

BabyVLM-V2: Toward Developmentally Grounded Pretraining and Benchmarking of Vision Foundation Models

Shengao Wang^{*†}, Wenqi Wang^{*}, Zecheng Wang^{*}, Max Whitton^{*}

Michael Wakeham, Arjun Chandra, Joey Huang, Pengyue Zhu

Helen Chen[‡], David Li[‡], Jeffrey Li[‡], Shawn L. Li[‡], Andrew Zagula[‡], Amy Zhao[‡], Andrew Zhu[‡]

Sayaka Nakamura², Yuki Yamamoto², Jerry Jun Yokono²

Aaron Mueller, Bryan A. Plummer, Kate Saenko, Venkatesh Saligrama, Boqing Gong

Boston University, ²Sony Group Corporation, {wsashawn, wqwang, vicwang0, maxwh, bgong}@bu.edu

<https://shawnking98.github.io/BabyVLM-v2/>

Abstract

Early children’s developmental trajectories set up a natural goal for sample-efficient pretraining of vision foundation models. We introduce BabyVLM-V2, a developmentally grounded framework for infant-inspired vision-language modeling that extensively improves upon BabyVLM-V1 through a longitudinal, multifaceted pretraining set, a versatile model, and, most importantly, DevCV Toolbox for cognitive evaluation. The pretraining set maximizes coverage while minimizing curation of a longitudinal, infant-centric audiovisual corpus, yielding video-utterance, image-utterance, and multi-turn conversational data that mirror infant experiences. DevCV Toolbox adapts all vision-related measures of the recently released NIH Baby Toolbox® into a benchmark suite of ten multimodal tasks, covering spatial reasoning, memory, and vocabulary understanding aligned with early children’s capabilities. Experimental results show that a compact model pretrained from scratch can achieve competitive performance on DevCV Toolbox, outperforming GPT-4o on some tasks. We hope the principled, unified BabyVLM-V2 framework will accelerate research in developmentally plausible pretraining of vision foundation models.

1. Introduction

We formalize our objective: Given a longitudinal, infant-centric audiovisual sample of early children’s sensory experiences (e.g., Figure 1a), can we learn a foundation model

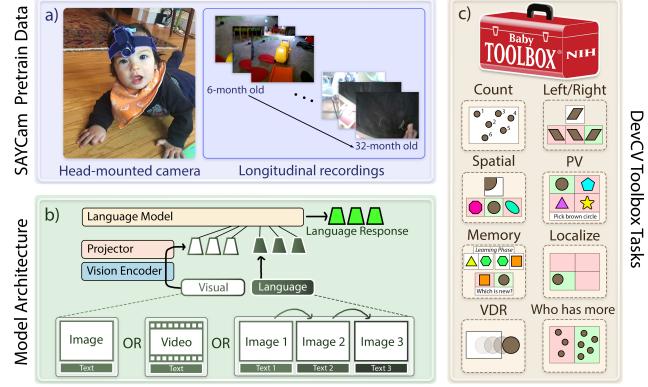


Figure 1. **BabyVLM-V2:** An extensive, versatile, and developmentally plausible framework for research in vision foundation models. Its (a) pretraining set is diverse in format (video, image-utterance, and multiple turns), enabling (b) a flexible model. Its (c) benchmark developmentally aligns with the pretraining set’s age span by grounding on the newly released NIH Baby Toolbox®.

(FM) that is as versatile and capable as the early children’s perception? As a further challenge, can we leverage principles of developmental psychology to create a benchmark as an initial step toward artificial developmental intelligence (ADI), in both *what* it is and *how* to achieve it within the constraints of early children’s limited sensory intake? We consider a resultant model and benchmark *developmentally plausible* if the training data and desired model performance closely mirror those of early children.

We envision that our answer to this objective, BabyVLM-V2, will have a threefold impact. First, by making the limited training data accessible to independent researchers and friendly to university resources, we will broaden research engagement in developing FMs [18, 59] in a time when the scaling law [23] causes research on FMs

^{*}Equal contribution.

[†]Project lead.

[‡]Equal contribution; work done as interns at Boston University.

Table 1. BabyVLM-V2 extensively extends BabyVLM-V1 [56].

	BabyVLM-V1	BabyVLM-V2 (Ours)
Pretraining	67k img-utterance 181k video-utterance 63k interleaved	768k img-utterance 181k video-utterance 63k interleaved
Instruction	None	150k examples
Benchmarks	4 tasks, intuitive Visual vocabulary, captioning	10 tasks, grounded on NIH Baby Toolbox® Visual vocabulary, counting, memory, attention, spatial reasoning, localization, spatiotemporal reasoning, executive function
Models	Input: text, single img Output: logits	Input: text, img, multi-img, video, multi-turn Output: language

to be dominated by industry. Second, we envision that ADI could advance studies in cognitive science and psychology by allowing scientists to read into early children’s minds in an unprecedented way. Lastly, we believe that the broadened engagement in FMs will improve public understanding, trust, and safe use of FMs and AI in general.

Previously, Wang *et al.* proposed BabyVLM-V1 [56], a scaffold for studying ADI from the lens of vision-language models (VLMs). It consists of 1) an image-text pretraining set extracted from SAYCam’s head-mounted camera recordings from three children for approximately two hours per week from age 6 to 32 months [50], 2) four intuitive and developmentally inspired benchmark tasks, and 3) a public codebase for pretraining and evaluation. BabyVLM-V1 pretrained a baseline VLM from scratch, whose performance, unfortunately, fell far behind the remarkable capabilities of early children [7, 34]. Similarly, Vong *et al.* [54] trained a CLIP-style [44] contrastive model using SAYCam, but with a narrower focus on word-referent mappings rather than general perception. More related work is in Section 2.

While BabyVLM-V1 sets up a basic framework, it lacks crucial elements. Its pretraining set only leverages about a third of SAYCam’s recordings, causing it to cover only a tiny portion of the total visual intake time of a three-year-old since birth [38]. It does not support instruction tuning [64], which is crucial for a pretrained model to articulate its capabilities to user instructions. Importantly, its evaluation benchmarks are not based on any established psychology tests. Finally, the models trained in BabyVLM-V1 have near-zero open-set performance, and one has to postprocess their logits for evaluation.

This work extends BabyVLM-V1 to a comprehensive, extensive, and developmentally plausible framework, BabyVLM-V2 (see Figure 1), for studying the objective

posed at the beginning of the paper. Table 1 contrasts the two frameworks in pretraining, instruction tuning, benchmarks, and baseline models. Notably, we provide *DevCV Toolbox* (see Figure 3), a benchmark of ten tasks designed using the NIH Baby Toolbox® [11, 16], which was publicly released in February 2025 as a “universal assessment for developmental and pediatric communities”. We make minimal changes while adapting all of its vision-related measures to *DevCV Toolbox* in order to maintain developmental fidelity.

Interestingly, the *DevCV Toolbox* tasks are naturally diverse in format, desiring FMs to understand individual videos and images, reason across multiple images, and solve a task in multiple turns. To account for these requirements in the pretraining data, we compile video, image-utterance, and multi-turn data from the longitudinal, infant-centric videos in SAYCam [50]. As in BabyVLM-V1, we include a minimal curation process to bring our pretraining data as close to the children’s sensory intake as possible.

We validate BabyVLM-V2 through extensive experiments and human performance surveys. A model trained from scratch within our BabyVLM-V2 framework outperforms GPT-4o in math tasks, highlighting the potential of developmentally grounded pretraining.

2. Related work

Vision FMs refer to general-purpose models [3] often pre-trained on massive visual data [4, 37, 46, 58]. They can tackle many vision tasks via a unified interface, such as CLIP [44], ALIGN [20], BLIP [26, 27], SAMs [24, 45], and vision LLMs [1, 6, 28, 40]. The development of these powerful models hinges critically on pretraining [3, 5, 10], a process that trains a model on a large, generic dataset before tuning it to any downstream tasks.

Sample-efficient pretraining. While FMs have been relying on the scaling law, sample-efficient pretraining has gained momentum recently in the language [59] and medical [51] domains. To the best of our knowledge, BabyVLM-V1 was the first of this kind in vision, and we further their effort with a more comprehensive and extensive framework.

Cognitively plausible benchmarking. BabyVLM-V1 [56] designs four developmentally plausible tasks, which unfortunately lack grounding on established psychological tests. DevBench [52] and KIVA [62] draw inspiration from kid-oriented tests, yet they are more age-advanced than our pretraining data. Other cognitively plausible benchmarks have a narrower focus, such as Zorro [19], LRS [25], InfLevel [60], CoreCognition [29], and MEWL [21], and ModelVsBaby [48]. Table 2 summarizes the differences.

Tools assessing neurodevelopment in children. Our benchmark tasks are grounded on the NIH Baby Toolbox® [11], a standardized tool released in February 2025 for assessing neurodevelopment in children. It is not

Table 2. Comparison of existing developmentally inspired benchmarks.

Benchmark	Developmental	Task Diversity	Multimodal	Train	Val	Test	In-Domain	OOD	Human Data	Model
DevBench [52]	✓	✓	✓	✗	✗	✓	✗	✓	✓	✗
Labeled-S [54]	✓	✗	✓	✓	✗	✓	✓	✗	✗	✗
ModelVsBaby [48]	✓	✗	✓	✓	✓	✓	✗	✓	✓	✗
MEWL [21]	✗	✓	✓	✓	✓	✓	✓	✗	✓	✗
Zorro [19]	✓	✗	✗	✓	✗	✓	✓	✗	✗	✓
InfLevel [60]	✓	✗	✓	✗	✗	✓	✗	✓	✓	✗
LRS [25]	✓	✓	✗	✗	✗	✓	✗	✓	✗	✗
CoreCognition [29]	✓	✓	✓	✗	✗	✓	✗	✓	✓	✗
BabyVLM [56]	✗	✓	✓	✓	✗	✓	✓	✗	✗	✓
DevCV Toolbox (Ours)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

only more recent but also more comprehensive and normed than alternatives, such as the Bayley Scales Of Infant and Toddler Development [2], Mullen Scales of Early Learning [9], and Battelle Developmental Inventory [39]. Besides, its design for clinical use validates its credibility over the psychological tests used in research settings.

3. BabyVLM-V2

3.1. Data source & the pretraining set

We describe SAYCam, the developmental data source, followed by our minimal process to curate the pretraining set. **SAYCam:** The developmental plausibility of our work hinges on the use of a visual-audio-text corpus that faithfully samples what early children have seen and heard by a certain age, which requires the corpus to be 1) longitudinal and 2) infant-centric. To accomplish this, we use the SAYCam dataset [50], which is accessible to all nonprofit institutes, and will include BabyView [32] in future work. SAYCam contains egocentric recordings from three infants (left of Figure 1a) taken once every week from roughly 6 to 32 months old. Each recording is approximately two hours, and the recordings total 478 hours (see bottom of Figure 2 for the recorded time *vs.* wake and sleep time [38]). Notably, the utterances found in SAYCam are mostly from caregivers providing simple verbal instructions and descriptions to the infants (top of Figure 2). BabyView [32] is an ongoing effort in the same spirit as SAYCam, but at a larger scale and with extra gyroscope/accelerometer sensors.

Data split & the pretraining set. To maximize our use of the SAYCam corpus, we designate all video clips containing speech to the pretraining split, and evenly divide the remaining clips into validation and test splits. Their relative sizes are approximately 3:1:1, respectively. We then apply minimal processing to facilitate model pretraining while observing the children’s sensory intake as much as possible. Specifically, we transcribe all utterances, which are almost all from caregivers, using Azure Speech Recognition [36]. We then construct three types of pretraining data.

- **Video–utterance pairs.** We segment the camera recordings into short clips based on transcript boundaries, with

each clip corresponding to exactly one utterance. We then drop the video clips shorter than 0.5 seconds or with a transcript confidence score below 0.3. Further, we compute video-utterance similarities using X-CLIP [33] and only retain the video-utterance pairs with similarities greater than 0.1. This process leaves approximately 181k video clips in our pretraining set, a total of 138 (out of 478) hours. We pad 1 second to either side of the clips.

- **Image–utterance pairs.** Following BabyVLM-V1, we sample at 1 FPS from the video-utterance pairs and compute the CLIP similarity [44] between each frame and its utterance. Only frames with CLIP similarities > 0.2 are retained, resulting in 768k image-utterance pairs in total.
- **Interleaved text and images.** We create sequences of interleaved images and utterance from consecutive video segments, aiming to enable downstream capabilities that involve conversations. For each video segment, we pair the frame with the highest CLIP similarity with its associated utterance and use a sliding window over the resulting image-utterance pairs to construct the interleaved sequences. We randomly choose a window size between 4 and 8 and employ a stride of half the window size, resulting in 63k interleaved sequences.

Unlike BabyVLM-V1’s image-utterance pairs, the mixing of three pretraining data formats prepares models for diverse downstream tasks, which can involve videos, multiple or single images, and even multi-turn conversations.

3.2. Pretraining & fine-tuning BabyLLaVA-V2

Using our pretraining split, we pretrain BabyLLaVA-V2, which uses a language model (LLaMA-1.1B [53, 63]) as a versatile interface to probe various capabilities of a visual encoder (ViT-L-16 [8], 300M parameters). A lightweight MLP connector [30] projects visual features into the language space. This model architecture (Figure 1b) is the same BabyVLM-V1’s BabyLLaVA-Llama. We pretrain the entire model from scratch using the three-stage pipeline described in Appendix A. Finally, we fine-tune the model using a small, curated instruction set consisting of the tasks as in *DevCV Toolbox*, which we describe next.

