BabyVLM: Data-Efficient Pretraining of VLMs Inspired by Infant Learning

Supplementary Material

Examples of Filtered SAYCam Dataset. The filtered SAYCam training dataset consists of 67,280 image-utterance pairs in total. We provide some examples below.

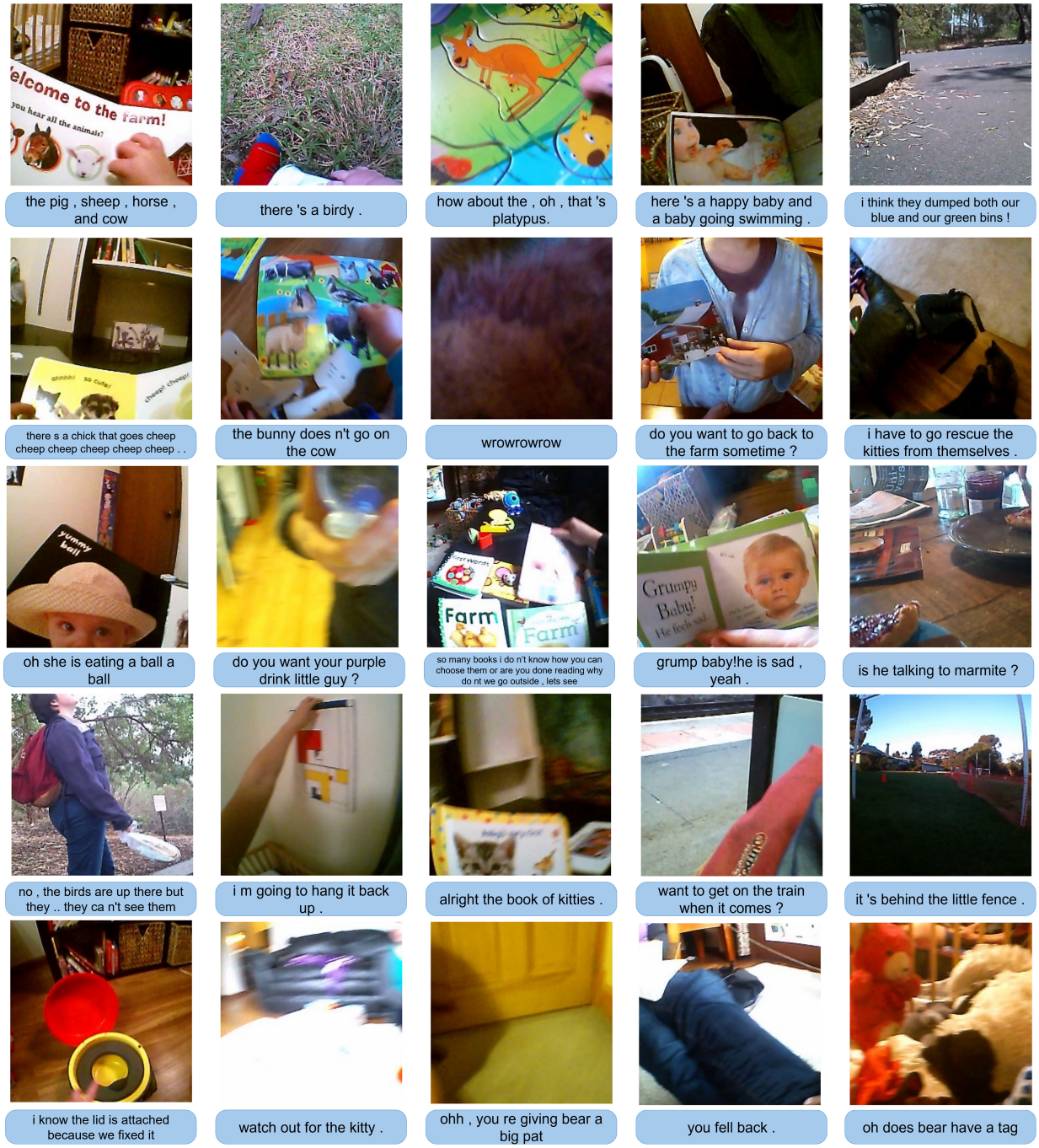

Figure 4. Examples of the filtered SAYCam dataset

Implementation Details for Creating Transferred Training Dataset. Starting from LLaVA's pretraining dataset [6], which includes approximate 558K image-caption pairs coming from CC3M [13], LAION [12], and SBU [9], we carefully design few-shot examples to prompt GPT-4o to transform original captions into simple, natural utterances that a caregiver might say to a two-year-old. Additionally, we instruct GPT-4o to identify captions misaligned with a child's daily experience by explicitly outputting an infeasibility flag in its JSON mode. We get 339,826 feasible samples after this step. The detailed prompt for GPT-4o is provided below.

```
messages = [ {"role":"system", "content": f""" You are an expert in child-directed speech and early childhood education.
Your task is to rewrite the captions below into single, simple utterances that a parent might say to a two-year-old child.
These utterances should:
1. Use simple, familiar words (e.g., "pretty," "big," "red"), mimicking the vocabulary of a two-year-old child (~300–500
   words)
2. Be short and straightforward, with no more than 5–10 words, avoid complex grammar, abstract concepts, or
   unnecessary details.
3. Focus on everyday objects, actions, and simple relationships that a child can understand.
4. Avoid specific names (e.g., people, brands, cities) and replace them with general terms (e.g., "a man," "a city").
5. Use a tone of curiosity or encouragement, as if engaging a child in a conversation.
6. Maintain alignment with the original image caption
Here are some examples of how to rewrite the captions:
Original: "An aerial view of Paris at night from a plane stock photo."
Modified: "What a nice view of the city light."
Original: "Two horses in a field canvas wall art."
Modified: "Do you see two horses in the grass?"
Original: "A beautiful sunset over a calm ocean."
Modified: "The sun is setting. Look at the water."
Original: "person skiing at a slalom event in the downhill."
Modified: "Look, someone skiing fast!"
It's possible that the given image caption is not something a baby would see in daily life and thus inappropriate to be
rewritten, for example, something not realistic or too abstract. In this case, just indicate the "feasible" flag to be False and
output N/A. Otherwise, leave it as True.
Now, rewrite the following caption accordingly:""" }
]
```

Figure 5. Full prompt for transferred dataset creation

To enhance visual consistency, we use CLIP similarity [11] to select a subset of samples matching the size of the filtered SAYCam training dataset. Specifically, we compute CLIP similarity between each image in the filtered SAYCam dataset and every image in the transferred LLaVA pretraining dataset. Given the significantly larger size of the latter, we retain only the top 1,000 most similar images for each SAYCam image, setting the similarity of all others to zero, resulting in a sparse similarity matrix. We then apply the sparse Hungarian algorithm [4] to establish a one-to-one match between images from the transferred dataset and the filtered SAYCam. Examples of the final transferred dataset can be seen in Figure 6 on the next page.

