latents, resulting in a total of 2,016 tokens—matching our token count. Additionally, we implemented our own version of **LLaVA-1.5-HD** (Liu et al., 2023a) and **LLaVA-UHD** (Xu et al., 2023) using the same ViT and LLM backbone as our model. These two are our closest baselines. LLaVA-1.5-HD processes low- and medium-resolution images and generates a total of 2,880 visual tokens, whereas, LLaVA-UHD process the images at their native resolution and generates at max 6 crops from the image, each of which generates 64 tokens. At max, LLaVA-UHD can generate 384 tokens.

Table 1 presents the results of these baselines. Empirically, we found that the mean pooling strategy consistently outperformed other methods across all benchmarks, demonstrating particularly strong performance in tasks requiring fine-grained visual detail, such as TextVQA and ChartQA. Notably, Dragonfly outperforms LLava-1.5-HD and LLaVA-UHD on all benchmarks. While advanced token-reduction methods like the Perceiver Resampler also performed well, the simplicity and effectiveness of mean pooling—combined with a robust vision encoder and high-resolution inputs—proved to be the most efficient approach in this supervised fine-tuning setting.

4.3 Ablation 2: How important are each image resolution?

To evaluate the impact of image resolution on downstream performance, we trained four separate models using different combinations of image resolutions. For low resolution, we used all 576 tokens; for medium resolution, 4×36 tokens; and for high resolution, 36×36 tokens. The results, as presented in Table 2, provide several key insights into the role of image resolution. First, models utilizing medium or high-resolution images generally outperform those relying solely on low-resolution inputs across most benchmarks, underscoring the significance of higher resolutions in capturing fine-grained visual details. Additionally, combining low resolution with medium or high resolution consistently performs better than using any individual resolution, particularly on tasks such as ChartQA and TextVQA. This indicates that blending global context from low-resolution images with detailed regional features from medium or high-resolution images is especially effective for tasks requiring both broad contextual understanding and fine-grained detail recognition. The best overall performance is achieved by integrating all three resolutions (low + medium + high), which yields the highest scores across most benchmarks, emphasizing the value of leveraging a full spectrum of image resolutions.

4.4 Ablation 3: Disentangling Resolution and Multi-Crop Benefits

Our previous results demonstrate improved performance from our multi-resolution encoding strategy. However, it remains unclear whether these gains are primarily due to the higher image resolution preserving more information or the multi-crop approach generating separate features for each sub-image. While our method provides both benefits over a single-crop, fixed-resolution approach, we now conduct an experiment to disentangle their relative importance. Specifically, we test: 1) the effect of generating multi-crop features from an image *already downsized to low resolution*, which limits the ability to preserve extra raw image information compared to the standard single-resolution approach, and 2) the effect of generating multi-crop features from an image that *retains its native*

Table 3: Ablation study results evaluating the impact of zooming in. The table compares performance using low resolution and medium resolution, pooled down to 576 tokens, with versions starting from the low-resolution image and starting from the native-resolution image.

Metric	**Low-Resolution**	**Medium-Resolution from Low-Resolution**	**Medium-Resolution from Native-Resolution**
AI2D	60.6	62.9	61.7
ScienceQA	76.0	77.6	76.9
ChartQA	21.6	52.4	56.6
POPE	83.4	85.1	86.8
GQA	49.5	54.7	54.9
TextVQA	40.0	57.4	61.2
VizWiz	57.4	58.0	56.7
MME Perception	1205.3	1398.9	1444.7

resolution, allowing it to preserve more raw image information than both the standard low resolution approach and 1).

For the first experiment, we rescaled all images to a low resolution of 336×336, with the low-resolution performance consistent with Table 2. From this baseline, we conducted an experiment where we zoomed in $2\times$, generating images of size 672×672 and producing four crops from the rescaled image. Each crop was passed through the ViT, generating 576 tokens (24×24), which we then pooled down to 144 tokens per crop, for a total of 576 tokens across all crops. This matches the total token count of the low-resolution model. In Table 3, this represents the column "Medium-Resolution from Low-Resolution", and it outperforms the "Low-Resolution" model in all benchmarks, particularly excelling in tasks like ChartQA and TextVQA, where localized information is critical. This suggests that the multi-crop approach itself, even without preserving additional raw image information, significantly contributes to improved performance, likely by enabling more focused processing of image sub-regions.

For the second experiment, without first rescaling to low resolution, we worked directly from the native-resolution image and resized it to 672×672, producing four crops from the resized image. Each crop was passed through the ViT, generating 576 tokens (24×24), which we then pooled down to 144 tokens per crop, for a total of 576 tokens across all crops. In Table 3, this represents the column "Medium-Resolution from Native-Resolution." There are two key observations here. First, as expected from previous results, this model outperforms the "Low-Resolution" baseline across all tasks. Second, it also outperforms the "Medium-Resolution from Low-Resolution" model on a majority of the tasks (5/8), highlighting the importance of preserving raw image information. However, these results indicate that most of the performance gains come from featurizing sub-crops, which remains the most important part of our approach.

4.5 Main Results

Table 4: Comparison of Dragonfly with existing Language-Image Multimodal Models (LMMs) across various benchmarks. The best performance is indicated in **bold**, while the second-best is underlined.

Model	**Backbone**	**#Data**	**VQA^{v2}**	**VQA^{T}**	**POPE**	**SQA**	**VizWiz**	**AI2D**	**ChartQA**	**MME**	**MMB/MMB^{CN}**
InstructBLIP	Vicuna-7B	130M	-	50.1	-	60.5	34.5	-	-	-	36.0/23.7
Qwen-VL-Chat	Qwen-7B	1.4B	78.2	61.5	-	68.2	38.9	62.3	65.7	1487.5	60.6/56.7
LLaVA-1.5	Vicuna-7B	1.2M	78.5	58.2	85.9	66.8	50.0	54.8	18.2	1510.7	63.4/58.3
VILA	Llama2-7B	61M	79.9	64.4	85.5	68.2	57.8	-	-	1533.0	68.9/61.7
LLaVA-NeXT	Vicuna-7B	1.2M	81.8	64.9	86.5	70.1	57.6	66.6	54.8	1519.0	67.4/60.6
MM1-7B-Chat	MM1-7B	>2B	**82.3**	72.8	86.6	72.6	45.3	-	-	1529.3	72.3/-
mPLUG-Owl2	Llama2-7B	401M	79.4	58.2	86.2	68.7	54.5	-	-	1450.2	63.5/-
Monkey	Qwen-7B	1B	80.3	-	67.6	69.4	**61.2**	62.6	65.1	-	-
SPHINX	Llama2-7B	1B	78.1	51.6	80.7	69.3	39.9	-	-	1476.1	66.9/56.2
SPHINX-2k	Llama2-7B	1B	80.7	61.2	87.2	70.6	44.9	-	-	1470.7	65.9/57.9
ShareGPT4V-7B	Vicuna-7B	1.8M	80.6	-	-	68.4	57.2	-	-	**1567.4**	68.8/62.2
VisionLLM v2-chat	Vicuna-7B	22M	81.4	66.3	87.5	**94.4**	54.6	-	-	1512.5	**77.1/67.6**
InternVL-7B	Vicuna-7B	>28.7B	79.3	57.0	86.4	66.2	52.5	-	-	1525.1	64.6/57.6
Dragonfly (Ours)	Llama3-8B	2.9M	81.0	**73.6**	**87.9**	79.5	59.0	**67.9**	**71.2**	1538.1	71.9/66.1