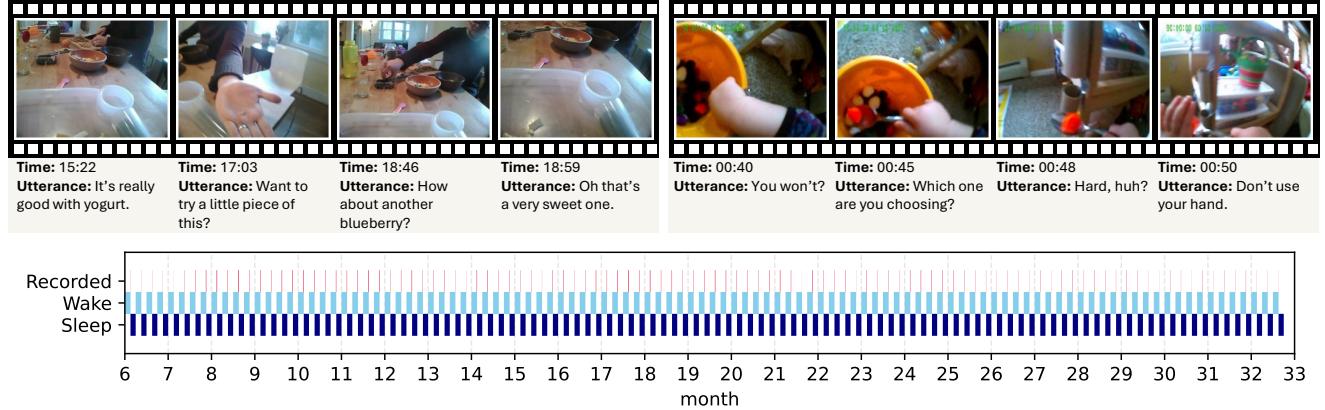


Figure 2. Top: Video frames and utterances recorded from the infants’ view. Bottom: Recorded wake time *vs.* wake/sleep time for the ages of 6 months to 32 months in SAYCam [50].

3.3. Age-appropriate benchmarking

Our objective with BabyVLM-V2 is to design benchmark tasks that test *age-appropriate* visual skills given our pre-training data’s age span. However, we acknowledge that developmental benchmarking is an ongoing and rapidly evolving field of research. Early children’s growth rates vary significantly, and among psychologists and cognitive scientists, substantial conceptual and methodological disagreements exist regarding the notion of developmental intelligence and how to properly probe, measure, and benchmark it [16]. How can we properly define ADI, then, given the inconsistent measurement techniques in human developmental research? To answer this, we consult with two experienced psychologists specializing in development and learning. Numerous meetings with them led us to the timely NIH Baby Toolbox®, over which we ground the design of our benchmark, *DevCV Toolbox*.

3.3.1. Background: NIH Baby Toolbox®

In February 2025, a multi-institutional team solicited by the NIH publicly released the NIH Baby Toolbox®, envisioning it as a standardized evaluation of neurodevelopmental intelligence in infants [15]. The NIH Baby Toolbox® divides developmental function into three domains: Cognition, Motor, and Social-Emotional, where the Cognition domain includes the subdomains of Language, Executive Function/Memory, and Math, each consisting of some number of specific tests, known in the toolbox as *measures*. See Table 3 for a summary of these measures and Appendix B for technical details.

3.3.2. DevCV Toolbox

In this section, we develop a computer vision counterpart, called *task* for clarity, for every vision-related *measure* in the NIH Baby Toolbox®, leading to ten tasks in our *DevCV Toolbox*, which are summarized in Table 3 and illustrated in Figure 3.

The need to adapt measures to tasks. Unlike the practice in computer vision, most of the measures originally found in the NIH Baby Toolbox® 1) have only a couple of test examples and 2) are human-oriented but not accessible to AI models. Additionally, the cartoon stimuli in NIH Baby Toolbox® are out-of-domain from our pretraining set, preventing their direct use. Hence, we adapt the measures to computer vision tasks by standardizing their format and equipping each task with thousands of naturalistic examples (see Table 3), separated into instruction and test sets according to the split defined in the pretraining stage.

We construct the tasks using SAYCam to ensure that the examples are in the same domain as the pretraining data, thereby focusing the benchmarking on the models’ in-domain cognitive capabilities. To provide an additional tool to evaluate models’ generalizability, we also compile an out-of-domain test set using Ego4D [14] with the same techniques. Below, we detail the construction of *Picture Vocabulary* as a representative example, and briefly describe the rest. See Appendix B for more details on the construction of *DevCV Toolbox*.

Picture vocabulary (≥ 25 months): The top right of Figure 3 shows the original picture vocabulary (PV) measure found in the NIH Baby Toolbox®, which assesses the Receptive Language of children aged 25 months and older. Participants are presented with four clipart images on an iPad, and an audio prompt instructs them to touch the named image.

We adapt PV to *DevCV Toolbox* using the pipeline in Figure 4, to replace the clipart in the NIH Baby Toolbox® manual with *objects and actions* detected from SAYCam video frames. Concretely, we sample frames at 1FPS, label all objects and actions present using manual transcripts and GPT-4o, and then crop out regions for each label using Grounding-DINO [31]. Low quality crops and labels beyond the child-oriented MAB-CDI vocabulary [35] are removed. Each PV example (*e.g.*, the top left of Figure 3) consists of a language prompt, a target image corresponding

DevCV Toolbox

Picture Vocabulary

"Which is a stroller?
Answer with A, B, C, or D."
Answer: A

Looking While Listening

"Which is a chair?
Answer A or B."
Answer: B

Localization

"Where is the knife?
Answer with top left, top right, bottom left, or bottom right."
bottom left

Left / Right

"Look at this cat. Which is exactly like it? Answer with A, B, or C."
Answer: B

Spatial Details

"Look at this hand. Which has a hand exactly like it?
Answer with A, B, or C"
Answer: B

Visual Delayed Response

"Where does the toy leave the frame from? Answer with top or bottom."
Answer: bottom

Memory

Round 1

Rounds 2-15: "Which is the new image? Answer with A or B."
Answer: A

Who Has More

"Which has more cups? Answer with A or B."
Answer: B

Who Has More

"Which has more balls? Answer A or B."
Answer: A

Subitizing

"How many chairs do you see? Answer with a number 1-12."
Answer: 3

Object Counting

"How many balls do you see? Answer with a number 1-12."
Answer: 7

NIH Baby Toolbox

Picture Vocabulary

"Touch the picture of chair."
Answer: *touch*

Looking While Listening

"Touch the picture of table."
Answer: *touch*

Mullen RL #19

"Look at these. Where is the cat?"
Answer: *point*

Mullen VR #29

"Look at this boy. Find another boy that is just like this one."
Answer: *point*

Mullen VR #25

"Look at this flower. Look for one just like it. Look for it here."
Answer: *point*

Visual Delayed Response

"Look at that."
"Where is it hiding?"
Answer: *touch*

Memory

"The animal is hungry. Touch the animal."

"Now you see two animals. One of them we already fed. Let's touch the new animal."

Who Has More?

"Who has more eggs?"
Answer: *touch*

Subitizing

"How many did you see?"
Answer: 2

Object Counting

"Count the blocks."
Answer: 4

Figure 3. DevCV Toolbox tasks and their corresponding NIH Baby Toolbox® measures

to the prompt, and three distractor images, and we construct the examples in a round-robin manner for diversity. The target and distractor images are related either semantically or phonologically in NIH Baby Toolbox®; therefore, we derive a distractor distribution over phonology and semantics from the toolbox and then sample distractor images accordingly. We manually screen the process to ensure quality and diversity. Appendix B presents more details.

Other tasks. We describe the other tasks in DevCV Toolbox briefly. Construction details are in Appendix B.

1. **Looking while listening** (6–24 months) shows infants two clipart objects, and plays an audio prompt describing one of them. Eye tracking is used to detect the participant’s response. We replace clipart with natural objects from SAYCam, and eye tracking with multiple choice.
2. **Localization / Mullen visual receptive language #19**

5

Table 3. DevCV Toolbox tasks and their corresponding NIH Baby Toolbox® measures (EF/M stands for Executive Function/Memory).

DevCV Toolbox tasks	#Instruct/#Test	Model Input	NIH Baby Toolbox® measures	Months	Subdomain
Looking While Listening	0/1.2k	2 images	Looking While Listening	6-24	Language
Picture Vocabulary	63.9k/1.2k	4 images	Picture Vocabulary	25+	Language
Localization	12.3k/2.1k	1 image	Mullen Receptive Language #19	1-42	Language
Left/Right	12.3k/2.3k	4 images	Mullen Visual Reception #29	1-42	EF/M
Spatial Details	11.8k/1.2k	4 image	Mullen Visual Reception #20	1-42	EF/M
Visual Delayed Response	5.2k/0.9k	5-8 images	Visual Delayed Response	22-42	EF/M
Memory	10.0k/0.5k	29 images	Delayed Memory	22-42	EF/M
Who Has More (synthetic)	11.2k/1.8k	2 images	Who Has More	25-42	Math
Who Has More (naturalistic)	6.9k/2.2k	2 images	Who Has More	25-42	Math
Subitizing (synthetic)	0/1.9k	3 images	Subitizing	25-42	Math
Subitizing (naturalistic)	0/0.2k	3 images	Subitizing	25-42	Math
Object Counting	13.7k/3.0k	1 image	Object Counting	25-42	Math

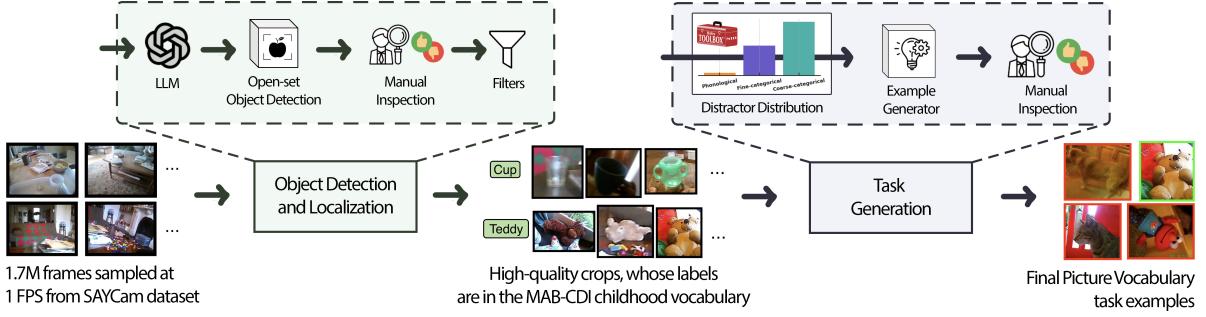


Figure 4. Pipeline to adapt the picture vocabulary measure in NIH Baby Toolbox® to DevCV Toolbox.

(1–42 months) tests an infant’s ability to point at sketched objects as they are named. We task a model with localizing an object in a natural video frame.

3. **Left/Right / Mullen visual reception #29** (1–42 months) measures an infant’s attention to detail by instructing them to match objects by orientation.
4. **Spatial details / Mullen visual reception #29** (1–42 months) measures attention to detail in identical objects among distractors of the same type.
5. **Visual delayed response** (22–42 months) shows infants a creature moving behind one of two occluders, and after a short pause, instructs them to tap the target occluder. We use video clips with prominent objects moving out of the field of view.
6. **(Delayed) memory** (22–42 months) involves multiple turns, each presenting a pair of animals. Participants are asked to “feed” the new animal appearing for the first time, and they receive corrective feedback during the early rounds.
7. **Who has more** (25–42 months) shows two images with the same shape in different quantities and asks which image has more. We replace the shape with natural objects as one sub-task, and use entire natural video frames for

the other sub-task.

8. **Subitizing** (25–42 months) refers to the rapid identification of the number of items in a small set. An infant sees one to four identical shapes for one second, and then an audio prompt requests the count.
9. **Object counting** (25–42 months) evaluates a child’s ability to count up to 12 colored shapes on a screen. During evaluation, we employ accuracy as the metric. These tasks cover all cognitive measures in NIH Baby Toolbox® except the *non-visual* MacArthur-Bates language (9–30 months, 7–18 months), familiarization (6–21 months), verbal counting (25–42 months), and verbal arithmetic (37–42 months). Adult performance data on these tasks confirms the validity of our DevCV Toolbox (see *Human performance* in Tables 4 and Appendix C for details). In future work, we hope to complete a survey of children’s performance.

4. Experiments

We design experiments about the key elements of BabyVLM-V2 framework, aiming to validate the quality of the DevCV Toolbox, as well as illustrate the effectiveness of our training data and training recipe. Meanwhile, the exper-

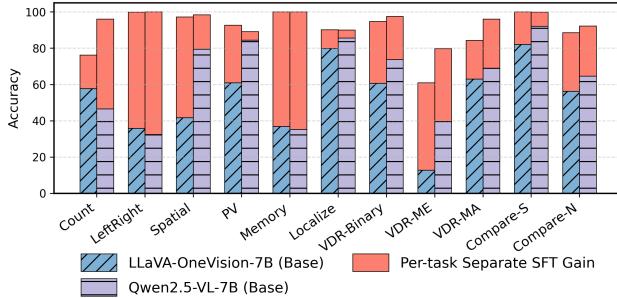


Figure 5. Task-specific supervised finetuning of LLaVA-OneVision-7B and Qwen2.5-VL-7B.

iments position our BabyLLaVA-V2 in context across three cognitive subdomains and ten tasks. Note that we exclude two tasks, *Subitizing* and *Looking While Listening*, from the majority of the experiments to test our models’ generalization on unseen tasks near the end. Implementation details are in Appendix D.

4.1. Examining DevCV Toolbox

Overall quality. We validate the quality of *DevCV Toolbox* by conducting human surveys, detailed in Appendix C. As shown in Table 4, the human volunteers recruited in our home institute achieved near-perfect accuracy on the executive functioning/memory subdomain (*Spatial Details*, *Memory*, *Visual Delayed Response*) and the math tasks of *Object Counting* and *Who Has More*. Their accuracy on *Localization* is slightly low (87.3%), and a follow-up revealed that it could improve when the volunteers were instructed to spend more time on the task.

Differentiating capability. Table 4 also demonstrates that, between Human performance and Random guess, there is a sufficiently big room for differentiating various models. Indeed, the proprietary GPT and Gemini models are on the upper end, while our BabyLLaVA-V2 and the open-source models of about the same size as ours are on the lower end, indicating that the tasks in *DevCV Toolbox* are challenging but solvable.