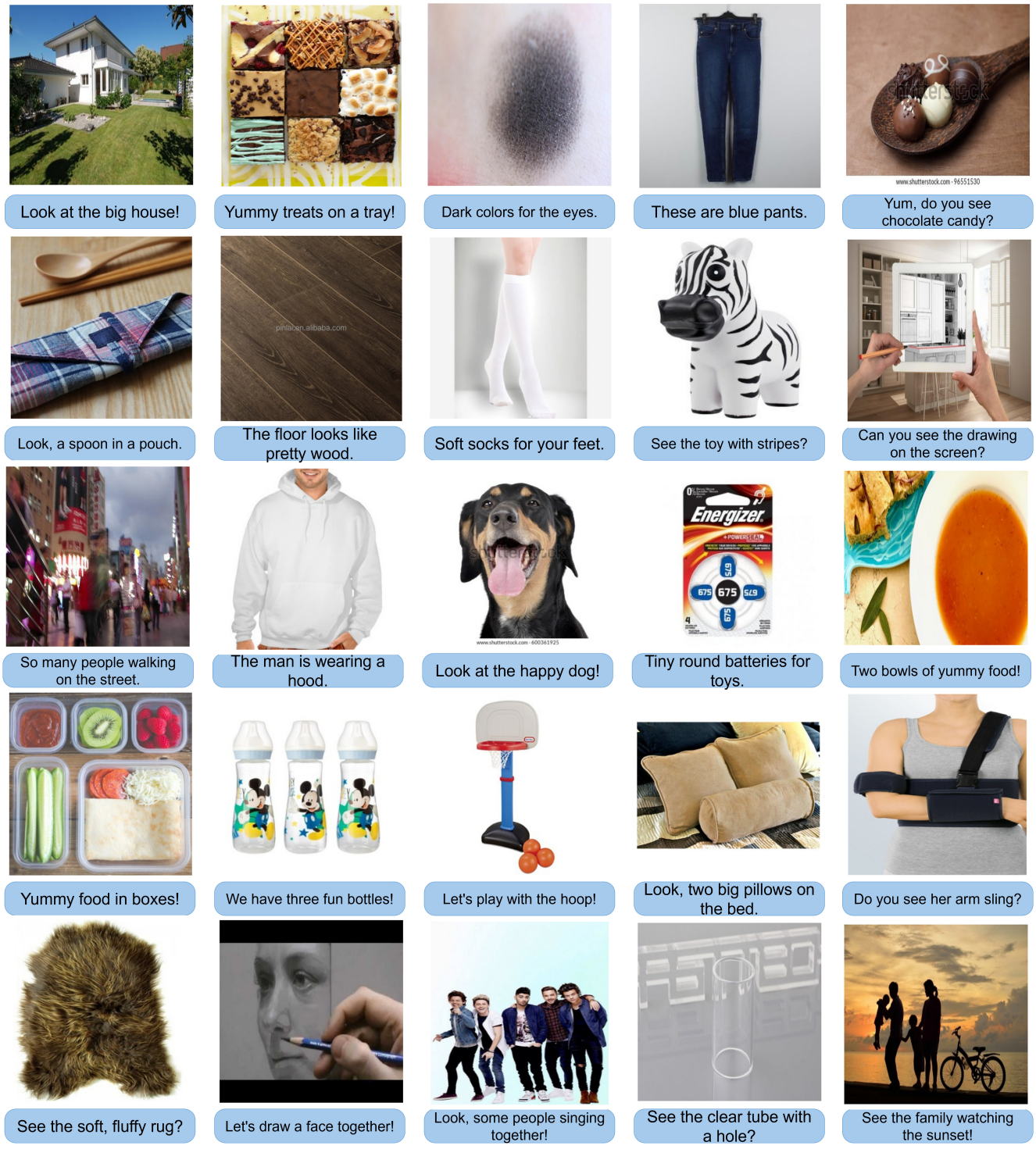

Figure 6. Examples of the transferred dataset

Implementation Details for Creating the Visual Two-Word Test. We construct VTWT by sub-sampling the SAYCam test split and using GPT-4o to generate 5,117 candidate two-word phrases through structured prompts. These prompts incorporate few-shot examples and Chain-of-Thought (CoT) guidance to enhance phrase quality. Specifically, we first prompt GPT-4o to generate a detailed image description based on the input image and utterance. Using this description, the model then generates a pair of two-word phrases—one positive and one negative—that differ in noun, verb, or adjective, ensuring clear semantic distinctions. The detailed prompt, along with few-shot examples, is shown in Figure 8.
To ensure the quality of the test samples, each sample was manually reviewed by two expert annotators with experience in vision and language research to verify that: (1) the caption is correctly describing the image in detail, (2) the positive phrase is concretely depicted in the image, (3) the negative phrase is not depicted in the image, and (4) both the positive and negative phrases are linguistically plausible. After this review, 967 high-quality test samples remain in the benchmark. Examples of VTWT test samples are shown in Figure 7 below.

Figure 7. Examples of VTWT Task

messages = [{"role":"system", "content": f""" You are provided with an image taken from an infant's perspective, along with an utterance directed to the child during the scene. Follow these two steps:
1. **Generate a Caption:** Create a detailed description of the image, capturing the main objects, actions, and context. This caption should reflect what a 2-year-old child might perceive in the scene.
2. **Generate Two-Word Phrases:** Based on the generated caption, the provided utterance, and direct reference to the image, produce a sets of two-word phrases that a 2-year-old child might say when viewing the scene. Each set should include:
- A "positive_phrase" that accurately corresponds to the image.
- A "negative_phrase" that is contextually plausible but introduces a subtle difference or contradiction.
