

# *Mull*-Tokens: Modality-Agnostic Latent Thinking

## Supplementary Material

### 1. Appendix

In this supplementary document, we include further ablations in the training design choices, more details, and more insights into the shortcomings of related existing work versus our approach using qualitative examples. Finally, we also provide some qualitative examples to demonstrate our insights.

#### 1.1. Training Details

We train using Deepspeed [5] stage 2 on 8 H100 GPUs. For the fine-tuned versions, we use a batch size of 1. We use a max gradient norm of 5 and use float bf16 precision for all our experiments, with a learning rate of  $1e-6$ . We freeze the vision encoder since we see minimal differences with training the visual encoder in our early experiments. All other parts of the model are fully fine-tuned.

The GRPO variants are trained using Deepspeed stage 3 to accommodate the significant increase in memory requirements. For the GRPO we use a reference model which is a frozen version of the same model. We use a group size of 2 (we run out of memory with a higher group size). We use a gradient accumulation size of 32 with 8 GPUs (data parallel). We use a beta (weighing the KL divergence) of 0.04. These hyperparameters follow the standard settings in the Video-R1 codebase [2]. We do GRPO for 200 steps - hence, 50K iterations ( $200 \times 32 \times 8$ ).

**Data mix used for each stage.** For stage 1, our goal is to primarily pretrain the *Mull*-Tokens to hold intermediate reasoning information. Hence, we primarily use the Video-R1 and Zebra-COT datasets which include reasoning traces. Specifically, we use a mixture of 40% Video-R1, 40% Zebra-COT, and 20% SAT for 200K iterations. In Stage 2 and 3, we now want the model to optimize the reasoning chains to data primarily not seen in the pretraining SFT mixture. Hence, we use a higher proportion of SAT and still include some pretraining data to prevent domain forgetting following [6]. Specifically, we use a mixture of 60% SAT, and 20% each from Video-R1 and Zebra-COT for another 200K iterations.

For the baselines, we maintain the number of iteration steps to ensure fair comparison. For DirAns FT baseline - we also first train for 192K iterations (24K steps with data parallel on 8 GPUs) on 40% Video-R1, 40% Zebra-COT, and 20% SAT and then on 60% SAT, and 20% each from Video-R1 and Zebra-COT for another 192K iterations. We also tried training all in one stage for longer using 60% SAT, and 20% each from Video-R1 and Zebra-COT for 384K iterations and found the former to be a stronger baseline. For

the TextCOT baseline, we first train for 192K iterations on 40% Video-R1 with text reasoning traces, 40% Zebra-COT answers only, and 20% SAT and then do GRPO for 50K steps on 60% SAT, and 20% each from Video-R1 (answers only) and Zebra-COT (answers only). We also tried the baseline of only first training on the Video-R1 since it is the only dataset with text reasoning traces before doing GRPO. However, we found the base model trained first with the entire mix to be a stronger starting point (higher benchmark numbers) for the GRPO.

#### Prompting Strategies & Evaluation Settings

To strictly evaluate the reasoning capabilities of different model configurations, we employ three distinct prompting strategies. The specific template chosen determines whether the model relies on explicit textual generation, latent computation, or direct pattern matching. Additionally, a task-specific constraint (e.g., "Please provide only the single option letter" or "Please provide a numeric answer") is appended to all templates based on the question type.

##### 1. Text-Reasoning Baseline (Video-R1).

To reproduce text-based reasoning baselines, we utilize the template established in prior work [2, 4, 7].

```
{Question}
Please think about this question as
if you were a human pondering deeply.
Engage in an internal dialogue using
expressions such as 'let me think',
'wait', 'Hmm', 'oh, I see', 'let's break
it down', etc, or other natural language
thought expressions. It's encouraged to
include self-reflection or verification
in the reasoning process. Provide your
detailed reasoning between the <think>
</think> tags, and then give your final
answer between the <answer> </answer>
tags.
```

##### 2. Image-text interleaved and *Mull*-Tokens (Ours).

For models based on our proposed *Mull*-Tokens, we employ a similar template. Since we do not explicitly verbalize the thoughts, we avoid phrases that encourage expressions like "Hmm", "Oh, I see" etc.

