Digital Discovery

ROYAL SOCIETY OF CHEMISTRY

PAPER

View Article Online
View Journal

Cite this: DOI: 10.1039/d5dd00542f

CREOLab: a procedure captioning dataset for understanding creative tool use in object-rich laboratory videos

Shigeaki Goto * and Tatsuki Hasebe

"Creative tool use" refers to the flexible application of tools beyond their intended purpose. In scientific experiments, this behavior is described as a "lab hack," and its automatic documentation is valuable for accumulating experimental knowledge. Recently, vision-language models (VLMs) have shown promise for generating procedural descriptions from experimental videos. However, VLMs typically rely more on object-based knowledge than on understanding the manipulations. This issue is often overlooked in existing laboratory video datasets, as tools are typically used in standard, prescribed ways. Thus, the extent to which these models can interpret and describe actions that extend beyond object-based knowledge, such as creative tool use, remains uncertain. Moreover, laboratory environments often contain numerous items unrelated to the operation (*i.e.*, decoy objects), which can divert the model's attention and further complicate the accurate identification of creative manipulations. To address this limitation, we developed an evaluation dataset called "CREOLab" (CREative tool use in Object-rich Laboratories), consisting of 65 videos from 13 experimental scenarios featuring creative tool use, each recorded across five levels of decoy object density. Using a state-of-the-art, cloud-based VLM captioning system, we evaluated model performance. As the number of decoy objects increased, the model tended to insert redundant procedural steps or omit essential ones. As a result, it failed to document scenarios involving creative tool use accurately. These findings suggest that enhancing the reliability of automatic experimental recording with VLMs requires mechanisms for automated verification of generated outputs, as well as recording protocols that reduce the influence of decoy objects.

Received 6th December 2025
Accepted 8th May 2026

DOI: 10.1039/d5dd00542f

rsc.li/digitaldiscovery

Introduction

Maintaining detailed records of experimental work is crucial for ensuring reproducibility and preserving scientific knowledge. Traditionally, handwritten lab notebooks have been the primary medium used for this purpose. However, in recent years, the digital management of experimental data through electronic lab notebooks (ELNs) has been recommended.[1–4] The benefits of digitalization extend beyond reproducibility assurance.[5] Textual records and metadata tags are particularly advantageous for data retrieval and analysis and can serve as input for language models (LMs).[6,7] Consequently, ELNs are expected to form the foundation of data-driven research in informatics[8] and self-driving laboratories (SDLs),[9,10] thereby enabling new research paradigms and experimental automation.

To enhance the usefulness of ELNs, a part of the metadata entry process should be automated to reduce the workload.[1,6] Recent studies have explored the automatic labeling or captioning of experimental videos using recognition techniques. Previous research on automated experiment recording has focused mainly on predefined operational categories, such as reagent addition or stirring, rather than on diverse, undefined manipulations.[11–13] These efforts can be framed as temporal action segmentation problems, which are typically addressed through classification-based neural networks.[14,15]

However, laboratory operations are far more diverse and often display considerable creativity in practice. For instance, weighing paper may not only serve to measure chemicals but also be used as a temporary tool or even as a notepad. Such flexible, purpose-extending practices are commonplace and reflect the ingenuity of researchers who adopt tools to achieve experimental goals efficiently in the absence of specialized equipment. In a laboratory context, these behaviors are often referred to as lab hacks,[16] whereas in the automation studies, they are described as creative tool use.[17,18] Both embody valuable procedural knowledge that merits systematic documentation.

Recently, video captioning technologies based on vision LMs (VLMs) have been actively investigated to convert arbitrary video content into descriptive text.[19–22] Among these, cloud-based VLMs, such as GPT-based architectures, outperform standalone models on video captioning benchmarks.[23,24] However, VLMs generally rely more on prior object knowledge than on the

Toyota Central R&D Labs., Inc., 41-1, Yokomichi, Nagakute, Aichi, Japan. E-mail: sg-goto@mosk.tytlabs.co.jp

temporal dynamics of visual sequences, often resulting in overly simplistic interpretations of video content.[25–27] In particular, existing laboratory video datasets primarily capture standard experimental procedures in which tools are used for their conventional purposes.[11,13,28,29] Consequently, even when a model appears to recognize an action such as "pipetting," it is unsure whether it demonstrates a true understanding of the operation or simply relies on shortcut reasoning triggered by the presence of the pipette. Therefore, these object-knowledge biases often remain undetected under conventional evaluation settings. This limitation raises a critical question: is it possible to automatically record laboratory actions that embody researchers' creativity and ingenuity, such as lab hacks or creative tool use, that transcend object-level recognition?

