

Appendix

A Limitations

In this work, we pose a fundamental rethinking of vision-language reasoning, and introduce the concept of pixel-space reasoning. While we show the effectiveness of our approach to cultivating pixel-space reasoning, this improvement is still bottlenecked by limited data that spans across tasks and contents. In addition, we focus on two specific visual operations to handle primary media formats of images and videos. In the future, we endeavor to include more visual operations and examine the effectiveness of pixel-space reasoning on more diverse collections of tasks.

B Derivations of Curiosity-Driven Reward

The primary objective is to maximize the expected correctness outcome, formalized as a constrained optimization problem. Let $r(\mathbf{x}, \mathbf{y})$ be the original correctness reward for a query \mathbf{x} and response \mathbf{y} . The policy generating responses is denoted by $\pi_\theta(\mathbf{y}|\mathbf{x})$.

The optimization problem is:

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E} \left[r(\mathbf{x}, \mathbf{y}) \middle| \mathbf{x} \sim \mathcal{D}, \mathbf{y} \sim \pi_\theta(\mathbf{y}|\mathbf{x}) \right] & (6) \\ \text{subject to} \quad & C_1(\theta; \mathbf{x}) \equiv \text{RaPR}(\mathbf{x}) - H \geq 0 & (7) \\ & C_2(\mathbf{y}) \equiv N - \mathbf{n}_{\text{vo}}(\mathbf{y}) \geq 0 & (8) \end{aligned}$$

where:

- $\text{RaPR}(\mathbf{x}) \doteq \mathbb{E} \left[\mathbf{1}_{\text{PR}}(\mathbf{y}) \middle| \mathbf{y} \sim \pi_\theta(\mathbf{y}|\mathbf{x}) \right]$ is the Rate of Pixel-space Reasoning for query \mathbf{x} .
- H is a predefined minimum threshold for $\text{RaPR}(\mathbf{x})$.
- $\mathbf{n}_{\text{vo}}(\mathbf{y})$ is the number of visual operations in response \mathbf{y} .
- N is a predefined upper bound on $\mathbf{n}_{\text{vo}}(\mathbf{y})$.

Constraint (7) is an expectation-level constraint for a given query \mathbf{x} , while constraint (8) applies to each individual response \mathbf{y} .

To incorporate these constraints into the objective, a common technique is the method of Lagrangian Relaxation. For a maximization problem, this typically involves subtracting terms proportional to the constraint violations (when constraints are written as $g(x) \leq 0$) from the original objective function $r(\mathbf{x}, \mathbf{y})$. If we rewrite our constraints as $g_1(\theta; \mathbf{x}) \equiv H - \text{RaPR}(\mathbf{x}) \leq 0$ and $g_2(\mathbf{y}) \equiv \mathbf{n}_{\text{vo}}(\mathbf{y}) - N \leq 0$, the standard Lagrangian modification to the per-instance reward would be:

$$r_{\text{Lagrangian}}(\mathbf{x}, \mathbf{y}; \theta) = r(\mathbf{x}, \mathbf{y}) - \lambda_1(H - \text{RaPR}(\mathbf{x})) - \lambda_2(\mathbf{n}_{\text{vo}}(\mathbf{y}) - N) \quad (9)$$

where $\lambda_1, \lambda_2 \geq 0$ are Lagrange multipliers. The overall optimization objective would then be to maximize $\mathbb{E} [r_{\text{Lagrangian}}(\mathbf{x}, \mathbf{y}; \theta)]$ with respect to θ , and to minimize with respect to the multipliers.

However, directly applying this standard formulation has two problems. Firstly, this formulation has an over-satisfaction issue. The term $-\lambda_2(\mathbf{n}_{\text{vo}}(\mathbf{y}) - N)$ would provide a positive reward if $\mathbf{n}_{\text{vo}}(\mathbf{y}) < N$ (i.e., the constraint is "over-satisfied"), potentially encouraging the policy to use far fewer visual operations than necessary. Secondly, the term $-\lambda_1(H - \text{RaPR}(\mathbf{x}))$ operates on the expectation-level and does not properly reward individual responses $y \sim \pi_\theta$.

Therefore, we adopt the following modified reward function:

$$r'(\mathbf{x}, \mathbf{y}) = r(\mathbf{x}, \mathbf{y}) + \alpha \cdot \max(H - \text{RaPR}(\mathbf{x}), 0) \cdot \mathbf{1}_{\text{PR}}(\mathbf{y}) + \beta \cdot \min(N - \mathbf{n}_{\text{vo}}(\mathbf{y}), 0) \quad (10)$$

where $\alpha \geq 0, \beta \geq 0$ are fixed hyperparameters.