The phrases don't have to directly describe the image; as long as they're somewhat related to the scene, they are acceptable. Pay attention to the given utterance, as it may suggest the keys or what's happening in the scene. When generating the phrases, be creative and consider:
- What activities could occur in this scene?
- What actions could be performed with the main object?
- What stands out or is special about the scene?
Ensure that the negative phrases remain contextually relevant by adhering to one of the following constraints:
- Describing the same object with a different attribute or action.
- Describing the same action with a different object.
- Presenting opposites of the same aspect (e.g., inside vs. outside, on vs. off, open vs. closed, etc.).
- Substituting a related object or action that is plausible within the scene context.
Ensure all phrases are grammatically correct and semantically plausible, reflecting typical two-word combinations used by 2-year-old children. They must capture certain content of the image so that one cannot distinguish them without looking at the image. Format each set as a JSON object with the key's "caption", "positive_phrase" and "negative_phrase".
Example 1:
Image:
[image]
Utterance:
want me to draw a picture ?
Output:
{
"caption": "An image of a pile of crayons on the ground. There are several crayons in the pile, with various colors and sizes. The crayons are scattered around, with some on top of each other and others next to each other. The scene appears to be a playful and creative environment, likely in a child's room or play area.",
"positive_phrase": "draw ball",
"negative_phrase": "chase ball"
}
Example 2:
Image:
[image]
Utterance:
juice ?
Output:
{
"caption": "An image capturing a scene with a dining table and various objects. On the table, there is a bowl, a cup, a spoon, and a bottle. The cup is placed next to the bowl, and the spoon is resting on the table. The bottle is located towards the left side of the table. The dining table occupies a significant portion of the image, extending from the left to the right side.",
"positive_phrase": "drink cup",
"negative_phrase": "eat cup"
}Now, based on the following image and utterance, generate a similar set of caption and two-word phrases: """ }
]