This study aims to identify the challenges associated with automatically recording experimental procedures from laboratory videos, including atypical operations within a video captioning framework using competitive and representative VLMs. To achieve this goal, we propose an evaluation dataset named CREOLab (CREative tool use in Object-rich Laboratory), which comprises 13 scientific experimental scenarios involving creative tool use (Fig. 1). The dataset is specifically designed to progressively introduce multiple irrelevant laboratory items (decoy objects) into each scene. This design enables rigorous evaluation of whether VLMs can genuinely interpret and document manipulative actions rather than relying excessively on correlations between visible objects and their typical uses.

The main contributions of this study are summarized as follows:

- Development of the CREOLab video captioning evaluation dataset based on 13 scientific experimental scenarios focusing on creative tool use.
- Introduction of multiple decoy objects in each scenario to assess the robustness of caption generation, systematically increasing the number of decoys to conduct an in-depth analysis of object-knowledge biases.
- Development of a fully automated evaluation protocol for VLMs incorporating a pipeline that integrates caption generation with cloud-based VLMs and checklist-based evaluation.
- Execution of 1000 caption generations and evaluations for each of multiple cloud-based VLMs using a procedural documentation system, revealing the limitations of current VLMs in procedural recording.

The remainder of this paper is organized as follows. Section 2 reviews research related to procedural recording and automation. Section 3 provides detailed information about the dataset. Section 4 describes the evaluation methodology. Section 5 presents and discusses the results, including future challenges in the automatic recording of scientific procedures. Finally, Section 6 presents the conclusion.

Related works

In this section, we first review existing scientific video datasets from experiments. Then we discuss benchmarking video captioning techniques. In both cases, we frame the discussion around our main research question: do VLMs genuinely understand scientific experimental procedures or do they rely primarily on object-knowledge biases? we further identify the limitations of prior studies that our proposed dataset is designed to address.

Laboratory video datasets

Most existing video datasets focusing on scientific experiments emphasize the correspondence between predefined action labels and video scenes. Sasaki *et al.*[11] constructed a dataset containing approximately 500 short clips, segmented at the action level, and classified them into three categories: "adding,"

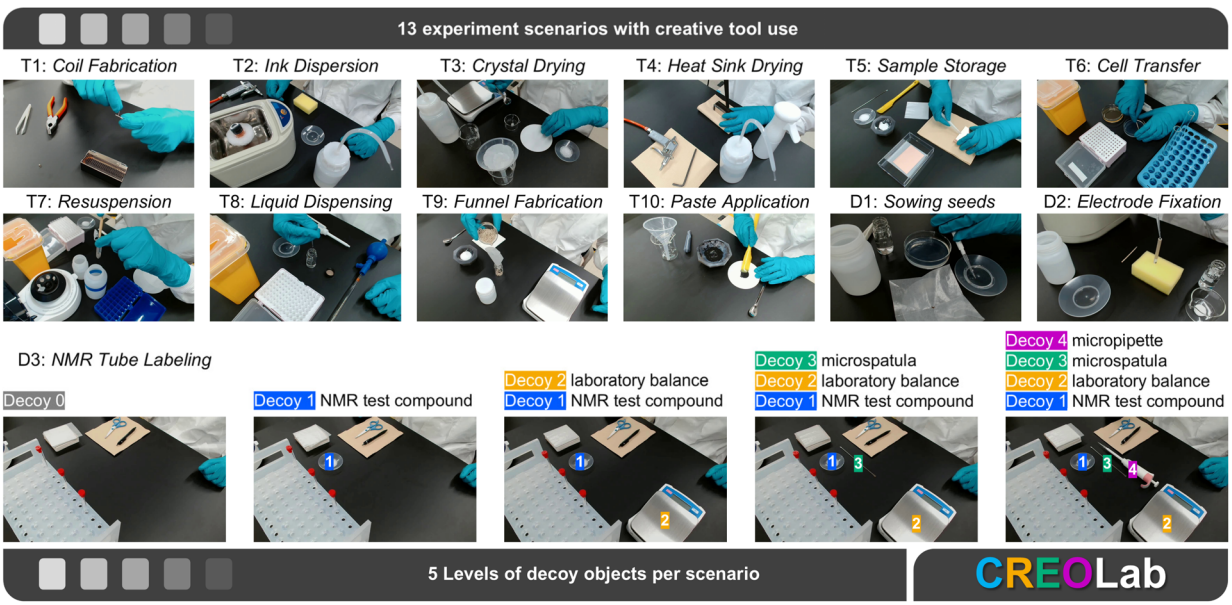

Fig. 1 Overview of the proposed CREOLab dataset. This video captioning dataset consists of 65 original videos covering 13 creative tool use scenarios based on scientific experiments, each featuring 5 levels of decoy objects. NMR: nuclear magnetic resonance.