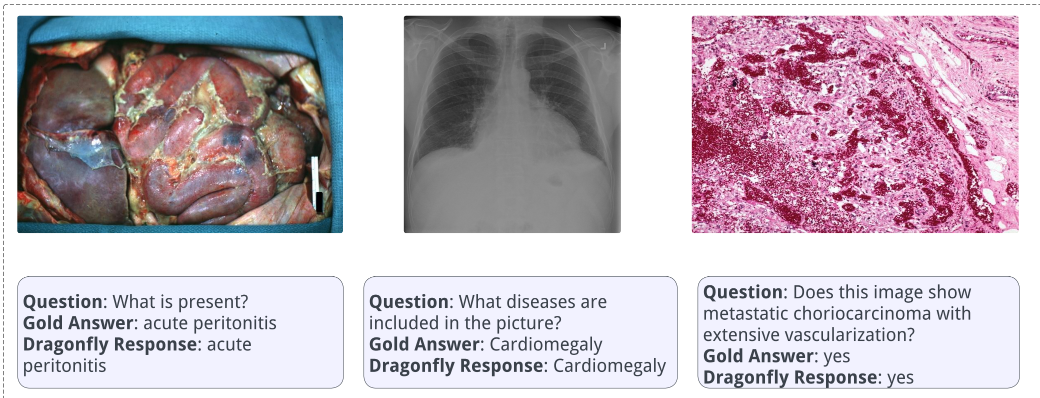

Figure 2: Examples of Biomedical Visual Question Answering (VQA). The figure shows three questions along with their gold standard answers and the corresponding responses from the Dragonfly-Med model.

Table 4 presents the performance of Dragonfly across multiple benchmarks in comparison to other off-the-shelf VLMs. We evaluate the models across ten established benchmarks, including general visual question answering datasets (ScienceQA (Lu et al., 2022), VQA$^{v^2}$ (Antol et al., 2015), VizWiz (Gurari et al., 2018)), chart interpretation and OCR-based VQA datasets (ChartQA (Masry et al., 2022) and TextVQA (Singh et al., 2019)), hallucination assessment datasets (POPE (Yifan et al., 2023)), and other standard benchmarks such as AI2D (Kembhavi et al., 2016), MME (Fu et al., 2023), MMB (Liu et al., 2023c), and MMB^{CN}, which is the chinese version of MMB.

One of the key areas where Dragonfly excels is in tasks that require fine-grained visual understanding, such as TextVQA and ChartQA. For instance, Dragonfly achieves a score of 73.6 on TextVQA and 71.2 on ChartQA, outperforming all other models in the table. By comparison, Qwen-VL-Chat Bai et al. (2023b), trained on over 400 times more data, achieves only 61.5 on TextVQA and 65.7 on ChartQA. This result aligns with previous research (Beyer et al., 2024), which emphasizes the importance of high-resolution images for tasks involving intricate visual details, such as text recognition and chart interpretation.

In addition to these tasks, Dragonfly achieves best performance on POPE-f1 (87.9) and ranks second-best on VizWiz (59.0), MME (1538.1), ScienceQA (79.5), and MMB^{CN} (66.1). The models that often outperform Dragonfly on certain benchmarks, such as MM1-7B-Chat, and Monkey Li et al. (2024c), are trained on significantly larger datasets, with over 1 billion samples.

As shown in Supplementary Table 14, Dragonfly competes strongly against 13B-17B models across various benchmarks. It outperforms all comparable 13B models on TextVQA, ChartQA, and MMB^{CN}, while also achieving second-best performance on POPE, ScienceQA, VizWiz, AI2D, and MMB, competing against powerful models such as CogVLM-17B-Chat Wang et al. (2023a). This underscores Dragonfly's efficiency in leveraging high-resolution, zoomed-in image features and a powerful visual encoder without requiring extensive pretraining data.

4.6 Biomedical Domain Adaptation

We employed a domain adaptation strategy to evaluate our model's ability to specialize to the biomedical domain and assess its fine-grained image understanding. Starting with a model checkpoint instruction tuned on a general domain dataset, we implemented a three-step training process tailored specifically for the biomedical domain to create Dragonfly-Med.

The first stage involved tuning the vision encoder, which is critical given the limited exposure of the standard CLIP vision encoder to biomedical images. The training dataset for this phase primarily comprised short caption datasets from sources like LLaVA-Med (Li et al., 2024a), Openpath (Huang et al., 2023a), and MedICaT (Subramanian et al., 2020), supplemented by general domain datasets from LLaVA-Pretrain (Liu et al., 2024c). This phase included approximately 1.16 million image-text pairs, split roughly evenly between the general and biomedical domains. Stage 1 took approximately 24 hours to train on 8 NVIDIA H100 GPUs.

In the second stage, we jointly trained the vision encoder, language model, and projection layer. We used a diverse set of datasets, including LLaVA-Med-Instruct (Li et al., 2024a), MIMIC-III-CXR (Johnson et al., 2019), Openpath (Huang et al., 2023a), ROCO (Pelka et al., 2018), Kaggle DR, and DDR (Li et al., 2019). Additionally, we included training sets from benchmark datasets such as VQA-RAD (Lau et al., 2018), SLAKE (Liu et al., 2021), Path-VQA (He et al., 2020), IU X-Ray, and Peir Gross (Demner-Fushman et al., 2016). The dataset totaled 723K image-text pairs, with approximately 15% from the general domain and 85% from the biomedical domain. General domain datasets included SVIT Zhao et al. (2023b), ShareGPT4V Chen et al. (2023b), and ArXivCap Li et al. (2024b). Stage 2 took about 30 hours on 8 NVIDIA H100 GPUs.

The final stage involved supervised finetuning using combined training datasets from our benchmark tasks: VQA-RAD, SLAKE, Path-VQA, IU X-Ray, Peir Gross, and subsets of ROCO and MIMIC-CXR. We finetuned a single model end-to-end on this aggregated training data to optimize performance across all tasks simultaneously. Stage 3 required approximately 4 hours of training on 8 NVIDIA H100 GPUs.

Table 5: Medical image captioning and clinical report generation evaluation results. For MIMIC-CXR, we specifically focus on generating the findings section of the radiology report.

Dataset	Metric	BiomedGPT	SOTA	Dragonfly-Med (Ours)
IU X-Ray	ROUGE-L	28.5	44.8 (Zhou et al., 2021)	29.1
	METEOR	12.9	24.2 (Huang et al., 2023b)	**30.5**
	CIDEr	40.1	43.5 (Wang et al., 2023b)	**61.7**
Peir Gross	ROUGE-L	36.0	36.0 (Zhang et al., 2023a)	**42.0**
	METEOR	15.4	15.4 (Zhang et al., 2023a)	**40.2**
	CIDEr	122.7	122.7 (Zhang et al., 2023a)	**198.5**
ROCO	ROUGE-L	18.2	18.2 (Zhang et al., 2023a)	**19.2**
	METEOR	7.8	7.8 (Zhang et al., 2023a)	**15.5**
	CIDEr	24.2	24.2 (Zhang et al., 2023a)	**45.2**
MIMIC-CXR	ROUGE-L	23.8	33.5 (Zhou et al., 2021)	25.2
	METEOR	14.2	19.0 (Zhou et al., 2021)	**23.6**
	CIDEr	14.7	**50.9** (Miura et al., 2020)	**50.9**

Table 6: Biomedical VQA evaluation results.

