

PaliGemma 2: A Family of Versatile VLMs for Transfer

Andreas Steiner^{*,†}, André Susano Pinto^{*}, Michael Tschannen^{*}, Daniel Keysers, Xiao Wang, Yonatan Bitton, Alexey Gritsenko, Matthias Minderer, Anthony Sherbondy, Shangbang Long, Siyang Qin, Reeve Ingle, Emanuele Bugliarello, Sahar Kazemzadeh, Thomas Mesnard, Ibrahim Alabdulmohsin, Lucas Beyer and Xiaohua Zhai

Google DeepMind, ^{*}Core team, [†]Project lead

PaliGemma 2 is an upgrade of the PaliGemma open Vision-Language Model (VLM) based on the Gemma 2 family of language models. We combine the SigLIP-So400m vision encoder that was also used by PaliGemma with the whole range of Gemma 2 models, from the 2B one all the way up to the 27B model. We train these models at three resolutions (224px², 448px² and 896px²) in multiple stages to equip them with broad knowledge for transfer via fine-tuning. The resulting family of base models covering different model sizes and resolutions allows us to investigate factors impacting transfer performance (such as learning rate) and to analyze the interplay between the type of task, model size, and resolution. We further increase the number and breadth of transfer tasks beyond the scope of PaliGemma including different OCR-related tasks such as table structure recognition, molecular structure recognition, music score recognition, as well as long fine-grained captioning and radiography report generation, on which PaliGemma 2 obtains state-of-the-art results.

1. Introduction

PaliGemma [9] is a 3B vision-language model (VLM) for transfer combining the SigLIP [108] vision encoder and the 2B Gemma language model [21]. It matches the performance of much larger prior VLMs consisting of a range of different vision encoders and language models. We now upgrade PaliGemma by replacing its language model component with the more recent and more capable language models from the Gemma 2 family [22], producing new PaliGemma 2 base VLMs at 3 different sizes (3B, 10B, 28B) and 3 different resolutions (224px², 448px², 896px²). To equip these VLMs with broad capabilities we use the same 3-stage training recipe as PaliGemma. The resulting models are designed to be fine-tuned, and when evaluated on the 30+ transfer tasks considered in [9] (which include common captioning and VQA tasks, and some video and referring expression tasks), PaliGemma 2 slightly outperforms PaliGemma at the same resolution and model size, and obtains substantial improvements at larger model sizes. We release the PaliGemma 2 VLMs as open-weight models which can serve as drop-in replacement for PaliGemma.

Having a family of models at hand that are all derived from comparable building blocks and are trained according to the same recipe allows us to analyze the effect of model size and resolution on the downstream performance in a controlled setting (see Sec. 4.1). For example, while almost every task benefits from added compute, we identify which transfer tasks benefit more from compute due to increased resolutions, and which from compute due to a larger, more capable language model. We also show that larger models tend to have a lower optimal *transfer* learning rate.

We also explore new tasks which were not explored in depth in [9], including text detection and recognition (Sec. 4.2), table structure recognition (Sec. 4.3), molecular structure recognition (Sec. 4.4), optical music score recognition (Sec. 4.5), long caption generation (Sec. 4.6), spatial reasoning (Sec. 4.7), and radiography report generation (Sec. 4.8). PaliGemma 2 obtains state-of-the-art results on many of those tasks. Finally, we benchmark and analyze low-precision variants of PaliGemma 2 for on-device deployment on CPU (Sec. 4.9).

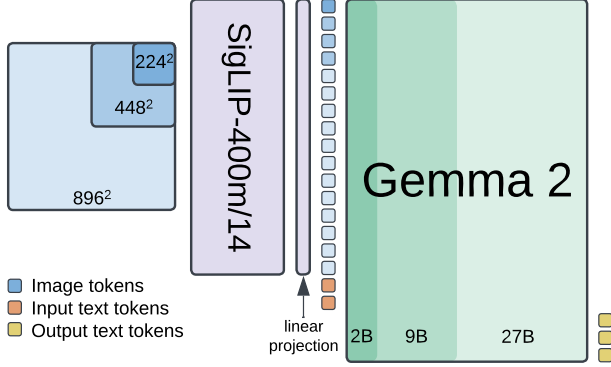


Figure 1 | PaliGemma 2 processes a $224\text{px}^2/448\text{px}^2/896\text{px}^2$ image with a SigLIP-400m encoder with patch size 14px^2 , yielding $256/1024/4096$ tokens. After a linear projection, the image tokens are concatenated with the input text tokens and Gemma 2 autoregressively completes this prefix with an answer.

2. Related work

Over the last few years, VLMs evolved rapidly from simple dual-encoder (contrastive) [31, 77, 108] or encoder-decoder (captioning) [20, 93, 94, 98] designs trained from scratch, to more capable designs combining a pretrained vision encoder with a pretrained language model [4, 5, 14, 16, 48, 72, 96, 103]. Broadly, three paradigms are used to transfer these models: zero-shot, few-shot, and fine-tuning. Another recent trend is “instruction tuning” which aims to make the models more user friendly [18, 54].

Several previous works [9, 19, 34, 35, 45, 66, 92, 109] have investigated the effect of scaling VLMs along different axes such as training data and compute, resolution, model size, and quality of components, in particular the vision encoder. However, we are not aware of prior work which jointly studies the effect of the image resolution and the size of the language models on *transfer via fine-tuning*. In particular, prior works relying on different language model sizes often use models with different architecture and training recipes from different labs, e.g. [35, 92] (with the notable exception of [47]).

3. Model

We follow exactly the same modeling, training, and data setup as PaliGemma [9] and briefly sum-

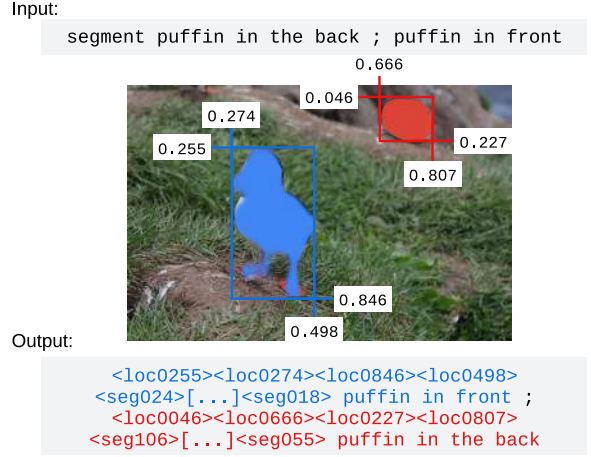


Figure 2 | Referring segmentation example from our PaliGemma demo^a. The model is pretrained with a vocabulary that includes localization tokens (for detection) and segmentation tokens (to define a binary mask inside a bounding box).

^a <https://huggingface.co/spaces/big-vision/paligemma>

marize the most important aspects here. We use the same pretrained SigLIP-So400m vision encoder [3, 108] and map its (sequence of) embeddings to the Gemma 2 input space with a linear projection. The visual embeddings are combined with a text prompt and fed to the Gemma 2 language model (prefill). Predictions are then obtained by autoregressively sampling from the language model (see Fig. 1).

We pretrain PaliGemma 2 in three stages (with stage 0 corresponding to unimodal pretraining of the components, see [108] and [21]).

- **Stage 1** combines the pretrained SigLIP-So400m and Gemma 2 checkpoints (raw checkpoints, without post-training steps) and trains them jointly on a multimodal task mixture of 1 billion examples designed to enable transferability to a wide range of tasks via fine-tuning. The image resolution is 224px^2 ; no parameters are frozen during this stage.
- **Stage 2** first trains for 50 million examples at resolution 448px^2 and then for 10 million examples at resolution 896px^2 . The task mixture has the same components but tasks benefiting from high resolution are upweighted, and the output sequence length is increased

					Training cost / example			
Vision Encoder		LLM	Params.	224px ²	448px ²	896px ²		
PaliGemma 2	3B	Gemma 2	2B	3.0B	1.0	4.6	23.5	
PaliGemma 2	10B	SigLIP-So400m	Gemma 2	9B	9.7B	3.7	18.3	67.7
PaliGemma 2	28B		Gemma 2	27B	27.7B	18.9	63.5	~155.6

Table 1 | The vision encoder parameter count is small compared to the LLM, but the compute is dominated by the vision tokens in the LLM. The last three columns show the relative training cost per example (as measured in our pre-training setup). Models are trained on Cloud TPUv5e [24], except the 28B model at 896px² is trained on TPUv5p, for which we assume a speed-up of 2.3× per chip.

(to promote e.g. learning of OCR for long sequences of visual text).

- **Stage 3** fine-tunes the checkpoints from stage 1 or 2 (depending on the resolution) to the target task. PaliGemma considered a range of academic benchmarks, including some involving multiple images and short videos. We consider the same set of benchmarks here (exploring the same set of hyperparameters from [9, Sec. 3.2.4]). In addition, we also explore new applications involving document-related tasks, long caption generation, and medical image understanding.

Following [22], we apply logits soft-capping [6] to the attention and output logits in the Gemma 2 component with the same parameters as [22] in Stages 1 and 2, but not in Stage 3, as this led to worse results for some transfer tasks. Further, we use the Adam optimizer [42] with default hyperparameters throughout, and adjust the learning rate based on the model size in Stages 1 and 2. Specifically, we multiply the learning rate of $2 \cdot 10^{-5}$ used in Stages 1 and 2 for PaliGemma by 0.5 for PaliGemma 2 3B and by 0.25 for PaliGemma 2 10B and 28B.

For details on the training data mixture we refer to [9, Sec. 3.2.5] and provide a brief summary here. The mixture involves captioning, grounded captioning (as in [94]), OCR, different machine generated visual question answering (VQA) tasks [11, 75], detection [13] and instance segmentation [15]. Many of the corresponding labels are machine generated, mostly relying on publicly available specialist models (see [9, Sec. 3.2.5]), and none uses a large commer-

cial VLM as common among other open VLMs such as LLaVA [54].

Similar to PaliGemma, we train PaliGemma 2 models on Cloud TPUv5e Pod slices [24] (except TPUv5p for the 28B model at 896px²) of 256 to 1024 chips and use a fully-sharded data-parallel (FSDP [8, 110]) sharding strategy. PaliGemma 2 3B has roughly the same training cost as PaliGemma (3 days for Stage 1 using 256 chips); the cost for other variants and resolutions can be inferred from Table 1. It is worth noting that increasing resolution incurs a similar additional cost as increasing the language model size.

4. Experiments

In addition to the broad range of transfer tasks considered in [9], we also consider new tasks involving text detection and recognition (Sec. 4.2), table structure recognition (Sec. 4.3), molecular structure recognition (Sec. 4.4), optical music score recognition (Sec. 4.5), long caption generation (Sec. 4.6), spatial reasoning (Sec. 4.7), and radiography report generation (Sec. 4.8).

We provide examples for each new task in Appendix A and transfer details in Appendix B.

4.1. Investigating model size and resolution

To study the effect of model size and resolution on task performance we finetune the 3 model variants (3B, 10B and 28B) in two resolutions (224px² and 448px²) on the 30+ academic benchmarks used by [9], covering a broad range of captioning, VQA, and refer-

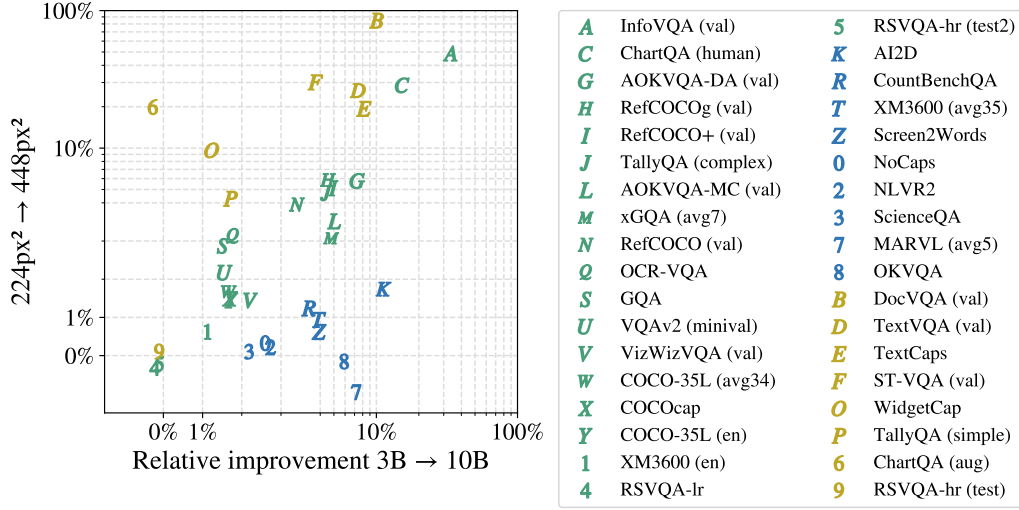


Figure 3 | Relative improvements of metrics after transfer, when choosing a pre-trained checkpoint with a larger LM, or with a higher resolution. The tasks are grouped into tasks sensitive to both model size and resolution (—), sensitive to model size (—), and sensitive to resolution (—). Note that some benchmarks are quite saturated (e.g. ScienceQA’s relative improvement of 2.2% corresponds to an error reduction of 53.8% – see Figure 13). Data used to create this plot available in Table 13.

ring segmentation tasks on natural images, documents, infographics, and videos. We reuse the optimal hyperparameters from the earlier PaliGemma work and only sweep the learning rate $\{0.03, 0.06, 0.1, 0.3, 0.6, 1.0, 3.0\} \cdot 10^{-5}$ for every model size. Since for most tasks the earlier work used the same hyperparameters for 224px² and 448px², we only sweep at 224px² resolution and reuse the selection for both resolutions. We select the best learning rate based on the respective validation split for each model size and task, then retrain the models and report the test metrics. Complete results are available in Table 13.