"stirring," and "transferring." Gabrieli *et al.*[13] developed the lab action dataset and performed a comparative evaluation of various model architectures, including VLMs, for classifying videos based on predefined operation labels. Similarly, Yagi *et al.*[28] created the FineBio dataset, in which experimental procedures are hierarchically annotated at the protocol, step, and atomic operation levels.

With advances in deep learning, particularly in LMs, recent research has shifted toward associating procedural step descriptions with corresponding video scenes based on semantic understanding. Nishimura *et al.*[29] introduced BioVL, a dataset of 16 bioscience experiment videos, to evaluate the ability of video-language embedding models to align procedural texts with visual events. Cui *et al.*[30] developed the ProBio dataset for molecular biology, defining a recognition task to identify which procedural step in a written protocol corresponds to an observed video action. They also introduced three difficulty levels based on the ambiguity of the procedural descriptions. Nishimoto *et al.*[31] developed BioVL-QR, a dataset distinguished by the use of QR codes attached to experimental items. By detecting and matching these QR codes within the video, they leveraged object-level information to predict the correspondence between procedural steps in known protocols and specific video segments.

In recent years, the rapid advancement of VLMs has attracted significant attention in generating procedural texts directly from video data. Nishimoto *et al.*[31] highlighted that their BioVL-QR dataset could serve as a valuable resource for automatic generation of experimental protocols through video understanding. Similarly, Chen *et al.*[32] employed curated online videos of scientific experiments and used VLMs to produce natural language descriptions, including procedures, underlying principles, and safety guidelines, by referencing Wikipedia.

One limitation of existing datasets is that they primarily emphasize experimental procedures in which tools are used conventionally, thereby limiting insight into whether VLMs genuinely interpret scientific procedures or instead rely predominantly on prior knowledge of standard tool usage. To address this, our CREOLab dataset emphasizes creative tool-usage scenarios, featuring atypical operations that cannot be comprehended through shortcuts based solely on object knowledge.

Benchmarking in video captioning

Generating procedural texts from videos can be viewed as a form of video captioning.[33–35] Traditional video captioning methods, which relied on template-based descriptive systems and classical classifiers, have evolved toward generative approaches using advanced LMs and VLMs.[36]

Large LMs (LLMs) possess extensive prior knowledge of tools and environments. Research has shown that leveraging this knowledge enhances performance on captioning benchmarks. For example, Chou *et al.*[37] demonstrated that pretrained models, such as GPT, can generate commonsense textual knowledge regarding the functions and purposes of tools within a scene, and incorporating this auxiliary information significantly improves the accuracy of action captioning. Furthermore, Niu *et al.*[38] proposed a method that inputs the initial and final frames of a video into an LLM to infer intermediate procedural steps using the model's chain-of-thought reasoning. More recently, cloud-based models, such as GPT, have been extended to process visual inputs, enabling their direct application as VLMs for video captioning and producing high-quality, contextually coherent results.[23,24]

However, studies have shown that current VLMs still struggle to capture motion and action details in videos accurately. Wang *et al.*[25] through ActionBench experiments involving video reversal and antonymic verb substitution, revealed that many VLMs rely excessively on prior object knowledge while underemphasizing dynamic motion. Similarly, Shvetsova *et al.*[26] identified limitations in existing benchmarks that permit correct predictions based primarily on object or background recognition. They proposed improved benchmarking methods to eliminate such biases. Ma *et al.*[27] also introduced "hard negative" captions that alter actions while retaining the same objects, thereby increasing the task's difficulty.

Based on aforementioned prior studies, in our CREOLab dataset, irrelevant scientific instruments are deliberately placed in the scenes as decoys. This serves as a highly challenging benchmark that closely reflects a realistic laboratory environment and is likely to induce captioning errors owing to object-knowledge bias. Furthermore, by introducing decoys incrementally within the same scenario, our dataset enables a more quantitative and in-depth analysis of object-knowledge bias than previous datasets.

Dataset

This section describes the configuration of the creative tool use scenarios and the decoy objects that define the proposed CREOLab dataset. Its design is based on two principles: (1) adopting creative tool-use scenarios that cannot be easily interpreted solely based on knowledge of objects' typical functions and (2) introducing decoy objects incrementally to allow for the quantitative analysis of object-knowledge bias. It also outlines the procedures for video recording and annotation used during the dataset's construction.