Developmental fidelity. *DevCV Toolbox* should developmentally align with the pretraining data’s age span (6–32 months). Hence, we are in the process of performing a large-scale children survey about *DevCV Toolbox* using the Children Helping Science platform [49], though this survey will take a couple of years per our estimation.

4.2. Validating the instruction tuning dataset

Instruction tuning addresses the mismatch between pre-training and downstream tasks, steering models towards the downstream. To validate the effectiveness of our instruction tuning data, we use it to supervise the fine-tuning of three models under two strategies. Figure 5 fine-tunes LLaVA-OneVision-7B and Qwen2.5-VL-7B on each task separately (see Appendix A for the experiment setup). The consistent

and relatively big gains from the fine-tuning are highlighted in the red top bars, signifying that the instruction data can effectively guide the models to the downstream tasks in *DevCV Toolbox*.

Furthermore, we experiment with the second fine-tuning strategy that combines the instruction data into a single set. Table 5 contrasts it against the first strategy, fine-tuning a model for each task separately, over our BabyLLaVA-V2. The results show that the overall difference between the two strategies is marginal. The results on most tasks decrease under the mixed-tuning setting, which produces a single unified model rather than multiple per-task models, but some tasks, such as *Memory* and *Spatial Details*, can actually benefit from the mixed fine-tuning, implying knowledge transfer or regularization from other tasks.

4.3. Ablating the pretraining data

The speech transcripts in our pretraining set could be noisy because the naturalistic child-directed utterances are often misaligned with the children’s visual intake. We study their impact on the pretrained models by replacing the transcripts with video captions generated by GPT-4o (see Appendix D for how we prompt GPT-4o). We train BabyLLaVA-V2-synthetic on this altered pretraining dataset and present the results in Table 6. Overall, the synthetic captions improve performance, especially on tasks that demand semantic reasoning (*Picture Vocabulary*) and a long attention window (*Memory*). However, the gains are modest, suggesting that our minimally curated pretraining set already provides strong supervision. In future work, novel pretraining algorithms can likely mine stronger supervision from this organic pretraining set.

4.4. Inspecting BabyLLaVA-V2

Our BabyLLaVA-V2’s overall performance in Table 4 is encouraging, on par with the open-source models whose size is about the same as ours. Of course, one could argue that those models are not fine-tuned under the BabyVLM-V2 framework, but they are probably trained on much larger datasets than ours.

To further stretch BabyLLaVA-V2, we study its generalization along two axes: 1) out-of-domain generalization and 2) performance over previously unseen tasks.

Out-of-domain generalization. We have created a sibling of *DevCV Toolbox* by replacing SAYCam with Ego4D. Both are about egocentric videos, but Ego4D is from the perspective of grown-ups. BabyLLaVA-V2’s overall accuracy on this sibling benchmark is 41.1% (vs. 31.8% of random guess), significantly lower than its in-domain performance (55.2%) on *DevCV Toolbox*. We conclude that BabyLLaVA-V2 can generalize beyond its training domain to some degree, but it is far from human infants’ remarkable generalization capabilities. Appendix D further

Table 4. **Performance comparison of different models on DevCV Toolbox (in-domain).** Different background colors denote different model families. We report accuracy (%) for all tasks; the higher, the better.

Model	Overall	Count	LeftRight	Spatial	PV	Memory	Localization	Visual Delay Response			Who Has More	
								binary	multi-exact	multi-adjacent	synthetic	naturalistic
Upper bound												
Human performance	93.0	99.1	94.5	100	91.8	97.9	87.3	98.2	63.6	95.5	98.2	96.4
Proprietary models												
Gemini-2.5-flash	72.7	71.1	34.9	73.8	91.2	96.9	84.8	75.9	42.4	70.3	87.5	70.7
GPT-4o	74.6	39.0	89.8	92.6	93.7	99.7	81.7	64.2	29.3	62.9	87.9	79.3
Gemini-2.5-pro	82.5	77.2	68.8	90.5	93.8	97.8	88.8	86.9	54.0	87.7	90.6	71.7
GPT-5	87.6	69.1	96.0	94.5	95.0	99.9	85.2	95.1	62.9	90.1	88.9	86.6
Open-source models												
LLaVA-OneVision-0.5B	33.2	43.5	33.7	28.7	23.5	24.0	12.3	58.9	7.31	49.2	37.3	46.2
InternVL3.5-1B	37.2	27.9	32.2	34.6	34.4	25.8	44.8	64.1	11.6	36.8	47.8	49.1
Qwen2.5-VL-3B	47.0	29.2	33.7	40.0	71.7	36.5	85.8	66.7	17.0	32.7	51.7	52.3
Baby models (Ours)												
BabyLLaVA-V2	55.2	44.6	42.3	91.3	27.4	75.3	38.8	57.6	33.1	45.6	98.4	52.8
Lower bound												
Random guess	31.8	8.33	33.3	33.3	25.0	25.0	25.0	50.0	12.5	37.5	50.0	50.0

Table 5. **Two supervised fine-tuning strategies.** BabyLLaVA-V2-separate denotes models fine-tuned on each task’s instruction dataset separately, and BabyLLaVA-V2-mixed is a single model fine-tuned on the mixed instruction set.

Model	Overall	Count	LeftRight	Spatial	PV	Memory	Localization	Visual Delay Response			Who Has More	
								binary	multi-exact	multi-adjacent	synthetic	naturalistic
BabyLLaVA-V2-separate	56.0	45.2	42.5	87.1	28.4	70.7	43.3	55.7	37.0	49.9	98.6	56.4
BabyLLaVA-V2-mixed	55.2	44.6	42.3	91.3	27.4	75.3	38.8	57.6	33.1	45.6	98.4	52.8

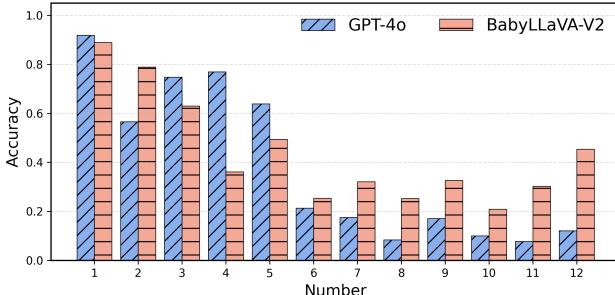


Figure 6. GPT-4o and our model’s counting performance by different object numbers.

tests BabyLLaVA-V2’s out-of-domain generalization on the original NIH Baby Toolbox®.

Unseen tasks. We have excluded *Looking While Listening* and *Subitizing* from the instruction tuning, which are thus unseen by BabyLLaVA-V2. While the two tasks are in spirit similar to *Picture Vocabulary* and *Object Counting*, respectively, BabyLLaVA-V2 yields near-random-guess results on them. We will address this issue in future work by improving the instruction tuning algorithm.

4.5. Intriguing findings

Finally, we draw some intriguing “byproduct” findings from Table 4, which can improve our understanding of the proprietary GPT and Gemini models.

GPT models struggle to count. *Object Counting* requires a model to count objects in an image (between 1 and 12), and GPT-4o can hardly count beyond 5 (see Figure 6).

BabyLLaVA-V2 can match or outperform GPT-4o on

some cognitive tasks. On *Spatial Details* and *Who Has More*, BabyLLaVA-V2 is on par with the four latest GPT and Gemini models. Moreover, it even outperforms GPT-4o on the math tasks of *Object Counting* and *Who Has More*. Figure 6 shows that BabyLLaVA-V2 counts better than GPT-4o given six or more objects.

GPT vs. Gemini. In general, the proprietary models give rise to similar results on DevCV Toolbox. However, when we zoom into the individual tasks, GPT-5 is significantly better than the rest on *Spatial Details*, while Gemini models are better at *Object Counting* than the GPT models.

5. Conclusion

We introduced BabyVLM-V2, a framework that features a developmentally plausible pretraining set derived from the longitudinal SAYCam corpus, a compact VLM (BabyLLaVA-V2) trained from scratch, and comprehensive developmental benchmarks (DevCV Toolbox). DevCV Toolbox adapts all vision-related measures from the newly published NIH Baby Toolbox®. It contains ten measures spanning three subdomains (language, executive function/memory, and math) and requires a flexible model interface that can process image, video, and multi-turn dialogue. We demonstrate the potential of developmentally plausible vision FMs through extensive experiments on our pre-training and instruction tuning datasets, and we confirm the quality of DevCV Toolbox through extensive benchmarking with proprietary and open-source models. This framework will serve as a principled platform to broaden research

Table 6. Two language sources for pretraining. BabyLLaVA-V2-original is pretrained on our pretraining set whose language is mainly caregivers’ speech transcripts, while BabyLLaVA-V2-synthetic is pretrained on synthetic utterances generated by GPT-4o.

Model	Overall	Count	LeftRight	Spatial	PV	Memory	Localization	Visual Delay Response			Who Has More	
								binary	multi-exact	multi-adjacent	synthetic	naturalistic
BabyLLaVA-V2-original	55.2	44.6	42.3	91.3	27.4	75.3	38.8	57.6	33.1	45.6	98.4	52.8
BabyLLaVA-V2-synthetic	57.4	46.7	35.3	92.0	30.7	87.7	36.9	57.8	38.1	49.0	99.2	57.6

engagement in vision FMs and accelerate progress toward developmentally plausible learning.

Acknowledgements

Special thanks to Chen Yu, Jessica Sullivan, and Michel C. Frank for their wholehearted support and feedback throughout the project!

References

- [1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhao-hai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL Technical Report, 2025. arXiv:2502.13923 [cs]. [2](#)
- [2] Palanikumar Balasundaram and Indirapriya Darshini Avula. Bayley Scales Of Infant and Toddler Development. In *StatPearls*. StatPearls Publishing, Treasure Island (FL), 2025. [3](#)
- [3] Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, and Emma Brunskill et al. On the Opportunities and Risks of Foundation Models, 2022. arXiv:2108.07258 [cs]. [2](#)
- [4] Mathilde Caron, Alireza Fathi, Cordelia Schmid, and Ahmet Iscen. Web-scale visual entity recognition: an ILM-driven data approach. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2024. Curran Associates Inc. [2](#)
- [5] Fei-Long Chen, Du-Zhen Zhang, Ming-Lun Han, Xiu-Yi Chen, Jing Shi, Shuang Xu, and Bo Xu. VLP: A Survey on Vision-language Pre-training. *Machine Intelligence Research*, 20(1):38–56, 2023. [2](#)
- [6] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blissein, Ori Ram, Dan Zhang, and Evan Rosen et al. Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multi-modality, Long Context, and Next Generation Agentic Capabilities, 2025. arXiv:2507.06261 [cs]. [2](#)
- [7] Gil Diesendruck and Paul Bloom. How specific is the shape bias? *Child Development*, 74(1):168–178, 2003. [2](#)
- [8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. [3, 13](#)
- [9] Ron Dumont, John O. Willis, Kathleen Viezel, and Jamie Zibulsky. Mullen Scales of Early Learning, AGS Edition, 1995. In *Encyclopedia of Special Education*. John Wiley & Sons, Ltd, 2014. [3](#)
- [10] Dumitru Erhan, Aaron Courville, Yoshua Bengio, and Pascal Vincent. Why Does Unsupervised Pre-training Help Deep Learning? In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 201–208. JMLR Workshop and Conference Proceedings, 2010. ISSN: 1938-7228. [2](#)
- [11] Richard Gershon, Miriam A. Novack, and Aaron J. Kaat. The NIH Infant and Toddler Toolbox: A new standardized tool for assessing neurodevelopment in children ages 1–42 months. *Child Development*, 95(6):2252–2254, 2024. [2, 15](#)
- [12] Richard C. Gershon, Molly V. Wagster, Hugh C. Hendrie, Nathan A. Fox, Karon F. Cook, and Cindy J. Nowinski. Nih toolbox for assessment of neurological and behavioral function. *Neurology*, 80(11_supplement_3):S2–S6, 2013. [15](#)
- [13] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18995–19012, 2022. [24](#)
- [14] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, and Xingyu Liu et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18995–19012, 2022. [4](#)
- [15] Y. Catherine Han, Courtney K. Blackwell, Elizabeth M. Dworak, Rachel M. Flynn, Maxwell A. Mansolf, Miriam A. Novack, Sarah Pila, and Aaron J. Kaat. *NIH Baby Toolbox® Technical Manual*. Northwestern University, Evanston, IL, version 1.1 edition, 2025. [4, 15, 16](#)
- [16] Y. Catherine Han, Elizabeth M. Dworak, Maxwell Mansolf, Hubert Adam, Lihua Yao, Miriam A. Novack, Sarah Pila, Rachel M. Flynn, Amanda M. Flagg, Vitali Ustsinovich, Kay Savio, Greg J. Byrne, Richard C. Gershon, and Aaron J. Kaat. NIH Baby Toolbox® methodology and norms development. *Infant Behavior and Development*, 80:102117, 2025. [2, 4](#)
- [17] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon

- Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models, 2022. [25](#)
- [18] Michael Y. Hu, Aaron Mueller, Candace Ross, Adina Williams, Tal Linzen, Chengxu Zhuang, Ryan Cotterell, Leshem Choshen, Alex Warstadt, and Ethan Gotlieb Wilcox. Findings of the Second BabyLM Challenge: Sample-Efficient Pretraining on Developmentally Plausible Corpora, 2024. arXiv:2412.05149 [cs] version: 1. [1](#)
- [19] Philip A. Huebner, Elior Sulem, Fisher Cynthia, and Dan Roth. BabyBERTa: Learning More Grammar With Small-Scale Child-Directed Language. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 624–646, Online, 2021. Association for Computational Linguistics. [2, 3](#)
- [20] Chao Jia, Yinfai Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021. ISSN: 2640-3498. [2](#)
- [21] Guangyuan Jiang, Manjie Xu, Shiji Xin, Wei Liang, Yujia Peng, Chi Zhang, and Yixin Zhu. MEWL: Few-shot multimodal word learning with referential uncertainty. In *Proceedings of the 40th International Conference on Machine Learning*, pages 15144–15169. PMLR, 2023. ISSN: 2640-3498. [2, 3](#)
- [22] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020. [25](#)
- [23] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling Laws for Neural Language Models, 2020. arXiv:2001.08361 [cs]. [1](#)
- [24] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3992–4003, 2023. [2](#)
- [25] Eliza Kosoy, Emily Rose Reagan, Leslie Lai, Alison Gopnik, and Danielle Krettek Cobb. Comparing machines and children: Using developmental psychology experiments to assess the strengths and weaknesses of LaMDA responses. SSRN Electronic Journal, 2024. Available at SSRN: <https://ssrn.com/abstract=4696693> or <http://dx.doi.org/10.2139/ssrn.4696693>. [2, 3](#)
- [26] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation. In *Proceedings of the 39th International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022. ISSN: 2640-3498. [2](#)
- [27] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. In *Proceedings of the 40th International Conference on Machine Learning*, pages 19730–19742. PMLR, 2023. ISSN: 2640-3498. [2](#)
- [28] Songtao Li and Hao Tang. Multimodal Alignment and Fusion: A Survey, 2025. arXiv:2411.17040 [cs] version: 2. [2](#)
- [29] Yijiang Li, Qingying Gao, Tianwei Zhao, Bingyang Wang, Haoran Sun, Haiyun Lyu, Robert D. Hawkins, Nuno Vasconcelos, Tal Golan, Dezhil Luo, and Hokin Deng. Core knowledge deficits in multi-modal language models. In *Forty-second International Conference on Machine Learning*, 2025. [2, 3](#)
- [30] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2023. Curran Associates Inc. [3, 13](#)
- [31] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding DINO: Marrying DINO with Grounded Pre-training for Open-Set Object Detection. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part XLVII*, pages 38–55, Berlin, Heidelberg, 2024. Springer-Verlag. [4, 14](#)
- [32] Bria Long, Robert Z. Sparks, Violet Xiang, Stefan Stojanov, Zi Yin, Grace E. Keene, Alvin W. M. Tan, Steven Y. Feng, Chengxu Zhuang, Virginia A. Marchman, Daniel L. K. Yamins, and Michael C. Frank. The BabyView dataset: High-resolution egocentric videos of infants' and young children's everyday experiences, 2025. arXiv:2406.10447 [cs]. [3](#)
- [33] Yiwei Ma, Guohai Xu, Xiaoshuai Sun, Ming Yan, Ji Zhang, and Rongrong Ji. X-clip: End-to-end multi-grained contrastive learning for video-text retrieval. In *Proceedings of the 30th ACM International Conference on Multimedia*, page 638–647, New York, NY, USA, 2022. Association for Computing Machinery. [3](#)
- [34] Maya Malaviya, Ilia Sucholutsky, Kerem Oktar, and {Thomas L.} Griffiths. Can humans do less-than-one-shot learning? pages 997–1003, 2022. Publisher Copyright: © 2022 The Author(s). This work is licensed under a Creative Commons Attribution 4.0 International License (CC BY); 44th Annual Meeting of the Cognitive Science Society: Cognitive Diversity, CogSci 2022 ; Conference date: 27-07-2022 Through 30-07-2022. [2](#)
- [35] Virginia A. Marchman and Philip S. Dale. The MacArthur-Bates Communicative Development Inventories: updates from the CDI Advisory Board. *Frontiers in Psychology*, 14, 2023. Publisher: Frontiers. [4, 14, 15](#)
- [36] Microsoft. Azure AI Speech | Microsoft Azure. [Accessed 13-11-2025] <https://azure.microsoft.com/en-us/products/ai-services/ai-speech>. [3](#)
- [37] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. HowTo100M: Learning a Text-Video Embedding by Watching Hundred Million Narrated Video Clips. In *2019*

- IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2630–2640, 2019. ISSN: 2380-7504. 2
- [38] National Heart, Lung, and Blood Institute. How Sleep Works - How Much Sleep Is Enough? NHLBI, National Institutes of Health (NIH) website, 2022. 2, 3
- [39] Jean. Newborg and Riverside Publishing Company. Battelle developmental inventory. *BDI-2*, 2005. Edition: 2nd ed. Place: Itasca, Ill Publisher: Riverside Pub. 3
- [40] OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, A. J. Ostrow, Akila Welihinda, Alan Hayes, and Alec Radford et al. GPT-4o System Card, 2024. arXiv:2410.21276 [cs]. 2
- [41] Maxime Oquab, Timothée Dariset, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning Robust Visual Features without Supervision, 2024. arXiv:2304.07193 [cs]. 13
- [42] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving Language Understanding by Generative Pre-Training. . 13
- [43] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. . 13
- [44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. ISSN: 2640-3498. 2, 3, 15
- [45] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädele, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollar, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. In *The Thirteenth International Conference on Learning Representations*, 2025. 2
- [46] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: an open large-scale dataset for training next generation image-text models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, pages 25278–25294, Red Hook, NY, USA, 2022. Curran Associates Inc. 2
- [47] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, 2016. Association for Computational Linguistics. 13
- [48] Saber Sheybani, Sahaj Singh Maini, Aravind Dendukuri, Zoran Tiganj, and Linda B. Smith. ModelVsBaby: a Developmentally Motivated Benchmark of Out-of-Distribution Object Recognition, 2024. 2, 3
- [49] Melissa Kline Struhl, Laura Schulz, and Mark Sheskin et al. Children Helping Science. [Accessed 13-11-2025] <https://childrenhelpingscience.com/>. 7, 24
- [50] Jessica Sullivan, Michelle Mei, Andrew Perfors, Erica Wojcicki, and Michael C. Frank. SAYCam: A Large, Longitudinal Audiovisual Dataset Recorded From the Infant’s Perspective. *Open Mind*, 5:20–29, 2021. 2, 3, 4
- [51] Yuqi Sun, Weimin Tan, Zhuoyao Gu, Ruian He, Siyuan Chen, Miao Pang, and Bo Yan. A data-efficient strategy for building high-performing medical foundation models. *Nature Biomedical Engineering*, 9(4):539–551, 2025. Publisher: Nature Publishing Group. 2
- [52] Alvin Wei Ming Tan, Chunhua Yu, Bria Lorelle Long, Wan-jing Anya Ma, Tonya Murray, Rebecca D. Silverman, Jason D Yeatman, and Michael Frank. Devbench: A multi-modal developmental benchmark for language learning. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. 2, 3
- [53] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and Efficient Foundation Language Models, 2023. arXiv:2302.13971 [cs]. 3, 13
- [54] Wai Keen Vong, Wentao Wang, A. Emin Orhan, and Brenden M. Lake. Grounded language acquisition through the eyes and ears of a single child. *Science (New York, N.Y.)*, 383(6682):504–511, 2024. 2, 3
- [55] Ao Wang, Lihao Liu, Hui Chen, Zijia Lin, Jungong Han, and Guiguang Ding. Yoloe: Real-time seeing anything, 2025. 20
- [56] Shengao Wang, Arjun Chandra, Aoming Liu, Venkatesh Saligrama, and Boqing Gong. Babylm: Data-efficient pre-training of vlms inspired by infant learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1380–1390, 2025. 2, 3, 13
- [57] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2020. Curran Associates Inc. 15
- [58] Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, Ping Luo, Ziwei Liu, Yali Wang, Limin Wang, and Yu Qiao. Internvid: A large-scale video-text dataset for multimodal understanding and generation. In *The Twelfth International Conference on Learning Representations*, 2024. 2
- [59] Alex Warstadt, Aaron Mueller, Leshem Choshen, Ethan Wilcox, Chengxu Zhuang, Juan Ciro, Rafael Mosquera, Bhargavi Paranjape, Adina Williams, Tal Linzen, and Ryan Cotterell. Findings of the BabyLM Challenge: Sample-Efficient Pretraining on Developmentally Plausible Corpora. In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages

- 1–34, Singapore, 2023. Association for Computational Linguistics. [1](#), [2](#)
- [60] Luca Weihs, Amanda Yuile, Renée Baillargeon, Cynthia Fisher, Gary Marcus, Roozbeh Mottaghi, and Aniruddha Kembhavi. Benchmarking Progress to Infant-Level Physical Reasoning in AI. *Transactions on Machine Learning Research*, 2022. [2](#), [3](#)
- [61] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017. [13](#)
- [62] Eunice Yiu, Maan Qraitem, Anisa Noor Majhi, Charlie Wong, Yutong Bai, Shiry Ginosar, Alison Gopnik, and Kate Saenko. KiVA: Kid-inspired visual analogies for testing large multimodal models. In *The Thirteenth International Conference on Learning Representations*, 2025. [2](#)
- [63] Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. TinyLlama: An Open-Source Small Language Model, 2024. arXiv:2401.02385 [cs]. [3](#), [13](#)
- [64] Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. Instruction Tuning for Large Language Models: A Survey, 2025. arXiv:2308.10792 [cs]. [2](#)
- [65] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. 2022. [20](#)

BabyVLM-V2: Toward Developmentally Grounded Pretraining and Benchmarking of Vision Foundation Models

Supplementary Material

A. Model training

A.1. BabyLLaVA-V2 architecture

We build upon the original BabyLLaVA-Llama model introduced in BabyVLM-V1 [56], by giving it the capability to process multiple images as input and conduct multi-turn visual–linguistic interactions. The model architecture consists of a compact language backbone, a visual encoder, and a lightweight multilayer perceptron (MLP) connector that projects visual features into the language space. Unlike BabyVLM-V1, which also experimented with smaller backbones (GPT-2 [43] + ResNeXt-50 [61]), we only adopt the larger variant composed of a LLaMA-1.1B [53, 63] language model and a ViT-L-16 [8] visual encoder (300M params). We find that the smaller variant often struggles to complete complex downstream tasks such as memory, primarily due to its limited model capacity, whereas the larger configuration achieves a better balance between developmental plausibility and expressive capability.

A.2. BabyLLaVA-V2 training paradigm

We train the entire model from scratch using a four-stage pipeline, as summarized in Table 7.

Stage 0: Unimodal Training. In the first stage, the language and vision backbones are trained independently to acquire the basic representational abilities for each modality. The language backbone is trained on all transcribed utterances using a standard autoregressive loss [42]. Its tokenizer is initialized via Byte-Pair Encoding (BPE) [47] trained on the same corpus, with a fixed vocabulary size of 6000. The vision backbone is trained using a DINOv2 [41] objective on SAYCam frames. We do not apply any filtering during this stage—except restricting samples to the training split—since the filtering procedures are primarily designed to enforce image–utterance alignment, which is irrelevant to unimodal representation learning.

Stage 1: Feature Alignment. This stage corresponds to Phase 1 training in LLaVA [30]. Both the vision and language backbones are frozen, and only the MLP connector is optimized using an autoregressive loss. The objective is to align visual features with the language embedding space, effectively bridging the two modalities. To maintain training stability, we use only the image–utterance subset of the pretraining data in this stage, postponing exposure to multi-image inputs until later phases.

¹The training dataset, the model checkpoints, the training scripts and the evaluation samples will be released to the public in the near future.

Stage 2: Joint Pretraining. In this stage, the vision backbone remains frozen, while the MLP connector and language backbone are trained jointly on the full mixed-format pretraining dataset, as described in Section 3.1. This allows the model to learn multimodal grounding over diverse input structures.

Stage 3: Instruction Fine-tuning. Finally, we fine-tune the model using the mixed instruction dataset, which is a combination of all the instruction samples mentioned in Table 3. This step enables the model to perform various downstream tasks through natural-language prompts. The vision backbone, MLP connector and language backbone are all updated to learn instruction-following behavior and context-dependent reasoning. We apply two different learning rates for different modules in this stage: the learning rate of the vision backbone is 1e-5, while that of the MLP connector and language backbone is 5e-5.

Main hyperparameters of all 4 stages are summarized in Table 7. All experiments are conducted on four NVIDIA A6000 GPUs with 48 GB of VRAM each. Language backbone training completes in less than one hour, while the vision backbone completes in 4 days. Next, training the MLP connector requires approximately five hours. Joint pretraining on the mixed-format dataset takes roughly 34 hours to converge. Finally, instruction tuning takes 60 hours.

A.3. Open-source model fine-tuning

We conduct LoRA finetuning experiments on two open-source models, LLaVA-OneVision-7B and Qwen2.5-VL-7B, to evaluate the effectiveness of our instruction-finetuning dataset. Each task is finetuned separately. We set the LoRA rank to 64, use a scaling factor of 64, and apply a dropout rate of 0.05. Training is performed for 5 epochs with a global batch size of 128, a learning rate of 1e-4, a weight decay of 0.1, a warmup ratio of 0.03, and a cosine learning-rate schedule.

B. Developmentally aligned benchmarks

In Appendix B, we adopt the following organization: Sub-section B.1 describes general implementation details that are shared by several tasks, including details on the vocabulary used in *DevCV Toolbox*, acquisition of SAYCam annotations, acquisition of Ego4d annotations, and important distinction between SAYCam and Ego4d. Then, each subsection between 2 and 11 describes how these annotations are used to construct one task in *DevCV Toolbox* each, and are each broken up into *Original Toolbox Task*, *Adaptation*,

Table 7. Training stage specification of BabyLLaVA-V2. Note that for stage 3, different modules have different learning rate, as mentioned in Section A.2.

Stage	Trained modules	Frozen modules	Dataset	Loss	Learning rate	Epoch	Global batch size
0-language	Language backbone	N/A	283k utterance only	Autoregressive	2e-4	10	16
0-vision	Vision backbone	N/A	1085k image only	DINOv2	1e-4	100	64
1	MLP connector	Language backbone + vision backbone	768k image-utterance	Autoregressive	3e-3	5	128
2	MLP connector + language backbone	Vision backbone	768k image-utterance + 181k video-utterance + 63k multi-turn	Autoregressive	2e-4	5	128
3	MLP connector + language backbone + vision backbone	None	150k instruction finetune	Autoregressive	5e-5 1e-5	5	128

Data Collection, and Example Prompt. Some of these also include information on *Evaluation* or *Data Composition*.