Figure 8. Full prompt for VTWT

Implementation Details for Creating Baby Winoground. We construct Baby Winoground using the 967 test samples from the Visual Two-Word Test (VTWT). Our goal is to modify the original image from VTWT such that the modified image is associated exclusively with the negative phrase while preserving most of the original content. To achieve this, we leverage the *search and replace* functionality of Stability AI's Stable Image Ultra model [3] as our image-editing tool. This process requires two prompts:

- Search Prompt: Describes the object, subject, or scene to be replaced in the image.
- Replace Prompt: Specifies the new object, subject, or scene replacing the original.

A direct approach would be to use the positive and negative phrases as search and replace prompts, respectively. However, the two-word constraint often omits crucial details, making it difficult for the image-editing model to generate accurate edits. To address this, we prompt GPT-4o to dynamically generate more descriptive search and replace prompts. We provide few-shot examples and specify key characteristics empirically found to improve edit quality; the full prompt is shown in Figure 9. As in VTWT, expert annotators manually review all test samples, ensuring that the edited images align exclusively with the negative phrases. After filtering, 365 high-quality test samples remain. Examples are shown in Figure 10.

messages = [{"role":"system", "content": f""" You are an expert in evaluating image-caption pairs which will be fed into a diffusion model to generate new synthetic data. Your task is defined as follows:
You will be given an image, a two-word positive caption, and a two-word negative caption. The positive caption is associated with the provided image and either describes something directly present in the scene or something plausible. The negative caption describes something which is not present or directly associated with the provided image. Your goal is to edit the image so that the negative caption is concretely and accurately depicted in the image. To achieve this, you will be writing prompts for a diffusion model which can edit images using a search and replace function. The search and replace function requires you to generate two prompts:
Search prompt - This describes the object, scene, or subject you want to replace in the generated image.
Replace prompt - This specifies the replacement object, scene, or subject.
When generating the search prompt and replace prompts, you should adhere to the following guidelines:
- Use 2-6 simple, concise, and descriptive words that accurately describe what to search for in the image and what to replace it with.
- Include additional details in the search prompt to help localize the subject being described such as color, shape, and location.
- Ensure that the replace prompt completely describes and contextualizes what the subject of the search prompt should be replaced with.
Format your output as a JSON object with the keys "search_prompt" and "replace_prompt".
Use the following examples as reference:
Example 1:

Positive Caption: kitty book
Negative Caption: doggy book
Expected Response: { "search_prompt": "book with cat on cover", "replace_prompt": "book with dog on cover" }
Example 2:

Positive Caption: happy baby
Negative Caption: sad robot
Expected Response: { "search_prompt": ”happy baby", "replace_prompt": ”sad robot" }
Now, based on the following image, positive caption, and negative caption, respond accordingly: """ }
]

Figure 9. Full prompt for Baby Winoground. We uses few-shot examples to generate search and replace prompts for the image-editing model.