Creative tool use scenarios

In this study, creative tool use refers to the innovative act of employing tools for purposes beyond their original intended functions. These actions exemplify researchers' ingenuity in overcoming complex experimental challenges where specialized instruments are unavailable or impractical.

We conducted interviews with seven in-house researchers actively engaged in experimental work. Their areas of expertise covered diverse disciplines, including electrochemistry, inorganic chemistry, polymer chemistry, biology, and thermal engineering. Through discussions with these participants, we collected 23 instances of creative tool use. From this set, we excluded three scenarios for safety reasons, such as those

involving heating operations, four scenarios because the required tools were not readily available, and three scenarios that did not align with the dataset's concept, specifically those in which the proposed creative tool use corresponded to the standard use of the tool at the verb level. For each scenario described in the interviews, we abstracted research-specific details and reconstructed the scenario with a different performer. The reenactments were recorded on video, and the original researchers subsequently reviewed them. Whenever inconsistencies were identified, the procedures or tools were modified accordingly. Ultimately, we constructed 13 creative tool use scenarios (Table 1).

Each scenario was assigned an ID with the prefix "D" or "T." The prefix "D" indicates the development split used for video captioning and prompt design. The prefix "T" denotes the test split used for evaluating previously developed captioning systems.

In every scenario, the primary tool is intentionally used in a creative manner distinct from its conventional application. For example, in scenario D1, tiny seeds about 0.2 mm in diameter, which cannot be grasped with tweezers, are soaked in water and individually sown by drawing them up one by one with a micropipette, an instrument initially designed for dispensing liquids. In scenario T9, because a funnel with a flow path wide enough for pellets cannot fit into a short vial bottle, a simple funnel is improvised by rolling a sheet of weighing paper, typically used for packaging reagents.

Thus, none of the scenarios represent misuse of tools; instead, they depict realistic and rational instances of creative problem-solving. It is worth noting that these scenarios are abstract examples and have not undergone formal training in experimental safety.

Decoy object variations

In each scenario, objects not directly involved in the task were deliberately placed in the video. These objects, termed "decoy objects," were intended to divert the attention of the VLM. As presented in Table 1, multiple types of decoy objects were defined for each scenario. Under the zero-decoy condition, none of these objects appears. In contrast, under the N decoys condition, all decoys from Decoy 1 to Decoy N were present. To ensure that all objects were captured within the camera frame, the maximum number of decoy object types was limited to four. Accordingly, five experimental conditions were established for each of the 13 scenarios, ranging from 0 to 4 decoys, yielding a total of 65 videos. The core task performed in each video remained identical, regardless of the number of decoy objects. The decoy objects were deliberately chosen for their high contextual relevance, *e.g.*, items commonly used alongside those featured in the main scene. This selection strategy is designed to reflect real-world experimental conditions better and to elicit misdescriptions of standard operations that depend more on object knowledge.

Table 1 List of scenarios and decoy objects. Thirteen scenarios are prepared, comprising development splits D1–3 and test splits T1–10, where the same GT procedure is performed within each scenario. For each scenario, four decoy objects are introduced across five levels; at level *N*, all decoy items numbered up to *N* are present

	Scenario overview		Additional decoy object			
ID	Tool	Creative use	Decoy 1	Decoy 2	Decoy 3	Decoy 4
D1	Micropipette	Sowing tiny seeds	Sucrose solution	Tweezers	Seed	Distilled water
D2	Sponge	Fixing a working electrode	Electrode polishing agent	Neutral detergent	Ultrasonic cleaner	Ultrapure water
D3	Weighing paper	Preparing a label for an NMR tube	NMR test compound	Laboratory balance	Microspatula	Micropipette
T1	Screwdriver	Winding a platinum wire into a coil	Pliers	Cross-recessed screw	Heat sink	Tweezers
T2	Ultrasonic cleaner	Dispersing catalyst ink in a vial	Distilled water	Air blow gun	Neutral detergent	Sponge
T3	Filter paper	Covering the crystallizing dish	Ethanol	Powder reagent	Beaker	Laboratory balance
T4	Alcohol	Promoting the drying of a heat sink	Air blow gun	Laboratory wipe	Hex key	Spray bottle
T5	Weighing paper	Partitioning metal plate samples	Chemical paste	Plastic spatula	Powder reagent	Microspatula
T6	Toothpick	Transferring microbial cells from an agar plate	Micropipette	Pipette tip box	Pipette tip waste container	Test tube
T7	Paint brush	Resuspending contents in a microtube	Microcentrifuge	Filter paper	Microtube rack	Powder reagent
T8	Pipette tip	Dispensing liquid (without using a micropipette)	Micropipette	Vial cap	Pipette tip waste container	Volumetric pipette
T9	Weighing paper	Pouring pellets into a vial; using it as a conical funnel	Laboratory spatula	Powder reagent	Laboratory balance	Ultrapure water
T10	Filter paper	Serving as a substrate for reagent application	Laboratory spatula	Filter funnel	Beaker	Pestle

Video recording setup

This scenario focuses exclusively on manual operations involving small objects, such as wires and seeds. The filming emphasized detailed depictions of these fine movements, at the expense of wide-angle coverage. As a result, the worker's entire body is not visible within the frame. Nevertheless, care was taken to ensure that all objects appearing in the scenario remained within view whenever possible.