B.1. Data collection procedures common to all tasks

Vocabulary filtering

To ensure that all benchmarks in this work focus on developmentally appropriate vocabulary, we draw on the *MacArthur–Bates Communicative Development Inventories (MAB–CDI): Words and Gestures* [35]. The MAB–CDI is a standardized instrument assessing early vocabulary comprehension and production in infants and toddlers, covering familiar words across core semantic categories (e.g., animals, foods, body parts, actions).

Because it is widely regarded as a gold-standard reference for early lexical development, we restrict our benchmark vocabulary to words that appear in—or are closely aligned with—those in the MAB–CDI. Accordingly, during visual concept mining from SAYCam and Ego4D, we retain only crops whose labels fall within this developmentally grounded lexical domain, ensuring that every keyword used across tasks reflects concepts young children could plausibly understand.

SAYCam annotations

To support all SAYCam-based benchmarks in this work, we build the following unified preprocessing pipeline that extracts high-quality image crops for every object and action concept appearing in the corpus. This pipeline is reused (with task-specific modifications described in the corresponding benchmark sections) across tasks and provides consistent visual grounding for all downstream datasets.

- **Frame-level detection and indexing:** We first sample SAYCam videos at 1 FPS and run an open-vocabulary detector (Grounding–DINO [31]) using the GPT-annotated labels associated with each frame as the open set. Let \mathcal{S} denote the set of all such SAYCam labels. For each label $s \in \mathcal{S}$, we construct an index $\text{Index}(s)$ that maps s to all frames in which it is detected, together with its

proportionally buffered bounding boxes and GPT-derived blurriness scores. This $\text{Index}(s)$ structure serves as the master lookup table for retrieving visual instances of any concept.

- **Normalizing label variants:** Raw SAYCam labels $s \in \mathcal{S}$ often include plural forms, paraphrases, or compositional descriptions. To ensure consistent visual grounding, we cluster lexically or semantically equivalent labels into small groups based on lexical similarity, plural equivalence, and phrase containment heuristics. Each label s is assigned to its cluster $\mathcal{M}(s)$. This allows us to treat variants of s such as “shoes”, “a shoe”, or “pair of shoes” as a single underlying concept by retrieving visual instances from $\{\text{Index}(s') | s' \in \mathcal{M}(s)\}$.
- **Quality filtering:** Because SAYCam contains naturalistic video frames from children’s head-cam footage, many detections are of low-quality due to motion blur, wrong/irrelevant detector predictions, small/partial bounding boxes. Therefore, we score each detection result using four broad signals: (1) detector confidence, (2) CLIP image–text alignment, (3) crop size, and (4) spatial clarity (e.g., centeredness). These signals are normalized per-concept and combined into a single quality measure. We also employ additional light-weight adjustments to ensure that within $\mathcal{M}(s)$, rare labels are not overwhelmed by frequent ones and that exact label matches are preferred over looser variants.
- **Ensuring lexical and visual diversity:** To avoid selecting many near-duplicate frames of the same scene, we apply simple diversity controls. We first ensure that different lexical variants of a concept are represented, and then enforce a minimal temporal spacing between chosen frames. From this diversified pool, we keep only a small number (≤ 10) of final crops per concept, prioritizing clarity and representativeness.
- **Final output:** The result is a compact, high-quality set of image crops for every object or action concept in SAYCam. These curated crops act as the visual foundation for the majority of benchmarks built from SAYCam in this paper. They guarantee concept fidelity, diversity of visual

Table 8. Comparison between SAYCam and Ego4d. Object Size is reported in terms of the average % of the frame’s area filled.

Data Source	participants	Number of Pixels	Object Size
SAYCam	infants	307k (fixed)	57%
Ego4d	adults	over 2M (average)	4%

contexts, and consistent quality standards across tasks.

Ego4D Annotations

For Ego4D, we do not perform any heavy data cleaning or processing due to the native, high-quality annotations of the dataset. For most of our benchmarks, we use image data from the egotracks split, which contains densely annotated egocentric video tracks. In addition, *Picture Vocabulary* (see Section B.2) also draws image crops from `fho_1ta`—a subset of Ego4D focused on future hand–object interactions—providing additional object-centric visual diversity.

Overall differences between SAYCam and Ego4d

Here, we analyze the differences between SAYCam and Ego4d which result in BabyLLaVA-V2’s very poor generalization to Ego4d. Specifically, SAYCam was filmed by 3 babies across 4 homes, while Ego4d was filmed by 923 participants across 74 sites. There’s also a significant domain shift in the size of the frames and the sizes of the objects relative to the frames. See Table 8 for a summary.

In addition, although all of the Ego4d examples constructed in *DevCV Toolbox* are directly based on objects listed in SAYCam’s vocabulary, their backgrounds may still include objects that BabyLLaVA-V2 never saw in its training, and thus detract from its’ overall understanding of the scene. For example, we might construct an example from Ego4d that asks about the location of a *hand*, and although BabyLLaVA-V2 saw examples of *hand* during training, the frame is full of other objects to which BabyLLaVA-V2 can attribute no meaning. In such a case, context clues learned by BabyLLaVA-V2 about where gloves are usually found relative to their scene, such as at the end of an arm or holding onto a known object, are lost, and performance drops correspondingly. Further, we conjecture that this lack of generalization stems from not only the explicit action categories included in Ego4d that a baby would never have seen (like fixing a car or performing a laboratory experiment), but also from the inherently wider field of view captured adult demonstrators relative to babies. Further, we argue that even if Ego4d had been filmed of the same locations and actions as SAYCam, we would still observe a domain shift caused solely because the demonstrators are adults, perceiving the world from a higher point of view than babies. This point reinforces the uniqueness of the baby domain in the space of egocentric computer vision.

B.2. Picture Vocabulary

Original Toolbox Task

Our task is directly adapted from the NIH Baby Toolbox® Picture Vocabulary Test (PVT), which evaluates a participant’s receptive vocabulary by presenting a spoken target word alongside four images (one correct, three distractors) [11]. The goal is to touch the picture matching the target word. Distractors in the original PVT are designed to be *plausible but incorrect*, typically encompassing coarse-categorical, fine-categorical, or phonological similarity. While the full PVT, taken directly from the NIH Toolbox® [12, 15], includes 373 examples, we identify 52 examples intended for early childhood receptive vocabulary evaluation through combining all-MiniLM-L6-v2 embedding similarity [57] comparison to vocabulary in [35] and manual inspection.

While the Baby Toolbox PVT uses an IRT-based computer-adaptive score that converts response patterns into age-normed ability estimates [15], our adaptation simplifies this to straightforward 4-way accuracy since all items in the benchmark are evaluated rather than adaptively selected.

Our adaptation preserves the original developmental intent while replacing controlled illustrations with naturalistic egocentric visual inputs (SAYCam/Ego4D), providing a grounded benchmark for modeling baby-level vocabulary comprehension in realistic developmental environments.

Adaptation

To adapt the original PVT design to naturalistic corpora, we first map MAB-CDI words $r \in \mathcal{R}$ to corpus vocabularies \mathcal{S} : GPT-annotated labels for SAYCam and native objects/actions labels for Ego4D. This produces a set of visually grounded targets $\mathcal{G}_r \subset \mathcal{S}$ for each CDI anchor r , forming a one-to-many mapping $r \rightarrow \mathcal{G}_r$.

We then analyze the 52 baby-level NIH PVT items to quantify the original distractor structure. We define three categories: fine-categorical, coarse-categorical, and phonological. We manually annotate every NIH distractor to one or more of these types accordingly. Note that the original PVT includes unrelated distractors and we exclude those given the difficulty in controlling the quality of unrelated distractors in naturalistic imagery. We obtain the unnormalized distractor-type weights:

$$w_{\text{coarse}} = 0.5643, \quad w_{\text{fine}} = 0.1472, \quad w_{\text{phon}} = 0.0321.$$

Using these proportions, we construct corpus-specific distractor pools from the entire corpus \mathcal{S} (because the model is only required to identify the correct target concept, not to correctly recognize or label the distractors):

- **Fine-categorical:** We use similarity scoring based on CLIP text embeddings [44]. For SAYCam, candidates

above a similarity threshold of 0.7 is considered belonging to the same fine-grained category while for Ego4D we use a quantile band [0.997, 0.99973] to also filter out overly similar and thus indistinguishable words.

- **Coarse-categorical:** We use Kmeans clustering based on CLIP text embeddings (SAYCam: $K = 100$; Ego4D: $K = 150$).
- **Phonological:** We use Soundex-based string similarity for both datasets.

For each CDI anchor r , we select a ground-truth label $g \in \mathcal{G}_r$ and sample three distinct distractors from these pools using the weights. For SAYCam examples, we perform a final round of manual screening to filter out the infeasible examples, while Ego4D examples are filtered with a hybrid procedure combining Gemini2.5-flash checks with a lightweight manual review.

Data Collection

To produce high-quality 4-way visual choices, we collect image crops corresponding to every target and distractor label.

For SAYCam, because *Picture Vocabulary* requires extremely precise, semantically clear images, we modify the fully automated pipeline in Section B.1 with the following changes:

1. Candidates come *directly* from $\text{Index}(g)$ where $g \in \mathcal{G}_r$ given anchor r .
2. Human annotators manually filter irrelevant, ambiguous, or blurry crops and refine bounding boxes, replacing automated quality scores.

This process yields a compact, high-precision crop inventory used for all SAYCam examples.

For Ego4D, for the objects, we use the bounding boxes from the `visual_crop` field of the EgoTracks benchmark, applying a deterministic buffer ($1.2 \times + 8\text{px}$ margin) and requiring a post-buffer normalized area > 0.03 . For actions, we use the `fho_lta` benchmark which contains abundant action annotations. As there are no explicit bounding box annotations, we sample frames from the middle 25% of each action frame interval and apply minimal center-biased cropping to maintain clarity. For each label, we keep 10 candidates while preserving diversity and visual fidelity, and we apply a Gemini2.5-flash pass to eliminate unusable crops.

Dataset composition. As shown in Figure 7, We obtain 1181 SAYCam examples, covering 344 unique GT labels, 1311 unique distractor labels, and 1660 unique crops. Due to manual filtering, its distractor distribution only loosely follows NIH proportions (shown in Figure 8). Similarly, we obtain 346 Ego4D examples over 124 unique GT labels, 343 unique distractor labels, and 633 unique images (shown in Figure 7) with the corresponding distractor distribution

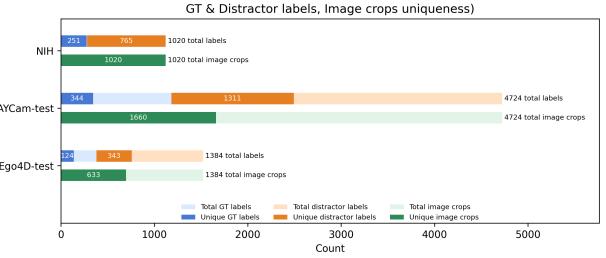


Figure 7. Label/Image crop uniqueness comparison between Picture Vocabulary Test in NIH Baby Toolbox®, SAYCam, and Ego4D.

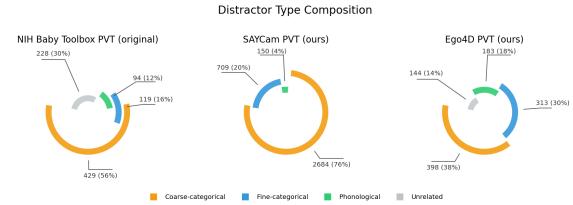


Figure 8. Distractor type composition for the Picture Vocabulary Test in NIH Baby Toolbox®, SAYCam, and Ego4D. The original PVT contains multi-type overlaps, while our sampling assigns each distractor a single type even though some satisfy multiple cues. Ego4D uses unrelated distractors only as a rare fallback.

shown in Figure 8.

Example Prompt

Each finalized example is a prompt embedded with 4 image choices for which the following is an example:

```
" Touch the image of 'foot' (A)
<image> (B) <image> (C) <image>
(D) <image>"
```

The model needs to output one of A, B, C, or D to be evaluated.

B.3. Looking While Listening

Original Toolbox Task

The Looking while listening test (LwL) from NIH Baby Toolbox® aims to evaluate comprehension for object labeling and receptive language [15]. The infant is shown two clipart images which is followed by an audio prompt describing one of them. Eye tracking is used to detect whether the participant is looking at the ground-truth image. Similar to PVT, we simplify the original metric to accuracy only.

Adaptation

To adapt LwL to our benchmark in SAYCam, We replace clipart with naturalistic image crops from SAYCam, and eye tracking with multiple choice, similar to *Picture Vocabulary*.

Data collection

Examples for *Looking While Listening* are taken directly from *Picture Vocabulary* examples.

Example Prompt

Each finalized example is a prompt embedded with 2 image choices for which the following is an example:

" Touch the image of 'foot'
(A) <image> (B) <image>"

The model needs to output one of A or B to be evaluated.

B.4. Localization

Original Toolbox Task

Much like Picture Vocabulary, the Mullen Receptive Language test #19 tests infants on their ability to point at sketched *target* objects as they are named, avoiding confusing them with the *distractor* objects. Specifically, after gesturing to a group of sketched objects, the psychologist asks: *Look at these. Where is the cat?* If the child points in the direction of the cat, they pass the test.

Adaptation

Localization makes a significant modification to the original NIH Baby Toolbox® measure- In *DevCV Toolbox*, we find it meaningful to test pointing to objects *in their naturalistic environments*, namely, we treat the objects naturally occurring in the background of the frame as distractors rather than inserting unrelated objects. Additionally, because it is infeasible to ask a model to 'point', the answer choices are always *top left*, *top right*, *bottom left*, *bottom right*.