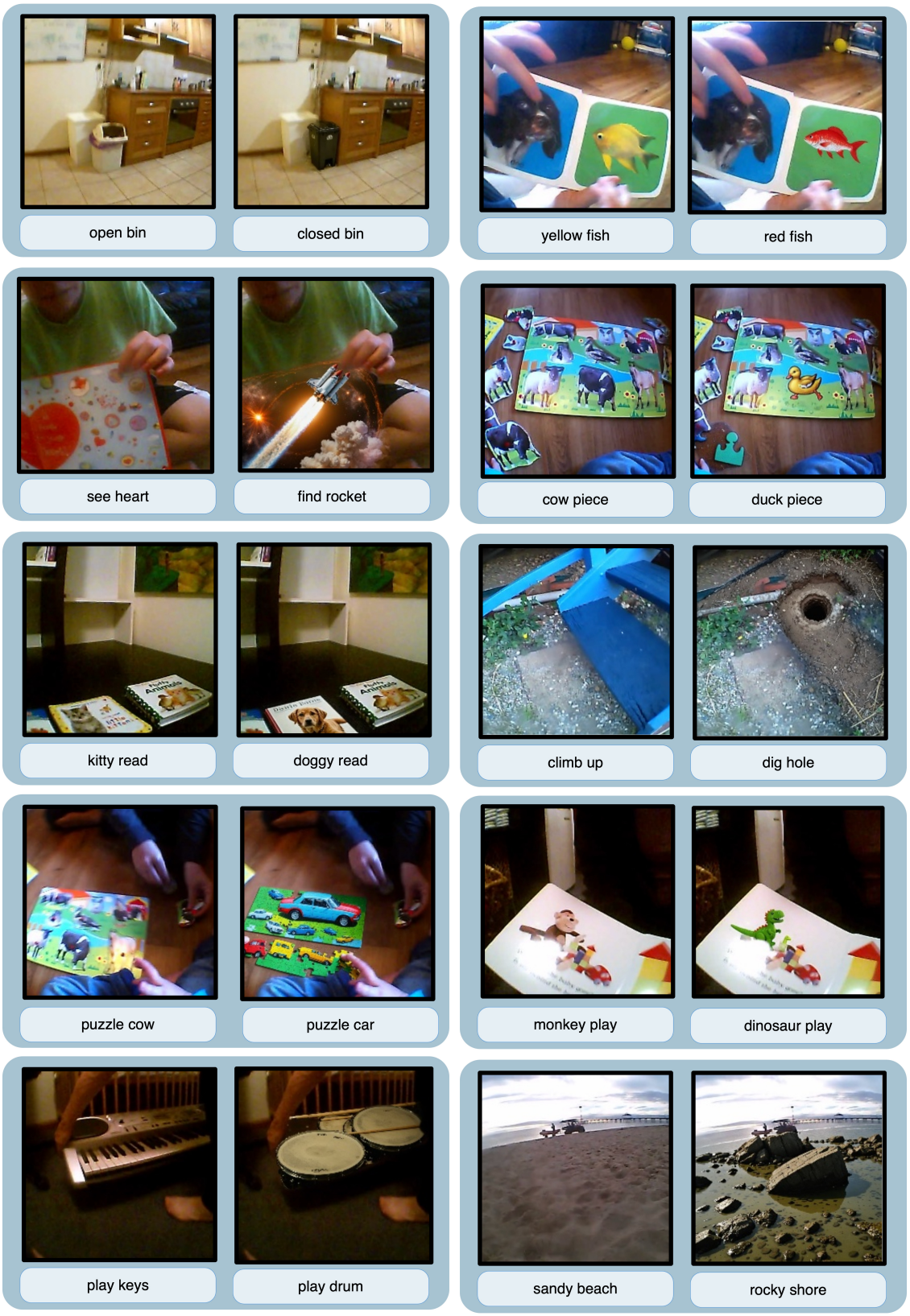

Figure 10. Examples of Baby Winoground Task

Examples of SAYCam Caption. The SAYCam Caption task consists of 294 test samples in total. We provide some examples below.