Although some scenarios were designed to simulate work in a draft chamber, all videos in this dataset were recorded under standard, air-conditioned conditions. For safety reasons, no actual chemicals were used; instead, visually similar household materials served as substitutes. Specifically, water was used in place of alcohol, and seasonings represented various chemicals. For nonchemical instruments, such as micropipettes and disk working electrodes, all items, including decoy objects, were genuine laboratory tools.

Each recording was captured using a single RGB camera mounted on a tripod, with the viewing angle adjusted for each scenario. Within the same scenario, the task content was standardized, and the same operation was performed and recorded across five decoy levels. The demonstrations were restricted to scenes involving creative tool use, which is the focus of our investigation, while preceding and subsequent tasks were excluded. Consequently, the dataset comprises 65 videos, with durations ranging from 10.5 to 60.0 s and an average length of 22.9 s. No audio was recorded.

Annotations

Manual annotations were created for each of the 65 recorded videos, producing ground truth (GT) data in JSON format, as shown in Fig. 2. Captions were freely composed following the format [Action verb] [Specific object name] [Specific action details] for each procedural step, typically comprising two to four steps per video. All visible objects, including decoys, were also listed. For each object, its name and the normalized coordinates of its center position in the initial frame were documented. The work procedures were independent of the presence or absence of dummy objects; therefore, identical caption strings were used for scenarios representing the same task.

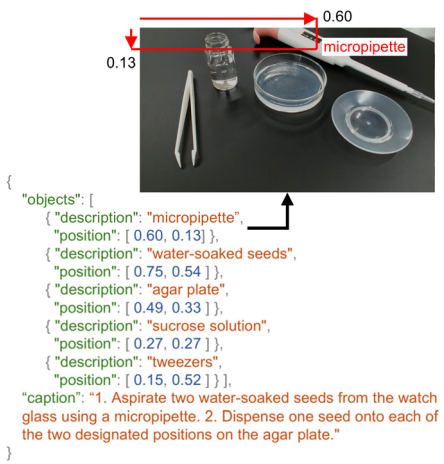

Fig. 2 An example of GT data. For each video, the names of objects (including decoys), their relative positions in the image, and a caption common to the scenario are provided.

Experimental setup

This section presents the experimental setup used for verification with the constructed CREOLab dataset. The primary objective of this experiment was not to develop an optimal video captioning system but to analyze the limitations and failure modes of current VLMs. To achieve this, we designed an automated experimental pipeline comprising two components: a captioning module and an evaluation module, as illustrated in Fig. 3. The following sections provide a detailed explanation of each module.

Captioning module

Recent studies have shown that cloud-based VLMs have achieved high performance on several video captioning benchmarks.[23,24] Based on this progress, our experiment employs GPT-5, LLaMA-4, Gemini-3, and Claude-4, which are well-known advanced VLMs available at the time of experimentation, to perform video captioning. To the best of our knowledge, few published papers have addressed video captioning in the context of scientific experimental procedures, and no baseline

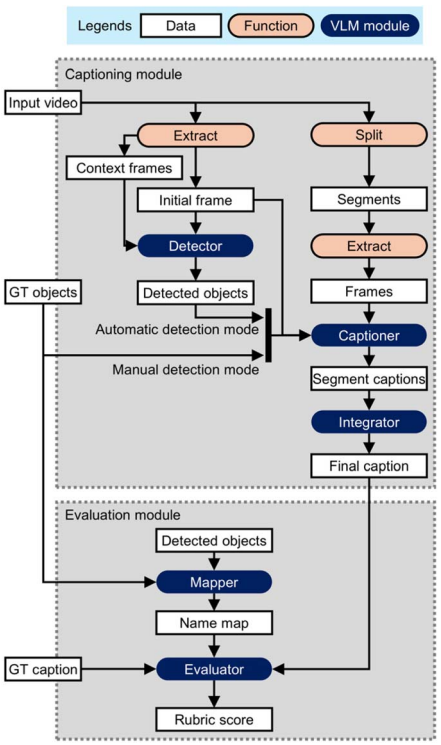

Fig. 3 Overview of the system developed to highlight the challenges faced by VLMs in experimental procedure captioning. The system comprises a captioning module and an evaluation module, which operate automatically across all experimental iterations.

systems have been established using cloud-based VLMs. Therefore, it is necessary to design a customized captioning system. In this study, we prioritized simplicity to facilitate the interpretation and discussion of the results.