Dataset	Metric	LLaVA-Med	Med-Gemini	SOTA	Dragonfly-Med (Ours)
VQA-RAD	Acc (closed)	84.2	69.7	87.1 (Tanwani et al., 2022)	78.1
	Token F1	-	50.1	62.1 (Tu et al., 2024)	61.4
SLAKE	Acc (closed)	83.2	84.8	**91.6** (Yuan et al., 2023)	**91.6**
	Token F1	-	75.8	**89.3** (Tu et al., 2024)	**89.3**
Path-VQA	Acc (closed)	91.7	83.3	91.7 (Li et al., 2024a)	90.6
	Token F1	-	58.7	62.7 (Tu et al., 2024)	**67.1**

The results, as reported in Table 5 and 6, are based on this finetuned model and evaluated against the official held-out test sets of the respective benchmarks (details of the biomedical benchmarks are provided in Appendix Section E). For VQA tasks, we use accuracy and token-level F1 (Tu et al., 2024), while for image captioning and radiology report generation tasks, we use metrics such as ROUGE-L (Lin, 2004), METEOR (Banerjee & Lavie, 2005), and CIDEr (Vedantam et al., 2015). These metrics evaluate the fluency of text, the sequence of content, and the recognition of synonyms and word stems, with CIDEr specifically tailored for assessing text descriptions of images.

Dragonfly-Med achieves competitive performance across multiple benchmarks. On the image captioning task, Dragonfly-Med delivers state-of-the-art or competitive results on several metrics across these datasets. Notably, on the Peir Gross and ROCO datasets, Dragonfly-Med outperforms existing methods on all three metrics: ROUGE-L, METEOR, and CIDEr. On the other two captioning benchmarks (IU X-Ray and MIMIC-CXR), Dragonfly-Med achieves state-of-the-art performance on two out of three evaluation metrics. Some baseline models are significantly larger than our current implementation.

For VQA tasks, Dragonfly-Med attains an accuracy of 91.6% and a token F1 score of 89.3% on the SLAKE dataset, matching the current state-of-the-art. Similarly, on Path-VQA, Dragonfly-Med sets a new state-of-the-art performance with a token F1 score of 67.1, surpassing the much larger Med-PaLM-M model, which scores 62.7. Additionally, Dragonfly-Med consistently outperforms Med-Gemini, a significantly larger model, on all VQA tasks. These results further highlight the fine-grained understanding and reasoning capabilities of the Dragonfly-Med architecture for image region tasks. Figure 2 presents a few examples from our evaluation tasks, along with Dragonfly-Med's responses.

5 Discussion and Conclusion

High-resolution image inputs are crucial for capturing fine-grained visual details, particularly in tasks requiring complex understanding. Our study demonstrates that leveraging powerful vision encoders and pushing image resolutions beyond native sizes enhances the model's ability to identify subtle visual cues. Zooming in beyond native resolution allows the model to capture fine-grained details that might otherwise be missed, particularly in small objects, dense text, and intricate visual patterns. We show that a simple mean pooling strategy, when paired with high-resolution inputs, provides an effective and computationally efficient solution, preserving both global context and fine details. Dragonfly outperforms models using more complex reduction methods and even surpasses larger models in several benchmarks while utilizing fewer tokens and less data. The effectiveness of mean pooling likely lies in its simplicity: it distills redundant visual information and aggregates key features without introducing additional parameters or biases that might require extensive data to optimize. This non-parametric approach appears to be particularly advantageous in low-data regimes, where the limited supervision can hinder the training of parameter-heavy methods. By avoiding the complexities of learning a compression mechanism, mean pooling ensures a robust, data-efficient integration of high-resolution features, enabling better generalization with fewer resources.

Despite the strong performance of Dragonfly, there are several limitations to our approach. First, we only explored supervised fine-tuning and did not evaluate these strategies at the pretraining stage. Therefore, while our results show promise, we cannot make broad generalizations about the effectiveness of high-resolution, multi-crop techniques or mean pooling across other phases of training. Second, although we have demonstrated competitive performance using much smaller datasets than other models, it remains unclear whether our approach will continue to scale as effectively with larger supervised fine-tuning datasets. Further investigation is needed to determine whether the model's performance improvements hold up with increasing data volume. Third, while the increased resolution and multiple image crops enhance the model's visual understanding, they come at the cost of higher computational demands in the vision encoder. However, it is important to note that, compared to the LLM, the computational overhead in the ViT is relatively smaller. Moreover, by applying mean pooling, we ensure that the context length passed to the LLM remains manageable, helping mitigate the impact of these additional FLOPs. In future, we aim to scale up our fine-tuning dataset and explore the benefits of zoomed-in features more comprehensively.

Interestingly, the strong performance of our simple approach—zooming in beyond native resolution and mean pooling the tokens—highlights a broader issue: the fixed-resolution approach of current vision transformers is inherently limiting. While multi-crop strategies offer some improvement, they introduce complexity and increased computational demands. Moving forward, VLMs should adopt native-resolution architectures that can process images at various scales in a single pass, preserving all the information without requiring multiple crops. Additionally, improved training strategies are needed to ensure that models retain the same level of detail as if magnified sub-crops were processed individually.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, et al. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, 2023.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023a.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023b.

Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pp. 65–72, 2005.

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. Paligemma: A versatile 3b vlm for transfer. *arXiv preprint arXiv:2407.07726*, 2024.

Guiming Hardy Chen, Shunian Chen, Ruifei Zhang, Junying Chen, Xiangbo Wu, Zhiyi Zhang, Zhihong Chen, Jianquan Li, Xiang Wan, and Benyou Wang. Allava: Harnessing gpt4v-synthesized data for a lite vision-language model. *arXiv preprint arXiv:2402.11684*, 2024a.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. *arXiv preprint arXiv:2311.12793*, 2023a.

Lin Chen, Jisong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. *arXiv preprint arXiv:2311.12793*, 2023b.

Xi Chen, Josip Djolonga, Piotr Padlewski, Basil Mustafa, Soravit Changpinyo, Jialin Wu, Carlos Riquelme Ruiz, Sebastian Goodman, Xiao Wang, Yi Tay, et al. Pali-x: On scaling up a multilingual vision and language model. *arXiv preprint arXiv:2305.18565*, 2023c.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24185–24198, 2024b.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv preprint arXiv:2303.12345*, 2023. 2, 3.

Dina Demner-Fushman, Marc D Kohli, Marc B Rosenman, Sonya E Shooshan, Laritza Rodriguez, Sameer Antani, George R Thoma, and Clement J McDonald. Preparing a collection of radiology examinations for distribution and retrieval. *Journal of the American Medical Informatics Association*, 23(2):304–310, 2016.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3608–3617, 2018.

Xuehai He, Yichen Zhang, Luntian Mou, Eric Xing, and Pengtao Xie. Pathvqa: 30000+ questions for medical visual question answering. *arXiv preprint arXiv:2003.10286*, 2020.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, et al. Cogagent: A visual language model for gui agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14281–14290, 2024.

Zhi Huang, Federico Bianchi, Mert Yuksekgonul, Thomas Montine, and James Zou. Leveraging medical twitter to build a visual–language foundation model for pathology ai. *bioRxiv*, pp. 2023–03, 2023a.

Zhongzhen Huang, Xiaofan Zhang, and Shaoting Zhang. Kiut: Knowledge-injected u-transformer for radiology report generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19809–19818, 2023b.

Baoyu Jing, Pengtao Xie, and Eric Xing. On the automatic generation of medical imaging reports. *arXiv preprint arXiv:1711.08195*, 2017.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317, 2019.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. *arXiv preprint arXiv:1603:07396*, 2016.

Jason J Lau, Soumya Gayen, Asma Ben Abacha, and Dina Demner-Fushman. A dataset of clinically generated visual questions and answers about radiology images. *Scientific data*, 5(1):1–10, 2018.

Hugo Laurençon, Léo Tronchon, Matthieu Cord, and Victor Sanh. What matters when building vision-language models? *arXiv preprint arXiv:2403.11703*, 2024.

Bo Li, Peiyuan Zhang, Jingkang Yang, Yuanhan Zhang, Fanyi Pu, and Ziwei Liu. Otterhd: A high-resolution multi-modality model. *arXiv preprint arXiv:2311.04219*, 2023a.

Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. Otter: A multi-modal model with in-context instruction tuning. *arXiv preprint arXiv:2305.03726*, 2023b.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *Advances in Neural Information Processing Systems*, 36, 2024a.

Lei Li, Yuqi Wang, Runxin Xu, Peiyi Wang, Xiachong Feng, Lingpeng Kong, and Qi Liu. Multimodal arxiv: A dataset for improving scientific comprehension of large vision-language models. *arXiv preprint arXiv:2403.00231*, 2024b.

Tao Li, Yingqi Gao, Kai Wang, Song Guo, Hanruo Liu, and Hong Kang. Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening. *Information Sciences*, 501:511–522, 2019.

Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. Monkey: Image resolution and text label are important things for large multi-modal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26763–26773, 2024c.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26689–26699, 2024.

Ziyi Lin, Chris Liu, Renrui Zhang, Peng Gao, Longtian Qiu, Han Xiao, Han Qiu, Chen Lin, Wenqi Shao, Keqin Chen, et al. Sphinx: The joint mixing of weights, tasks, and visual embeddings for multi-modal large language models. *arXiv preprint arXiv:2311.07575*, 2023.

Bo Liu, Li-Ming Zhan, Li Xu, Lin Ma, Yan Yang, and Xiao-Ming Wu. Slake: A semantically-labeled knowledge-enhanced dataset for medical visual question answering. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pp. 1650–1654. IEEE, 2021.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *NeurIPS 2023*, 2023b.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26296–26306, 2024a.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge. *arXiv preprint arXiv:2401.12345*, 2024b. 7, 35, 36.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024c.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*, 2023c.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *The 36th Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. *ACL 2022 Findings*, 2022.

Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruti Shah, Xianzhi Du, Futang Peng, Floris Weers, Anton Belyi, Haotian Zhang, Karanjeet Singh, Doug Kang, Ankur Jain, Hongyu Hè, Max Schwarzer, Tom Gunter, Xiang Kong, Aonan Zhang, Jianyu Wang, Chong Wang, Nan Du, Tao Lei, Sam Wiseman, Guoli Yin, Mark Lee, Zirui Wang, Ruoming Pang, Peter Grasch, Alexander Toshev, and Yinfei Yang. Mm1: Methods, analysis & insights from multimodal llm pre-training. *arXiv preprint arXiv:2403.09611*, 2024a.

Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruti Shah, Xianzhi Du, Futang Peng, Floris Weers, et al. Mm1: Methods, analysis & insights from multimodal llm pre-training. *arXiv preprint arXiv:2403.09611*, 2024b.

Meta AI. Introducing meta llama 3: The most capable openly available llm to date. *https://ai.meta.com/blog/meta-llama-3/*, 2024.

Yasuhide Miura, Yuhao Zhang, Emily Bao Tsai, Curtis P Langlotz, and Dan Jurafsky. Improving factual completeness and consistency of image-to-text radiology report generation. *arXiv preprint arXiv:2010.10042*, 2020.

Michael Moor, Qian Huang, Shirley Wu, Michihiro Yasunaga, Yash Dalmia, Jure Leskovec, Cyril Zakka, Eduardo Pontes Reis, and Pranav Rajpurkar. Med-flamingo: a multimodal medical few-shot learner. In *Machine Learning for Health (ML4H)*, pp. 353–367. PMLR, 2023.

Obioma Pelka, Sven Koitka, Johannes Rückert, Felix Nensa, and Christoph M Friedrich. Radiology objects in context (roco): a multimodal image dataset. In *Intravascular Imaging and Computer Assisted Stenting and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis: 7th Joint International Workshop, CVII-STENT 2018 and Third International Workshop, LABELS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Proceedings 3*, pp. 180–189. Springer, 2018.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18082–18091, 2022.

Khaled Saab, Tao Tu, Wei-Hung Weng, Ryutaro Tanno, David Stutz, Ellery Wulczyn, Fan Zhang, Tim Strother, Chunjong Park, Elahe Vedadi, et al. Capabilities of gemini models in medicine. *arXiv preprint arXiv:2404.18416*, 2024.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8317–8326, 2019.

Sanjay Subramanian, Lucy Lu Wang, Sachin Mehta, Ben Bogin, Madeleine van Zuylen, Sravanthi Parasa, Sameer Singh, Matt Gardner, and Hannaneh Hajishirzi. Medicat: A dataset of medical images, captions, and textual references. *arXiv preprint arXiv:2010.06000*, 2020.

Ajay K Tanwani, Joelle Barral, and Daniel Freedman. Repsnet: Combining vision with language for automated medical reports. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 714–724. Springer, 2022.

Tao Tu, Shekoofeh Azizi, Danny Driess, Mike Schaekermann, Mohamed Amin, Pi-Chuan Chang, Andrew Carroll, Charles Lau, Ryutaro Tanno, Ira Ktena, et al. Towards generalist biomedical ai. *NEJM AI*, 1(3):AIoa2300138, 2024.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4566–4575, 2015.

Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, et al. Cogvlm: Visual expert for pretrained language models. *arXiv preprint arXiv:2311.03079*, 2023a.

Zhanyu Wang, Lingqiao Liu, Lei Wang, and Luping Zhou. Metransformer: Radiology report generation by transformer with multiple learnable expert tokens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11558–11567, June 2023b.

Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. Cris: Clip-driven referring image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11686–11695, 2022.

Jiannan Wu, Muyan Zhong, Sen Xing, Zeqiang Lai, Zhaoyang Liu, Wenhai Wang, Zhe Chen, Xizhou Zhu, Lewei Lu, Tong Lu, et al. Visionllm v2: An end-to-end generalist multimodal large language model for hundreds of vision-language tasks. *arXiv preprint arXiv:2406.08394*, 2024.

Size Wu, Wenwei Zhang, Lumin Xu, Sheng Jin, Xiangtai Li, Wentao Liu, and Chen Change Loy. Clipself: Vision transformer distills itself for open-vocabulary dense prediction. *arXiv preprint arXiv:2310.01403*, 2023.

Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In *European Conference on Computer Vision*, pp. 736–753. Springer, 2022.

Ruyi Xu, Yuan Yao, Zonghao Guo, Junbo Cui, Zanlin Ni, Chunjiang Ge, Tat-Seng Chua, Zhiyuan Liu, Maosong Sun, and Gao Huang. Llava-uhd: an lmm perceiving any aspect ratio and high-resolution images. *arXiv preprint arXiv:2403.11703*, 2023.

Ruyi Xu, Yuan Yao, Zonghao Guo, Junbo Cui, Zanlin Ni, Chunjiang Ge, Tat-Seng Chua, Zhiyuan Liu, Maosong Sun, and Gao Huang. Llava-uhd: an lmm perceiving any aspect ratio and high-resolution images. *arXiv preprint arXiv:2403.11703*, 2024.

Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. Mm-react: Prompting chatgpt for multimodal reasoning and action. *arXiv preprint arXiv:2303.11381*, 2023.

Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Guohai Xu, Chenliang Li, et al. Ureader: Universal ocr-free visually-situated language understanding with multimodal large language model. *arXiv preprint arXiv:2310.05126*, 2023.

Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, and Fei Huang. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13040–13051, 2024.

Li Yifan, Du Yifan, Zhou Kun, Wang Jinpeng, Xin Zhao Wayne, and Wen Ji-Rong. Evaluating object hallucination in large vision-language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. URL `https://openreview.net/forum?id=xozJw0kZXF`.

Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity. *arXiv preprint arXiv:2310.07704*, 2023.

Zheng Yuan, Qiao Jin, Chuanqi Tan, Zhengyun Zhao, Hongyi Yuan, Fei Huang, and Songfang Huang. Ramm: Retrieval-augmented biomedical visual question answering with multi-modal pre-training. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 547–556, 2023.