4.1.1. Effect on task performance

Increasing image resolution and increasing LM size both lead to an increase in the FLOPs spent on the prediction (and training, see Table 1) of our PaliGemma 2 models. Thus, we generally expect most tasks to benefit from both these changes. On the other hand, some tasks might benefit from more detail in the input (higher resolution) or better language understanding and increased world knowledge provided by a larger LM. To get a more fine-grained understanding of these aspects we visualize in Fig. 3 the relative improvement in transfer metrics when equipping PaliGemma 2

3B (224px²) with either the bigger 9B LM while keeping the resolution (3.7× more FLOPs), or keeping the model size but increasing the resolution to 448px² (4.6× more FLOPs).

As expected, most tasks similarly benefit from a resolution and model increase (green markers). There is a group of tasks (yellow markers) focused on text, document, screen and chart understanding which mainly benefit from a resolution increase. The images in the corresponding benchmarks often have a native resolution significantly larger than 224px², which is aligned with this observation. Another group of tasks (blue markers) mostly benefits from LM size increase. Some of these tasks involve multilingual data (XM3600 (avg35)), or require advanced visual reasoning (AI2D, CountBenchQA, NLVR2).

Fig. 4 provides additional detail on the scaling behavior as a function of resolution and model size. Compared to increasing model size from 3B to 10B, increasing it further to 28B often only leads to moderate improvements, or no improvements at all. Using the largest PaliGemma 2 can thus be useful if one wants to get the best possible performance and has no compute or latency constraints. A possible factor related to the relatively worse transferability of PaliGemma 2 28B

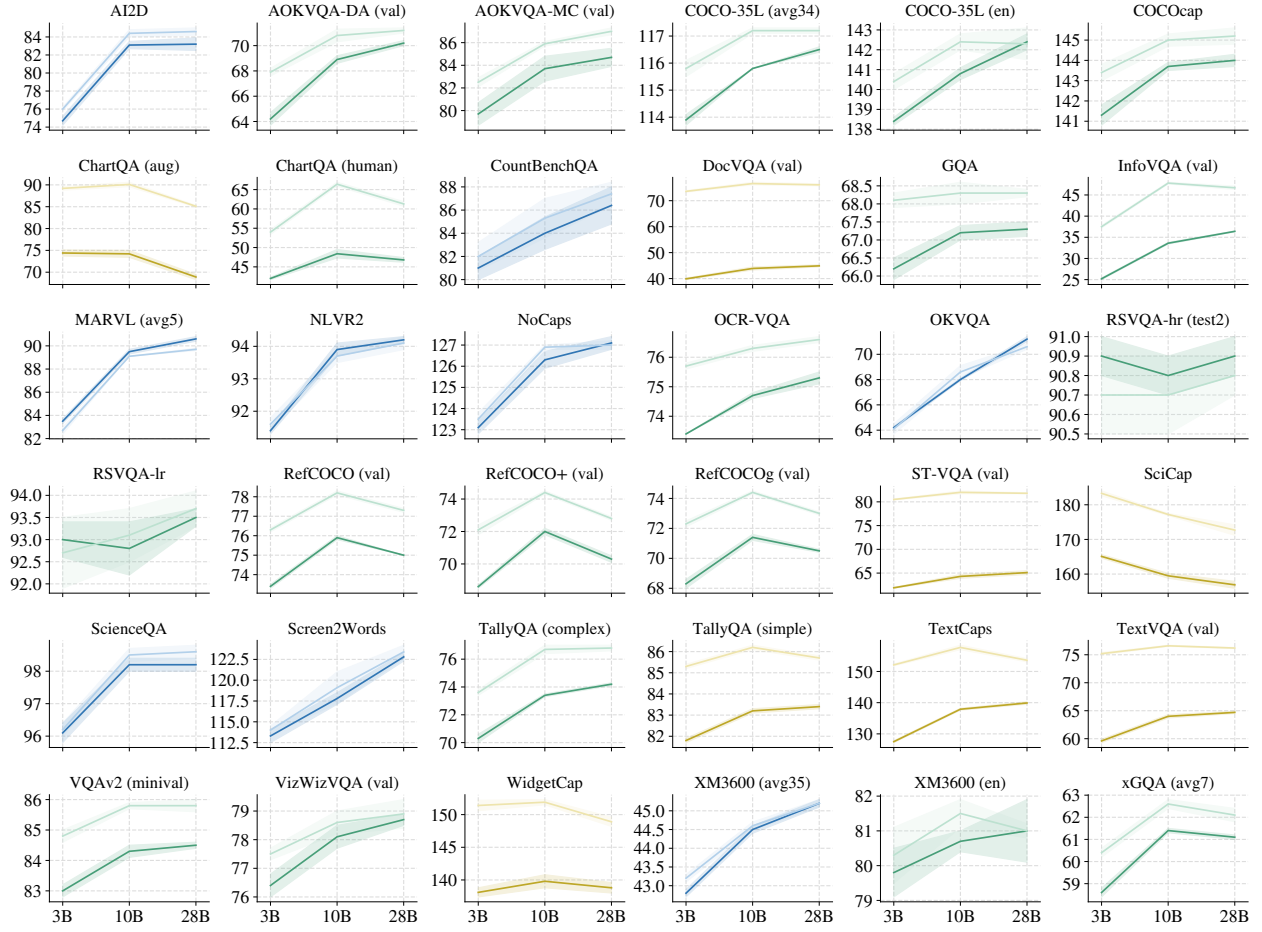


Figure 4 | Transfer performance as a function of model size and resolution (median over 5 transfer runs). The shaded area marks standard deviation to reported value. Lighter lines correspond to higher resolution (448px²). The tasks are grouped into tasks sensitive to both model size and resolution (—), sensitive to model size (—), and sensitive to resolution (—). Data for this plot is available in Table 13.

is that the underlying Gemma 2 27B model is trained from scratch, as opposed to the 2B and 9B models, which are distilled [22, Sec. 6.1].

4.1.2. Model size and transfer learning rate

Figure 5 visualizes the (normalized) task performance as a function of the transfer learning rate. As a general trend we observe that the optimal learning rate for larger models tends to be lower than for smaller models (diagonal patterns in the heat map). We thus recommend to sweep smaller learning rates when increasing model size. Additionally, we found that the new PaliGemma 2 3B generally has a smaller optimal transfer learning rate when compared to PaliGemma.

4.1.3. Using Gemma 2 instead of Gemma 1

We also compare with PaliGemma in Table 15. It can be seen that for the same resolution and model size (i.e. 3B) PaliGemma 2 models perform slightly better than the corresponding PaliGemma models. On average over the 30+ academic benchmarks the scores were 0.65 better for 224px² and 0.85 for 448px².

4.2. Text detection and recognition

We apply PaliGemma 2 to advanced OCR involving localization and recognition of individual words from images. Specifically, the outputs are pairs of {transcription, bounding box}. Following the HierText competition [57], we use word level precision, recall, and F1 as the metrics. A word

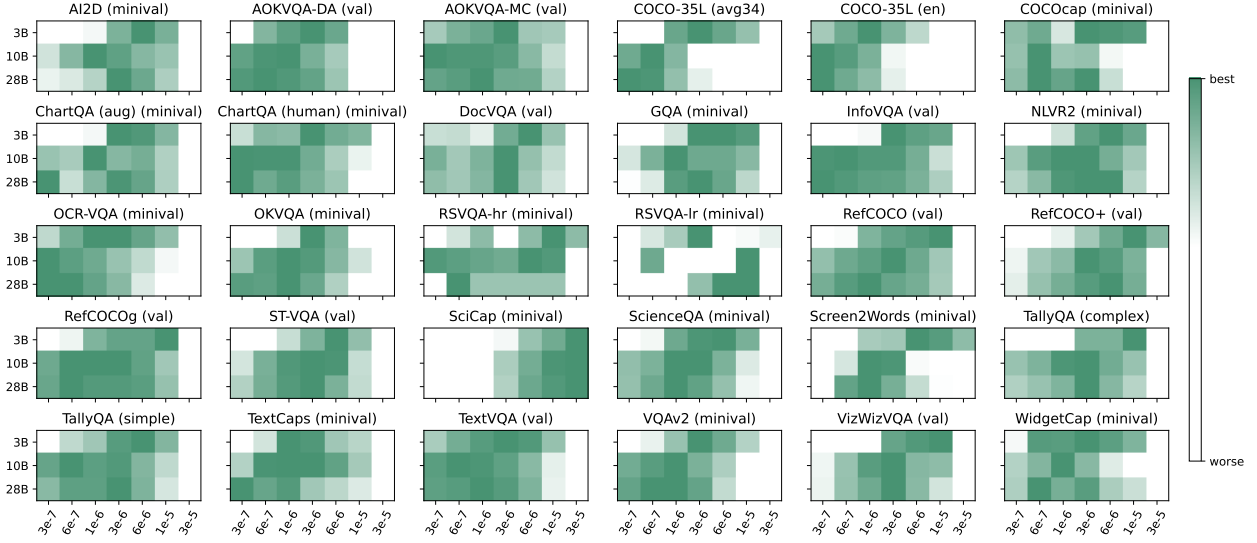


Figure 5 | Per-task performance as a function of model size and learning rate for several of the downstream tasks. Values are normalized for each task and model size, with darker color indicating better task performance. Larger models tend to have a lower optimal transfer learning rate. Zero-shot tasks not shown as their values were not used to select learning rates. The data used for this plot is provided in Table 14.

result is considered true positive if the IoU with the ground-truth bounding box is greater than or equal to 0.5 and the transcription matches the ground-truth. Note that the HierText protocol does not normalize letter cases, punctuation symbols, or filter based on text lengths but directly compares predictions against ground-truth.

We fine-tune PaliGemma 2 on a mixture of the train splits of ICDAR’15 [36], Total-Text [17], MLT17 and MLT19 [68], HierText [56], TextOCR [84], IntelOCR [44] and evaluate on the ICDAR’15 and Total-Text test sets, which are the most commonly used OCR benchmarks. Table 2 shows the results: PaliGemma 2 3B at 896px² outperforms the state of the art HTS [58]. We emphasize that this result is obtained simply by fine-tuning a general-purpose VLM which does not rely on task-specific architecture components as common in the OCR literature. This highlights PaliGemma 2’s versatile interface, and shows the benefits of OCR-related pretraining in Stages 2 and 3. We further tried reducing the resolution which led to substantially lower prediction quality, while increasing the model size did not lead to improvements.

4.3. Table structure recognition

The goal of table structure recognition is to extract table text content, corresponding bounding box coordinates, and the table structure in HTML format from document images. To transfer PaliGemma 2 to this task we finetune on (the train splits of) two popular data sets, PubTabNet [112] containing 516k images of tabular data from the PubMed Central Open Access Subset (commercial use collection) and FinTabNet [111], consisting of 113k financial report tables from annual reports of S&P 500 companies. We remove examples with obviously corrupted ground truth (e.g. a bounding box extending outside the image frame) from the training data and further apply the refinements from [86] to FinTabNet. Images are resized to the target resolution while preserving the aspect ratio, and padded to square size to match the target input resolution.

We assess model quality with the Tree Edit Distance Similarity (TEDS) [112] and the Grid Table Similarity (GriTS) [85], two families of metrics which measure cell text content, cell topology/structure, and bounding box quality. PaliGemma 2 sets a new state of the art for most of these metrics (Table 3). We further tried in-

	ICDAR’15 Incidental			Total-Text		
	P	R	F1	P	R	F1
HTS	81.9	68.4	74.5	75.7	69.4	72.4
PaliGemma 2 3B 896px ²	81.9	70.7	75.9	73.8	74.5	74.2

Table 2 | Text detection and recognition performance: The 896px² PaliGemma 2 model outperforms the state-of-the-art model HTS [58] on ICDAR’15 Incidental and Total-Text, under the evaluation protocol of HierText [57].

	FinTabNet				PubTabNet			
	S-TEDS	TEDS	GriTS-Top	GriTS-Con	S-TEDS	TEDS	GriTS-Top	GriTS-Con
SOTA	98.9	98.2	99.0	98.6	97.9	96.9	-	-
PaliGemma 2 3B 896px ²	99.2	98.9	99.4	99.2	97.6	97.3	98.0	97.8

Table 3 | PaliGemma 2 results for table structure recognition on FinTabNet [111] and PubTabNet [112], compared to the state of the art. The reference metrics are from [28, 38, 60, 86].

creasing the model size which did not lead to additional benefits, and using a lower image resolution led to a small regression in quality.

4.4. Molecular structure recognition

We explore PaliGemma 2 for molecular structure recognition, the task of inferring the molecule graph structure (represented as a SMILES string [99]) from molecular drawings. As training data we use 1 million molecules from the PubChem dataset [41], rendered using the Indigo toolkit [71], and augmented with a variety of drawing styles and random perturbations, following MolScribe [76]. We then evaluate on the same eval set as [76] consisting of 5.7k synthetic molecule images rendered with the ChemDraw library. We use exact match percentage as a metric, shown in Table 4. PaliGemma 2 outperforms the state of the art MolScribe when using 448px² resolution; further increasing the resolution did not lead to a higher exact match percentage.

4.5. Optical music score recognition

We apply PaliGemma 2 to optical music score recognition: translating images of single-line pianoform scores into their digital score representation in the ****kern** format¹. The ****kern** representation encodes pitch and duration along with

other common score-related information such as articulation and barlines.

We use the GrandStaff dataset [79] containing 53.7k images and employ the official train, validation and test splits. During training we use both the original images and synthetically augmented versions. Evaluation is done on the original images without distortion. The metrics are the same as in [80] and are based on the the normalized mean edit distance. More specifically, the Character Error Rate (CER) counts errors at the character level, the Symbol Error Rate (SER) measures errors at the symbol level (combining multiple characters), and the Line Error Rate (LER) is based on full lines in the ****kern** encoding.

The results are shown in Table 5 along with those of the current state of the art method [80]. The error rates decrease with increasing resolution, with the best error rates obtained at 896px² resolution. Increasing the model size from 3B to 10B did not lead to further error reduction.