Again, the objects in this task are real objects from SAYCam and Ego4d rather than the sketches used in the NIH Baby Toolbox®, and just like in the NIH Baby Toolbox®, the prompt is the full frame and the name of the object to be localized.

Data collection

The examples for both SAYCam and Ego4d are generated using the centers of the bounding boxes annotated in [B.1](#) and Ego4d's egotracks, respectively.

To avoid including test examples where a bounding box stretches across two answers ambiguously (for example, an object in the bottom middle that could reasonably be called either *bottom left* or *bottom right*), we 1) crop each frame so that its closest corner is flush with the edges of the object's bounding box, and 2) enforce a maximum bounding box area of 1/4 of the frame's area (see [Figure 9](#)), which filters out 5.2k of the 7.3k possible test examples. In practice, we find that both of these steps are needed to ensure fair, reasonably unambiguous examples. We enforce no minimum confidence in the SAYCam object annotations

Localization Data Collection

Case 1: Answer is ambiguous

"Where is the cat?
Answer with *top left*, *top right*,
bottom left, or *bottom right*."

Answer: *top left* or *top right*



Crop frame about bounding box

"Where is the cat?
Answer with *top left*, *top right*,
bottom left, or *bottom right*."

Answer: *top right*



Case 2: Object too large to be localized

"Where is the giraffe?
Answer with *top left*, *top right*,
bottom left, or *bottom right*."

Result: Remove Example



Figure 9

and use all object names generated in [B.1](#).

Example Prompt

Each finalized example is a prompt embedded with one image and the same four choices, for which the following is an example:

"<image>
Point at the cup. Is it in (A)
the top left of the image, (B)
the top right, (C) the bottom
left, or (D) the bottom right?"

The model needs to output one of *top left*, *top right*, *bottom left*, or *bottom right* to be evaluated.

B.5. Left/Right

Original Toolbox Task

Left/Right is adapted directly from Mullen Visual Reception test #29, in which a psychologist shows a child an object, then instructs the child to match it with the identical one. If the child correctly points to the identical object,

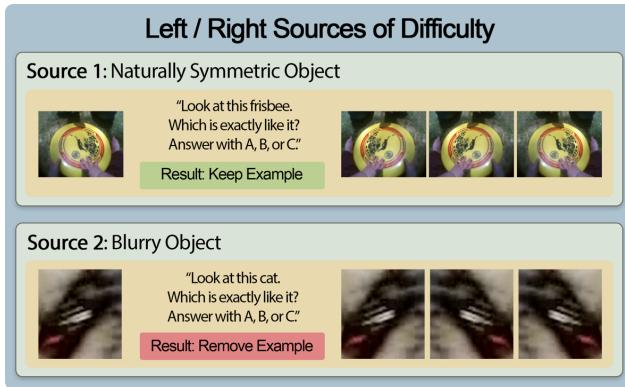


Figure 10

avoiding confusing it with its own mirror image, they pass the test.

Adaptation

The only modification made while adapting VR test #29 to *DevCV Toolbox* is replacing the clipart objects with real objects from SAYCam and Ego4d. In *DevCV Toolbox*, the basic format is preserved: a prompt image, followed by a correct answer and two distractor choices in some random order, are presented to the model. The target image is a duplicate of the prompt, and the incorrect answers are the mirror image of the target.

Some examples in *Left/Right* are harder than others; we conjecture that difficulty in this task can result from either 1) naturally symmetric objects, or 2) low resolution objects (see Figure 10). Naturally symmetric objects are difficult because they require an encoding of fine-grained details. However, low resolution objects are difficult because even though there might be some spatial clues to discriminate the target from its mirror image, if have models can't ascribe any semantic meaning to the image, they won't encode any semantic meaning to its details. By filtering out small bounding boxes, we aim to remove the examples that are difficult solely due to low resolution.

Data collection

For the SAYCam variant, use the object names and bounding boxes generated in B.1. We enforce no minimum or maximum object size, and for the val and test splits, we enforce a minimum confidence in the bounding box of .85. In both variants, all object crops are zero-padded to (640, 480).

For the Ego4d variant, we use object names and bounding boxes from the published Ego4d egotracks annotations and include only objects that belong to the vocabulary defined in B.1. To remove examples with poor resolution, we require either a minimum bounding box height or width of one fifth of the frame, which filters out about half of the otherwise qualifying examples.

Example Prompt

Each finalized example is a prompt embedded with 1 image prompt and 3 image choices for which the following is an example:

```
"<image>
Which of the following is the
same as this? (A) <image> (B)
<image>, or (C) <image>?"
```

The model needs to output one of A, B, or C to be evaluated.

B.6. Spatial Details

Original Toolbox Task

Similarly, Mullen Visual Reception test #25 also tests understanding of details in images. In this test, the child is presented with a sketch of a *tulip*, and the psychologist asks: *See this flower. Find one just like this. Look for it here*, while tracing their finger along a page filled with sketches of a *tulip*, a *sunflower*, a *clover*, and a *daisy*. The child is allowed to refer back to the *tulip* while choosing their answer. If the child points to the *tulip*, they pass the test.

Adaptation

Again, the objects in our benchmark are real, cropped objects from SAYCam and Ego4d rather than clipart, and they come from more categories than just *flower*. Additionally, because the models cannot "point" to the choices, the choices are passed as separate images and the correct answer is the index (A, B, or C) of the matching image.

Our final modification to the original measure is that to make it more difficult for a computer, we present the answer choices in their naturalistic backgrounds rather than cropped as in the NIH Baby Toolbox®. In practice, we find the final modification necessary to make *Spatial Details* require a fine-grained understanding of detail, as matching identical images is trivial even for a small vision model.

Data collection

To construct examples from both SAYCam and Ego4d, we match objects with the label, but require that they come from different videos. The labels for each come from B.1 and egotracks, respectively. Note that the same object can appear multiple times within an example- for instance, the same *chair*, captured in two separate videos, can show up as two of the choices. In such cases, the model is forced to rely on spatial details such as orientation, perspective, and lighting, to match identical occurrences.

To ensure quality, we enforce a minimum object confidence of .92 in the SAYCam annotations. To increase difficulty, we also require that objects have an area of less than half of the frame's area.

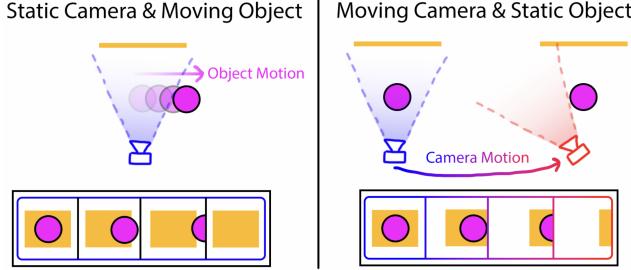


Figure 11. Comparison between sources of occlusion. **Left:** object occlusion from a static camera and moving object. **Right:** object occlusion from a moving camera and static object. Each panel shows a top-down view of the scene along with the corresponding projected 2D video depicting the occlusion event.

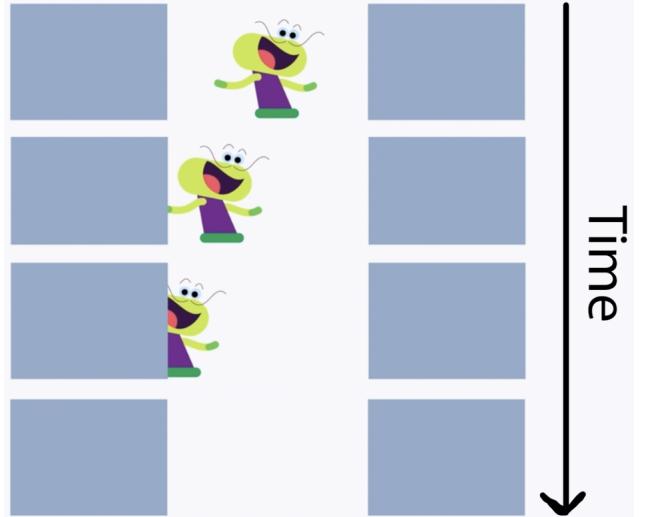


Figure 12. Example of Visual Delayed Response task, taken directly from the NIH Baby Toolbox.

Example Prompt

Each finalized example is a prompt embedded with 1 image prompt and 3 image choices for which the following is an example:

```
"<image>
Which of the following is the
same as this? (A) <image> (B)
<image>, or (C) <image>?"
```

The model needs to output one of A, B, or C to be evaluated.

B.7. Visual Delayed Response

Original Toolbox Task

Inspired by Visual Delayed Response in the NIH Baby Toolbox, we introduce an evaluation task designed to assess the spatiotemporal reasoning capabilities of vision-language models. More specifically, our task focuses on object tracking and spatial localization over time, requiring models to process multi-frame/video input to infer spatial trajectory and disappearance of a designated object.

Adaptation

In the original NIH Baby Toolbox task, a cartoon creature is placed in a frame with grey walls to its left and right. The creature moves from the center of the frame to behind either wall, and the child must identify which wall the creature hid behind. (See Figure 12)

Translating this task to real-world videos is challenging, as the synthetic examples from the toolbox portray an unrealistically ideal scenario. Each toolbox example depicts a moving object observed from a static camera perspective, with simplified backgrounds and perfectly smooth motion trajectories. Such controlled scenarios are rare in real-world footage, especially in egocentric videos captured from a toddler’s perspective.

To address this challenge we exploit the frequent head movements captured in SAYCam footage, together with the

fact that many objects in real-world scenes are largely stationary. By inverting the source of 2D object motion from a static camera with moving objects to a moving camera with stationary objects (see Figure 11), we are able to expand the dataset by over an order of magnitude.

Formally, the model is provided a video $V = \{f_1, f_2, \dots, f_T\}$ and designated key object k . The video depicts the key object k moving within the field of view and eventually exiting the visible frame at time $t^* \leq T$. The model’s objective is to predict the exit region $r \in \mathcal{R}$, where \mathcal{R} denotes the set of possible frame boundaries through which the object may leave.

We define two variants of this task, which differ in the set of selectable exit regions \mathcal{R} provided to the model:

- **Multi-choice setting:** $\mathcal{R}_m = \{\text{left}, \text{right}, \text{top}, \text{bottom}, \text{top-left}, \text{top-right}, \text{bottom-left}, \text{bottom-right}\}$
- **Binary setting:** $\mathcal{R}_b = \{\text{correct}, \text{opposite}\}, \mathcal{R}_b \subseteq \mathcal{R}_m$

The multi-choice variant provides a comprehensive set of possible exit regions, where the model is given eight regions as selectable options. The binary variant is a simplified version of the task, where the model only chooses between two options: the correct exit region or the region directly opposite to it.

The overall task can be summarized as a mapping $f_{VDR}(V, k) \rightarrow r$, where $r \in \mathcal{R}$. Here, f_{VDR} represents the function that, given a video V and designated key object k , predicts the exit region $r \in \mathcal{R}$ through which the object leaves the frame.

Data Collection

SAYCam. Collecting examples for the SAYCam variant of Visual Delayed Response can be split into 3 stages: fil-

tering with GPT annotations, filtering with object tracking, and manual labeling.

Stage 1. We first use the 1 FPS annotations provided by GPT in B.1, where each frame is labeled with a "key object" and "objects" attribute. The "key object" denotes a singular object being attended to in a particular frame (if any), and "object" denotes a list of all visible objects within the frame of view. We do an initial filtering for candidate clips by using a sliding window over the 1 FPS frames of each long-range video. For a clip to pass the filter, the first half of frames in the window must have the same "key object", k . In addition, the second half of frames must not have k listed as a "key object" or be present in the "objects" list. From the 422990 initial clips, 17443 are passed as candidate clips to the next stage.

Stage 2. We then perform open-set object detection [55] over the 1 FPS frames sampled from each candidate clip, where the only object class to be detected is the "key object" itself. An object tracking algorithm [65] is also used to track the "key object" over the full fps video. The clips are filtered according to the object tracks, where each track must satisfy all of the following:

- Start within the middle 70% of the frame
- Appear in at least 10 consecutive frames
- Disappear for at least 10 consecutive frames before the full clip ends

To help account for errors in the object detection/tracking, we purposefully loosen the filters and add additional measures for sporadic/false detections. From the 17443 initial clips, 3908 are passed as candidate clips to the next stage.

Stage 3. The final stage involves manually reviewing and hand-labeling each candidate example from the previous stage. We label not only for the ground truth exit direction, but also for a variety of annotations related to overall quality of the clip. In total, we annotate for camera motion, scene visibility, camera stability, occlusion, exit direction, and presence of multiple objects. A breakdown for each is provided as follows:

- **Occlusion:** {Fully Occluded, Partially Occluded, Remains in View}
- **Camera Motion:** {Static, Moving}
- **Direction of Exit:** {Up, Down, Left, Right}
- **Scene Visibility:** {Excellent, Good, Fair, Poor}
- **Camera Stability:** {Very Stable, Stable, Shaky, Very Shaky}
- **Multiple Objects:** {True, False}

We then filter for valid high-quality clips according to the following criteria:

- Object must become fully occluded
- Direction of exit cannot be contradicting (both left & right, or both up & down)
- Scene visibility better than "Poor"
- Camera stability better than "Very Shaky"

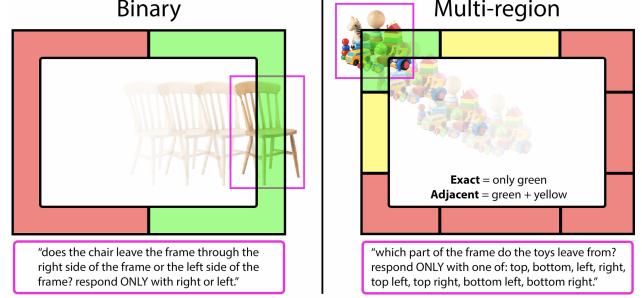


Figure 13. Visualization of evaluation methods for Visual Delayed Response task. **Left:** Binary evaluation for the binary setting, where there is only a correct and opposite incorrect option. **Right:** Exact and Adjacent evaluation for the multi-region setting, where the correct region for Exact is defined by only the green region, and the correct region for Adjacent is defined by both the green and yellow regions.