This formulation offers several benefits. Firstly, the clipping mechanism addresses the over-satisfaction issue while preserving equivalence to the original constrained objective [Wang et al., 2022]. The clipping ensures the penalties are active only when the respective constraints are violated, otherwise the penalties are zero, thus avoiding over-satisfaction.

Secondly, this structure allows α, β to be treated as fixed hyperparameters. In standard Lagrangian methods (Eq. 9), multipliers are often dynamically adjusted; for example, Karush-Kuhn-Tucker (KKT) conditions imply that multipliers for inactive constraints (those satisfied with slack) are zero. The clipping zeros out the penalties when constraints are satisfied, thereby obviating the need for dynamic adjustment of α, β based on constraint satisfaction levels.

In addition, the inclusion of the indicator $\mathbf{1}_{\text{PR}}(\mathbf{y})$ converts the query-level expectation constraint into a response-level reward. Intuitively, this term acts as a targeted incentive: it rewards the specific behavior of engaging in pixel-space reasoning precisely when the average rate of such reasoning is below the desired threshold. The multiplier $\alpha \geq 0$ scales this incentive. It provides an implicit penalty for missing out on the potential bonuses the policy could have earned by employing pixel-space reasoning.

C Data and Training Details

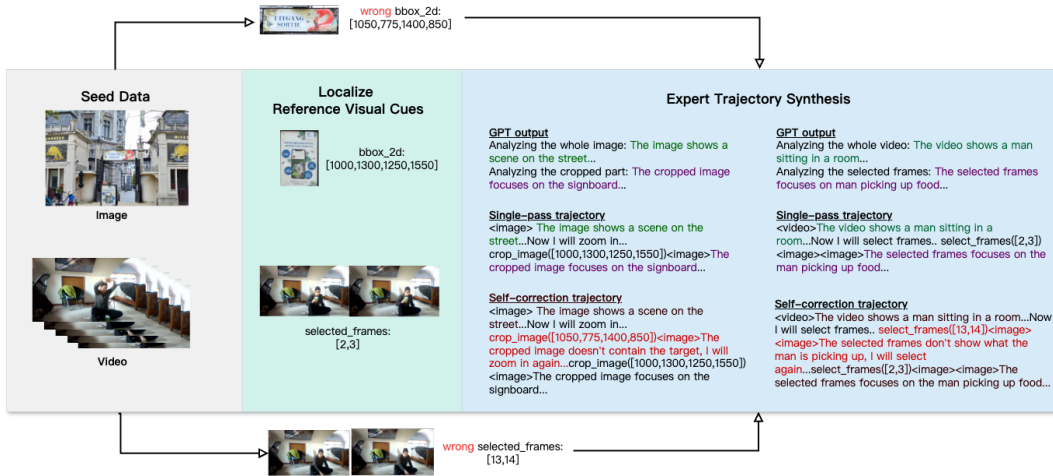


Figure 8: A detailed illustration of our data generation pipeline.

C.1 Protocols of Visual Operations

We include two primary visual operations: cropping an image and selecting frames from a video.

CropImage This operation allows the model to zoom in on a specific region of an image by providing a bounding box. The input includes a two-dimensional bounding box `bbox_2d`—a list of numeric coordinates $[x_1, y_1, x_2, y_2]$ constrained within the image dimensions—and a `target_image` index indicating which image to operate on (indexed from 1, where 1 refers to the original image). This operation helps the model focus on fine-grained details.

SelectFrames This operation enables the model to select a subset of frames from a video. The input `target_frames` is a list of integer indices specifying which frames to extract from a 16-frame sequence, with a limit of no more than 8 frames. This allows the model to focus on key temporal moments relevant to the query.

C.2 Instruction Tuning Data

Details of Seed Datasets We selected datasets based on two key attributes: high visual complexity requiring fine-grained analysis, and the presence of explicit annotations that can serve as targets or anchors for visual operations. Based on these criteria, our data sources include:

- **SA1B** [Kirillov et al., 2023]: A large-scale dataset of high-resolution natural scenes offering rich visual detail and complexity.

- **FineWeb** [Ma et al., 2024]: Consists of webpage screenshots paired with Question-Answering (QA) instances and precise bounding box annotations for answer regions, offering explicit spatial targets for visual analysis.
- **STARQA** [Wu et al., 2024]: Provides video data with QA pairs and annotated temporal windows indicating relevant visual contents for answers, offering both visual and temporal context for potential video-specific operations.