Haotian Zhang, Haoxuan You, Philipp Dufter, Bowen Zhang, Chen Chen, Hong-You Chen, Tsu-Jui Fu, William Yang Wang, Shih-Fu Chang, Zhe Gan, and Yinfei Yang. Ferret-v2: An improved baseline for referring and grounding with large language models. *arXiv preprint arXiv:2403.11703*, 2024.

Kai Zhang, Jun Yu, Zhiling Yan, Yixin Liu, Eashan Adhikarla, Sunyang Fu, Xun Chen, Chen Chen, Yuyin Zhou, Xiang Li, et al. Biomedgpt: A unified and generalist biomedical generative pre-trained transformer for vision, language, and multimodal tasks. *arXiv preprint arXiv:2305.17100*, 2023a.

Sheng Zhang, Yanbo Xu, Naoto Usuyama, Hanwen Xu, Jaspreet Bagga, Robert Tinn, Sam Preston, Rajesh Rao, Mu Wei, Naveen Valluri, et al. Biomedclip: a multimodal biomedical foundation model pretrained from fifteen million scientific image-text pairs. *arXiv preprint arXiv:2303.00915*, 2023b.

Bo Zhao, Boya Wu, Muyang He, and Tiejun Huang. Svit: Scaling up visual instruction tuning. *arXiv preprint arXiv:2307.04087*, 2023a.

Bo Zhao, Boya Wu, and Tiejun Huang. Svit: Scaling up visual instruction tuning. *arXiv preprint arXiv:2307.04087*, 2023b.

Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. Regionclip: Region-based language-image pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16793–16803, 2022.

Yi Zhou, Lei Huang, Tao Zhou, Huazhu Fu, and Ling Shao. Visual-textual attentive semantic consistency for medical report generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3985–3994, 2021.

A Appendix

B General Domain Training Data Description

We curated a vision instruction-tuning dataset using samples from ShareGPT4V (Chen et al., 2023a), ALLaVA (Chen et al., 2024a), SVIT (Zhao et al., 2023a), and selected tasks from Cauldron (Laurençon et al., 2024). Initially, we combined the samples from these four sources, resulting in nearly 9 million data points. Through experimentation with the training data, we derived several key insights:

- Increasing the number of training samples during visual instruction tuning improves the model's performance on commonsense reasoning tasks but also increases the likelihood of hallucination. To mitigate this, the model benefits from training on specialized data.
- Deduplicating the training samples is crucial. Duplicate samples can introduce bias during training, negatively impacting model performance.
- Question-answering data enhances benchmark performance but can reduce the detail and length of generated text.

Based on these insights, we first deduplicated the image-instruction pairs. Since SVIT and ShareGPT4V share the same image set, and SVIT generates multiple instructions per image, we randomly selected eight instructions per image to scale the dataset. The Cauldron dataset, a vast collection of 50 high-quality datasets converted to user/assistant format, included some datasets related to math or coding, which caused misalignment during training. As a result, we excluded five datasets from Cauldron. After processing and deduplication, our final training set contained 2.4 million image-instruction pairs. Additionally, we included text-only data from OpenHermes and MathInstruct to maintain the model's zero-shot capabilities.

C Impact of Token Compression on Model Performance

Token compression, determined by pooling stride and kernel size, plays a critical role in balancing the preservation of visual detail with computational efficiency. To evaluate its impact, we experimented with varying levels of token compression using mean-pooling in two settings: Low + Medium resolution and Low + High resolution models. For each configuration, we adjusted the number of tokens per sub-image (9, 16, 36, 64, or 144), as shown in Tables 7 and 8.

In the Low + Medium resolution configuration (Table 7), each image is divided into four medium-resolution sub-images, and each sub-image is compressed to 16, 36, 64, or 144 tokens using mean-pooling. A significant performance jump is observed across most benchmarks when increasing from 16 to 36 tokens. This suggests that extreme compression (16 tokens per sub-image) overly simplifies the representation, likely discarding fine-grained features critical for tasks like ChartQA and TextVQA, which rely on detailed visual understanding. Beyond 36 tokens, the performance gains taper off, with 36 tokens often outperforming higher counts such as 64 and 144 tokens. This highlights 36 tokens per sub-image as an effective balance for preserving detail while avoiding unnecessary redundancy.

In the Low + High resolution configuration (Table 8), each high-resolution image is divided into 36 sub-images, with each sub-image compressed to 9, 16, 36, or 64 tokens. Due to the computational burden of handling 5184 tokens per image, we did not evaluate 144 tokens in this setting. Similar to the Low + Medium resolution ablations, we observe a significant performance improvement when increasing from the aggressively pooled 9 tokens per sub-image to 16 tokens per sub-image. Another notable observation is that lower token counts (16 or 36) often outperform higher counts (64). Since each high-resolution image is cropped into 36 non-overlapping sub-images, each sub-image covers only a small portion of the original image, making a small token count sufficient to capture detailed features from that region. In fact, increasing the number of tokens could negatively impact performance by introducing more background information.

For simplicity, we used a uniform compression level of 36 tokens per sub-image for both medium- and high-resolution sub-images in our final experiments. However, using different compression ratios for medium- and high-resolution sub-images could potentially yield better results. We present this result this in Supplementary Table 12.

Table 7: Performance comparison for different pooling strides or compression levels for mean-pooling in Low + Medium resolution. Starting from no compression (576 tokens per sub-image) and descending order: 144, 64, 36, and 16 tokens.

Benchmark	**576 tokens**	**144 tokens**	**64 tokens**	**36 tokens**	**16 tokens**
AI2D	63.8	62.7	62.8	**64.5**	60.7
ScienceQA	79.3	77.9	**79.5**	79.2	78.5
ChartQA	**54.0**	53.4	52.2	52.9	26.2
POPE-f1	85.7	87.4	86.3	**87.5**	83.1
GQA	54.1	**55.2**	55.1	54.6	50.4
TextVQA	**64.0**	62.6	61.3	60.9	43.9
VizWiz	56.1	57.0	56.1	**58.7**	53.4
MME	1414.0	1413.0	**1420.6**	1227.4	1285.9

Table 8: Performance comparison for different pooling strides or compression levels for mean-pooling in the Low + High resolution model. Starting from 64 tokens per sub-image and descending to: 36, 16, and 9 tokens.

Benchmark	**64 tokens**	**36 tokens**	**16 tokens**	**9 tokens**
AI2D	62.9	**63.6**	62.0	61.4
ScienceQA	**80.1**	79.0	79.5	77.5
ChartQA	**56.9**	56.4	55.2	45.4
POPE-f1	86.7	87.7	**88.4**	85.9
GQA	54.9	55.2	**55.9**	52.9
TextVQA	**66.8**	65.2	64.6	59.1
VizWiz	57.7	**59.7**	59.1	59.1
MME	1421.1	1397.8	**1434.9**	1309.9

Table 9: Summary of the evaluation benchmarks for general domain.

Task	**Dataset**	**Description**	**Split**	**Metrics**
General VQA	VQA^{v2}	VQA on natural images.	test-dev	Accuracy (↑)
	ScienceQA	Multi-choice VQA on a diverse set of science topics.	test	Accuracy (↑)
	VizWiz	VQA on images taken by visually impaired users.	test	Accuracy (↑)
	AI2D	VQA on diagrams and other artificial images.	test	Accuracy (↑)
Text-oriented VQA	TextVQA	VQA on natural images containing text.	val	Exact Match (↑)
	ChartQA	VQA on various types of charts and graphs.	test	Accuracy (↑)
LVLM Benchmarks	MMBench	Multi-choice VQA on a diverse set of topics.	test	Accuracy (↑)
	MMBench^{CN}	Multi-choice VQA on a diverse set of topics in Chinese.	test	Accuracy (↑)
	POPE	Multi-choice VQA for testing hallucinations.	overall	Accuracy (↑)
	MME	Multi-modal evaluation benchmark for general VQA abilities.	test	Accuracy (↑)

D Biomedical Training Data Description

Many public datasets were used in the training and evaluation of Dragonfly. All datasets were de-identified. Open datasets were used following their existing licenses.