4.6. Generating long, fine-grained captions

Generating long image captions with fine-grained detail has many use cases in multimodal learning, for example to train text-to-image generation models with good controllability [7, 105]. To adapt PaliGemma 2 for this task we fine-tune on

¹<https://www.humdrum.org/rep/kern/>

	Full Match \uparrow
MolScribe [76]	93.8
PaliGemma 2 10B ^{448px²}	94.8

Table 4 | PaliGemma 2 performance for molecule structure recognition on ChemDraw data [76].

	CER \downarrow	SER \downarrow	LER \downarrow
Sheet Music Tr. [80]	3.9	5.1	13.1
PaliGemma 2 3B ^{896px²}	1.6	2.3	6.7

Table 5 | PaliGemma 2 performance for music score recognition on the GrandStaff data set [80]. Character Error Rate (CER), Symbol Error Rate (SER), and Line Error Rate (LER) in [%].

the DOCCI (Descriptions of Connected and Contrasting Images) [69] data set which contains 15k images with detailed human-annotated English descriptions with an average length of 7.1 sentences (639 characters, 136 words). The descriptions provide object spatial relations, object counting, text rendering, world knowledge, etc.

We first fine-tune PaliGemma 2 on DOCCI’s train split, exploring the hyperparameter range suggested in [9, Sec. 3.2.4]. We select the most performant models by perplexity scores based on the test split, and generate image captions on the 100-image `qual_dev` split, with a maximum decoding length of 192. We then conduct human evaluations assessing whether each generated sentence is factually aligned with (entailed by) the image content (see Appendix B.5 for details on the evaluation protocol). Based on these evaluations we select the most factually aligned models and retrain them on the union of train and test splits, followed by another round of human evaluation (on the `qual_dev` split). The results, shown in Table 6 indicate that the fine-tuned PaliGemma 2 model produces more factually aligned sentences than many popular VLMs, which are often instruction-tuned on 10–100 \times larger high-quality captioning sets than PaliGemma 2. Unsurprisingly, we observe that increasing model size and resolution both improve factual alignment.

	#par.	#char.	#sent.	NES \downarrow
MiniGPT-4	7B	484	5.6	52.3
mPLUG-Owl2	8B	459	4.4	48.4
InstructBLIP	7B	510	4.0	42.6
LLaVA-1.5	7B	395	4.2	40.6
VILA	7B	871	8.6	28.6
PaliGemma	3B	535	8.9	34.3
PaLI-5B	5B	1065	11.3	32.9
PaliGemma 2 ^{448px²}	3B	529	7.7	28.4
PaliGemma 2 ^{448px²}	10B	521	7.5	20.3

Table 6 | PaliGemma 2 results for long captioning on the DOCCI data [69]. Pali* models are models fine-tuned on DOCCI at 448px²; the other baselines are instruction-tuned on a broad range of tasks. Average prediction length in characters and sentences, and percentage of Non-Entailment Sentences (NES), measuring factual inaccuracies.

4.7. Spatial reasoning

VLMs like PaliGemma 2 obtain strong performance in vision-language tasks which involve object localization, such as referring expression comprehension and segmentation [9, 15, 94, 104]. These tasks and the associated benchmarks often rely on machine-generated annotations and are blind to complex failure modes, e.g. those involving negations.

The Visual Spatial Reasoning (VSR) benchmark [53] is designed to overcome these issues and we use it here to assess the spatial reasoning capabilities of PaliGemma 2. It is formulated as a classification task, where a model needs to determine whether a statement about the spatial relationship of objects in the image is correct or not. To use PaliGemma 2’s flexible text interface we frame this benchmark as a QA task with True / False answers. The results in Table 7 show that PaliGemma 2 outperforms prior fine-tuned models, and fine-tuning also provides a significant improvement over InstructBlip [18], a strong zero-shot model from the literature. We observe significant benefits from larger model size, indicating benefits from improved language understanding, whereas going beyond resolution 224 did not lead to improvements.

	zs. split	rand. split
Human [53]	95.4	
InstructBLIP (zs.) [18]	65.6	-
LXMERT [89]	70.1	61.2
PaliGemma 2 3B 224px ²	74.8	81.6
PaliGemma 2 10B 224px ²	79.8	86.8

Table 7 | PaliGemma 2 accuracy on VSR [53] on the zeroshot and random test splits. We show a fine-tuned (LXMERT) and zero-shot (InstructBLIP) baseline from the literature.

4.8. Radiography report generation

To explore the capabilities of PaliGemma 2 models in the medical domain, we apply it to automatic chest X-ray report generation, which can be cast as a (long) captioning task on X-ray images. We fine-tune PaliGemma 2 on the MIMIC-CXR dataset [23, 33] which contains 377k images (originating from 228k radiographic studies at the Beth Israel Deaconess Medical Center in Boston, MA) with free-text radiology reports. We use the same train, validation, and test splits as [90]. To improve quality, we use an LLM (Gemini 1.5 pro) to remove mentions of prior X-rays as the model does not have access to those.

We measure the RadGraph F1-score [30], which is the F1 score between the entities extracted from the reference report and the generated one using RadGraph. RadGraph takes into account the absence or presence of findings in the report, as well as their relationships to image features. Results are reported on test data held out during training and tuning.

Table 8 shows the performance of PaliGemma 2 models along with baselines from the literature. PaliGemma 2 obtains a state-of-the-art RadGraph score. Increasing resolution and model size both lead to modest improvements.

4.9. CPU inference and quantization

In some cases we may want to run inference of PaliGemma 2 on devices without accelerators. We are interested in the resulting runtimes and quality when running inference on

	C↑	B↑	R↑	F1↑
Flamingo-CXR [90]	13.8	10.1	29.7	20.5
Med-Gemini-2D [102]	17.5	20.5	28.3	24.4
PaliGemma 2 3B 896px ²	19.9	14.6	31.9	28.8
PaliGemma 2 10B 896px ²	17.4	15.0	32.4	29.5

Table 8 | PaliGemma 2 performance for radiography report generation on the on the MIMIC-CXR data [23, 33]. We report CIDEr (C), BLEU4 (B), Rouge-L (R), and RadGraph F1-scores [%] [30] (a clinical metric).

CPUs, and briefly present experiments using the `gemma.cpp`² framework here. `gemma.cpp` is a lightweight, portable C++ inference engine that supports 8-bit switched-floating-point quantization (alternative options for CPU inference include `llama.cpp`³, `XNNPack`⁴, and others).

To assess the inference speed for CPU-only inference, we run PaliGemma 2 inference on four different architectures with `gemma.cpp`. We use a checkpoint of PaliGemma 2 3B (224px²) fine-tuned on COCOcap and the example image for PaliGemma in `gemma.cpp`. The prompt “describe this image” results in a prefill length of $256 + 4 = 260$ tokens (for image + text). The output response “A large building with two towers on the water” consists of 11 tokens. All runs used batch size 1. The results are presented in Table 9 and give an overview of what can be expected on different processors (for this particular setting).

From evaluations on PaliGemma [9] we already know that going from 32-bit floating point (f32) to 16-bit (bf16) weights is possible without a loss of quality. Here we compare to the `gemma.cpp` mixed quantization. Table 10 shows a quality comparison for five of the fine-tuning datasets (chosen for coverage of various tasks). We fine-tuned PaliGemma 2 3B (224px²) once for each of these five datasets. (Noticeable differences to Table 13 for the Jax version are the result of using greedy decoding for COCOcap and TextCaps.) We then evaluated the resulting checkpoints both in Jax and in `gemma.cpp` after quantization. The

²<https://github.com/google/gemma.cpp>

³<https://github.com/ggerganov/llama.cpp>

⁴<https://github.com/google/XNNPACK>

Processor	Threads	Walltime [s]			Tokens/sec	
		ViT	Prefill	Extend	Prefill	Extend
Apple M1 Max	4+1	1.6	8.2	0.9	32	12
Apple M3 Pro	7+1	0.8	4.4	0.5	59	22
AMD Milan	8+1	0.82	4.9	0.64	53	17
AMD Milan	32+1	0.39	1.8	0.34	144	32
AMD Genoa	8+1	0.36	1.8	0.29	147	37
AMD Genoa	32+1	0.17	0.8	0.27	323	41

Table 9 | CPU-only inference speed measurements with `gemma.cpp`-based implementation on different architectures. Inference of finetuned PaliGemma 2 3B (224px²) with greedy decoding. Prefill is done with 260 tokens and followed by 11 calls to extend during decoding.

	COCOpac	TextCaps	AI2D	OKVQA	DocVQA(val)
Jax, F32, 12.1GB	140.0	126.3	75.4	64.0	39.8
<code>gemma.cpp</code> , quantized, 4.0GB	139.8	126.6	75.6	64.1	39.8
relative metric values [%]	99.9	100.2	100.1	100.1	99.9

Table 10 | Quality comparison between Jax/f32 inference on TPU and quantized `gemma.cpp`-based inference on CPU. Inference of one fine-tuned PaliGemma 2 3B (224px²) run. Noticeable differences to Table 13 for the Jax version are the result of using greedy decoding for COCOpac and TextCaps. Relative numbers based on metric values before rounding to one decimal.

relative quality after quantization shows no practical quality difference.

5. Conclusion

With PaliGemma 2 we present a new family of open-weight models spanning a broad range of model sizes and input resolutions. PaliGemma 2 obtains strong transfer performance across a broad range of captioning, VQA, and video tasks. In particular, the newly added larger variants lead to significant improvements compared to PaliGemma for users with a larger compute budget. Furthermore, we show that PaliGemma 2 excels in applications beyond what was considered in PaliGemma, including domains like music, molecules, and medical imaging.

References

- [1] M. Acharya, K. Kafle, and C. Kanan. TallyQA: Answering complex counting questions. In *AAAI*, 2019.
- [2] H. Agrawal, K. Desai, Y. Wang, X. Chen, R. Jain, M. Johnson, D. Batra, D. Parikh, S. Lee, and P. Anderson. NoCaps: Novel object captioning at scale. In *ICCV*, 2019.
- [3] I. Alabdulmohsin, X. Zhai, A. Kolesnikov, and L. Beyer. Getting vit in shape: Scaling laws for compute-optimal model design. In *NeurIPS*, 2023.
- [4] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, R. Ring, E. Rutherford, S. Cabi, T. Han, Z. Gong, S. Samangooei, M. Monteiro, J. Menick, S. Borgeaud, A. Brock, A. Nematzadeh, S. Sharifzadeh, M. Binkowski, R. Barreira, O. Vinyals, A. Zisserman, and K. Simonyan. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022.
- [5] J. Bai, S. Bai, S. Yang, S. Wang, S. Tan, P. Wang, J. Lin, C. Zhou, and J. Zhou. Qwen-VL: A versatile vision-language model for understanding, lo-

- calization, text reading, and beyond. *arXiv:2308.12966*, 2023.
- [6] I. Bello, H. Pham, Q. V. Le, M. Norouzi, and S. Bengio. Neural combinatorial optimization with reinforcement learning. *arXiv:1611.09940*, 2016.
- [7] J. Betker, G. Goh, L. Jing, T. Brooks, J. Wang, L. Li, L. Ouyang, J. Zhuang, J. Lee, Y. Guo, et al. Improving image generation with better captions. Technical Report, 2023.
- [8] L. Beyer, X. Zhai, and A. Kolesnikov. Big vision. https://github.com/google-research/big_vision, 2022.
- [9] L. Beyer, A. Steiner, A. S. Pinto, A. Kolesnikov, X. Wang, D. Salz, M. Neumann, I. Alabdulmohsin, M. Tschannen, E. Bugliarello, T. Unterthiner, D. Keysers, S. Koppula, F. Liu, A. Grycner, A. Gritsenko, N. Houlsby, M. Kumar, K. Rong, J. Eisenschlos, R. Kabra, M. Bauer, M. Bošnjak, X. Chen, M. Minderer, P. Voigtlaender, I. Bica, I. Balazevic, J. Puigcerver, P. Papalampidi, O. Henaff, X. Xiong, R. Soricut, J. Harmsen, and X. Zhai. PaliGemma: A versatile 3B VLM for transfer. *arXiv:2407.07726*, 2024.
- [10] A. F. Biten, R. Tito, A. Mafla, L. Gomez, M. Rusinol, C. Jawahar, E. Valveny, and D. Karatzas. Scene text visual question answering. In *ICCV*, Oct. 2019.
- [11] S. Changpinyo, D. Kukliansy, I. Szpektor, X. Chen, N. Ding, and R. Soricut. All you may need for VQA are image captions. In *NAACL*, 2022.
- [12] D. L. Chen and W. B. Dolan. Collecting highly parallel data for paraphrase evaluation. In *ACL*, 2011.
- [13] T. Chen, S. Saxena, L. Li, D. J. Fleet, and G. E. Hinton. Pix2seq: A language modeling framework for object detection. In *ICLR*, 2022.
- [14] X. Chen, X. Wang, S. Changpinyo, A. J. Piergiovanni, P. Padlewski, D. Salz, S. Goodman, A. Grycner, B. Mustafa, L. Beyer, A. Kolesnikov, J. Puigcerver, N. Ding, K. Rong, H. Akbari, G. Mishra, L. Xue, A. Thapliyal, J. Bradbury, W. Kuo, M. Seyedhosseini, C. Jia, B. K. Ayan, C. Riquelme, A. Steiner, A. Angelova, X. Zhai, N. Houlsby, and R. Soricut. PaLI: A jointly-scaled multilingual language-image model. *arXiv:2209.06794*, 2022.
- [15] X. Chen, X. Wang, L. Beyer, A. Kolesnikov, J. Wu, P. Voigtlaender, B. Mustafa, S. Goodman, I. Alabdulmohsin, P. Padlewski, D. Salz, X. Xiong, D. Vlasic, F. Pavetic, K. Rong, T. Yu, D. Keysers, X. Zhai, and R. Soricut. PaLI-3 vision language models: Smaller, faster, stronger. *arXiv:2310.09199*, 2023.
- [16] X. Chen, J. Djolonga, P. Padlewski, B. Mustafa, S. Changpinyo, J. Wu, C. R. Ruiz, S. Goodman, X. Wang, Y. Tay, S. Shakeri, M. Dehghani, D. Salz, M. Lucic, M. Tschannen, A. Nagrani, H. Hu, M. Joshi, B. Pang, C. Montgomery, P. Pietrzyk, M. Ritter, A. J. Piergiovanni, M. Minderer, F. Pavetic, A. Waters, G. Li, I. Alabdulmohsin, L. Beyer, J. Amelot, K. Lee, A. P. Steiner, Y. Li, D. Keysers, A. Arnab, Y. Xu, K. Rong, A. Kolesnikov, M. Seyedhosseini, A. Angelova, X. Zhai, N. Houlsby, and R. Soricut. PaLI-X: On scaling up a multilingual vision and language model. In *CVPR*, 2024.
- [17] C. K. Ch’ng and C. S. Chan. Total-Text: A comprehensive dataset for scene text detection and recognition. In *ICDAR*, 2017.
- [18] W. Dai, J. Li, D. Li, A. M. H. Tiong, J. Zhao, W. Wang, B. Li, P. Fung, and S. Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. *arxiv:2305.06500*, 2023.
- [19] M. Deitke, C. Clark, S. Lee, R. Tripathi, Y. Yang, J. S. Park, M. Salehi, N. Muenighoff, K. Lo, L. Soldaini, et al. Molmo and PixMo: Open weights and open data