From the 3908 initial clips, 2380 are passed as final clips for the dataset.

Ego4D. Data collection for Ego4D follows a very similar structure to the SAYCam process, with the addition of tracked object annotations being already provided by the Ego4D dataset. We use a sliding window over each long-range video's object tracks and filter for all of the following:

- Object is present in first half of window and disappears in second half
- Object bounding box ≥ 40000 pixels (13% of screen)
- Starts within the middle 50% frame

Each clip is also manually reviewed/labelled according to the same procedure as SAYCam data collection

Multi-frame versions. Since the average clip can range from 100-150 frames, we manually create multi-frame counterparts to each example. More specifically, we look to obtain 1 representative object frame and 3-9 linearly sampled frames that best showcase the object motion/disappearance in a given clip. To do this, we first find the specific frame for three different fields: full object frame, start occlusion frame, and end occlusion frame. The full object frame is always used as the first frame in the multi-frame sequence, and shows the key object in clear view. The start/end occlusion frames mark the interval with which the key object becomes occluded. A random number of frames (3-9) are linearly sampled along this interval to complete the multi-frame sequence for a given clip.

Evaluation

Evaluation is performed over three separate variants: **Exact** and **Adjacent** in the multi-choice setting, and **Binary** in the binary setting (see Figure 13). Accuracy is used as the metric for evaluation, defined as the fraction of predictions considered correct across all trials for a given variant. In the multi-choice setting:

- **Exact:** Only the labelled ground truth region is counted as correct.
- **Adjacent:** Both the labelled ground truth region and its two adjacent regions are counted as correct. This helps account for small ambiguities in the ground truth label.

In the binary setting:

- **Binary:** A prediction is correct if it matches the "correct" region rather than the "opposite" region.

Example Prompt

Each finalized example includes a series of `<image>` tags or singular `<video>` tag, followed by the prompt. To be properly evaluated, the model must output exactly one option from the choices given in the prompt.

Example from binary setting with multi-frame input:

```
"<image><image><image><image>
does the bottle leave the frame
through the right side of the
frame or the left side of
the frame? respond ONLY with
'right' or 'left'."
```

Example from multi-choice setting with video input:

```
"<video>
which part of the frame do the
toys leave from? respond ONLY
with one of: 'top', 'bottom',
'left', 'right', 'top right',
'top left', 'bottom right', or
'bottom left'."
```

B.8. Memory

Original Toolbox Task

The Memory task in the NIH Toolbox is designed to measure how well toddlers (22–42 months old) learn and remember new information using a touchscreen. Children play a short game where they “feed” hungry cartoon animals by touching them on the screen. The test is divided into the learning phase and the test phase.

- **Learning phase:** children see pairs of animals and are told to touch the new animal—the one they have not fed before. They complete 10 trials and receive feedback so they can learn the rules and memorize the animals seen in this phase.
- **Testing phase:** children again see pairs of animals and told to touch the new animal, where each old animal from the learning phase appears twice, each time paired with a different new animal. They complete 20 trials and receive no feedback so correct responses reflect their memory for animals in learning phase.

The animals were selected based on how many 24-month old infants were familiar with them according to data from

the MB-CDI Wordbank. Performance is scored based on whether the child touches the correct animal in the testing phase, along with optional reaction time measures to show how quickly they respond.

Adaptation

To simplify the problem and enlarge the potential dataset size, we define the set of word labels used in the learning phase as

$$\mathcal{W}_{\text{learn}} = \{w_1, w_2, \dots, w_k\},$$

where each w_i corresponds to an image $x_i \in \mathcal{X}_{\text{learn}}$. These image–label pairs (x_i, w_i) serve as the stimuli to be memorized during the learning phase. We further sample $2k$ additional word labels for the testing phase,

$$\mathcal{W}_{\text{test}} = \{w_{k+1}, w_{k+2}, \dots, w_{3k}\},$$

each associated with a novel image $x_j \in \mathcal{X}_{\text{test}}$.

At each round t , the Vision–Language Model (VLM) receives an input consisting of two images and a text prompt:

$$I_t = \{x_{p_t}, x_{q_t}, P_t\},$$

where x_{p_t}, x_{q_t} are the image inputs and P_t is the corresponding prompt.

- **Learning phase:** The learning phase contains k rounds:

$$I_t^{\text{learn}} = \begin{cases} \{x_1, P_1\}, & t = 1, \\ \{x_{t-1}, x_t, P_t\}, & 2 \leq t \leq k, \end{cases}$$

where the two images in the second case are presented in random order. This setup enables the model to incrementally associate visual concepts across consecutive rounds within a single context window.

- **Testing phase:** The testing phase consists of $2k$ rounds, each comparing a learned stimulus with a new one:

$$I_t^{\text{test}} = \{x_{i(t)}, x_{j(t)}, P_t^{\text{test}}\}, \quad x_{i(t)} \in \mathcal{X}_{\text{learn}}, \quad x_{j(t)} \in \mathcal{X}_{\text{test}}.$$

Here, $x_{i(t)}$ is a previously seen image and $x_{j(t)}$ a novel one. The model must identify which image corresponds to the new concept described in P_t^{test} .

Evaluation

Each learned concept $w_i \in \mathcal{W}_{\text{learn}}$ is paired with two distinct new concepts:

$$(w_i, w_{a(i)}), (w_i, w_{b(i)}), \quad a(i), b(i) \in \{k+1, \dots, 3k\}, \\ a(i) \neq b(i), \tag{1}$$

forming two dyads per old stimulus and a total of $2k$ dyads in the testing phase. To mitigate the influence of random guessing, an old stimulus w_i is considered successfully *remembered* only if both of its dyads are answered correctly:

$$r_i = \begin{cases} 1, & \text{if both dyads for } w_i \text{ are correct,} \\ 0, & \text{otherwise.} \end{cases}$$

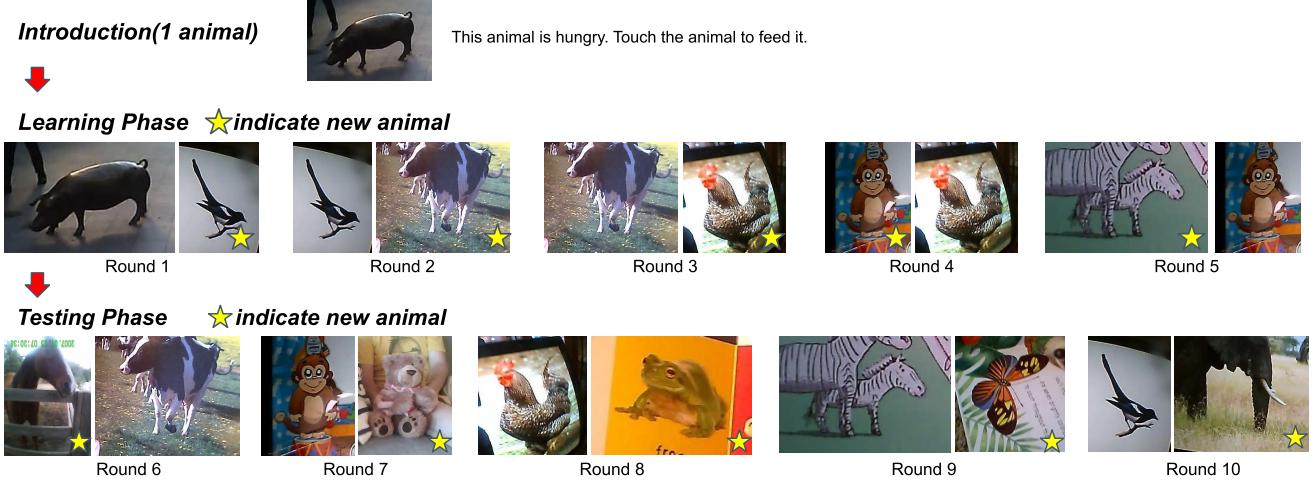


Figure 14. A sample of our memory task adaptation. We use the MAB-CDI words detected in SAYCam as the images to be memorized.

The overall memory accuracy is then computed as

$$\text{Acc}_{\text{mem}} = \frac{1}{k} \sum_{i=1}^k r_i.$$

In all experiments, we set $k = 5$, resulting in a total of $3k = 15$ distinct word–image pairs. This design preserves the spirit of the original Toolbox while adapting the procedure to the VLM’s limited context window. When designing the evaluation metric, we follow the structure of the original Toolbox with appropriate simplifications. Specifically, we remove the original intermediate 6–8 min delay settings between the learning and testing phases in our benchmark design. Future extensions may incorporate external memory mechanisms such as Retrieval-Augmented Generation (RAG), or introduce irrelevant contexts between the two phases to simulate real-world temporal gaps. In this work, however, we focus exclusively on assessing the model’s in-context retrieval ability.

Data collection

For the scalability of the memory task, we expanded the image set from the cartoon animals in the original Toolbox to the objects in the SAYCam dataset, which also ensures that the items are familiar to children. We used a combination of annotation-based search scripts and automated vision models, including CLIP for object–text similarity and SAM for object segmentation as shown in B.1, to find and isolate frames where these objects appeared clearly. Manual screening was also done after auto-filtering. This process allowed us to gather real-world visual examples of common objects seen by young children, supporting the creation of new learning and memory trials for our benchmark. The visual objects collected from SAYCam

dataset will serve as our stimuli in the memory task.

Example Prompt

Each finalized example is a list of prompts each embedded with 2 image choices, for which the following is an example:

"Let's try more.
Touch the new image.
(A) <image> or (B) <image>."

The model needs to output one of A or B to be evaluated.

B.9. Who Has More

Original Toolbox Task

In the NIH Baby Toolbox®, the Who Has More Measure is poised as a simple narrative: there are two animals; each of them is pictured with some number of the same object. Which animal has more?

Adaptation

In DevCV Toolbox, we remove the narrative aspect and replace the clipart objects with naturalistic SAYCam and Ego4d objects. In the *Naturalistic* adaptation, the objects are not necessarily identical and appear in their naturalistic backgrounds; in the *Synthetic* adaptation, the objects are perfectly identical, cropped, and pasted onto black backgrounds in matching layouts. The model is prompted to identify whether the *first* or *second* has more.

Data collection

In the synthetic variants, to pick the two quantities to compare, we first sample a number between one and ten. Then, from the numbers remaining that are *lower than the first one*, we sample the second quantity. We do this to ensure

a balanced distribution in the *differences in numbers being compared* for each answer. The objects being compared come from the annotations in B.1 and egotracks for SAY-Cam and Ego4d, respectively.

For the test sets in the naturalistic adaptations, each example is hand annotated by two separate human experts to cross-validate annotation quality. Specifically, the first human expert labels video frames with an object type and the number of that object. Next, for each the frames that the first annotator labeled, the second annotator labels the number of the named object in each, *without access to the first's annotation*.

With both labels for each frame, we construct an example for every pair of frames of with objects of the same type for which *both annotators would have arrived at the same answer as to which has more had they based their decision solely on their count annotation*. As an example, say the first annotator labels frame A as having 5 cups, and frame B as having 6 cups. If the second annotator labels 5 cups in frame A and 7 cups in frame B, we construct a *Who Has More* example from frames A and B (despite the annotators giving frame B two different labels) because $5 < 7$ and $6 < 7$. However, if the second annotator instead labeled frame B as having 5 cups, we *do not* construct a *Who Has More* example from frames A and B, because the two annotators would have given different answers for such an example.

In constructing *Who Has More*, we observe that some objects occur in multiples more than others, and each object follows a unique (and usually nonuniform) distribution of quantity- for example, the number of hands visible in a frame is usually one or two and rarely another number, while an object like books could reasonably be seen in any quantity between one and ten. Additionally, we observe that given the differences in settings and scene perspective, the distributions of object types as well as quantity per object is inherently different for SAYCam and Ego4d.

Example Prompt

Each finalized example is a prompt embedded with 2 image choices for which the following is an example:

Which of the following has more
of shoe? (A) <image>, or (B)
<image>?"

The model needs to output one of A or B to be evaluated.

B.10. Subitizing

Original Toolbox Task

In the NIH Baby Toolbox[®], the infant sees one to four colored dots for only one second, then an audio prompt requests the number of dots. Importantly, the dots are not shown for long enough to be counted one at a time- Subitize is intended to measure the ability to *quickly identify small*

quantities, without counting.

Adaptation

To construct *Subitizing* in *DevCV Toolbox*, we paste objects onto random locations on black frames, in random quantities between one and four. To simulate the "one second flash", we insert empty frames before and after the frame including the objects.

Data collection

In the SAYCam variant, the objects being pasted come from frames cropped by the bounding boxes obtained in Section B.1, subjected to a minimum confidence of .95. In the Ego4d variant, the bounding boxes come from egotracks, and only objects in the MAB-CDI vocabulary are included.

Example Prompt

Each finalized example is a prompt embedded with 1 blank frame, 1 image prompt, and 1 blank frame for which the following is an example:

```
<image> <image> <image>
How many of apple did you see?
Answer with 1, 2, 3, or 4."
```

The model needs to output one of 1, 2, 3, or 4 to be evaluated.

B.11. Object Counting

Original Toolbox Task

In the NIH Baby Toolbox[®], infants are shown some number of an object on a screen, and asked to count them. Unlike the Subitize measure, there is no time limit- participants have time to count each item individually.

Adaptation

In *DevCV Toolbox*, the examples are constructed in the same way as the *Subitizing* examples, except the quantities are between one and twelve, and there are no blank frames corresponding with the lack of a time limit.

Data collection

The data collection for *Object Counting* is the same as for *Subitizing*.

Example Prompt

Each finalized example is a prompt embedded with 1 image prompt, for which the following is an example:

```
<image>
How many of chair did you see?
Answer with a number 1-12."
```

The model needs to output a number between 1 and 12 to be evaluated.