Since such VLMs cannot process raw video files, discrete frames extracted from each video were used as sequential image inputs. A shorter frame interval minimizes the likelihood of missing scene transitions but results in a larger number of frames. Although the application programming interface supports numerous image inputs, not all may be considered during inference.[39] To address this trade-off, following previous studies,[40–42] the input videos were divided into short segments. Each video was segmented into 6 s units with a 1 s overlap between consecutive segments. From each segment, seven evenly sampled frames (four in the case of LLaMA 4 owing to input limitations) were used as prompts for the VLM to generate captions. The model's segment-level captions were integrated and summarized into a concise procedural description of approximately three steps.

During captioning, an object list containing the names and spatial positions of all objects was incorporated into the prompt alongside the initial frame image. This addition helped reduce inconsistencies in terminology across segments. To differentiate the model's captioning capability from its object detection performance, two operational modes were implemented:

- Manual detection mode: this mode uses the "objects" attribute from the GT data, allowing captioning to proceed without relying on automatic object detection.
- Automatic detection mode: this mode did not use any GT data. Instead, information corresponding to the "objects" attribute was automatically generated by the VLM by analyzing context frames sparsely sampled from the entire video. In this case, the captioning outcomes directly reflected the VLM's inherent ability to recognize objects.

Evaluation module

Evaluating generative tasks, such as video captioning, has long posed a considerable challenge. Traditional metrics, like BLEU,[43] rely on surface-level word-sequence matching and fail to capture semantic meaning. In contrast, embedding-based methods, such as BertScore,[44] better reflect semantic similarity but remain insensitive to negation and word order variations,[45] leading to serious misjudgments in procedural evaluations. Although these metrics can visualize overall performance differences, they offer limited insight into why one result outperforms another.

In this study, a precise absolute evaluation was unnecessary because our primary goal was to clarify the limitations of current VLMs. However, to efficiently visualize and explore a large number of experimental results, it was desirable to assign scores that enabled relative comparisons. Therefore, we employed a checklist-based point-deduction method and constructed an automated evaluation pipeline using the same VLM as that used in the captioning module as the evaluator. The deduction criteria were as follows:

- Critical step omissions: −15 points per instance.
- Incorrect step sequence: −12 points per instance.
- Unnecessary additional steps: −8 points per instance.
- Incomplete step descriptions: −5 points per instance.
- Incorrect terminology: −10 points per instance.
- Ambiguous terminology: −5 points per instance.

We determined the deduction values according to their impact on task reproducibility. Each evaluation began with a perfect score of 100 points, from which deductions were applied for corresponding errors to compute the final score. Analyzing each deduction factor enables a statistical discussion of failure patterns.

Deduction judgments were made by comparing the generated captions with the annotated GT procedures. However, automatically generated captions do not always use identical terminology. For instance, if a scene depicts "holding distilled water," the automatic recognition system might detect the object as a bottle and describe it as "holding a bottle". When the correspondence between "distilled water" and "bottle" is predefined, both expressions are treated as equivalent. Otherwise, "holding distilled water" is considered missing and penalized under critical step omissions, while "holding a bottle" is treated as an unnecessary action under unnecessary additional steps. To mitigate this issue, we created a name-mapping scheme that aligned the lists, thereby accommodating terminological variations during the evaluation process.

Results and discussion

This section presents the results of evaluation experiments conducted using the constructed experimental pipeline and the CREOLab dataset, along with a discussion of the findings, focusing mainly on the case where GPT-5 (gpt-5-2025-08-07) is used as the VLM. Because the experiments relied on a cloud-based generative model, caption generation and its evaluation were affected by random seeds, resulting in substantial variation in the final scores. To ensure the reliability of the findings, ten iterations were conducted for each of the 50 test scenario videos under both the automatic and manual detection modes, resulting in a total of 1000 experiments. When reporting the scores, the mean values obtained under identical conditions were used, and 95% confidence intervals were represented as error bars to illustrate the range of variation. For the generated examples, one representative case was systematically selected from each condition group, corresponding to the median score (the upper median in even-numbered groups), thereby minimizing randomness and arbitrariness.