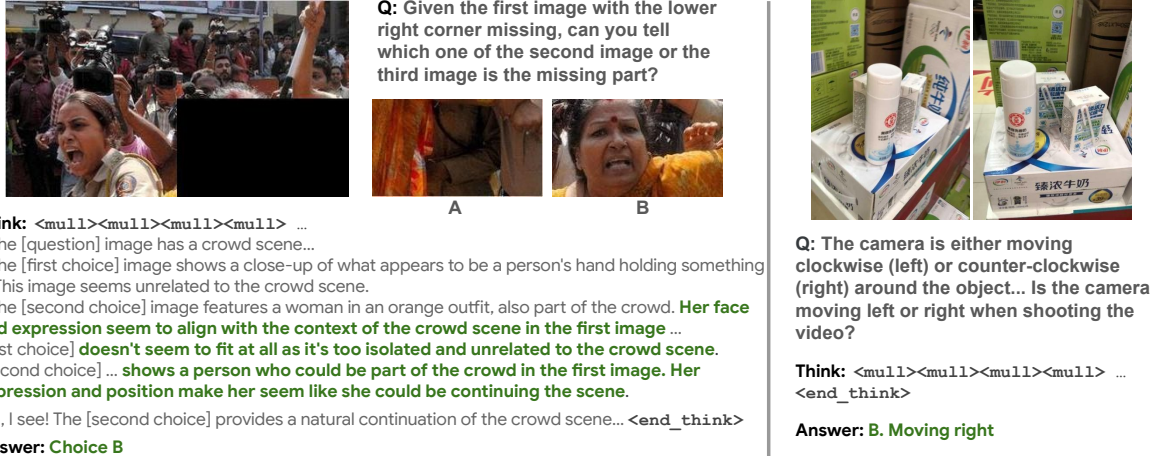


Figure 1. *Mull*-Tokens tokens can be used in conjunction with text reasoning and the model can effectively decide when to also avoid using text reasoning depending on the task. For the example on the left, the model accurately uses *Mull*-Tokens along with some textual descriptive cues to reason about the missing piece. For the example on the right, the model decides to simply use the *Mull*-Tokens to directly predict the answer since textual descriptions are likely not helpful to detect camera motion.

Table 1. Performance with different prompt strategies interleaving image and text thoughts. Explicitly instructing the model to think with images hurts performance compared to letting the model choose.

| Model                        | BLINK |        |       |        |      |      |        | SAT-R | VSI  |        | ERQA | All Avg |
|------------------------------|-------|--------|-------|--------|------|------|--------|-------|------|--------|------|---------|
|                              | MV    | RelDep | SpRel | Jigsaw | IQT  | Avg  | Reason |       | Avg  | Reason |      |         |
| Qwen 2.5 VL 7B               | 39.0  | 61.3   | 92.4  | 58.7   | 25.3 | 55.3 | 41.0   | 59.0  | 24.0 | 23.0   | 38.9 | 44.3    |
| + Img-Txt (free-form)        | 57.1  | 69.4   | 75.5  | 68.7   | 25.3 | 59.2 | 50.4   | 74.0  | 30.3 | 33.0   | 38.5 | 50.5    |
| + Img-Txt (force im thought) | 49.6  | 73.4   | 73.4  | 69.3   | 24.7 | 58.1 | 47.9   | 65.0  | 27.2 | 30.7   | 39.0 | 47.3    |
| + <i>Mull</i> -Tokens (our)  | 63.1  | 83.1   | 81.8  | 74.0   | 32.0 | 66.8 | 56.4   | 77.7  | 33.0 | 32.9   | 38.2 | 53.9    |

```
{Question}
Please think about this question
deeply. It's encouraged to include
self-reflection or verification in the
reasoning process. Provide your final
answer between the <answer> </answer>
tags.
```

For image-text interleaved thoughts, we use Zebra-CoT [3] dataset. Each image is first encoded by the Qwen-VL [1] encoder, then all tokens are average-pooled into a single image feature vector. While allocating more latents to each image can likely help at the cost of higher compute, we leave such an ablation to future work. We use a discrete token mapped to the image feature, rather than using continuous embeddings for two reasons. First, we observed a slight degradation in performance. Second, continuous embeddings are slower due to recurrence loops during inference and training. During training, we use an explicit token <im start> followed by the <latent> token. The output of the <latent> is supervised to be the image feature.