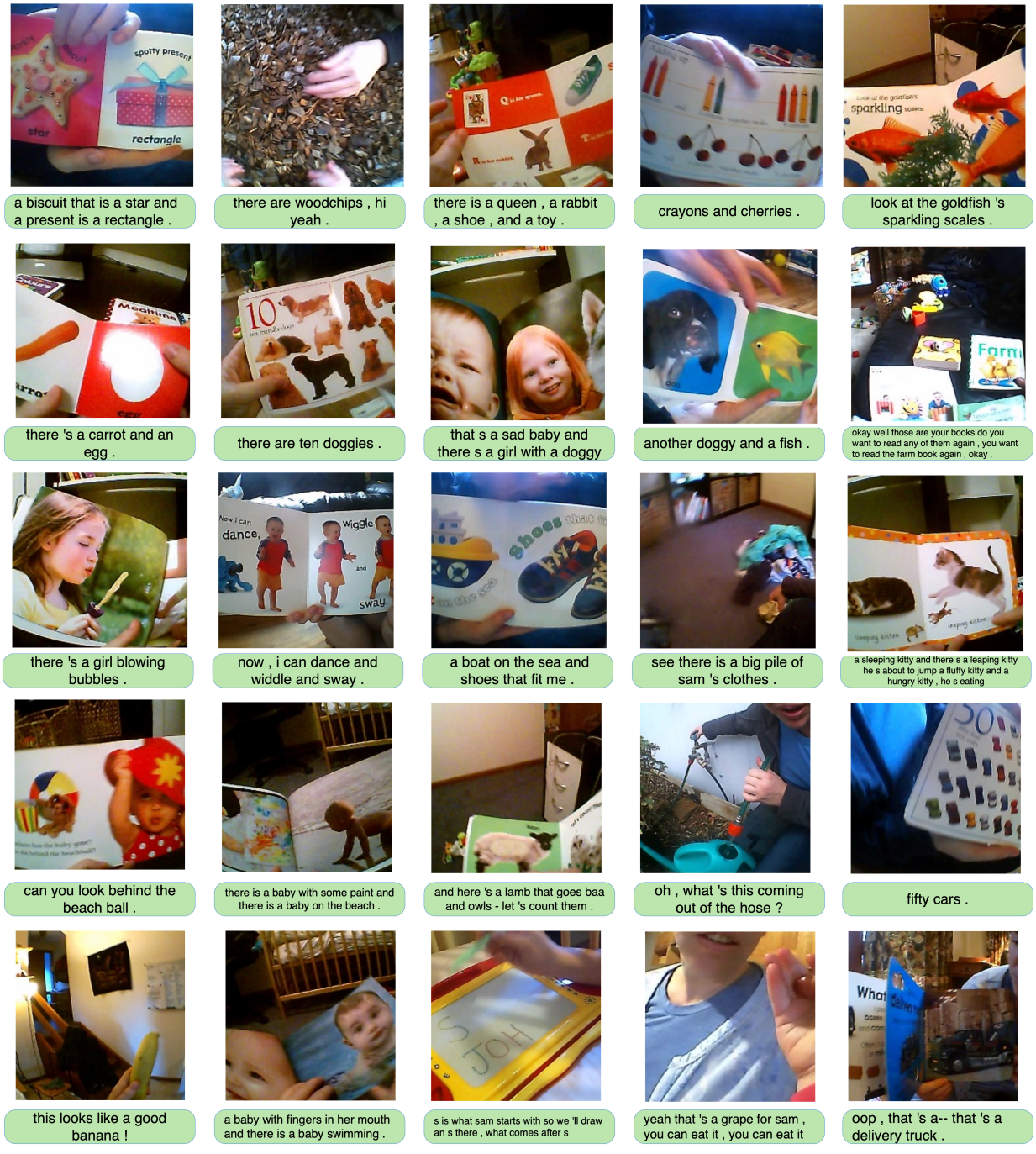

Figure 11. Examples of SAYCam Caption task

BabyLLaVA Training and Evaluation Details. BabyLLaVA follows the architecture and training strategy of LLaVA [5, 6], consisting of a language backbone, a vision backbone, and a two-layer MLP connector.
For the language backbone, we train a small GPT-2 model with 7.18M parameters from scratch using the language portion of our training corpus. The vision backbone is directly adopted from Orhan et al. [10] and is based on a ResNeXt-50 [15] model with 23 million parameters, trained from scratch using DINOv2 [8] on all SAYCam video clips, including those without utterance transcriptions. These clips are subsampled into 9 million frames at 5 FPS. The connector is a simple two-layer MLP, identical to LLaVA-v1.5.
Our training framework closely follows LLaVA but introduces Stage 0, an additional unimodal pretraining stage for the language and vision backbones. Unlike LLaVA, which initializes from pretrained CLIP and Vicuna v1.5 [16], BabyLLaVA requires this extra stage since both backbones are trained from scratch. The full training process consists of the following three stages. All the stages can be finished within 2 hours on four A6000 GPUs.