- for state-of-the-art multimodal models. *arXiv:2409.17146*, 2024.
- [20] K. Desai and J. Johnson. Virtex: Learning visual representations from textual annotations. In *CVPR*, 2021.
- [21] Gemma Team. Gemma: Open models based on gemini research and technology. *arXiv:2403.08295*, 2024.
- [22] Gemma Team. Gemma 2: Improving open language models at a practical size. *arXiv:2408.00118*, 2024.
- [23] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley. PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. *Circulation*, 101(23), 2000.
- [24] Google Cloud. Introduction to Cloud TPU. <https://cloud.google.com/tpu/docs/intro-to-tpu>, 20xx. Accessed: 2024-07-04.
- [25] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *CVPR*, 2017.
- [26] D. Gurari, Q. Li, A. J. Stangl, A. Guo, C. Lin, K. Grauman, J. Luo, and J. P. Bigham. VizWiz Grand Challenge: Answering visual questions from blind people. In *CVPR*, 2018.
- [27] T.-Y. Hsu, C. L. Giles, and T.-H. Huang. Scicap: Generating captions for scientific figures. *arXiv:2110.11624*, 2021.
- [28] Y. Huang, N. Lu, D. Chen, Y. Li, Z. Xie, S. Zhu, L. Gao, and W. Peng. Improving table structure recognition with visual-alignment sequential coordinate modeling. In *CVPR*, 2023.
- [29] D. Hudson and C. Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. *CVPR*, 2019.
- [30] S. Jain, A. Agrawal, A. Saporta, S. Truong, T. Bui, P. Chambon, Y. Zhang, M. P. Lungren, A. Y. Ng, C. Langlotz, et al. Rad-Graph: Extracting clinical entities and relations from radiology reports. In *NeurIPS Datasets and Benchmarks Track*, 2022.
- [31] C. Jia, Y. Yang, Y. Xia, Y. Chen, Z. Parekh, H. Pham, Q. V. Le, Y. Sung, Z. Li, and T. Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021.
- [32] G. Jocher, J. Qiu, and A. Chaurasia. Ultralytics YOLO, 2023. URL <https://github.com/ultralytics/ultralytics>.
- [33] A. E. Johnson, T. J. Pollard, S. J. Berkowitz, N. R. Greenbaum, M. P. Lungren, C.-Y. Deng, R. G. Mark, and S. Horng. MIMIC-CXR, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317, 2019.
- [34] O. F. Kar, A. Tonioni, P. Poklukar, A. Kulshrestha, A. Zamir, and F. Tombari. BRAVE: Broadening the visual encoding of vision-language models. *arXiv:2404.07204*, 2024.
- [35] S. Karamcheti, S. Nair, A. Balakrishna, P. Liang, T. Kollar, and D. Sadigh. Prismatic VLMs: Investigating the design space of visually-conditioned language models. *arXiv:2402.07865*, 2024.
- [36] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. K. Ghosh, A. D. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu, F. Shafait, S. Uchida, and E. Valveny. ICDAR 2015 competition on robust reading. In *ICDAR*, 2015.
- [37] K. Karkkainen and J. Joo. Fairface: Face attribute dataset for balanced race, gender, and age for bias measurement and mitigation. In *WACV*, 2021.
- [38] T. Kawakatsu. Multi-cell decoder and mutual learning for table structure and character recognition. In *ICDAR*, 2024.

- [39] S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg. ReferItGame: Referring to objects in photographs of natural scenes. In *EMNLP*, Oct. 2014.
- [40] A. Kembhavi, M. Salvato, E. Kolve, M. Seo, H. Hajishirzi, and A. Farhadi. A diagram is worth a dozen images. In *ECCV*, 2016.
- [41] S. Kim, P. A. Thiessen, E. E. Bolton, J. Chen, G. Fu, A. Gindulyte, L. Han, J. He, S. He, B. A. Shoemaker, et al. Pubchem substance and compound databases. *Nucleic acids research*, 44(D1):D1202–D1213, 2016.
- [42] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2017.
- [43] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. Carlos Niebles. Dense-captioning events in videos. In *ICCV*, 2017.
- [44] I. Krylov, S. Nosov, and V. Sovrasov. Open images v5 text annotation and yet another mask text spotter. In *ACCV*, 2021.
- [45] H. Laurençon, L. Tronchon, M. Cord, and V. Sanh. What matters when building vision-language models? *arXiv:2405.02246*, 2024.
- [46] A. Lees, V. Q. Tran, Y. Tay, J. Sorensen, J. Gupta, D. Metzler, and L. Vasserman. A new generation of perspective API: Efficient multilingual character-level transformers. *arXiv:2202.11176*, 2022.
- [47] B. Li, H. Zhang, K. Zhang, D. Guo, Y. Zhang, R. Zhang, F. Li, Z. Liu, and C. Li. LLaVA-NeXT: What else influences visual instruction tuning beyond data?, May 2024. URL <https://llava-vl.github.io/blog/2024-05-25-llava-next-ablations/>.
- [48] J. Li, D. Li, S. Savarese, and S. C. H. Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023.
- [49] Y. Li, G. Li, L. He, J. Zheng, H. Li, and Z. Guan. Widget Captioning: Generating natural language description for mobileuser interface elements. In *EMNLP*, 2020.
- [50] Y. Li, H. Mao, R. Girshick, and K. He. Exploring plain vision transformer backbones for object detection. In *ECCV*, 2022.
- [51] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: common objects in context. *arXiv:1405.0312*, 2014.
- [52] F. Liu, E. Bugliarello, E. M. Ponti, S. Reddy, N. Collier, and D. Elliott. Visually grounded reasoning across languages and cultures. In *EMNLP*, Nov. 2021.
- [53] F. Liu, G. E. T. Emerson, and N. Collier. Visual spatial reasoning. *TACL*, 11:635–651, 2023.
- [54] H. Liu, C. Li, Q. Wu, and Y. J. Lee. Visual instruction tuning. In *NeurIPS*, 2023.
- [55] S. Lobry, D. Marcos, J. Murray, and D. Tuia. RSVQA: Visual question answering for remote sensing data. *IEEE Trans. on Geoscience and Remote Sensing*, 58(12), Dec. 2020.
- [56] S. Long, S. Qin, D. Panteleev, A. Bissacco, Y. Fujii, and M. Raptis. Towards end-to-end unified scene text detection and layout analysis. In *CVPR*, 2022.
- [57] S. Long, S. Qin, D. Panteleev, A. Bissacco, Y. Fujii, and M. Raptis. ICDAR 2023 competition on hierarchical text detection and recognition. In *ICDAR*, 2023.
- [58] S. Long, S. Qin, Y. Fujii, A. Bissacco, and M. Raptis. Hierarchical text spotter for joint text spotting and layout analysis. In *WACV*, 2024.
- [59] P. Lu, S. Mishra, T. Xia, L. Qiu, K.-W. Chang, S.-C. Zhu, O. Tafjord, P. Clark, and A. Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *NeurIPS*, 2022.

- [60] N. T. Ly and A. Takasu. An end-to-end multi-task learning model for image-based table recognition. *arXiv:2303.08648*, 2023.
- [61] J. Mao, J. Huang, A. Toshev, O. Camburu, A. L. Yuille, and K. Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, 2016.
- [62] K. Marino, M. Rastegari, A. Farhadi, and R. Mottaghi. OK-VQA: A visual question answering benchmark requiring external knowledge. In *CVPR*, 2019.
- [63] A. Masry, X. L. Do, J. Q. Tan, S. Joty, and E. Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *ACL*, May 2022.
- [64] M. Mathew, D. Karatzas, R. Manmatha, and C. V. Jawahar. DocVQA: A dataset for VQA on document images. *arXiv:2007.00398*, 2020.
- [65] M. Mathew, V. Bagal, R. Tito, D. Karatzas, E. Valveny, and C. V. Jawahar. InfographicVQA. In *WACV*, 2022.
- [66] B. McKinzie, Z. Gan, J. Fauconnier, S. Dodge, B. Zhang, P. Dufter, D. Shah, X. Du, F. Peng, F. Weers, A. Belyi, H. Zhang, K. Singh, D. Kang, A. Jain, H. Hè, M. Schwarzer, T. Gunter, X. Kong, A. Zhang, J. Wang, C. Wang, N. Du, T. Lei, S. Wiseman, G. Yin, M. Lee, Z. Wang, R. Pang, P. Gräsch, A. Toshev, and Y. Yang. MM1: methods, analysis & insights from multimodal LLM pre-training. *arXiv:2403.09611*, 2024.
- [67] A. Mishra, S. Shekhar, A. K. Singh, and A. Chakraborty. OCR-VQA: Visual question answering by reading text in images. In *ICDAR*, 2019.
- [68] N. Nayef, F. Yin, I. Bizid, H. Choi, Y. Feng, D. Karatzas, Z. Luo, U. Pal, C. Rigaud, J. Chazalon, et al. ICDAR2017 robust reading challenge on multi-lingual scene text detection and script identification - RRC-MLT. In *ICDAR*, 2017.
- [69] Y. Onoe, S. Rane, Z. Berger, Y. Bitton, J. Cho, R. Garg, A. Ku, Z. Parekh, J. Pont-Tuset, G. Tanzer, S. Wang, and J. Baldridge. DOCCI: Descriptions of Connected and Contrasting Images. In *ECCV*, 2024.
- [70] H. Pang. YOLO-DocLayNet, Jan. 2024. URL <https://github.com/ppaanngggg/yolo-doclaynet>.
- [71] D. Pavlov, M. Rybalkin, B. Karulin, M. Kozhevnikov, A. Savelyev, and A. Churinov. Indigo: Universal cheminformatics API. *Journal of Cheminformatics*, 3(Suppl 1):P4, 2011.
- [72] Z. Peng, W. Wang, L. Dong, Y. Hao, S. Huang, S. Ma, and F. Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv:2306.14824*, 2023.
- [73] J. Pfeiffer, G. Geigle, A. Kamath, J.-M. Steitz, S. Roth, I. Vulić, and I. Gurevych. xGQA: Cross-lingual visual question answering. In *ACL*, 2022.
- [74] B. Pfitzmann, C. Auer, M. Dolfi, A. S. Nassar, and P. Staar. DocLayNet: A large human-annotated dataset for document-layout segmentation. In *SIGKDD*, 2022.
- [75] A. Piergiovanni, W. Kuo, and A. Angelova. Pre-training image-language transformers for open-vocabulary tasks. *arXiv:2209.04372*, 2022.
- [76] Y. Qian, J. Guo, Z. Tu, Z. Li, C. W. Coley, and R. Barzilay. MolScribe: Robust molecular structure recognition with image-to-graph generation. *J. Chem. Inf. Model.*, 63(7), 2023.
- [77] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.
- [78] H. Rashkin, V. Nikolaev, M. Lamm, L. Aroyo, M. Collins, D. Das, S. Petrov, G. S. Tomar, I. Turc, and D. Reitter. Measuring