For the *Mull*-Tokens approach, we simply append  $k$

<*Mull*> after the question prompt, providing the model with additional tokens as a multimodal thinking scratchpad.

### 3. Direct Answering (SFT & Base Model).

Finally, to evaluate the base model (zero-shot) and the standard Supervised Fine-Tuning (SFT) baseline, we use a *Direct* template. This is the most barebones setting to assess the model's ability to map inputs directly to answers without intermediate computation steps.

```
{Question}
Provide your final answer between the
<answer> </answer> tags.
```

#### Task-Specific Output Constraints.

To ensure the generated answers can be reliably parsed and evaluated against ground-truth, we append a format constraint to every prompt during evaluation. These constraints vary by task definition (e.g., Multiple Choice vs. Regression) and dictate the expected content within the <answer> tags. The specific suffixes used are:

- **Multiple Choice:** *"Please provide only the single option letter (e.g., A,*

*B, C, D, etc.) within the <answer> </answer> tags."*

- **Numerical / Regression:** *"Please provide the numerical value (e.g., 42 or 3.14) within the <answer> </answer> tags."*
- **OCR:** *"Please transcribe text from the image/video clearly and provide your text answer within the <answer> </answer> tags."*
- **Free-form:** *"Please provide your text answer within the <answer> </answer> tags."*

### Fuzzy Logic Used to Extract Answers

To ensure robust evaluation, particularly when models produce verbose CoT outputs, we implement a fuzzy matching strategy to parse the final prediction as follows:

- **Pattern Extraction:** We first attempt to extract the answer by matching the prediction against a prioritized list of regular expression patterns. These include standard conversational markers and our enforced XML tags:
  - `r"<answer>\s*(. *?)\s*</answer>"`
  - `r"(?:the answer is|the correct answer is|final answer:)\s+([a-zA-Z0-9]+)"`
  - `r"I counted a total of\s+(\d+)"`
  - `r"count is\s+(\d+)"`
- **Validation & Numerical Extraction:** If a match is found, we check if the extracted content is a valid multiple-choice option (single letter A-Z). If the extracted text is not a letter, we attempt to parse it as a numerical value (integer or float) for regression or counting tasks.
- **Fallback Strategy:** In cases where the extraction fails or yields no numerical/categorical value, we perform a final fallback search for the last isolated capital letter (e.g., `r"\b[A-Z]\b"`) appearing in the raw text, assuming it represents the final selection.

Empirically, we observe that the models usually adhere to the formatting instructions, with the vast majority of valid answers being successfully parsed directly from the `<answer>` tags via the first pattern.

## 1.2. Further Analysis and Ablations

In this section, we examine the shortcomings of modality switching, contrasting the inductive biases of interleaved image-text generation against our proposed joint reasoning approach. We identify specific issues, including ungrounded modality transitions and suboptimal scheduling, which hinder the performance of explicit switching baselines. Finally, we offer a preliminary analysis of the latent dynamics within our *Null-Tokens*.

## The failure points with interleaving image-text

To investigate the feasibility of explicit modality switching for visual reasoning, we conducted a pilot study using a baseline model trained to interleave textual reasoning traces with discrete image latent tokens. We hypothesize that distinct processing steps for visual and textual thinking should allow for better interpretability and performance. However, our experiments reveal critical limitations in this paradigm. **Bias Towards Textual Reasoning.** We observed a strong inductive bias in the model to rely solely on textual reasoning. When trained simply to solve problems, the model rarely invoked image latents at test time, preferring to reason exclusively via text. To address this, we introduced specific system prompts during training: one explicitly instructing the model to “think only using text” when the dataset included text-based reasoning traces and another to “think using both text and images” for the image-text interleaved training data.

Now, at test-time, we tried two approaches - one where we let the model decide by simply including the question without instructions for how to reason, and one where we append “think using both text and images” after the question to *force* the model to think using images. The results are shown in Table 1.

We observe two issues during inference:

- **Bias towards text-only reasoning:** When given the autonomy to choose its reasoning modality (i.e., without a forcing prompt), the model mostly reverts to text-only reasoning, and rarely switches to image latent thinking.
- **Forcing image thoughts degrades performance:** When we explicitly forced the utilization of image latents via the prompt, we observed a degradation in downstream task performance compared to the text-only baseline, as shown in Table 1.