- **Stage 0: Unimodal Pretraining.** The language backbone is trained independently on textual data, while the vision backbone remains unchanged, as we adopt the pretrained backbone directly from Orhan et al.
- **Stage 1: Feature Alignment.** Both backbones are frozen, and only the MLP connector is trained to align vision and language features.
- **Stage 2: End-to-End Training.** The vision backbone remains frozen while the connector and language backbone are trained jointly. We also experiment with different freezing strategies (freezing only the vision backbone, only the language backbone, or neither) and find that freezing only the vision backbone yields the best overall performance.

For evaluation, we observe that the choice of input prompt significantly impacts performance, a phenomenon noted in prior research [1, 7]. To investigate this effect, we test various prompts, including common patterns of child-directed utterances (e.g., “Look at” or “What’s that”), as well as the absence of a prompt. Interestingly, omitting the input prompt yields the best results, likely because it aligns with the model’s training setup, which does not incorporate fixed prompts.

CVCL Training and Evaluation Details. We train and evaluate two variants of the CVCL model from [14] (CVCL-filtered-aug & CVCL-filtered-random). CVCL-filtered-aug is trained on our filtered SAYCam dataset and our transferred dataset (section **??**), both of which contain approximately 67k image-caption pairs. Similarly, CVCL-filtered-random is trained on our filtered SAYCam dataset plus a randomly sampled unprocessed subset of approximately 67k image-caption pairs from LLaVA’s pretraining dataset [6]. We train both variants on a single A100 GPU for 12 hours each using the default hyperparameters specified in the supplemental info of [14]. For evaluation, we use the model checkpoint from the last training epoch for each variant.

Out-of-Domain Generalization. A primary aim of our approach is to ensure baby models align with the cognitive and linguistic limitations of early-stage learners. To empirically validate this property, we explicitly assess baby models on tasks that exceed typical infant-level developmental capacities, such as advanced visual reasoning (Winoground) and general-purpose tasks (VQA and BLiMP). As shown in Table 7, baby models (e.g., BabyLLaVA, CVCL) perform significantly below upper-bound models, affirming their constrained generalization capabilities. This limitation ensures developmental authenticity, preventing baby models from inadvertently solving complex tasks beyond their intended cognitive stage.

Interestingly, we find that the performance gap between BabyLLaVA and the larger LLaVA-v1.5-7B model is significantly greater on these complex, out-of-domain tasks compared to simpler, in-domain tasks such as VTWT (Table **??**). This indicates that observed differences in performance cannot be attributed solely to differences in model capacity (i.e., parameter count), but also arise from the complexity and alignment of tasks and datasets with the developmental stage being modeled. Thus, baby models' constraints are multidimensional, encompassing not only architectural limitations but also deliberate choices in task design and dataset construction.

Category	**Model**	**BLiMP$_{\text{filtered}}$**	**BLiMP$_{\text{supplement}}$**	**Winoground**	**VQA**	**DevBench**
Upper Bound Models	LLaVA-v1.5-7B	0.7299	0.8300	0.6327	0.6273	0.8570
	LLaVA-v1.5-7B-ft	0.7205	0.8032	0.5992	0.4941	0.6300
	CLIP-large	N/A	N/A	0.5638	0.2397	0.7172
Baby Models	BabyLLaVA-Llama	0.6772	0.5903	0.5214	0.2312	0.3907
	CVCL	N/A	N/A	0.5221	0.1600	0.3993
Random Guess	-	0.5000	0.5000	0.5000	0.1250	0.3750

Table 7. Evaluation results on out-of-domain benchmarks. For BLiMP, Winoground and VQA, please refer to [2] for implementation details. For DevBench, we report the average score of TROG, WG, LWL and VV.

Out-of-Domain Tasks Ablation Study. We also evaluate both our CVCL and BabyLLaVA model variants on several out-of-domain benchmarks, including general purpose benchmarks like VQA, and developmental benchmarks such as DevBench. We make several observations. First, we see that all model variants perform around random chance on Winoground, indicating that none of the models achieve robust compositional reasoning ability. For VQA and DevBench, however, both CVCL and BabyLLaVA variants trained on our transferred dataset (CVCL-filtered-aug & BabyLLaVA-filtered-aug) achieve superior performance, reinforcing the value of our developmentally adapted general-domain data. In addition, even the weakest BabyLLaVA model outperform all the CVCL variants on VQA, indicating the advanced reasoning ability of generative VLMs over discriminative VLMs. Finally, the modest performance across all model variants and tasks reinforces one of our main results that appropriate developmental modeling naturally constrains generalization.