C. Human survey

C.1. Small-scale human adult test

To confirm the validity of *DevCV Toolbox*, we collect small-scale adult performance data on eight of the ten tasks. We omit *Looking While Listening* and *Subitizing* as their examples are directly taken from *Picture Vocabulary* and *Object Counting*, respectively. In total, we have data from n=11 adult participants, each completing 10 trials per task for the SAYCam variants of *Picture Vocabulary*, *Localization*, *Left/Right*, *Spatial Details*, *Visual Delayed Response*, and *Object Counting*, and 5 trials per task for the SAYCam variants of naturalistic *Who Has More* and synthetic *Who Has More*, and as well as the Ego4d variants of all tasks other than *Memory*. Participants completed 30 consecutive rounds of each *Memory* variant, requiring a maximum memory of 29 distinct images.

Results for each task can be found in the *Human performance* rows of Tables 4 and 9. In summary, our participants achieved an average accuracy of 93.0 on all SAYCam tasks and 93.5 on all Ego4d tasks, for both of which they far outperform any model. From this, we conclude 1) *DevCV Toolbox* is a valid discriminator of vision FMs with adult performance as a strong upper bound, and 2) the SAYCam and Ego4d variants have roughly similar complexity and ambiguity for humans.

C.2. Children Helping Science tests

To further examine the developmental fidelity of *DevCV Toolbox*, an IRB review process is currently underway to extend this survey to a *large scale children survey*, where we plan to collect response data for each task from children of the ages recommended for the corresponding NIH Baby Toolbox® measure.

To this end, we collaborated with expert psychologists to develop child-friendly web interfaces for selected tasks and prepared them for deployment on the online developmental research platform Children Helping Science (CHS) [49]. CHS is a widely used, home-based platform through which families can participate in browser-based developmental studies run by researchers worldwide. By adapting our SAYCam-based tasks (PV, VDR and Memory) to CHS, we aim to collect performance from young children under conditions analogous to the NIH Baby Toolbox®. At the time of writing, the studies are under review and not yet live. We show two examples of our task UI design in Figures 15 and 16.

Taking PV as an example (Figure 15), to approximate the modality of the original NIH Baby Toolbox® task, which relies on audio-visual interaction with spoken prompts and observed child responses, we design an *audio&video test page* to verify that instructions and target words can be delivered clearly via audio and that the child’s webcam setup

is functioning for basic participation monitoring. The *instruction page* provides caregiver-friendly guidance in both text and spoken form. Finally, the *trial pages* present each example in a clean 2×2 grid of four large image options, paired with an audio prompt of the target word, optimizing engagement and accessibility for infants and toddlers while staying faithful to the original task format.

Following the PV setup, VDR also has an initial audio & video test page, along with an instruction page to provide context of the experiment to the caregiver. The trial page for this task (see Figure 16) displays the object that should be tracked, along with the video clip itself and two selectable arrows to submit an answer. Since MP4 with interactive display is not yet supported on the website, a GIF is created in its place. The beginning 5 seconds of the GIF show the first frame with a countdown, then the clip is played as normal and followed with another 5 second buffer to show that the video has ended. To help the caregiver and child understand the experiment, an interactive demo is played as the first 3 trials to showcase how each one should be properly done.

D. Additional experiments & details

D.1. Out-Of-Domain evaluation

To test BabyLLaVA-V2’s capability of generalizing to unseen data domain, we further evaluate it on a set of out-of-domain (OOD) tasks that share the same structure as the in-domain benchmarks but differ in their visual domains. We consider two OOD settings: (1) **Ego4D-based tasks** use egocentric videos from the Ego4D dataset [13], which remain first-person and naturalistic but introduce distinct environments and contexts. (2) **BabyToolbox-based tasks** correspond directly to standardized developmental psychology and clinical assessments, where the visual stimuli are abstract, non-egocentric cartoon images. The detailed test results are reflected in Table 9 and Table 10.

D.2. Importance of the pretraining stage

To evaluate the contribution of the pretraining stage, we compare two variants of BabyLLaVA-V2: (1) the full model trained with Stage 0–2 pretraining before instruction tuning, and (2) a randomly initialized model that skips pretraining and is trained only with Stage 3. For both variants, we fine-tune using different fractions of the instruction dataset and evaluate each model on in-domain tasks.

As shown in Figure 17, the pretrained model consistently outperforms the non-pretrained variant across all data fractions. The gap is especially pronounced when the instruction data is limited, demonstrating that pretraining provides a strong and sample-efficient initialization for downstream instruction tuning. As the instruction data fraction increases, both models improve, reflecting a clear scaling-like trend qualitatively consistent with observations in large-

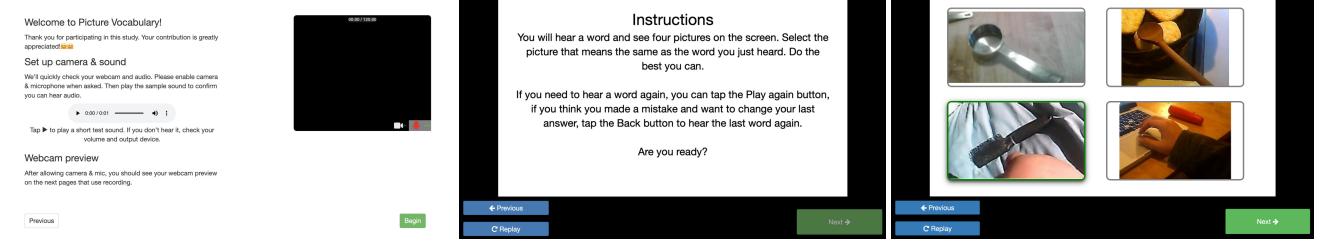


Figure 15. User interface design for our CHS-adapted Picture Vocabulary task.

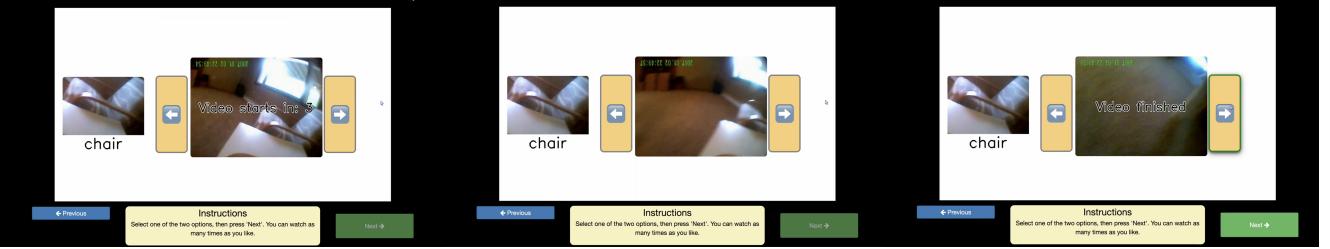


Figure 16. User Interface design for the trial page of Visual Delayed Response task on CHS.

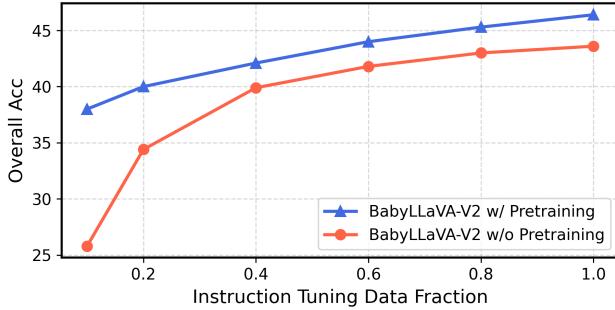


Figure 17. DevCV Toolbox overall performance on different instruction tuning data fraction.

scale model studies [17, 22]. This suggests that data-dependent performance gains also exist in compact, developmentally inspired models, while pretraining remains a crucial component for achieving data-efficient learning.

D.3. Synthetic caption generation

We study the impact of noisy visual-alignment in the naturalistic child-directed utterances transcribed in the pretraining dataset by replacing them with video captions generated by GPT-4o. To encourage diversity in the generated captions and ensure they remain close to the style of the original dataset, we include 10 randomly sampled transcriptions in each prompt. The transcriptions are sampled from a pool of the 1,000 highest confidence transcriptions in the original dataset that contain at least one noun and more than three words. These heuristic filters help ensure that the sampled transcriptions contain stylistic information rather than

simple phrases that are common in the dataset like "wow" or "let's go". The pool of 1,000 transcriptions are manually screened to remove uninformative transcriptions that passed the filtering step. The full prompt to GPT-4o is shown in Figure 18 and an example of a generated caption is shown in Figure 19.

```
messages =
[{"role": "system", "content": "You will be given the frames from a short video clip. Your task is to write a short, accurate caption for the video (1 sentence max). In addition, you should only use realistic child-directed language, like something a parent would say to a toddler. The sentences should be easy to understand for a two-year-old child. Avoid specific names of people, places, or brands. You should also stay in the style of the existing child-directed captions that we have transcribed. To help you with this, you will be given 10 captions from the dataset. Make sure to adhere to their style while still grounding in the visual content of the frames. Here are the 10 captions: {sampledCaptions}", "type": "image_url", "image_url": {video_frames}}, {"role": "user", "content": "Do not use ambiguous or hedging phrases. You should also not output phrases like 'I cannot see anything' or 'I don't know'. If you cannot tell what is happening in the video, make your best guess. Output only the caption (1 sentence max) with no other text: "}]
```

Figure 18. Full prompt for pretraining data ablation

Table 9. **Performance comparison across models on DevCV Toolbox out-of-domain tasks (Ego4D).** Different background colors denote different model families. We report accuracy (%) for all tasks.

Model	Overall	Count	LeftRight	Spatial	PV	Memory	Localization	Visual Delay Response			Who Has More	
								binary	multi-exact	multi-adjacent	synthetic	naturalistic
Upper Bound												
Human performance	93.5	96.4	98.2	96.4	96.4	98.8	90.9	100	58.2	100	100	92.7
Proprietary models												
GPT-4o	67.6	62.1	45.1	94.7	85.3	100	80.4	45.5	13.2	48.3	84.3	84.6
GPT-5	86.7	77.5	88.0	96.8	91.9	100	88.7	94.4	50.3	82.6	94.6	88.5
Gemini-2.5-flash	77.7	72.9	49.6	86.7	92.5	99.2	88.4	80.6	37.1	70.2	97.8	80.1
Gemini-2.5-pro	88.2	81.9	88.0	94.8	91.9	100	90.2	91.3	50.3	87.9	96.5	97.8
Open-source models												
LLaVA-OneVision-0.5B	39.4	43.9	32.6	33.3	27.7	22.6	21.6	73.0	15.2	67.7	46.8	49.4
InternVL3.5-1B	43.7	34.7	34.0	34.1	33.8	24.9	60.7	73.9	16.9	68.5	49.0	49.9
Qwen2.5-VL-3B	48.1	35.7	32.6	44.1	41.9	25.7	86.7	79.8	28.9	51.1	50.2	53.4
Baby models (Ours)												
BabyLLaVA-V2	41.1	33.9	32.9	42.4	29.8	40.7	30.0	55.3	17.7	37.1	86.0	45.8
Lower Bound												
Random guess	31.8	8.33	33.3	33.3	25.0	25.0	25.0	50.0	12.5	37.5	50.0	50.0

Table 10. **Performance on NIH Baby Toolbox out-of-domain tasks.** We report the #correct/#total for all tasks.

Model	Who Has More	Count	Mullen Visual Reception
BabyLLaVA-V2	13/24	2/6	3/12



GPT-4o: Look at the cute chalk drawing!

Original: No I drew his coat on so I protected him from the mud.

Figure 19. Example of caption generated by GPT-4o

Table 11. **Comparison between Gemini-2.5-flash performance with different prompting strategies**

Prompt Type	Count	LeftRight
standard	69	55
one-shot	66	82
alternate prompt 1	55	54
alternate prompt 2	67	56

D.4. Prompting Experiment

Finally, we complete a prompting experiment to show the stability of *DevCV Toolbox* examples with respect to commercial models, the results of which are shown in Table 11. We select *Left/Right* and *Object Counting* for this experiment, as we found that commercial models had the lowest and most variable performance on these. For both tasks, 100 examples are randomly selected and presented to Gemini-2.5-flash with a standard prompt, a one-shot prompt, and two variations of the standard prompt, called *alternate prompt 1* and *alternate prompt 2*. The standard prompt is the one used in all other experiments, and the one shot-prompt is a prompt that includes one other example, with its correct answer, prepended to the standard prompt.

For *Object Counting*, *alternate prompt 1* does not give the object’s name to be counted, e.g. "<image> How many objects do you see?", which we see drops performance, which is intuitive because large models thrive on context, in this case the name of the object to be counted. *Alternate prompt 2* gives more detail, e.g. <image> count the flora very closely, starting from one. Keep track of which ones have already been counted and what number you’ve counted to thus far. Then, report how many flora you counted.". Unsurprisingly, *alternate prompt 2* does not improve performance, showing that 1) the *standard* prompt was sufficient and 2) Gemini-2.5-flash has capable instruction-following capabilities. For *Object Counting*, we find that a one-shot prompt does not boost performance.

For *Left/Right*, the *standard* prompt gives each image token interleaved with their answer labels, e.g. "<image> Which of the following

is the same as this? (A) <image> (B) <image> (C) <image>". In *alternate prompt 1*, we undo this interleaving, resulting in "<image><image><image><image> Which of the following is the same as the first one? (A) the second one, (B) the third one, or (C) the fourth one?". In *alternate prompt 2*, we interleave even more, by giving some descriptive text before the prompt image, e.g. "Here is an image: <image>. Which of the following is the same as it? (A) <image>, (B) <image>, or (C) <image>?". Intuitively we expect *alternate prompt 2* to be the easiest, *alternate prompt 1* to be the hardest, and the *standard* prompt to fall in between. However, we find that none of these prompts elicits significantly different performance, however, the one-shot prompt *significantly* boosts performance. These two findings show the robustness of Gemini-2.5-flash, and the complexity of *Left/Right*, respectively.