### Performance Degradation with Forcing image

**Thoughts.** Comparing the *Img-Txt (free-form)* baseline with the *Img-Txt (force im thought)* setting reveals that prompting the model to use image thoughts leads to a reduction in overall performance. The average score across all benchmarks drops from 50.5% to 47.3%. Despite the aggregate decline, the forced image thought setting yields marginal but distinct improvements on specific tasks that inherently require visual simulation or geometric imagination- such as Jigsaw (68.7  $\rightarrow$  69.3) and Relative Depth (RelDep) (69  $\rightarrow$  73) - tasks that may require mentally manipulating visual patches to solve spatial puzzles, or visualizing depth maps or 3D structures from the 2D RGB inputs. However, while specific cases may benefit, we see an overall decline in performance compared to letting the model decide. These results suggest that while explicit image thoughts intuitively should contain valuable signals, the mechanism of explicit switching may be too brittle for general reasoning.

**Qualitative Analysis of Switching Failures.** To investigate performance degradation during explicit switching, we also qualitatively analyzed the model’s generated traces. We identified two prominent patterns in the failure cases that we show in Figure 2:

1. **Ungrounded Modality Transitions:** As shown in the top row of Figure 2, even when the model successfully generates an image latent (intended to process visual cues like fitting in the puzzle piece), the subsequent textual reasoning is often not properly grounded in that latent. In this example, The model correctly identifies that it needs to analyze the image choice’s edges and hence switches to the image thoughts. However, the text following it is incorrect. This suggests that the image latent did not contain enough information for the text thought to be properly grounded.
2. **Suboptimal Switching Scheduling:** The bottom row of Figure 2 demonstrates that the model struggles to identify the optimal moment to switch modalities. In this example, the model should have switched to latent after identifying the need to examine the options.

These findings suggest that rigid, explicit switching between modalities may be difficult and require more data. This limitation motivates our proposed approach: *modality-agnostic thinking tokens* that can jointly reason in a continuous space without explicit switching.

This motivates our modality agnostic *Mull-Tokens*. *Mull-Tokens* offers a simpler alternative where we do not need to switch between modalities, but rather think using latents trained to hold both visual and textual information jointly. Our proposed *Mull-Tokens* approach achieves the highest performance overall (Avg 53.9%). We posit that by enabling joint reasoning in a shared space, *Mull-Tokens* capture the benefits of visual imagination (e.g., high performance on RelDep and Jigsaw) without suffering the grounding losses associated with explicit modality switching. However, more interpretability studies will need to be done as future work to rigorously validate the function of *Mull-Tokens*.

### A preliminary attempt to interpret latents

While our primary focus has been on performance, analyzing the semantic content of the  $\langle Mull \rangle$  latents presents a compelling avenue for future work. We conducted a preliminary analysis of the Euclidean distance between subsequent latents to see if latents encode different meanings. For *Mull-Tokens* without a warm-up of image-text traces, the Euclidean distance between latents tapers off after a few steps before it converges, suggesting they become similar and add no new information. Our image-text warmed up latents maintains distances between steps. This indicates that the model avoids representational collapse and that each latent continues to contribute information (albeit useful or

not). However, more analysis needs to be done by decoding or mapping the latents to actual reasoning images and closely controlled counterfactual images to understand and interpret the “meaning” of the *Mull-Tokens*. With the absence of such data, we leave this analysis to future work. Our focus in the paper is to provide a more accurate, simple, and fast alternative to text reasoning or explicit image thoughts that require 100’s of tokens as reasoning before arriving at the answer.