Generated examples

Among the 1000 generation and evaluation experiments, Fig. 4 shows a representative example under the conditions of scenario T2, four decoy objects, and the automatic object detection mode. First, the detector outputs the names and coordinates of multiple objects in the scene with sufficient accuracy for referencing each object. Then, the captioner generates captions for each segment (a six-second video clip) using the detected object names. Subsequently, the integrator

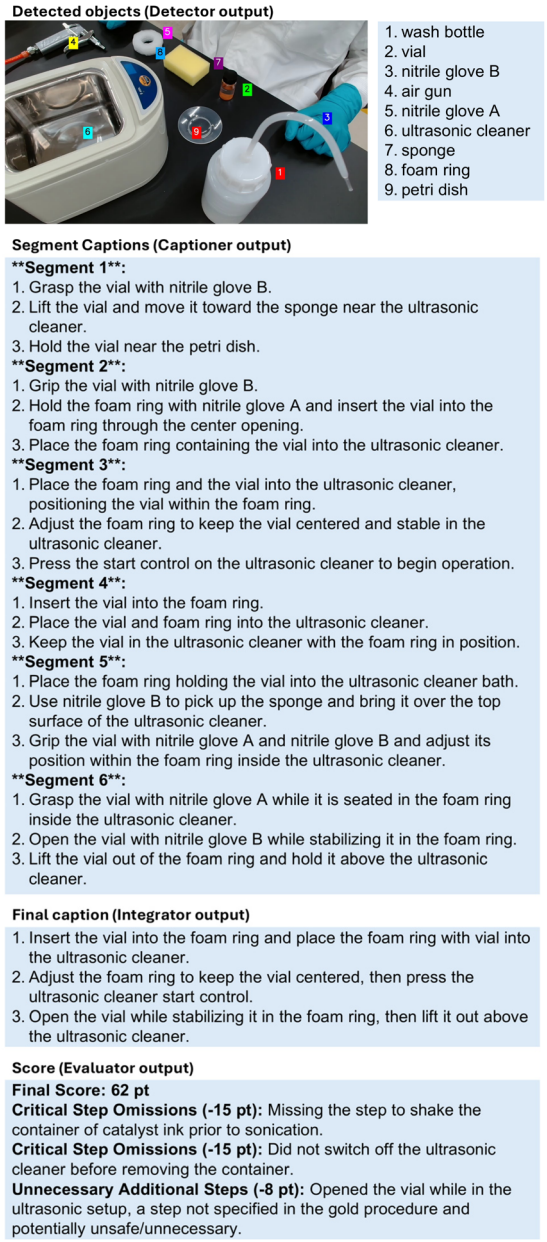

Fig. 4 Example of video caption generation and evaluation results obtained using the constructed system (Scenario T2, Decoy 4. GT caption: 1. shake the container of catalyst ink. 2. Insert the container of catalyst ink into the polystyrene foam with a hole. 3. Place the container and polystyrene foam into the ultrasonic cleaner and switch it on. 4. Switch off the ultrasonic cleaner and remove the container filled with catalyst ink.).

successfully summarizes these captions into a concise, three-step procedure. Finally, the evaluator records in detail which penalty items were applied and how they were assigned based on a comparison with the GT procedure. These results confirm that the experimental pipeline operated as intended.

Statistical performance evaluation

Fig. 5 summarizes the results from 1000 trials, illustrating variations in scores with respect to the number of decoy objects

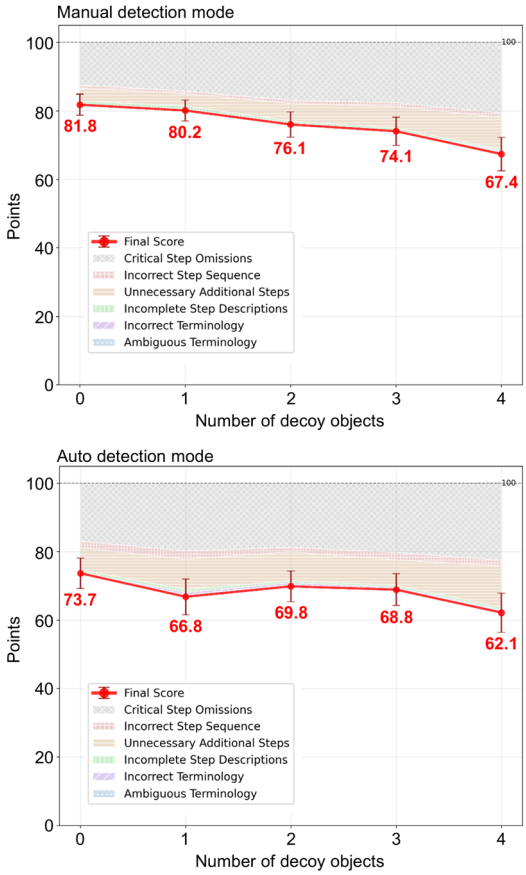

Fig. 5 Variation in captioning quality as a function of the number of decoy objects. Top: manual detection mode; Bottom: automatic detection mode. Each point represents the mean over 100 trials (10 test scenarios × 10 iterations), and the error bars indicate the 95% confidence interval of the mean final score.