Table 10: Selected Hyperparameters for Stage 1 and Stage 2 training of Dragonfly.

Hyperparameter	Stage 1	Stage 2
Batch Size	64	16
Learning Rate	2e-5	2e-6
LR Scheduler	cosine	cosine
Warmup Steps Ratio	0.01	0.01
Max Sequence Length	4096	4096
Tune Projection Layer	✓	✓
Tune Vision Encoder	×	✓
Tune LLM	×	✓

Table 11: Comparison of TFLOPs and maximum resolution between Dragonfly and baseline methods. FLOPs are calculated for processing a single image at the maximum resolution supported by each method. Calculations are based on the FLOPs accounting approach in (Hoffmann et al., 2022), with details provided in Appendix Section F. Note: Dragonfly* is a more aggressively pooled version of Dragonfly, with 64 tokens for low resolution, 36 tokens per patch for medium resolution, and 16 tokens per patch for high resolution, resulting in a total of 784 image tokens. This performs only slightly worse than the main Dragonfly version. The performance comparison is shown in Table 12.

Model	**Max Resolution**	**TFLOPs**
LLaVA-HD	672 × 672	40.33
LLaVA-UHD	672 × 1008	6.91
Dragonfly	2016 × 2016	41.65
Dragonfly*	2016 × 2016	25.10

Table 12: Performance comparison of multiple token reduction strategies for encoding high-resolution images. The first model, LLaVA-1.5-HD, uses CLIP-ViT-L/14 for both low and medium resolutions, producing 2,880 image tokens. The second model, LLaVA-UHD, results in a variable number of image crops based on the original image size, with each crop producing 64 tokens. The total number of tokens for LLaVA-UHD is variable, with a maximum of 6 crops allowed, resulting in a maximum of 384 image tokens. The third model, Dragonfly, generates 2,016 image tokens using a balanced multi-resolution pooling strategy, with 577 tokens for low resolution and 36 tokens per sub-image for medium and high resolution. The fourth model, Dragonfly*, is a more aggressively pooled version of Dragonfly, with 64 tokens for low resolution, 36 tokens per patch for medium resolution, and 16 tokens per patch for high resolution, resulting in a total of 784 image tokens. All models share the same LLM backbone, LLaMA-3.1-8B-chat, and are trained on the same dataset.

Benchmark	**LLaVA-1.5-HD**	**LLaVA-UHD**	**Dragonfly**	**Dragonfly***
AI2D	63.8	59.9	**64.2**	62.7
ScienceQA	79.3	76.3	**79.7**	79.3
ChartQA	54.0	37.2	56.4	**57.3**
POPE-f1	85.7	85.3	87.7	**88.1**
GQA	54.1	51.0	**55.7**	**55.7**
TextVQA	64.0	51.5	**66.5**	64.5
VizWiz	56.1	51.8	**61.7**	60.6
MME	1414.0	1302.1	**1438.9**	1423.3

Figure 3: Examples generated by Dragonfly, showcasing its diverse capabilities, including world knowledge and humor, multi-turn question-answering, OCR, and chart understanding.

D.1 LLaVA-Med

LLaVA-Med is a dataset for instruction-following tasks involving multi-round conversations about biomedical images, generated using the language-only model GPT-4 (Li et al. (2024a)). Specifically, the model is prompted to generate questions and answers in multi-round formats based on an image caption, as if it could view the image itself. To assemble the image captions and their contexts, LLaVA-Med utilizes PMC-15M (Zhang et al. (2023b)) to select images that contain a single plot. From these, it samples 60,000 image-text pairs from the five most prevalent imaging modalities: CXR (chest X-ray), CT (computed tomography), MRI (magnetic resonance imaging), histopathology, and gross pathology. The dataset also extracts sentences referencing the image from the original PubMed articles to provide additional context to the captions. LLaVA-Med offers two primary versions of the dataset: (i) 60K-IM, which includes inline mentions as context, and (ii) 60K, a similar-sized dataset that excludes inline mentions in its self-instruct generations. Furthermore, a supplementary dataset of 500,000 image-caption pairs is available for alignment purposes. Data link: `https://github.com/microsoft/LLaVA-Med`

D.2 Medicat

Medicat (Subramanian et al. (2020)) is a dataset of medical figures, captions, subfigures/subcaptions, and inline references that enables the study of these figures in context. It consists of 217,000 images from 131,000 open-access PubMed Central and includes captions, inline references for 74%

Table 13: Model architectures and data usage details for our model and baseline models.

Model	**LLM Backbone**	**Vision Base**	**#Data**	**MaxRes**
InstructBLIP (Dai et al., 2023)	Vicuna-7B	CLIP-g/14	130M	224×224
Qwen-VL-Chat (Bai et al., 2023a)	Qwen-7B	CLIP-bigG	1.4B	448×448
LLaVA-1.5 (Liu et al., 2024a)	Vicuna-7B	CLIP-L/14	1.2M	336×336
VILA (Lin et al., 2024)	Llama2-7B	CLIP-L/14	51M	364×364
LLaVA-NeXT (Liu et al., 2024b)	Vicuna-7B	CLIP-L/14	1.2M	672×672
MM1-7B-Chat (McKinzie et al., 2024b)	MM1-7B	CLIP-H	>2B	378×378
mPLUG-Owl2 (Ye et al., 2024)	Llama2-7B	CLIP-L/14	401M	448×448
Monkey (Li et al., 2024c)	Qwen-7B	CLIP-BigG	1B	896×1344
SPHINX (Lin et al., 2023)	Llama2-7B	Mixed Encoders	1B	448×448
SPHINX-2k (Lin et al., 2023)	Llama2-7B	Mixed Encoders	1B	762×762
ShareGPT4V-7B (Chen et al., 2023b)	Vicuna-7B	CLIP-L/14	1.8M	336×336
VisionLLM v2-chat (Wu et al., 2024)	Vicuna-7B	CLIP-L/14	22M	336×336
InternVL-7B (Chen et al., 2024b)	Vicuna-7B	InternViT-6B	>28.7B	224×224
InstructBLIP (Dai et al., 2023)	Vicuna-13B	CLIP-g/14	130M	224×224
LLaVA-1.5 (Liu et al., 2024a)	Vicuna-13B	CLIP-L/14	1.2M	336×336
VILA (Lin et al., 2024)	Llama2-13B	CLIP-L/14	51M	364×364
LLaVA-NeXT (Liu et al., 2024b)	Vicuna-13B	CLIP-L/14	1.2M	672×672
LLaVA-UHD (Xu et al., 2024)	Vicuna-13B	CLIP-L/14	1.2M	672×1008
InternVL-13B (Chen et al., 2024b)	Vicuna-13B	InternViT-6B	>28.7B	364×364
CogVLM-17B-Chat (Wang et al., 2023a)	Vicuna-7B	EVA2-CLIP-E	>1.5B	490×490
Dragonfly (Ours)	Llama3-8B	ViT-L/14	2.9M	2016×2016 or 1008×4032

Table 14: Comparison between Dragonfly and existing LMMs across various benchmarks. Bold numbers indicate the best performance among all the 13B models, while underlined numbers represent the second-best performance.