## References

- [1] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023. 2
- [2] Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Benyou Wang, and Xiangyu Yue. Video-R1: Reinforcing video reasoning in MLLMs. *arXiv preprint arXiv:2503.21776*, 2025. 1
- [3] Ang Li, Charles Wang, Deqing Fu, Kaiyu Yue, Zikui Cai, Wang Bill Zhu, Ollie Liu, Peng Guo, Willie Neiswanger, Furong Huang, et al. Zebra-cot: A dataset for interleaved vision language reasoning. *arXiv preprint arXiv:2507.16746*, 2025. 2
- [4] Mi Luo, Zihui Xue, Alex Dimakis, and Kristen Grauman. When thinking drifts: Evidential grounding for robust video reasoning, 2025. 1
- [5] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, page 3505–3506, New York, NY, USA, 2020. Association for Computing Machinery. 1
- [6] Arijit Ray, Jiafei Duan, Reuben Tan, Dina Bashkurova, Rose Hendrix, Kiana Ehsani, Aniruddha Kembhavi, Bryan A Plummer, Ranjay Krishna, Kuo-Hao Zeng, et al. Sat: Spatial aptitude training for multimodal language models. *arXiv preprint arXiv:2412.07755*, 2024. 1
- [7] Zeyuan Yang, Xueyang Yu, Delin Chen, Maohao Shen, and Chuang Gan. Machine mental imagery: Empower multimodal reasoning with latent visual tokens, 2025. 1

## Text is not grounded after switching to image latent



[Jigsaw puzzle: Which option fits the black patch?]  
**GT Correct Ans: A**

THOUGHT 0: The problem asks to determine which of the second or third image is most suitable to fill the missing lower right corner of the first image. To decide, let's examine the content and edges of all three images. ✓

THOUGHT 1: The first image is incomplete in the bottom right, indicated by a black box. We need to find the other two images' parts that could logically fit there. Let's analyze the second image's edges. ✓

<IMLATENT>

*This text following the image latent is not correct.*

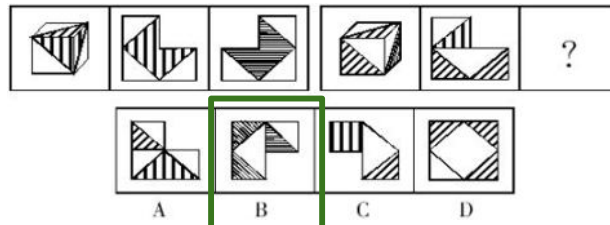
THOUGHT 2: The second image appears to show a continuation of the scene with signs and people, but its bottom edge does not align with the missing area in the first image. Let's look at the third image instead. ✗

<IMLATENT>

THOUGHT 3: The third image shows a close-up of a sign and part of a person, including an arm and hand. Comparing its edges to the missing part in the first image, it seems like this section could seamlessly fit into the gap, as it contains elements (the arm, hand, and sign) that are also present in the larger scene of the first image. ✗

...  
 <answer>B</answer>

## Incorrect time of switching to image latents



[IQ Test: Which option fits the "?" ?]  
**GT Correct Ans: B**

...  
 THOUGHT 1: Let's break down the pattern in the first sequence. Comparing the first two images, the shaded triangle moves position. The third image shows a rotation or flip of the triangle compared to the second. This suggests a sequential transformation.

<IMLATENT>

THOUGHT 2: For the second sequence, the cube and the triangle it contains undergo a similar transformation. The triangle's position and orientation change systematically from the first to the second image, and presumably, the third image continues this pattern.

<IMLATENT>

THOUGHT 3: Based on the identified pattern, the missing image should show a cube with a triangle inside it, where the triangle is in a configuration that follows the rotation or flipping rule observed in the examples. **Let me examine the options.**

THOUGHT 4: Option A shows a cube with a triangle that fits the expected pattern of transformation. The triangle's position and the cube's orientation align with the established sequence. Therefore, option A is the most likely correct choice. ✗

<answer>A</answer>

*Should have switched to image latents here instead of before. Text reason is wrong after*

Figure 2. Some qualitative examples of failures we observe with trying the existing approach of using explicit switching between image and text thoughts, motivating the need for our modality-agnostic *Mull-Tokens*. Upper half: the image thought may not be enough for the text following it to be accurately grounded. The model correctly identifies that it needs to think in the image latent, but the text after is inaccurate. With our *Mull-Tokens*, the model can free-form decide how many latents to allocate to the image since it doesn't need to explicitly switch back to text. Bottom half: Sometimes, the model cannot correctly identify when to switch to image latents. It switches at suboptimal times and doesn't switch to image latent even after identifying it needs to now compare the visual options (green bolded text). Our modality-agnostic *Mull-Tokens* removes the need to explicitly switch between modalities.