Detailed Data Pipeline Illustration. As the Fig. 8 depicts, after we obtain reference visual cues from seed data, we input both the whole HR image or video and the corresponding localized reference visual cues to gpt. Then we use template-based method to extract whole visual input analysis and local detailed analysis before we concatenate the whole analysis, localized reference visual cue and the partial analysis to form the single-pass trajectory. We utilize the reference visual cue to get the wrong visual cues to insert in the obtained single-pass trajectories to get self-correction trajectory.

Single-pass and Self-correction Data Synthesis Details

Category	Trajectory Type	Proportion
Image	single-pass	30%
	Recrop once	20%
	Recrop twice	20%
	Further zoom-in	30%
Video	single-pass	90%
	Reselect	10%

Table 2: Self-correction trajectory types and corresponding proportions.

Here single-pass means no error is inserted in the trajectory. Recrop once means we randomly select a bbox that has no intersection with the reference visual cue and insert it before the correct visual operation. Recrop twice means we randomly select 2 bboxes that have no intersection with the reference visual cue and insert them sequentially before the correct visual operation. Further zoom-in means we select an inaccurate bbox that contains the reference visual cue but is excessively larger than it, and we insert it before the correct visual operation. Reselect means we sample frame indexes that have no intersection with the reference visual cue’s frame indexes, and we insert it before the correct visual operation.

C.3 Training Details

Implementation Details. For Instruction Tuning, we adapt the Open-R1 code to implement SFT loss with loss masks. For RL, we implement based on OpenRLHF. We adopt GRPO [DeepSeek-AI et al., 2025] with selective sample replay [Wang et al., 2025], because we witness significant issues of vanishing advantages. As shown in Fig. 9, our reward scheme incorporates curiosity bonus and efficiency penalty in addition to correctness rewards, which provides more variance in rewards. However, the ratio of queries that suffer from reward uniformity steadily increases to 90% as training progresses, leading to a drastic plunge in performance evidenced by the ratios of "response-all-incorrect" queries. During RL training, we employed a near on-policy RL paradigm, where the behavior policy was synchronized with the improvement policy after every 512 queries, which we define as an episode. The replay buffer for SSR persisted for the duration of each episode before being cleared. For each query, we sampled 8 responses. The training batch size was set to 256 query-response pairs. Our 7B model is trained on 4×8 sets of A800 (80G) for 20 hours .

Training Hyperparameters. For Instruction Tuning, we use a batch size of 128. The learning rate is $1e^{-6}$ with 10% warm up steps. For RL, we set employ a cosine learning rate schedule with initial learning rate $1e^{-6}$ and 3% warm up iterations. During RL training, we sample 8 trajectories per training query and set hyperparameters to $\alpha = 0.5$, $\beta = 0.05$, $H = 0.3$, and $N = 1$. This configuration reflects our objectives: the threshold $H = 0.3$ encourages the policy to utilize pixel-space reasoning in approximately 30% of responses generated for a given query, while $N = 1$ promotes efficiency by favoring responses that require at most one visual operation. Under these parameters, a response can receive a maximum exploration bonus of approximately