Model	BLiMP$_{\text{filtered}}$	BLiMP$_{\text{supplement}}$	Winoground	VQA	DevBench
CVCL-filtered	N/A	N/A	0.5221	0.1600	0.3993
CVCL-filtered-aug	N/A	N/A	0.4714	0.1641	0.6086
CVCL-filtered-random	N/A	N/A	0.4935	0.1173	0.5198
BabyLLaVA-filtered	0.6772	0.5903	0.5214	0.2312	0.3907
BabyLLaVA-filtered-aug	0.6646	0.5061	0.5455	0.4064	0.5303
BabyLLaVA-filtered-random	0.6746	0.4778	0.5335	0.3659	0.4722

Table 8. Ablation study results on out-of-domain benchmarks. For BLiMP, Winoground and VQA, please refer to [2] for implementation details and metrics. For DevBench, we report the average score of TROG, WG, LWL and VV.

Transferred Data Efficiency Ablation Study. To further investigate the extra data efficiency brought by our introduced transferred dataset, we subsample *filtered-aug* and *filtered-random* datasets by different fractions, then perform model training with identical training steps, and test on our in-domain benchmarks. Figure 12 shows the result. The *filtered-aug* curve rises more steeply, and using only 25–50% of *filtered-aug* already equals the full *filtered-random* on in-domain tasks, confirming substantial sample-efficiency gains.

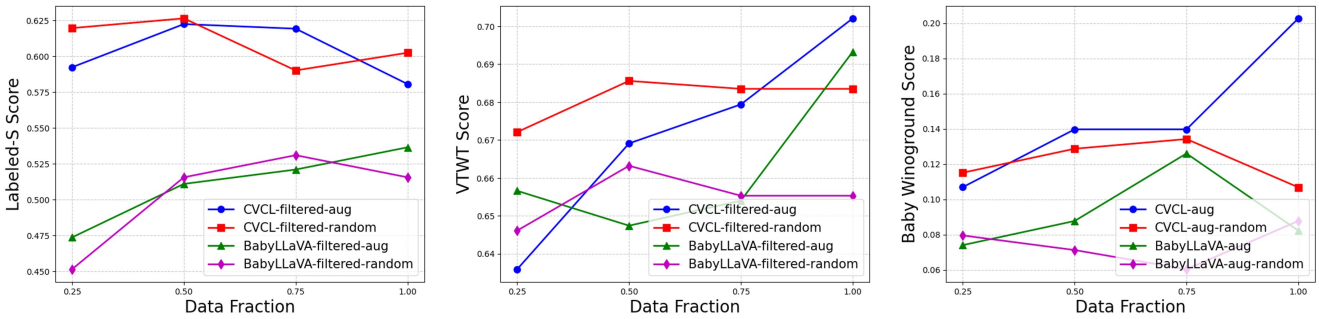

Figure 12. Performance on different fractions of datasets.

References

[1] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 10

[2] Leshem Choshen, Ryan Cotterell, Michael Y Hu, Tal Linzen, Aaron Mueller, Candace Ross, Alex Warstadt, Ethan Wilcox, Adina Williams, and Chengxu Zhuang. [call for papers] the 2nd babylm challenge: Sample-efficient pretraining on a developmentally plausible corpus. *arXiv preprint arXiv:2404.06214*, 2024. 11

[3] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024. 6

[4] Roy Jonker and Ton Volgenant. A shortest augmenting path algorithm for dense and sparse linear assignment problems. In *DGOR/NSOR: Papers of the 16th Annual Meeting of DGOR in Cooperation with NSOR/Vorträge der 16. Jahrestagung der DGOR zusammen mit der NSOR*, pages 622–622. Springer, 1988. 2

[5] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024. 10

[6] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024. 2, 10

[7] Vivian Liu and Lydia B Chilton. Design guidelines for prompt engineering text-to-image generative models. In *Proceedings of the 2022 CHI conference on human factors in computing systems*, pages 1–23, 2022. 10

[8] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 10

[9] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24, 2011. 2

[10] A Emin Orhan and Brenden M Lake. Learning high-level visual representations from a child's perspective without strong inductive biases. *Nature Machine Intelligence*, 6(3):271–283, 2024. 10

[11] 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, 2021. 2

[12] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022. 2

[13] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of ACL*, 2018. 2

[14] 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*, 383(6682):504–511, 2024. 10

[15] 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. 10

[16] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36: 46595–46623, 2023. 10