- attribution in natural language generation models. *Computational Linguistics*, 49(4): 777–840, 2023.
- [79] A. Ríos-Vila, D. Rizo, J. M. Iñesta, and J. Calvo-Zaragoza. End-to-end optical music recognition for pianoform sheet music. *IJDAR*, 26(3):347–362, 2023.
- [80] A. Ríos-Vila, J. Calvo-Zaragoza, and T. Paquet. Sheet Music Transformer: End-to-end optical music recognition beyond monophonic transcription. In *ICDAR*, 2024.
- [81] D. Schwenk, A. Khandelwal, C. Clark, K. Marino, and R. Mottaghi. A-OKVQA: A benchmark for visual question answering using world knowledge. *arXiv:2206.01718*, 2022.
- [82] O. Sidorov, R. Hu, M. Rohrbach, and A. Singh. TextCaps: A dataset for image captioning with reading comprehension. In *ECCV*, 2020.
- [83] A. Singh, V. Natarjan, M. Shah, Y. Jiang, X. Chen, D. Parikh, and M. Rohrbach. Towards VQA models that can read. In *CVPR*, 2019.
- [84] A. Singh, G. Pang, M. Toh, J. Huang, W. Galuba, and T. Hassner. TextOCR: Towards large-scale end-to-end reasoning for arbitrary-shaped scene text. In *CVPR*, 2021.
- [85] B. Smock, R. Pesala, and R. Abraham. GriTS: Grid table similarity metric for table structure recognition. *arXiv:2203.12555*, 2022.
- [86] B. Smock, R. Pesala, and R. Abraham. Aligning benchmark datasets for table structure recognition. In *ICDAR*, 2023.
- [87] A. Suhr, S. Zhou, A. Zhang, I. Zhang, H. Bai, and Y. Artzi. A corpus for reasoning about natural language grounded in photographs. In *ACL*, 2019.
- [88] A. Susano Pinto, A. Kolesnikov, Y. Shi, L. Beyer, and X. Zhai. Tuning computer vision models with task rewards. In *ICML*, 2023.
- [89] H. Tan and M. Bansal. LXMERT: Learning cross-modality encoder representations from transformers. In *EMNLP-IJCNLP*, 2019.
- [90] R. Tanno, D. Barrett, A. Sellergren, S. Ghaisas, S. Dathathri, A. See, J. Welbl, K. Singhal, S. Azizi, T. Tu, M. Schaeckermann, R. May, R. Lee, S. Man, Z. Ahmed, S. Mahdavi, Y. Matias, J. Barral, A. Es-lami, D. Belgrave, V. Natarajan, S. Shetty, P. Kohli, P.-S. Huang, A. Karthikesalingam, and I. Ktena. Collaboration between clinicians and vision-language models in radiology report generation. *Nature Medicine*, 2024.
- [91] A. V. Thapliyal, J. Pont Tuset, X. Chen, and R. Soricut. Crossmodal-3600: A massively multilingual multimodal evaluation dataset. In *EMNLP*, 2022.
- [92] S. Tong, E. Brown, P. Wu, S. Woo, M. Middepogu, S. C. Akula, J. Yang, S. Yang, A. Iyer, X. Pan, A. Wang, R. Fergus, Y. LeCun, and S. Xie. Cambrian-1: A Fully Open, Vision-Centric Exploration of Multimodal LLMs. *arXiv:2406.16860*, 2024.
- [93] M. Tschannen, M. Kumar, A. Steiner, X. Zhai, N. Houlsby, and L. Beyer. Image captioners are scalable vision learners too. In *NeurIPS*, 2023.
- [94] B. Wan, M. Tschannen, Y. Xian, F. Pavetic, I. Alabdulmohsin, X. Wang, A. S. Pinto, A. Steiner, L. Beyer, and X. Zhai. LocCa: Visual pretraining with location-aware captioners. In *NeurIPS*, 2024.
- [95] B. Wang, G. Li, X. Zhou, Z. Chen, T. Grossman, and Y. Li. Screen2words: Automatic mobile ui summarization with multimodal learning. In *Symposium on User Interface Software and Technology*, 2021.
- [96] J. Wang, Z. Yang, X. Hu, L. Li, K. Lin, Z. Gan, Z. Liu, C. Liu, and L. Wang. GIT: A generative image-to-text transformer for vision and language. *TMLR*, 2022.

- [97] X. Wang, J. Wu, J. Chen, L. Li, Y.-F. Wang, and W. Y. Wang. VaTeX: A large-scale, high-quality multilingual dataset for video-and-language research. In *ICCV*, 2019.
- [98] Z. Wang, J. Yu, A. W. Yu, Z. Dai, Y. Tsvetkov, and Y. Cao. SimVLM: Simple visual language model pretraining with weak supervision. In *ICLR*, 2022.
- [99] D. Weininger. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, 1988.
- [100] D. Xu, Z. Zhao, J. Xiao, F. Wu, H. Zhang, X. He, and Y. Zhuang. Video question answering via gradually refined attention over appearance and motion. In *ACM Multimedia*, 2017.
- [101] J. Xu, T. Mei, T. Yao, and Y. Rui. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*, 2016.
- [102] L. Yang, S. Xu, A. Sellergren, T. Kohlberger, Y. Zhou, I. Ktena, A. Kiraly, F. Ahmed, F. Hormozdiari, T. Jaroensri, E. Wang, E. Wulczyn, F. Jamil, T. Guidroz, C. Lau, S. Qiao, Y. Liu, A. Goel, K. Park, A. Agharwal, N. George, Y. Wang, R. Tanno, D. G. T. Barrett, W.-H. Weng, S. S. Mahdavi, K. Saab, T. Tu, S. R. Kalidindi, M. Etemadi, J. Cuadros, G. Sorensen, Y. Matias, K. Chou, G. Corrado, J. Barral, S. Shetty, D. Fleet, S. M. A. Eslami, D. Tse, S. Prabhakara, C. McLean, D. Steiner, R. Pilgrim, C. Kelly, S. Azizi, and D. Golden. Advancing multimodal medical capabilities of Gemini. *arXiv:2405.03162*, 2024.
- [103] Q. Ye, H. Xu, J. Ye, M. Yan, A. Hu, H. Liu, Q. Qian, J. Zhang, and F. Huang. mPLUG-Owl2: Revolutionizing multi-modal large language model with modality collaboration. In *CVPR*, 2024.
- [104] H. You, H. Zhang, Z. Gan, X. Du, B. Zhang, Z. Wang, L. Cao, S.-F. Chang, and Y. Yang. Ferret: Refer and ground anything anywhere at any granularity. In *ICLR*, 2024.
- [105] J. Yu, Y. Xu, J. Y. Koh, T. Luong, G. Baid, Z. Wang, V. Vasudevan, A. Ku, Y. Yang, B. K. Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *TMLR*, 2022.
- [106] L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg. Modeling context in referring expressions. In *ECCV*, 2016.
- [107] Z. Yu, D. Xu, J. Yu, T. Yu, Z. Zhao, Y. Zhuang, and D. Tao. ActivityNet-QA: A dataset for understanding complex web videos via question answering. In *AAAI*, 2019.
- [108] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer. Sigmoid loss for language image pre-training. In *ICCV*, 2023.
- [109] H. Zhang, M. Gao, Z. Gan, P. Dufter, N. Wenzel, F. Huang, D. Shah, X. Du, B. Zhang, Y. Li, et al. MM1.5: Methods, analysis & insights from multimodal LLM fine-tuning. *arXiv:2409.20566*, 2024.
- [110] Y. Zhao, A. Gu, R. Varma, L. Luo, C. Huang, M. Xu, L. Wright, H. Shojanazeri, M. Ott, S. Shleifer, A. Desmaison, C. Balioglu, P. Damania, B. Nguyen, G. Chauhan, Y. Hao, A. Mathews, and S. Li. Pytorch FSDP: experiences on scaling fully sharded data parallel. *VLDB*, 2023.
- [111] X. Zheng, D. Burdick, L. Popa, P. Zhong, and N. X. R. Wang. Global Table Extractor (GTE): A framework for joint table identification and cell structure recognition using visual context. In *WACV*, 2021.
- [112] X. Zhong, E. ShafieiBavani, and A. Jimeno Yepes. Image-based table recognition: Data, model, and evaluation. In *ECCV*, 2020.

Contributions and Acknowledgments

Model development contributors

Core Contributors

Andreas Steiner
André Susano Pinto
Michael Tschannen

Contributors

Daniel Keysers
Xiao Wang
Yonatan Bitton
Alexey Gritsenko
Matthias Minderer
Anthony Sherbondy
Shangbang Long
Siyang Qin
Reeve Ingle
Emanuele Bugliarello
Sahar Kazemzadeh
Thomas Mesnard
Ibrahim Alabdulmohsin
Lucas Beyer
Xiaohua Zhai

Lead

Andreas Steiner

Acknowledgments

Jan Wassenberg
Basil Mustafa

Model release contributors and general support

Gemma Model

Tris Warkentin
Alek Andreev
Armand Joulin
Victor Cotruta
Sanah Choudhry
Nathan Byrd

Open Models Success

Luiz Gustavo Martins
Kat Black
Phil Culliton
Chris Perry
D. Sculley
Sara Smoot

Marketing

Glenn Cameron
Natalie Dao

Kaggle

D. Sculley
Nilay Chauhan
Brenda Flynn
Kinjal Parekh

Developer Relations

Jetha Chan
Joe Fernandez
Ju-yeong Ji

Keras

Divyashree Sreepathihalli
Hongyu Chiu

Vertex AI

Keelin McDonell

Ethics and Safety

Antonia Paterson
Pankil Botadra

Hugging Face Partners

Merve Noyan
Pedro Cuenca
Pablo Montalvo

Nvidia Partners

Dong Meng
Manoj Kilaru
Shyamala Prayaga
Ryan Timbrook
Anna Warno

Ollama Partners

Michael Chiang
Jeffrey Morgan

Executive Sponsors

Raia Hadsell
Joelle Barral
Jeremiah Harmsen
Mat Velloso
Allen Hutchison

A. Tasks

This section provides one training example for the transfer tasks that were added in PaliGemma 2 in addition to the tasks considered in [9].



Figure 6 | Test set example from Total-Text [17] with PaliGemma 2 3B 896px² predictions.

In millions	May 31, 2015					
	Securities in an unrealized loss position for less than twelve months		Securities in an unrealized loss position for more than twelve months		Total	
	Gross unrealized losses	Fair value	Gross unrealized losses	Fair Value	Gross unrealized losses	Fair Value
Type of issue:						
General obligation municipal bonds	\$(3.8)	\$535.1	\$(0.5)	\$26.3	\$(4.3)	\$561.4
Revenue municipal bonds	(3.2)	361.6	—	—	(3.2)	361.6
Total	\$(7.0)	\$896.7	\$(0.5)	\$26.3	\$(7.5)	\$923.0

	May 31, 2015					
	Securities in an unrealized loss position for less than twelve months		Securities in an unrealized loss position for more than twelve months		Total	
In millions	Gross unrealized losses	Fair value	Gross unrealized losses	Fair Value	Gross unrealized losses	Fair Value
Type of issue:						
General obligation municipal bonds	\$(3.8)	\$355.1	\$(0.5)	\$26.3	\$(4.3)	\$561.4
Revenue municipal bonds	\$(3.2)	361.6	—	—	(3.2)	361.6
Total	\$(7.0)	\$896.7	\$(0.5)	\$26.3	\$(7.5)	\$923.0

Figure 7 | Original image from FinTabNet [111] with predicted cell content boxes (green), and resulting PaliGemma 2 model prediction.

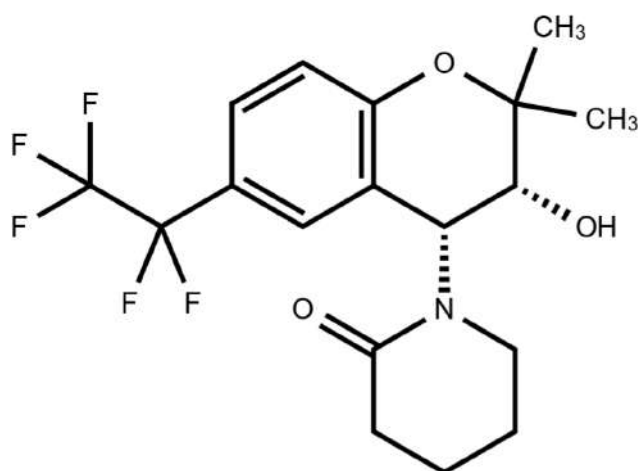


Figure 8 | Example of a rendered molecule with the corresponding SMILES string CC1([C@@H])([C@@H])(C2=C(O1)C=CC(=C2)C(C(F)(F)F)(F)F)N3CCCCC3=O)O)C.

**kern	**kern	
*staff2	*staff1	
*clefF4	*clefG2	
*k[b-]	*k[b-]	
*M3/4	*M3/4	
=1-	=1-	
2.r	8r	
.	8dL	
.	8g	
.	8b-	
.	8g	
.	8dJ	
=2	=2	
8r	4dd	
8GGL	.	
8BB-	4r	
8D	.	
8BB-	4r	
8GGJ	.	
=3	=3	
4GW	8r	
.	8ddL	
8GGL	8b-	
8BB-	8g	
8D	8gg	
8GJ	8b-J	
=4	=4	
4D	8aL	
.	8gg	
4d	8ff	
.	8ee	
4D	8ff	
.	8a-J	
=5	=5	
*-	*-	

Figure 9 | Example of a pianoform sheet with its **kern transcription (source <https://www.humdrum.org/guide/ch02/>).



Model	Caption
InstructBLIP	The image features a brown horse standing in a dirt area, with a saddle on its back. The horse appears to be well-trained, as it is standing still and paying attention to its surroundings. There are several people visible in the scene, including one person standing near the horse's left side, another person standing near the horse's right side, and a third person standing further away from the horse. In addition to the people, there are two cars parked in the background, one on the left side and another on the right side of the image.
mPLUG-Owl2	In this image we can see a horse on the ground. In the background there is a shed and we can see trees.
MiniGPT-4	This image is of a horse standing in the dirt near a building. The horse has a saddle on its back and is wearing a number 5 on its forehead. There is a barn or stable visible in the background, and some trees and shrubs in the foreground.
LLaVA-1.5	The image features a brown horse standing in a dirt field. The horse is wearing a saddle and bridle, indicating that it is ready for riding. The saddle is adorned with stars, adding a unique touch to the horse's appearance. The horse appears to be calmly standing in the field, possibly waiting for its rider or simply enjoying the outdoors.
VILA	In the image, a majestic brown horse stands in a fenced area, its coat gleaming under the sunlight. The horse is adorned with a black saddle and bridle, ready for a ride. The number 55 is prominently displayed on its side, perhaps indicating its racing number or a unique identifier. The horse's tail swishes gently, adding a sense of movement to the scene. The horse is positioned in front of a stone wall, which provides a rustic backdrop to the image. The wall's texture contrasts with the smoothness of the horse's coat, creating a visually interesting composition. The ground beneath the horse is covered in dirt, suggesting that this might be a training or riding area. The fence in the background further reinforces this idea, providing a safe and controlled environment for the horse. Overall, this image captures a moment of calm before what could be an exciting ride, with the horse poised and ready for action.
PaLI-5B (FT on DOCCI)	A side view of a brown horse that is standing on a dirt ground. The horse is facing to the right and has a black saddle on it. The horse has a white star on its face. The horse has a bridle around its neck. The horse is standing on a dirt ground that is covered in white rocks. There is a stone wall behind the horse. There is a metal shed behind the horse. There are trees behind the shed.
PaliGemma 3B 448px ²	A side view of a brown horse that is standing on a dirt ground. The horse is facing to the left and has a black saddle on its back. The saddle is black and has a brown leather strap on it. The horse has a black rope around its neck. The horse has a white star hanging from its bridle. The horse has a white number "55" written on its back. The horse has white hooves. The horse has a black rope wrapped around its legs. The horse has a brown rope wrapped around its head. The horse has a black rope wrapped around its tail. The horse is standing on a dirt ground that has small white rocks on it. Behind the horse is a stone wall that is made up of different sized rocks. Behind the stone wall is a metal gate that is being held up by metal beams. Behind the gate is a tree line that is made up of green trees.
PaliGemma 2 3B 448px ²	A brown horse is standing in a dirt area with rocks scattered around. The horse has a black saddle on its back and a white star painted on its chest. The number "55" is painted on the side of the horse in white. A stone building is behind the horse. A metal structure is in the background of the image with a wooden roof over it. Trees are in the background of the image as well.
PaliGemma 2 10B 448px ²	A brown horse is standing in a dirt area with small rocks. The horse has a black saddle on its back and a white star painted on its side. The horse has "55" written on its back in white. There is a pile of horse manure in front of the horse. There is a stone wall behind the horse. There is a wooden structure with a metal roof behind the stone wall. There are trees in the background.