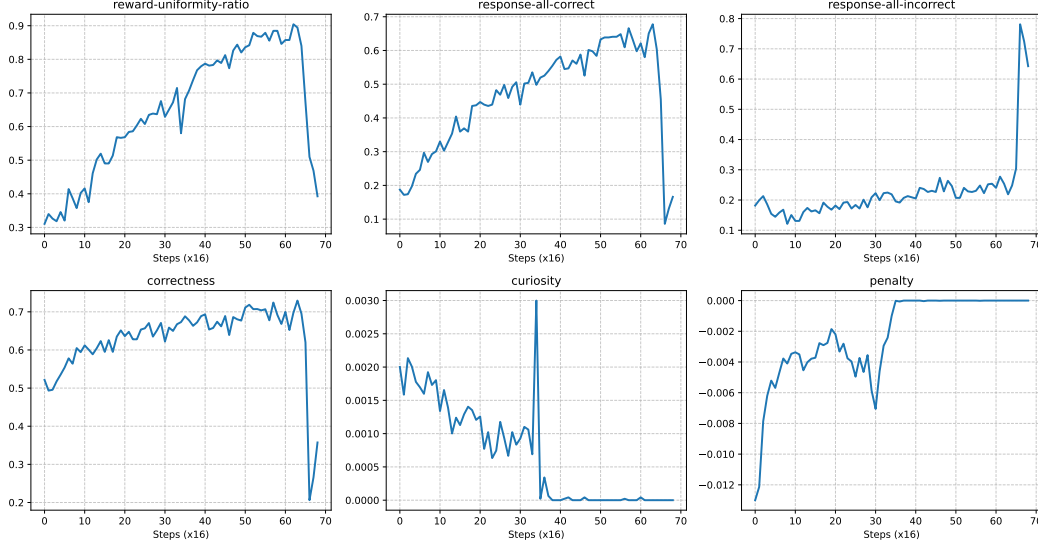


Figure 9: Training Dynamics of RL without SSR. The ratio of reward uniformity steadily saturates to 90%.

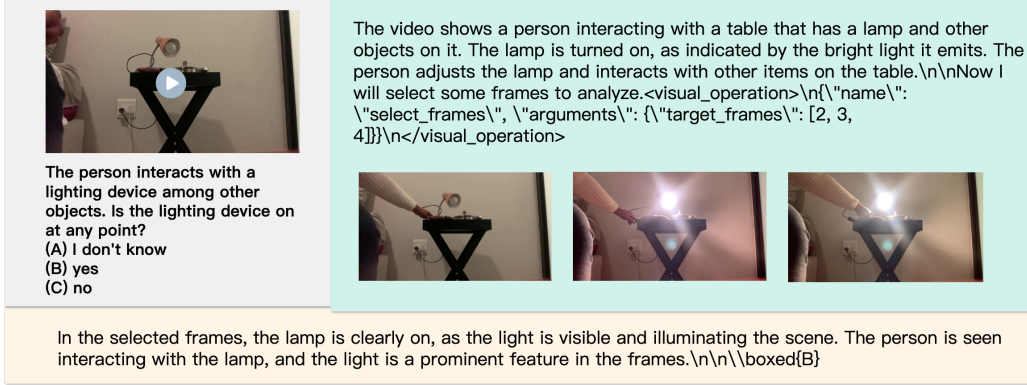


Figure 10: Pixel-Reasoner effectively captures key frames that are most informative for solving the query.

$0.5 \times (0.3 - 1/8) \approx 0.0875$, while each additional visual operation beyond the first incurs a penalty of -0.05 .

D Additional Analysis

D.1 Statistics

Qwen2.5-VL-Instruct shows limited Zero-Shot Ability in utilizing novel visual operations. We include the protocols of visual operations in the system prompt for Qwen2.5-VL-Instruct. Initially, it invokes visual operations in 20.2% training rollouts, where 40.6% of them incurs error and 36.2% leads to incorrect answer. This results in 23.2% average accuracy when utilizing pixel-space reasoning, in contrast to 49.5% average accuracy when utilizing textual reasoning.

RaPR of Pixel-Reasoner on the Evaluation Benchmarks. Our Pixel-Reasoner adaptively triggers pixel-space reasoning with a portion of 78.53% on V-Star, 57.78% on TallyQA-Complex, 58.95% on InfographicsVQA, and 66.95% on MVBench.

D.2 Case Analysis

We show successful trajectories in Fig. 11 and Fig. 10. We also provide concrete examples to show the failure modes when the policy suffers from "the learning trap".

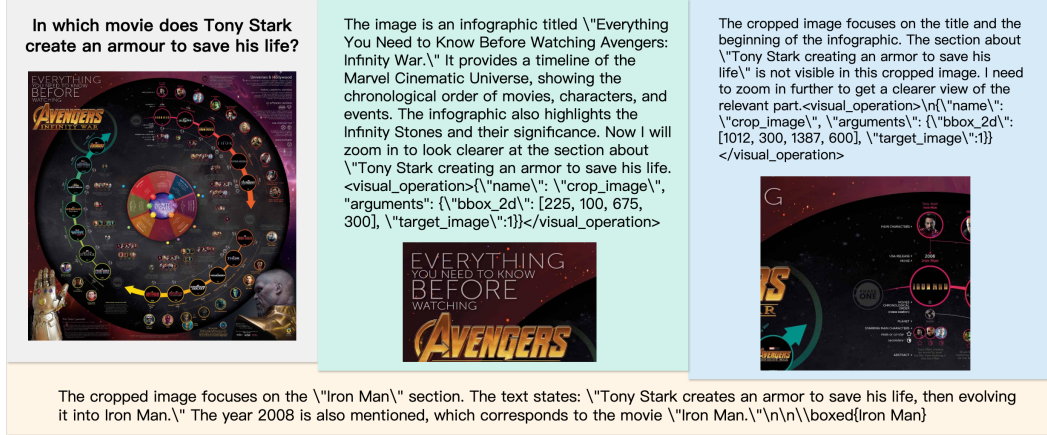


Figure 11: Pixel-Reasoner effectively identifies the relevant region within the infographic and performs targeted analysis, either by refining the crop or directly answering the query.