Model	**Backbone**	**#Data**	**VQA^{v2}**	**VQA^{T}**	**POPE**	**SQA**	**VizWiz**	**AI2D**	**ChartQA**	**MME**	**MMB/MMB^{CN}**
InstructBLIP	Vicuna-13B	130M	-	50.7	78.9	63.1	33.4	-	-	1212.8	-
LLaVA-1.5	Vicuna-13B	1.2M	80.0	61.3	85.9	71.6	53.6	59.5	18.2	1531.3	66.9/63.6
VILA	Llama2-13B	51M	80.8	66.6	84.2	73.7	**60.6**	-	-	<u>1570.1</u>	70.3/64.3
LLaVA-NeXT	Vicuna-13B	1.2M	**82.8**	67.1	86.2	73.6	**60.6**	**70.0**	<u>62.2</u>	**1575.0**	70.0/64.4
LLaVA-UHD	Vicuna-13B	1.2M	<u>81.7</u>	67.7	**89.1**	72.0	56.1	-	-	1535.0	68.0/<u>64.8</u>
InternVL-13B	Vicuna-13B	6B	80.2	58.7	87.1	70.1	54.6	-	-	1546.9	66.5/61.9
CogVLM-13B-Chat	Vicuna-7B	>1.5B	82.3	<u>70.4</u>	87.9	**91.2**	-	-	-	-	**77.6**/-
Dragonfly (Ours)	Llama3-8B	2.9M	81.0	**73.6**	<u>87.9</u>	<u>79.5</u>	<u>59.0</u>	<u>67.9</u>	**71.2**	1538.1	<u>71.9</u>/**66.1**

of figures, and manually annotated subfigures and subcaptions for a subset of figures. Data link: `https://github.com/allenai/medicat`.

D.3 MIMIC-III-CXR

The MIMIC-III-CXR dataset (Johnson et al. (2019)) is a substantial publicly available collection of chest radiographs, containing 377,110 images derived from 227,827 imaging studies conducted at the Beth Israel Deaconess Medical Center from 2011 to 2016. Each image in the dataset is paired with structured labels extracted from free-text radiology reports. The dataset is organized into training, validation, and testing subsets, with 368,960 images allocated for training, 2,991 for validation, and 5,159 for testing. To ensure patient confidentiality, all images have been de-identified. Data link: `https://physionet.org/content/mimic-cxr-jpg/2.1.0/`

D.4 OPENPATH

OpenPath dataset is an expansive collection of 208,414 pathology image-text pairs, making it the largest publicly available pathology image dataset annotated with text descriptions (Huang et al. (2023a)). This dataset was meticulously curated using popular pathology-related hashtags recommended by the United States and Canadian Academy for Pathology (USCAP) and the Pathology Hashtag Ontology projects. It spans images gathered from Twitter and other internet sites, including

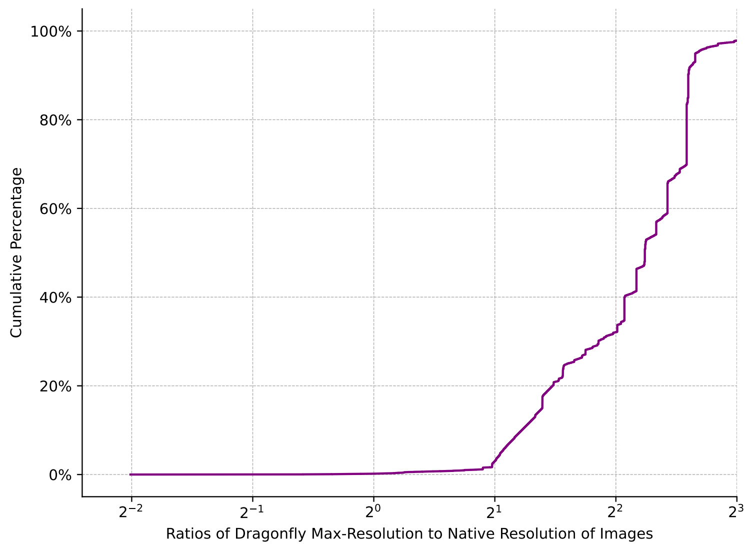

Figure 4: Ratio of maximum resolution of our high resolution image to the native resolution of the original image. We used all of our training dataset to calculate this ratio, which comprised data from multiple different sources and tasks. First, we matched each image into one of the aspect ratios with the algorithm mentioned in 4.1. Then, we calculated the ratio between the longest dimension in our max-res to the longest dimension in the native resolution of the image. From the plot, we can see that 65% of the images in our training cohort are zoomed-in by at least 4x the native resolution.

the LAION dataset, collected between March 21, 2006, and November 15, 2022. The dataset consists of three main components: (1) Tweets, with 116,504 image-text pairs; (2) Replies, comprising 59,869 pairs from highly liked responses; and (3) PathLAION, which adds 32,041 pairs from broader internet sources. Data link: `https://github.com/PathologyFoundation/plip`.

D.5 Kaggle DR (Diabetic Retinopathy)

The Kaggle website organized a DR detection challenge in 2015 Li et al. (2019). The California Healthcare Foundation sponsored the competition. The Kaggle DR dataset consists of 88,702 color fundus images, including 35,126 samples for training and 53,576 samples for testing. Different devices captured the images under various conditions (e.g., resolutions) at multiple primary care sites throughout California and elsewhere. For each subject, two images of the left and right eyes were collected with the same resolution. Clinicians rate each image for the presence of DR on a scale of 0–4 according to the ETDRS scale. Data link: `https://www.kaggle.com/c/diabetic-retinopathy-detection`.

D.6 DDR

DDR is a diabetic retinopathy dataset (Li et al. (2019)) that comprises 13,673 color fundus images collected from 147 hospitals across 23 provinces in China between 2016 and 2018, ensuring a broad demographic spread by including images from patients aged 1 to 100, averaging 54.13 years, and almost evenly split between males (48.23%) and females (51.77%). These images, derived from 9,598 patients and captured using 42 types of fundus cameras, adhere to stringent photographic standards to ensure clarity and appropriate exposure, focusing on crucial retinal structures and lesions. All images have been desensitized for widespread usage and graded for diabetic retinopathy (DR) severity by seven trained graders using the International Classification of Diabetic Retinopathy, supplemented by consensus and consultation with experienced specialists where necessary. Data link: `https://github.com/nkicsl/DDR-dataset`.

D.7 ROCO

The Radiology Objects in Context (ROCO) dataset is a comprehensive collection of over 81,000 radiology images derived from PubMedCentral's open-access biomedical literature (Pelka et al. (2018)). The dataset focuses on analyzing visual elements and semantic relationships within radiological imagery. It includes a variety of medical imaging modalities such as Computer Tomography (CT), Ultrasound, X-ray, Fluoroscopy, Positron Emission Tomography (PET), Mammography, Magnetic Resonance Imaging (MRI), and Angiography. Each image is accompanied by detailed metadata, including captions, keywords, and identifiers from the Unified Medical Language System (UMLS). The ROCO dataset also features an out-of-class set of approximately 6,000 images, ranging from synthetic radiology figures to digital art, to aid in improving prediction and classification tasks. The dataset is split into training, validation, and test sets with 70,308, 8,782, and 8,786 images, respectively.

D.8 VQA-RAD

The VQA-RAD dataset (Lau et al. (2018)) contains 314 radiology images and 2,244 question-answer pairs obtained from CT, MRI, and X-ray examinations, covering three anatomical regions: the head, abdomen, and chest. It features a diverse range of question styles, categorized into 11 types: modality, plane, organ system, abnormalities, etc. Among these, 58% of the question-answer pairs are closed-ended (yes/no), with the remaining 42% being open-ended. The dataset is segmented into a training set of 1,790 QA pairs and a testing set of 451 QA pairs. Our model was trained on the official training set and evaluated on the official test set. Data link: `https://huggingface.co/datasets/flaviagiammarino/vqa-rad`.