Figure 10 | Example DOCCI image and captions generated by PaliGemma 2 models and baselines, with non-entailment sentences highlighted in red.



Indication	Radiologist report	PaliGemma 2 3B 896px ² prediction
INDICATION: Woman with cardiomyopathy and cdiff with acute desaturation and dyspnea // PE, pulmonary edema, vs aspiration PE, pulmonary edema, vs aspiration.	IMPRESSION: Enlargement of the cardiac silhouette with pulmonary edema. Bilateral pleural effusions, more prominent on the left.	FINDINGS: There is substantial enlargement of the cardiac silhouette with pulmonary edema. Retrocardiac opacification is consistent with volume loss in the left lower lobe and pleural effusion. In the appropriate clinical setting, superimposed pneumonia would have to be considered.

Figure 11 | Example from the MIMIC-CXR [23, 33] validation set along with a PaliGemma 2 prediction.

B. Transfer and evaluation details

B.1. Text detection and recognition

In all experiments, we fine-tune the checkpoints for 15k steps with a batch size of 256 on 256 TPU-v5e. The maximum sequence length is set to 2048. We experiment with learning rates $\{0.01, 0.05, 0.1, 0.5, 1.0\} \cdot 10^{-4}$ and find that 10^{-5} gives the best results. We also found using a label-smoothing of 0.1 improves the results. The best results are obtained with resolution 896px^2 .

B.2. Table Structure Recognition

We use the same transfer setup and hyperparameter range as for text recognition described in Sec. B.1, except that we set maximum output length to 4096 and do not use label-smoothing. The optimal fine-tuning learning rate is 10^{-4} .

Preprocessing The cropped table input images are padded to square shape with white pixels and resized to the target image resolution. Cell bounding boxes of non-empty table cells are encoded using four PaliGemma location tokens of the form `<locDDDD>`, where DDDD encodes a quantized image location in the range 0000 to 1023. Boxes are specified using a special `coords="<locXMIN><locYMAX><locXMAX><locYMAX>"` attribute of table cell `<td>` HTML tags. Training examples with invalid table structure and overlapping cell bounding boxes are skipped. Additional correction of cell bounding box annotations and cell text annotations are applied to FinTabNet training examples using information from the source PDFs, following a similar approach as [86]. As is common in the literature [38], no filtering is applied to the test splits we report results on.

B.3. Molecule structure recognition

In all experiments, we fine-tune the pretrained checkpoint for 30k steps with batch size 256 using 256 TPU-v5e chips. The learning rate is set to 10^{-4} , label smoothing to 0.1, and the maximum output length is 256. We pad the images to square shape with white pixels and resize them to the target image resolution.

B.4. Optical music score recognition

We follow the training setup described in Sec. B.3 except that we use maximum output length 1024.

B.5. Generating long, fine-grained captions (DOCCI)

We rely on the transfer protocol and hyperparameters suggested in [9, Sec. 3.2.4.].

Human evaluation protocol To evaluate the factual grounding of the generated captions, we conduct human evaluations assessing the relationship between each sentence and the corresponding image. Raters are presented with highlighted sentences and asked, “What is the relationship of the highlighted sentence with respect to the image?”. They then select from four options: “Entailment”, “Neutral”, “Contradiction”, and “Nothing to assess”, categories adapted from the framework in [78] for evaluating the factual alignment of text and visual content. For example, the statement “The pig has black, rounded hooves on its front and back feet and a pink nose” (Fig. 12) would be rated as “Contradiction”, as the image clearly shows pink hooves. Figure 1 illustrates the annotation interface.

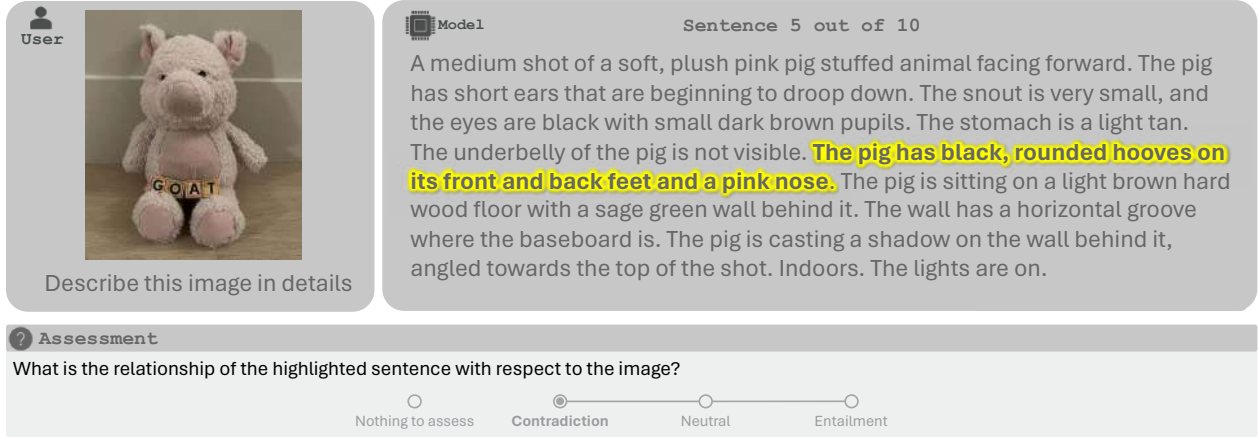


Figure 12 | Annotation interface used for human evaluation of image description accuracy. Raters assess the relationship between generated sentences and the corresponding image.

Each sentence was rated by five individuals and the majority agreement was used as the rating result. The overall binary agreement is 0.8407, indicating the proportion where all raters agree on the “Entailment” category. We refer to both “Contradiction” and “Neutral” as “Non-entailment”. Examples of human evaluation results can be found in Table 4. We use the proportion of “Non-entailment” sentences to select the most factually accurate models.

B.6. Spatial reasoning

We fine-tune the pretrained checkpoint with batch size 1024 using 64 TPU-v5e chips. The maximum output length is set to 18, which covers the training target outputs. We explore learning rates in $\{0.1, 0.2, 1.0, 3.0\} \cdot 10^{-6}$, weight decay in $\{0.1, 0.3, 1.0\} \cdot 10^{-6}$, dropout probability in $\{0.0, 0.1, 0.2\}$ and epochs in $\{1, 3, 5, 10, 15, 30\}$.

B.7. Radiography report generation

Reports in MIMIC-CXR dataset [23, 33] typically have the format INDICATIONS: . . . FINDINGS: { . . . }. IMPRESSIONS: { . . . }, where indications explain why the chest X-ray was ordered as clinical context for the radiologist, findings enumerate salient features of the image and impressions summarize the radiologist’s interpretation of the findings.

We train on the full reports and during prediction emulate the clinical workflow by providing the indications as a prefix to the model. The model then predicts findings and impressions sections.

After initial exploration based on the PaliGemma 2 at 448px² resolution we find that fine-tuning for 8 epochs with learning rate $5 \cdot 10^{-6}$ without label smoothing, dropout, and weight decay leads to good results when combined with greedy decoding. We fix these settings and sweep the learning rate again for higher resolutions and model sizes, considering learning rates in $\{0.03, 0.1, 0.3, 1.0, 5.0\} \cdot 10^{-4}$.

C. Object detection

Object detection has been used as a pre-training task in all members of the PaLI and PaliGemma family and improves downstream performance across a wide range of tasks [14]. In transfers, PaliGemma performs at or close to the state of the art on localization tasks such as referring expression comprehension and segmentation. This raises the question of how well PaliGemma performs on

	224px ²			448px ²			896px ²		
	PG1 3B	PG2 3B	PG2 10B	PG1 3B	PG2 3B	PG2 10B	PG1 3B	PG2 3B	PG2 10B
COCO	28.7	30.4	30.3	37.0	38.5	39.2	41.1	42.3	43.6
DocLayNet	50.8	46.7	50.4	64.1	62.5	63.5	66.5	66.1	66.0

Table 11 | Mean average precision (mAP) after transfer to detection tasks. PG1 and PG2 refer to PaliGemma [9] and PaliGemma 2, respectively.

classical object detection tasks. We tested this by transferring PaliGemma to MS COCO [51] and to the DocLayNet document layout detection benchmark [74].

For both tasks, we use a transfer strategy inspired by pix2seq’s *sequence augmentation* approach [13]. We use the prefix "detect all classes\n". In the suffix (target sequence), we first provide box coordinates and class names for all annotated objects, in random order. The suffix is then filled up to the maximum sequence length with *noise boxes*, where each noise box consists of random coordinates and a dedicated <noise> token in place of the class name. During training, no loss is applied to the coordinate tokens of the noise boxes, while the <noise> class tokens receive a loss as usual. This augmentation trains the model to output a larger number of boxes. In addition, it provides a mechanism for the model to represent the confidence that a prediction represents a real object, in form of the probability assigned to the <noise> token. During inference, the <noise> and <EOS> tokens are excluded from sampling. The likelihood of the class tokens is used as a confidence score.

For COCO, we train for 50 epochs. Results are provided in Table 11. As expected, performance strongly depends on resolution. We also observe small but consistent improvements from better language models. Performance at 896px² is roughly on par with prior sequence-based approaches [13], but lags behind specialized detection architectures like ViTDet [50].

For DocLayNet, we follow the same sequence augmentation approach and train for 50 epochs. Results are similar to COCO in that performance increases with resolution and Gemma 2 model size, although Gemma 1 performs on par with Gemma 2 on this task (Table 11). Similar to COCO, specialized detectors perform better on this task (e.g. YOLOv11 [32] reaches 79.5 mAP [70]).

These results show that, in contrast to many other tasks, classical detection poses a challenge to general-purpose VLMs like PaliGemma. We hypothesize that the limiting factor is not the model’s intrinsic object understanding, since it performs well on visual question answering and referring expression comprehension tasks. Instead, performance may be limited by a mismatch between the Average Precision metric, which rewards large numbers of predictions and accurate confidence scores, and the language modeling objective. Fine-tuning with a task-specific reward [88]) could address this limitation, but is beyond the scope of the simple transfer approach we propose for PaliGemma.

D. Ethics and Safety

Besides quality-related metrics, we also evaluate the new PaliGemma 2 VLMs with respect to a number of categories relevant to ethics and safety. These evaluations include prompts covering child safety, content safety and representational harms, following the approach used in Gemma 2 [22], but with image captioning and visual question answering (VQA) setups.

In addition, we also follow the setup used in [15] and use the Perspective API [46] with threshold > 0.8 to detect the presence of toxicity, profanity, among other potential issues in the image captions generated by PaliGemma 2 VLMs across images sourced from the Fairface dataset [37]. We report the

Metric	Perceived Gender			Ethnicity			Age Group		
	3B	10B	28B	3B	10B	28B	3B	10B	28B
Maximum									
Toxicity	0.14	0.15	0.19	0.29	0.39	0.39	0.26	0.18	0.32
Identity Attack	0.04	0.02	0.02	0.13	0.06	0.06	0.06	0.03	0.06
Insult	0.17	0.25	0.17	0.37	0.52	0.52	0.27	0.39	0.24
Threat	0.55	0.43	0.57	0.83	0.48	0.48	0.64	0.43	0.64
Profanity	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Median									
Toxicity	0.13	0.10	0.18	0.07	0.07	0.14	0.12	0.08	0.12
Identity Attack	0.02	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00
Insult	0.15	0.23	0.14	0.14	0.17	0.13	0.09	0.18	0.16
Threat	0.35	0.27	0.41	0.28	0.19	0.42	0.27	0.31	0.40
Profanity	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 12 | Safety statistics for captions generated by PaliGemma 2 VLMs on FairFace [37] using the Perspective API [46]. Numbers indicate the fraction of instances with thresholds ≥ 0.8 in [%], i.e. a value of e.g. 0.09 means 0.09%.

maximum and median values observed across subgroups for each of the perceived gender, ethnicity, and age attributes. Table 12 shows the overall results. Overall, we observe a low level of toxicity and profanity among others, across all slices and models. In addition, all PaliGemma 2 models perform comparably.