Failure mode 1: Hallucination. In the example below, the policy hallucinates as if the operations are successfully executed.

The video shows a table with various items, including vegetables and a bowl. The person is seen interacting with the vegetables, specifically placing them into the bowl. At one point, the person is seen shuffling some letters on the table. The video does not clearly show the sequence or order of the letters throughout the video. Now I will select some frames to look clearer at the sequence of the letters.

Execution error: max() arg is an empty sequence

The cropped video frames show the person continuing to prepare the salad and shuffle the letters. The letters appear to be in a different arrangement compared to the beginning of the video.

Failure mode 2: No-Reaction. In the example below, the policy ignores the error message and proceeds with textual reasoning.

The video shows a person in a kitchen. The person interacts with objects on the stove, including a pot. At one point, the person appears to handle an object on the counter and then moves away from the stove area. The sequence of actions suggests the person might have placed an item down or completed an action involving the counter.

Now I will select some frames to look clearer at which object was put down by the person.

Execution error: max() arg is an empty sequence

It seems there was an issue with selecting frames. Let me describe the scene instead:

The person in the video is standing near the stove with a pot on it. After interacting with the pot, the person moves to the counter area. It appears the person may have placed an object (like a box or a paper/notebook) down on the counter before moving away. The object is not visible in the provided frames, but the sequence of actions suggests it was placed there.

E Prompts

E.1 Prompts Used in RL Training

During RL training, we append the following prompt after the user query.

```
{question}
\n\nGuidelines: Understand the given visual information and the user query. Determine if
```

it is beneficial to employ the given visual operations (tools). For a video, we can look closer by `select_frames`. For an image, we can look closer by `crop_image`. Reason with the visual information step by step, and put your final answer within `\boxed{}`.

E.2 Prompts Used in Data Synthesis

E.2.1 Prompt for Question-answer Pair Generation for SA1B

Since SA1B lacks question-answer pairs and corresponding annotations and some pictures in SA1B have little content, we prompt gpt-4o to first determine if the image is information-rich. If yes, gpt-4o needs to use zoom-in tool to first crop a small part of the image, and then ask a question about objects in the small region. Otherwise, gpt-4o should reply Not valid. Here is the prompt for gpt-4o:

```
You are an expert in generating questions about small details in a
image. You will be given a HR image. First determin if the image is an
information-rich image. If it is not, return 'Not valid'. If it is,
choose a small region and use crop image tool to zoom in. According to
both cropped image and whole image. Generate a question about objects in
the small region. The question should be about the small object or its
color, material. Also generate 4 choices. One of them is the correct
answer. Others are wrong. It should not be ambiguous. For example if you
ask about the color of a person's shoes, there should either be only one
person or you specify which person you are referring to. Please make
sure the object is small. Don't ask about questions related to the
cropped image. For example, don't ask 'What is the color of the frame in
the cropped image?' because the cropped image will not be provided. Put
the question in the following format:
<question>
QUESTION HERE
</question>
Here is an example question:
<question>
question:What is the color of the person's shoes?
choices:
A: Red
B: Blue
C: Green
D: Yellow
correct_answer: A
</question>
<question>
question:What is the child on the crosswalk holding?
choices:
A: Ice cream
B: Ball
C: Book
D: None
correct_answer: C
</question>
Here is the tool description {tool_description}. For each tool call,
return a json object with function name and arguments within
<tool_call></tool_call> XML tags:
<tool_call>
{{"name": <function-name>, "arguments": <args-json-object>}}
</tool_call>
Stop generating after you call a tool.
Here is the image.
```

E.2.2 Prompts for Expert Trajectory Synthesis

For SA1B dataset:

You are an expert in generating trajectories involving image cropping and answering

questions. You will be given an image and one cropped part of it and a question. First, you need to briefly analyze the whole image, then generate: "Now I will zoom in to look clearer at 'query object or text'." Then you need to analyze the cropped part and answer the question. Put your answer choice in \boxed{ }.