D.9 SLAKE

The Slake-VQA dataset, annotated by expert physicians (Liu et al. (2021)), is a comprehensive bilingual (English and Chinese) VQA dataset. It includes 642 images and 14,028 question-answer pairs across three imaging modalities: CXR, CT, and MRI. This dataset spans various radiological areas, covering body regions such as the brain, neck, chest, abdomen, and pelvic cavity. It contains 9,849 VQA samples designated for training, 2,109 for validation, and 2,070 for testing. The questions vary widely, featuring both open-ended (free-form) and closed-ended (yes/no) types that assess different image characteristics like plane, quality, position, organ, abnormality, size, color, shape, and pertinent medical knowledge. We utilized only the English-language examples from the official dataset divisions, comprising 4,919 training, 1,053 validation, and 1,061 test examples. Our model was trained on the official training set and evaluated on the official test set. Data link: `https://www.med-vqa.com/slake/`

D.10 PATH-VQA

This dataset comprises question-answer pairs relating to pathology images (He et al. (2020)). It encompasses a variety of question formats, including open-ended and closed-ended (yes/no) questions. The dataset is constructed through automated techniques and draws from two open-access pathology textbooks and a digital library. It encompasses a total of 32,632 question-answer pairs derived from 4,289 images. The dataset is partitioned into official training, validation, and test subsets, containing 19,654, 6,259, and 6,719 QA pairs, respectively. Our model was trained on the official training set and evaluated on the official test set. Data link: `https://github.com/UCSD-AI4H/PathVQA/tree/master/data`

D.11 IU X-RAY

The IU X-ray dataset, detailed in Demner-Fushman et al. (2016), is available through the Open Access Biomedical Image Search Engine (OpenI). This collection includes radiological exams or cases, each associated with one or more images, a radiology report, and two sets of tags. The reports consist of four sections: Comparison, Indication, Findings, and Impression, with the latter two sections useful for image captioning. The dataset features two types of tags: MTI tags derived automatically from the report text by the Medical Text Indexer and manual tags assigned by two trained coders. Overall, it comprises 3,955 reports and 7,470 frontal and lateral X-ray images. The dataset is divided into 6,698 samples in the training set and 745 samples in the test set. Data link: `https://github.com/nlpaueb/bioCaption`

D.12 PEIR GROSS

The Peir Gross dataset, initially utilized for captioning in research by Jing et al. (2017), features photographs from medical cases sourced from the Pathology Education Informational Resource (PEIR) digital library intended for educational purposes in pathology. This dataset includes 7,443 images from the Gross collections across 21 pathology sub-categories in PEIR, with each image paired with a descriptive single-sentence caption. It is organized into two subsets: 5,172 images for training and 1,289 for testing. Data link: `https://github.com/nlpaueb/bioCaption`

E BIOMEDICAL BENCHMARKS

The details of our evaluation benchmarks are discussed in Section D. A benchmark summary table is also included in 15.

Table 15: Summary of the biomedical evaluation benchmark, which includes vision question answering, image captioning, and report generation across radiology and pathology modalities. We finetuned the model using a subset of the official training set and evaluated it on the official testing set. It should be noted that for MIMIC-CXR and ROCO, we utilized only a portion of the training dataset. Furthermore, for MIMIC-CXR, we selected only those subsets of the test set, including a findings section.

Task Type	**Modality**	**Dataset**	**Split**	
			Train	**Test**
Visual Question Answering	Radiology	VQA-RAD	1,790	451
	Radiology	Slake-VQA	4,919	1,053
	Pathology	Path-VQA	19,654	6,719
Report Generation	Chest X-ray	MIMIC-CXR	25,000	3,513
Image Captioning	Radiology	ROCO	25,000	8,786
	Radiology	IU X-RAY	6,698	745
	Pathology	Peir Gross	5,172	1,289

Table 16: Selected Hyperparameters for Stage 1 and Stage 2 training of Dragonfly-Med.

Hyperparameter	Stage 1	Stage 2
Batch Size	64	16
Learning Rate	2e-5	2e-6
LR Scheduler	cosine	cosine
Warmup Steps Ratio	0.01	0.01
Max Sequence Length	4096	4096
Tune Projection Layer	✓	✓
Tune Vision Encoder	✓	✓
Tune LLM	×	✓

F CODE EXAMPLE: FLOPS CALCULATION

We used DeepMind's Chinchilla scaling law paper to calculate flops (Hoffmann et al., 2022) and the code is given below.

Listing 1: Python code for calculating FLOPs for different approaches.

```python
import math

def format_flops(flops):
    if flops >= 1e12:
        return f"{flops/1e12:.2f} TFLOPs"
    elif flops >= 1e9:
        return f"{flops/1e9:.2f} GFLOPs"
    elif flops >= 1e6:
        return f"{flops/1e6:.2f} MFLOPs"
    return f"{flops:,} FLOPs"

def layer_flops(
    n_ctx=1024,
    d_model=1024,
    n_heads=16,
    d_ff=4096
):
    d_head = d_model // n_heads

    attn_qkv = 2 * n_ctx * 3 * d_model * (d_head * n_heads)
    attn_logits = 2 * n_ctx * n_ctx * (d_head * n_heads)
    attn_softmax = 3 * n_heads * n_ctx * n_ctx
    attn_reduce = 2 * n_ctx * n_ctx * (d_head * n_heads)
    attn_project = 2 * n_ctx * (d_head * n_heads) * d_model
    total_attn = attn_qkv + attn_logits + attn_softmax + attn_reduce +
        attn_project

    ff = 2 * n_ctx * (d_model * d_ff + d_model * d_ff)

    return total_attn + ff

def calculate_vit_flops(
    img_size=336,
    patch_size=14,
    n_channels=3,
    n_layers=24,
    n_heads=16,
    d_model=1024,
    d_ff=4096,
):
    n_patches = (img_size // patch_size) ** 2
    n_ctx = n_patches + 1

    embeddings = 2 * n_patches * (patch_size * patch_size) * n_channels *
         d_model

    total_flops = embeddings + (n_layers * layer_flops(n_ctx=n_ctx,
        d_model=d_model, n_heads=n_heads, d_ff=d_ff))
    return total_flops

def calculate_projection_flops(vision_dim=1024, projection_dim=4096,
    n_tokens=577):
    return 2 * vision_dim * projection_dim * n_tokens

def calculate_llm_flops(
    n_layers=32,
    n_heads=32,
    d_model=4096,
    n_ctx=577,
    d_ff=14336,
):
    d_head = d_model // n_heads

    embeddings = 2 * n_ctx * d_model

```

```
    total_flops = embeddings + (n_layers * layer_flops(n_ctx=n_ctx,
        d_model=d_model, n_heads=n_heads, d_ff=d_ff))

    return total_flops

# Llava-UHD
num_crops = 6
n_tokens = num_crops * 64

vit_flops = calculate_vit_flops() * num_crops
projection_flops = calculate_projection_flops(n_tokens=n_tokens)
llm_flops = calculate_llm_flops(n_ctx=n_tokens)
total_flops = vit_flops + projection_flops + llm_flops

# total_flops: 6.91 TFLOPs

# Llava-1.5
num_crops = 5
n_tokens = num_crops * 576

vit_flops = calculate_vit_flops() * num_crops
projection_flops = calculate_projection_flops(n_tokens=n_tokens)
llm_flops = calculate_llm_flops(n_ctx=n_tokens)
total_flops = vit_flops + projection_flops + llm_flops

# total_flops: 40.40 TFLOPs

# Dragonfly
num_crops = 41
n_tokens = 2016

vit_flops = calculate_vit_flops() * num_crops
projection_flops = calculate_projection_flops(n_tokens=n_tokens)
llm_flops = calculate_llm_flops(n_ctx=n_tokens)
total_flops = vit_flops + projection_flops + llm_flops

# total_flops: 41.65 TFLOPs
```