E. Detailed results

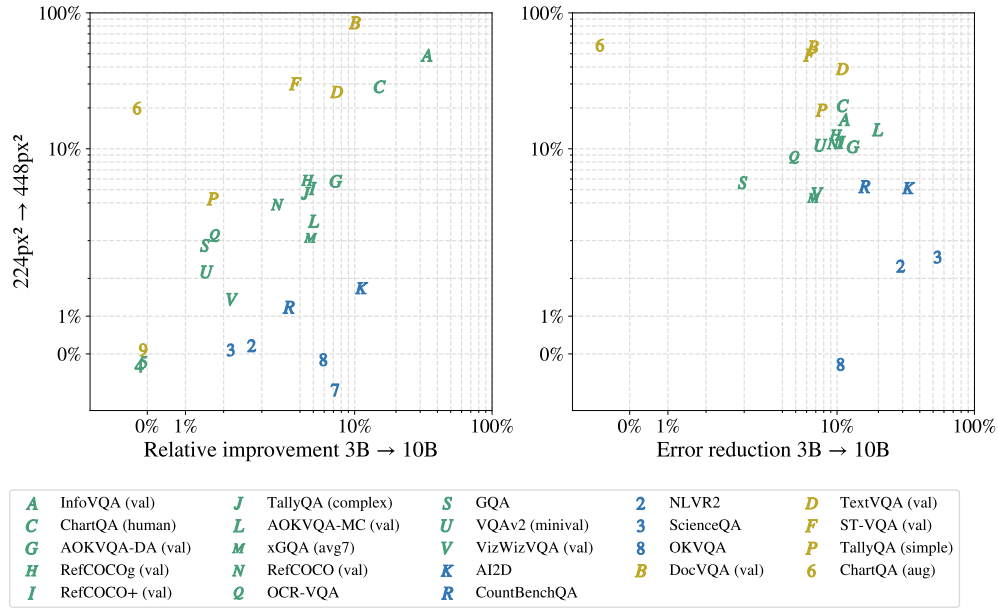


Figure 13 | Same data as in Figure 3 and Table 13. The left plot shows relative improvement when changing model size or resolution. The right plot shows the same improvements, but expressed in terms of error reduction. For saturated benchmarks, error reduction is a better metric for model improvement. Benchmarks without a clear normalization to a percentage (such as CIDEr scores) are not shown. Axes are in range $[-1, 100]$.

	224px ²			448px ²		
	3B	10B	28B	3B	10B	28B
AI2D [40]	74.7 (±0.5)	83.1 (±0.4)	83.2 (±0.7)	76.0 (±0.2)	84.4 (±0.4)	84.6 (±0.4)
AOKVQA-DA (val) [81]	64.2 (±0.5)	68.9 (±0.3)	70.2 (±0.2)	67.9 (±0.3)	70.8 (±0.5)	71.2 (±0.2)
AOKVQA-MC (val) [81]	79.7 (±1.0)	83.7 (±1.1)	84.7 (±0.8)	82.5 (±0.4)	85.9 (±0.2)	87.0 (±0.3)
ActivityNet-CAP [43]	34.2 (±0.3)	35.9 (±0.5)	-	-	-	-
ActivityNet-QA [107]	51.3 (±0.2)	53.2 (±0.4)	-	-	-	-
COCO-35L (avg34) [91]	113.9 (±0.2)	115.8 (±0.0)	116.5 (±0.1)	115.8 (±0.3)	117.2 (±0.1)	117.2 (±0.1)
COCO-35L (en) [91]	138.4 (±0.2)	140.8 (±0.3)	142.4 (±0.4)	140.4 (±0.4)	142.4 (±0.4)	142.3 (±0.8)
COCOCap[51]	141.3 (±0.5)	143.7 (±0.2)	144.0 (±0.3)	143.4 (±0.4)	145.0 (±0.3)	145.2 (±0.4)
ChartQA (aug) [63]	74.4 (±0.7)	74.2 (±0.8)	68.9 (±0.6)	89.2 (±0.4)	90.1 (±0.5)	85.1 (±0.2)
ChartQA (human) [63]	42.0 (±0.3)	48.4 (±1.1)	46.8 (±0.6)	54.0 (±0.6)	66.4 (±0.5)	61.3 (±0.6)
CountBenchQA [9]	81.0 (±1.0)	84.0 (±1.4)	86.4 (±1.6)	82.0 (±1.2)	85.3 (±1.7)	87.4 (±1.0)
DocVQA (val) [64]	39.9 (±0.3)	43.9 (±0.6)	44.9 (±0.4)	73.6 (±0.3)	76.6 (±0.5)	76.1 (±0.4)
GQA[29]	66.2 (±0.3)	67.2 (±0.2)	67.3 (±0.2)	68.1 (±0.2)	68.3 (±0.3)	68.3 (±0.1)
InfoVQA (val) [65]	25.2 (±0.2)	33.6 (±0.2)	36.4 (±0.1)	37.5 (±0.3)	47.8 (±0.2)	46.7 (±0.4)
MARVL (avg5) [52]	83.5 (±0.2)	89.5 (±0.2)	90.6 (±0.2)	82.7 (±0.3)	89.1 (±0.0)	89.7 (±0.1)
MSRVTT-CAP [101]	68.5 (±1.3)	72.1 (±0.5)	-	-	-	-
MSRVTT-QA [100]	50.5 (±0.1)	51.9 (±0.1)	-	-	-	-
MSVD-QA [12]	61.1 (±0.2)	62.5 (±0.2)	-	-	-	-
NLVR2 [87]	91.4 (±0.1)	93.9 (±0.2)	94.2 (±0.1)	91.6 (±0.2)	93.7 (±0.2)	94.1 (±0.2)
NoCaps [2]	123.1 (±0.3)	126.3 (±0.4)	127.1 (±0.3)	123.5 (±0.3)	126.9 (±0.1)	127.0 (±0.2)
OCR-VQA [67]	73.4 (±0.0)	74.7 (±0.1)	75.3 (±0.2)	75.7 (±0.1)	76.3 (±0.1)	76.6 (±0.1)
OKVQA [62]	64.2 (±0.1)	68.0 (±0.1)	71.2 (±0.2)	64.1 (±0.4)	68.6 (±0.5)	70.6 (±0.2)
RSVQA-hr (test) [55]	92.7 (±0.1)	92.6 (±0.0)	92.7 (±0.0)	92.8 (±0.0)	92.8 (±0.1)	92.8 (±0.1)
RSVQA-hr (test2) [55]	90.9 (±0.1)	90.8 (±0.1)	90.9 (±0.1)	90.7 (±0.2)	90.7 (±0.2)	90.8 (±0.1)
RSVQA-lr [55]	93.0 (±0.4)	92.8 (±0.6)	93.5 (±0.2)	92.7 (±0.8)	93.1 (±0.6)	93.7 (±0.4)
RefCOCO (testA) [106]	75.7 (±0.2)	77.2 (±0.1)	76.8 (±0.1)	78.6 (±0.3)	79.7 (±0.1)	79.3 (±0.1)
RefCOCO (testB) [106]	71.0 (±0.3)	74.2 (±0.3)	73.9 (±0.1)	73.5 (±0.1)	76.2 (±0.3)	74.8 (±0.1)
RefCOCO (val) [106]	73.4 (±0.1)	75.9 (±0.1)	75.0 (±0.0)	76.3 (±0.1)	78.2 (±0.1)	77.3 (±0.1)
RefCOCO+ (testA) [39]	72.7 (±0.2)	74.7 (±0.2)	73.6 (±0.2)	76.1 (±0.2)	77.7 (±0.2)	76.6 (±0.1)
RefCOCO+ (testB) [39]	64.2 (±0.2)	68.4 (±0.3)	67.1 (±0.1)	67.0 (±0.3)	71.1 (±0.2)	68.6 (±0.1)
RefCOCO+ (val) [39]	68.6 (±0.1)	72.0 (±0.2)	70.3 (±0.2)	72.1 (±0.3)	74.4 (±0.1)	72.8 (±0.1)
RefCOCOg (test) [61]	69.0 (±0.2)	71.9 (±0.1)	70.7 (±0.1)	72.7 (±0.1)	74.8 (±0.1)	73.7 (±0.1)
RefCOCOg (val) [61]	68.3 (±0.3)	71.4 (±0.2)	70.5 (±0.1)	72.3 (±0.2)	74.4 (±0.1)	73.0 (±0.1)
ST-VQA (val) [10]	61.9 (±0.1)	64.3 (±0.4)	65.1 (±0.4)	80.5 (±0.1)	82.0 (±0.3)	81.8 (±0.1)
SciCap [27]	165.1 (±0.5)	159.5 (±0.7)	156.9 (±1.0)	183.3 (±0.7)	177.2 (±0.3)	172.7 (±1.5)
ScienceQA [59]	96.1 (±0.3)	98.2 (±0.2)	98.2 (±0.2)	96.2 (±0.2)	98.5 (±0.2)	98.6 (±0.2)
Screen2Words [95]	113.3 (±0.8)	117.8 (±0.7)	122.8 (±0.5)	114.0 (±0.5)	119.1 (±1.9)	123.4 (±0.8)
TallyQA (complex) [1]	70.3 (±0.3)	73.4 (±0.1)	74.2 (±0.1)	73.6 (±0.2)	76.7 (±0.3)	76.8 (±0.2)
TallyQA (simple) [1]	81.8 (±0.1)	83.2 (±0.1)	83.4 (±0.1)	85.3 (±0.1)	86.2 (±0.1)	85.7 (±0.1)
TextCaps [82]	127.5 (±0.3)	137.9 (±0.3)	139.9 (±0.4)	152.1 (±0.3)	157.7 (±0.7)	153.6 (±0.5)
TextVQA (val) [83]	59.6 (±0.3)	64.0 (±0.3)	64.7 (±0.2)	75.2 (±0.2)	76.6 (±0.1)	76.2 (±0.1)
VATEX [97]	80.8 (±0.4)	82.7 (±0.5)	-	-	-	-
VQAv2 (minival) [25]	83.0 (±0.2)	84.3 (±0.2)	84.5 (±0.1)	84.8 (±0.2)	85.8 (±0.1)	85.8 (±0.2)
VizWizVQA (val) [26]	76.4 (±0.4)	78.1 (±0.4)	78.7 (±0.2)	77.5 (±0.2)	78.6 (±0.4)	78.9 (±0.5)
WidgetCap [49]	138.1 (±0.7)	139.8 (±1.0)	138.8 (±0.8)	151.4 (±0.8)	151.9 (±0.4)	148.9 (±0.7)
XM3600 (avg35) [91]	42.8 (±0.1)	44.5 (±0.1)	45.2 (±0.1)	43.2 (±0.1)	44.6 (±0.1)	45.2 (±0.1)
XM3600 (en) [91]	79.8 (±0.7)	80.7 (±0.3)	81.0 (±0.9)	80.3 (±0.8)	81.5 (±0.4)	81.0 (±0.2)
xGQA (avg7) [73]	58.6 (±0.2)	61.4 (±0.1)	61.1 (±0.1)	60.4 (±0.2)	62.6 (±0.2)	62.1 (±0.3)

Table 13 | Mean and std-deviation over 5 finetuning runs of PaliGemma 3B, 10B, 28B models at 224px² and 448px² resolutions on over 30+ academic tasks from [9]. Tasks splits, preprocessing, metrics and hyper-parameters following the 224px² versions according to previous work. Only the learning rate has been selected per model size based on validation splits.

Table 14 | Sweep of learning rates on the various tasks and model sizes at 224px² resolution. Although we report numbers in all metrics, learning rate selection was done based on the validation split and not on the zero-shot numbers.

Task	Model	3e-7	6e-7	1e-6	3e-6	6e-6	1e-5	3e-5
AI2D (minival)	3B	61.8	67.6	70.6	75.0	76.9	75.1	68.8
	10B	80.0	82.9	85.3	84.4	82.9	82.1	69.2
	28B	81.9	82.3	83.2	85.9	85.0	83.4	75.7
AOKVQA-DA (val)	3B	59.3	62.9	64.0	64.6	63.6	59.3	52.8
	10B	67.7	68.6	68.8	66.6	64.6	57.3	50.5
	28B	69.7	70.2	69.8	69.0	66.3	60.8	51.1
AOKVQA-MC (val)	3B	76.9	78.7	79.4	80.8	77.2	76.9	63.8
	10B	83.8	83.3	83.3	82.7	79.4	75.5	56.1
	28B	83.3	84.0	85.1	82.5	82.4	78.2	58.4
ActivityNet-CAP (minival)	3B	26.1	28.5	28.5	30.6	30.0	30.6	29.8
	10B	28.6	31.4	30.8	31.6	30.0	31.1	28.6
ActivityNet-QA (minival)	3B	43.3	46.8	49.4	52.6	53.8	53.5	52.0
	10B	49.9	52.2	53.9	55.0	55.3	54.6	51.2
COCO-35L (avg34)	3B	110.1	111.8	113.6	113.9	113.6	113.2	111.7
	10B	115.4	115.8	115.2	113.6	112.9	112.2	111.7
	28B	116.7	116.6	115.4	114.0	112.1	111.2	109.6
COCO-35L (en)	3B	137.9	138.6	139.1	138.4	137.6	136.5	133.8
	10B	140.6	140.3	139.6	137.3	135.5	133.8	132.5
	28B	142.5	141.3	140.4	137.7	134.5	133.2	129.9
COCOcap (minival)	3B	146.3	146.7	145.4	147.2	147.1	147.0	142.0
	10B	148.3	149.4	148.2	148.3	147.0	146.5	143.6
	28B	148.8	149.5	149.2	149.5	148.2	145.3	145.7
ChartQA (aug) (minival)	3B	60.8	64.3	66.0	69.7	69.5	68.4	63.6
	10B	69.0	68.6	71.1	69.5	69.9	68.4	60.4
	28B	66.8	63.4	65.2	66.7	66.0	64.1	55.9
ChartQA (human) (minival)	3B	41.4	42.8	42.7	44.1	43.2	42.9	35.4
	10B	50.9	50.8	50.8	49.2	47.0	44.5	34.6
	28B	48.3	46.9	47.7	46.5	45.3	41.8	33.8
CountBenchQA	3B	82.7	82.9	82.0	79.0	82.0	78.0	70.4
	10B	88.2	84.7	85.1	82.9	81.4	78.2	65.7
	28B	87.8	88.4	88.4	88.6	86.7	83.3	69.6
DocVQA (val)	3B	37.8	37.9	37.3	39.4	40.2	38.7	32.5
	10B	42.4	40.9	42.2	44.1	41.4	39.8	29.6
	28B	42.7	42.1	43.1	45.2	42.1	40.5	30.9
GQA (minival)	3B	70.9	72.2	72.9	73.9	73.9	73.8	72.4
	10B	73.6	74.3	74.7	74.4	74.4	74.2	71.5
	28B	73.7	73.9	74.7	74.8	74.6	74.1	72.3
InfoVQA (val)	3B	21.6	22.9	23.8	25.4	25.2	25.1	22.3
	10B	33.4	33.5	33.2	33.2	32.2	29.8	21.7
	28B	36.9	36.6	36.3	36.2	35.5	34.1	25.4
MARVL (avg5)	3B	69.9	73.4	77.1	81.2	83.0	82.4	69.9
	10B	86.5	88.2	89.2	89.4	89.1	87.4	67.6
	28B	86.7	88.5	89.5	90.3	90.8	89.2	76.2
MSRVTT-CAP (minival)	3B	62.8	66.1	67.8	67.6	72.6	74.0	68.3
	10B	70.4	71.5	75.3	74.0	66.2	69.4	67.2
MSRVTT-QA (minival)	3B	44.1	47.0	48.5	51.1	52.0	51.2	49.9
	10B	49.3	51.2	51.9	53.2	53.1	52.1	49.7

Continued on next page

Table 14 | Sweep of learning rates on the various tasks and model sizes at 224px² resolution. Although we report numbers in all metrics, learning rate selection was done based on the validation split and not on the zero-shot numbers.