Here is an example:

question: What is the price mentioned for renting the single house?

choices:

A: 9,000 Baht

B: 10,000 Baht

C: 8,500 Baht

D: 12,000 Baht

Analyzing the whole image: The image shows a lively street scene with people celebrating, possibly during a festival. There is a pickup truck with people on it, and others walking around. A signboard with text is visible in the background, which seems to contain information about renting or selling a house.

Now I will zoom in to look clearer at the text on the signboard.

Analyzing the cropped part: The cropped image focuses on the signboard. The text on the signboard mentions "SALE / RENT SINGLE HOUSE" and specifies the price for renting as **9,000 Baht**.

\boxed{A}

Here is the question, image and cropped part:
{text}

For Fineweb dataset:

You are an expert in generating trajectories involving image cropping and answering questions. You will be given an image and one cropped part of it and a question. First you need to briefly analyze the whole image, then generate: "Now I will zoom in to look clearer at the part about 'query'." Then you need to analyze the cropped part and answer the question. Put your answer in \boxed{ }. Final answer should be text from article. Don't change the original text or include irrelevant text from the article. The answer should be in one sentence.

Here are some examples:

question: What are the key responsibilities of a leader?

Analyzing the whole image: The document appears to be an article titled "Top 7 Skills a Leadership Training Should Teach Managers." It discusses various aspects of leadership training, including leadership essentials, change management, performance coaching, and conflict management. The article emphasizes the importance of leadership skills in managing teams effectively.

Now I will zoom in to look clearer at the part about "key responsibilities of a leader."

Analyzing the cropped part: The cropped part focuses on "Leadership Essentials," which outlines the basics of leadership, including understanding the role of a leader and the key responsibilities of a leader.

\boxed{building relationships, setting expectations, delegation, and developing a goal-oriented approach.}

question: Who won the first SEC championship in football?

Analyzing the whole image: The document is a Wikipedia article titled "SEC Championship Game." It provides an overview of the Southeastern Conference (SEC) Football Championship Game, including its history, format, results, and notable

moments. The article also includes a table summarizing the results of all SEC Championship games since its inception in 1992.

Now I will zoom in to look clearer at the part about "who won the first SEC championship in football."

Analyzing the cropped part: The cropped section includes a table of results from all SEC Championship games. The first game, held in 1992, lists #2 Alabama defeating #12 Florida with a score of 28-21 at Legion Field in Birmingham, Alabama.

\boxed{Alabama}

Here is the question and image:
{text}

For STARQA dataset:

You are an expert in generating trajectories involving frame selection and answering questions. You will be given 16 images (video frames) in chronological order and several selected frames from them and a question. First you need to briefly analyze the whole video, then generate: "Now I will select some frames to look clearer at 'query object or text'." Then you need to analyze the selected frames and answer the question. Put your answer choice in \boxed{ }.

Here are some examples:

question: why did the woman take the measuring spoons away from the boy?

choices:

- A: do not need it anymore
- B: feeding
- C: finish eating the piece
- D: so can take picture
- E: wants to play with it

Analyzing the video:

The video shows a woman and a boy in a kitchen setting. The boy is sitting on the counter, holding measuring spoons, while the woman appears to be engaged in a baking or cooking activity. The woman interacts with the boy, guiding him as they work with ingredients like flour and eggs. Toward the end, the woman takes the measuring spoons away from the boy.

Now I will select some frames to look clearer at why the woman took the measuring spoons away from the boy.

Analyzing the selected frames:

In the selected frames, the woman is seen taking the measuring spoons from the boy. The boy appears to have finished using the spoons to add ingredients to the bowl. The woman likely takes the spoons to proceed with the next step in the cooking process.

\boxed{A}

question: Which object was put down by the person?

choices:

- A: The cup/glass/bottle.
- B: The clothes.
- C: The bag.
- D: The book

Analyzing the video:

The video shows a person entering a room and sitting at a table. The person appears to be holding a sandwich and a book. She places the book on the table, eats the sandwich, and then picks up the book again to read. Toward the end of the video, the person leaves the table, leaving the book behind.

Now I will select some frames to look clearer at which object was put down by the

person.

Analyzing the selected frames:

In the selected frames, the person is seen entering the room holding a sandwich and a book. She places the book on the table before eating the sandwich. The book remains on the table as the person continues her activity and eventually leaves the room.

\boxed{D}

Here is the question and video:

{text}