Task	Model	3e-7	6e-7	1e-6	3e-6	6e-6	1e-5	3e-5
MSVD-QA (minival)	3B	55.2	57.8	60.7	63.3	63.1	61.3	57.0
	10B	61.1	63.9	65.4	64.2	63.2	63.0	56.3
NLVR2 (minival)	3B	82.5	86.2	88.2	90.4	90.9	90.2	85.9
	10B	91.8	93.0	93.3	93.3	92.5	91.7	86.1
	28B	92.2	92.8	93.6	93.7	93.7	92.2	88.0
NoCaps	3B	123.3	123.6	124.0	123.4	122.5	120.5	112.3
	10B	126.7	126.1	126.0	125.2	122.1	120.5	111.5
	28B	127.5	127.5	126.5	124.0	123.0	120.3	113.0
OCR-VQA (minival)	3B	72.6	73.1	73.4	73.4	73.2	72.9	70.6
	10B	74.7	74.5	74.3	73.9	73.5	73.0	70.6
	28B	75.5	75.5	75.2	74.8	73.9	72.5	71.0
OKVQA (minival)	3B	49.4	52.3	54.3	57.6	56.2	52.9	47.2
	10B	57.8	60.5	61.3	60.8	58.7	55.6	44.1
	28B	64.6	64.4	65.4	63.8	60.6	56.8	46.4
RSVQA-hr (minival)	3B	92.8	93.2	93.3	93.0	93.3	93.4	93.3
	10B	93.3	93.2	93.1	93.0	93.4	93.3	89.4
	28B	93.1	93.4	93.3	93.3	93.3	93.3	92.9
RSVQA-lr (minival)	3B	90.7	92.4	92.7	93.3	92.1	92.2	92.3
	10B	92.3	92.7	92.0	91.7	91.8	92.8	92.0
	28B	91.8	92.1	92.4	92.7	92.9	92.9	92.3
RefCOCO (testA)	3B	73.1	74.5	75.3	75.5	75.8	75.8	74.1
	10B	76.7	76.9	77.1	77.2	77.1	76.1	71.6
	28B	76.2	76.7	76.8	76.8	76.6	75.5	71.6
RefCOCO (testB)	3B	68.0	70.1	70.8	71.2	70.8	70.9	69.7
	10B	73.8	74.3	74.3	74.2	73.4	73.4	68.6
	28B	73.0	73.9	73.8	72.8	73.1	72.0	68.4
RefCOCO (val)	3B	70.4	72.1	73.0	73.2	73.3	73.4	71.6
	10B	75.1	75.6	75.8	76.1	75.6	74.9	70.6
	28B	74.6	75.0	75.2	74.8	74.6	74.0	69.9
RefCOCO+ (testA)	3B	67.6	70.1	70.8	71.8	72.2	72.7	71.0
	10B	72.9	73.5	74.0	75.0	74.9	74.2	69.0
	28B	72.7	73.4	73.4	74.0	74.3	72.9	69.3
RefCOCO+ (testB)	3B	55.3	58.6	60.5	62.9	63.2	64.6	63.8
	10B	66.0	67.1	67.3	68.4	68.2	67.9	62.6
	28B	65.3	66.4	67.1	67.5	67.8	67.0	62.7
RefCOCO+ (val)	3B	61.3	64.2	65.8	67.0	67.9	68.6	67.5
	10B	69.8	70.8	71.1	72.0	71.8	71.3	66.5
	28B	69.0	70.0	70.4	70.8	71.0	70.4	65.7
RefCOCOg (test)	3B	65.5	67.2	68.4	68.7	68.9	69.0	67.2
	10B	70.9	71.6	71.6	71.7	71.3	70.4	65.2
	28B	69.9	70.5	70.8	70.7	70.6	69.7	64.9
RefCOCOg (val)	3B	65.2	67.0	67.8	68.0	68.0	68.2	66.1
	10B	70.8	71.4	71.4	71.4	71.0	70.0	64.9
	28B	69.9	70.4	70.2	70.2	70.1	69.2	64.0
ST-VQA (val)	3B	56.1	58.8	60.4	61.5	62.3	61.2	57.0
	10B	60.9	62.9	63.8	64.0	63.9	61.2	54.8
	28B	63.0	64.4	65.2	65.5	64.3	62.6	55.7

Continued on next page

Table 14 | Sweep of learning rates on the various tasks and model sizes at 224px² resolution. Although we report numbers in all metrics, learning rate selection was done based on the validation split and not on the zero-shot numbers.

Task	Model	3e-7	6e-7	1e-6	3e-6	6e-6	1e-5	3e-5
SciCap (minival)	3B	55.2	67.4	76.9	109.4	130.3	138.8	148.1
	10B	78.6	92.5	106.2	128.1	136.9	143.2	143.8
	28B	80.3	94.7	104.0	125.9	136.2	140.1	141.7
ScienceQA (minival)	3B	87.7	92.1	94.5	95.1	95.2	94.3	91.4
	10B	96.9	97.1	97.6	97.6	97.1	96.2	93.7
	28B	96.8	97.1	97.4	97.2	96.8	96.1	94.2
Screen2Words (minival)	3B	95.1	104.2	109.0	109.3	113.2	112.5	110.1
	10B	110.9	115.4	118.2	118.1	114.7	113.0	110.0
	28B	113.0	119.5	120.4	118.8	116.2	114.2	106.3
TallyQA (complex)	3B	66.6	67.8	68.6	70.0	70.0	70.5	66.7
	10B	72.0	72.5	73.4	73.5	72.7	72.0	65.8
	28B	73.1	73.5	73.9	74.8	73.8	73.0	68.1
TallyQA (simple)	3B	80.4	81.1	81.3	81.8	81.9	81.5	79.1
	10B	83.0	83.3	83.1	83.2	82.7	82.1	79.1
	28B	82.9	83.3	83.3	83.5	83.0	82.2	79.7
TextCaps (minival)	3B	122.8	131.9	136.5	136.2	133.6	132.8	126.0
	10B	140.3	145.3	145.4	145.4	144.2	141.0	125.8
	28B	150.9	149.0	150.2	145.5	144.0	142.1	126.2
TextVQA (val)	3B	57.6	58.7	59.3	59.6	59.4	58.0	51.1
	10B	63.4	64.1	63.9	63.2	61.6	58.1	48.3
	28B	64.5	64.7	65.3	64.8	63.3	59.3	49.9
VATEX (minival)	3B	84.4	87.2	89.8	90.7	90.2	90.2	86.3
	10B	91.4	93.2	93.4	93.7	90.4	89.9	84.5
	28B	80.9	81.5	82.1	82.7	82.4	81.9	79.6
	10B	83.8	84.1	84.3	83.7	83.1	82.0	79.4
	28B	83.8	84.1	84.1	83.8	82.8	82.0	79.7
VizWizVQA (val)	3B	72.5	74.2	74.8	76.4	76.6	76.7	74.0
	10B	76.1	77.1	77.8	78.0	77.3	77.2	73.3
	28B	76.3	77.6	78.2	78.8	77.8	76.7	72.5
WidgetCap (minival)	3B	137.0	141.9	141.8	142.3	141.7	140.6	129.7
	10B	146.3	148.4	150.9	148.2	144.5	140.8	133.3
	28B	144.0	147.6	145.9	147.0	144.1	143.0	133.0
XM3600 (avg35)	3B	44.2	43.9	43.7	42.7	41.7	40.8	37.8
	10B	45.0	44.5	43.9	42.1	40.7	39.3	36.8
	28B	45.2	44.6	44.0	42.3	41.1	39.1	35.8
	3B	83.7	83.1	82.2	79.1	78.3	76.9	70.9
	10B	82.5	80.6	78.6	75.0	73.0	72.0	69.9
	28B	80.9	79.8	79.4	76.4	73.6	71.3	66.1
xGQA (avg7)	3B	51.7	54.0	55.3	58.0	58.7	57.8	49.1
	10B	58.5	60.5	61.4	61.3	61.8	60.2	38.0
	28B	58.8	59.2	60.8	62.3	61.9	61.7	49.4

Task	224px ²		448px ²	
	PG1	PG2	PG1	PG2
AI2D	72.1	74.7 (+2.6)	73.3	76.0 (+2.7)
AOKVQA-DA (val)	61.1	64.2 (+3.1)	65.7	67.9 (+2.2)
AOKVQA-MC (val)	78.5	79.7 (+1.2)	80.3	82.5 (+2.2)
ActivityNet-CAP	34.6	34.2 (−0.4)	-	-
ActivityNet-QA	50.8	51.3 (+0.5)	-	-
COCO-35L (avg34)	113.7	113.9 (+0.2)	115.8	115.8 (+0.0)
COCO-35L (en)	139.2	138.4 (−0.8)	141.2	140.4 (−0.8)
COCOpap	141.9	141.3 (−0.6)	144.6	143.4 (−1.2)
ChartQA (aug)	74.2	74.4 (+0.2)	88.5	89.2 (+0.7)
ChartQA (human)	40.0	42.0 (+2.0)	54.2	54.0 (−0.2)
CountBenchQA	81.9	81.0 (−0.9)	83.1	82.0 (−1.1)
DocVQA (val)	37.8	39.9 (+2.1)	74.1	73.6 (−0.5)
GQA	65.6	66.2 (+0.6)	67.0	68.1 (+1.1)
InfoVQA (val)	25.5	25.2 (−0.3)	37.0	37.5 (+0.5)
MARVL (avg5)	80.6	83.5 (+2.9)	76.8	82.7 (+5.9)
MSRVTT-CAP	70.5	68.5 (−2.0)	-	-
MSRVTT-QA	50.1	50.5 (+0.4)	-	-
MSVD-QA	60.2	61.1 (+0.9)	-	-
NLVR2	90.0	91.4 (+1.4)	88.9	91.6 (+2.7)
NoCaps	121.7	123.1 (+1.4)	123.6	123.5 (−0.1)
OCR-VQA	72.3	73.4 (+1.1)	74.6	75.7 (+1.1)
OKVQA	63.5	64.2 (+0.7)	63.2	64.1 (+0.9)
RSVQA-hr (test)	92.6	92.7 (+0.1)	92.8	92.8 (+0.0)
RSVQA-hr (test2)	90.6	90.9 (+0.3)	90.5	90.7 (+0.2)
RSVQA-lr	92.6	93.0 (+0.4)	93.1	92.7 (−0.4)
RefCOCO (testA)	75.7	75.7 (+0.0)	77.9	78.6 (+0.7)
RefCOCO (testB)	70.7	71.0 (+0.3)	72.4	73.5 (+1.1)
RefCOCO (val)	73.4	73.4 (+0.0)	75.6	76.3 (+0.7)
RefCOCO+ (testA)	71.9	72.7 (+0.8)	74.2	76.1 (+1.9)
RefCOCO+ (testB)	64.5	64.2 (−0.3)	64.5	67.0 (+2.5)
RefCOCO+ (val)	68.3	68.6 (+0.3)	69.8	72.1 (+2.3)
RefCOCOg (test)	68.2	69.0 (+0.8)	71.0	72.7 (+1.7)
RefCOCOg (val)	67.7	68.3 (+0.6)	70.1	72.3 (+2.2)
ST-VQA (val)	61.6	61.9 (+0.3)	79.7	80.5 (+0.8)
SciCap	162.3	165.1 (+2.8)	181.5	183.3 (+1.8)
ScienceQA	95.4	96.1 (+0.7)	95.9	96.2 (+0.3)
Screen2Words	117.6	113.3 (−4.3)	119.6	114.0 (−5.6)
TallyQA (complex)	69.6	70.3 (+0.7)	72.3	73.6 (+1.3)
TallyQA (simple)	81.7	81.8 (+0.1)	84.9	85.3 (+0.4)
TextCaps	127.5	127.5 (+0.0)	153.9	152.1 (−1.8)
TextVQA (val)	59.0	59.6 (+0.6)	74.6	75.2 (+0.6)
VATEX	79.7	80.8 (+1.1)	-	-
VQAv2 (minival)	82.1	83.0 (+0.9)	84.6	84.8 (+0.2)
VizWizVQA (val)	73.7	76.4 (+2.7)	75.5	77.5 (+2.0)
WidgetCap	136.1	138.1 (+2.0)	148.4	151.4 (+3.0)
XM3600 (avg35)	41.9	42.8 (+0.9)	42.4	43.2 (+0.8)
XM3600 (en)	78.0	79.8 (+1.8)	80.0	80.3 (+0.3)
xGQA (avg7)	57.3	58.6 (+1.3)	57.9	60.4 (+2.5)

Table 15 | Comparison of PaliGemma 3B and PaliGemma 2 3B at 224px² and 448px² resolutions. PG1 and PG2 refer to PaliGemma [9] and PaliGemma 2, respectively.