for each detection mode. Each plot represents the average of 100 trials (10 scenarios × 10 iterations) and the error bars denote the 95% confidence interval of the mean. Regardless of whether object detection was performed manually or automatically, increasing the number of decoy objects consistently led to more errors, such as omissions of essential steps and additions of unnecessary ones, thereby reducing the final scores.

Table 2 presents the mean scores stratified by the number of decoy objects. The Overall row shows the result obtained by combining all trials, which is equivalent to the result shown in Fig. 5. T1–T10 indicate the mean scores further stratified by scenario. For each row, we report the regression slope β of the mean score with respect to the number of decoy objects, the coefficient of determination R^2 and the one-sided p-value testing whether the slope is negative. In addition, Cohen's d relative to the no-decoy condition ($d = 0$) is shown below the mean score for each decoy level.

In the overall analysis, β was negative in the manual detection mode ($\beta = -3.49$) and the automatic detection mode ($\beta = -2.09$), with $p < 0.01$, indicating a significant negative effect of decoys on the score. However, the magnitude of this effect

Table 2 Mean final scores by number of decoy objects for Overall and each scenario (T1–T10). Values for 1–4 decoys are shown with Cohen's d relative to 0 decoys in parentheses. β, R^2, and p denote the slope, coefficient of determination, and the one-sided p-value for a test of a negative slope in the linear regression of score on decoy count. Gray-shaded conditions are discussed in Fig. 6 and 7

Scenario	Number of decoy objects					Regression statistics		
	0	1	2	3	4	β	R^2	$p(\beta<0)$
(a) Manual detection mode								
Overall	81.8	80.2 (−0.11)	76.1 (−0.34)	74.1 (−0.42)	67.4 (−0.70)	−3.49	0.062	<0.001
T1	81.9	71.2 (−1.16)	72.3 (−0.92)	60.8 (−2.00)	38.8 (−3.18)	−9.66	0.577	<0.001
T2	64.5	64.2 (−0.02)	52.2 (−1.03)	52.5 (−1.00)	46.6 (−1.16)	−4.75	0.287	<0.001
T3	82.9	73.5 (−1.51)	69.7 (−2.13)	77.2 (−0.92)	47.0 (−3.16)	−6.81	0.398	<0.001
T4	100.0	100.0 (+0.00)	98.7 (−0.45)	100.0 (+0.00)	97.9 (−0.65)	−0.42	0.047	0.066
T5	68.1	61.3 (−0.59)	65.0 (−0.37)	54.7 (−1.04)	61.6 (−0.61)	−1.96	0.057	<0.05
T6	91.7	92.4 (+0.07)	84.5 (−0.47)	96.0 (+0.52)	80.5 (−0.70)	−1.88	0.035	0.097
T7	64.4	69.6 (+0.46)	72.0 (+0.56)	60.4 (−0.24)	55.8 (−0.58)	−2.64	0.065	<0.05
T8	79.5	79.1 (−0.05)	55.0 (−2.12)	55.0 (−2.18)	57.0 (−2.35)	−6.91	0.390	<0.001
T9	94.2	96.7 (+0.38)	95.3 (+0.16)	94.0 (−0.03)	100.0 (+1.14)	+0.89	0.050	0.941
T10	91.2	93.6 (+0.39)	96.0 (+0.94)	90.4 (−0.17)	88.8 (−0.47)	−0.80	0.045	0.069
(b) Auto detection mode								
Overall	73.7	66.8 (−0.28)	69.8 (−0.17)	68.8 (−0.21)	62.1 (−0.45)	−2.09	0.014	<0.01
T1	69.9	59.2 (−0.74)	61.4 (−0.41)	58.6 (−0.70)	39.1 (−2.01)	−6.22	0.199	<0.001
T2	57.8	57.9 (+0.01)	56.4 (−0.14)	60.0 (+0.19)	52.3 (−0.44)	−0.89	0.010	0.242
T3	70.3	69.1 (−0.10)	69.6 (−0.06)	59.1 (−0.65)	59.3 (−0.79)	−3.20	0.112	<0.01
T4	98.7	95.8 (−0.73)	94.2 (−1.21)	98.4 (−0.10)	94.7 (−0.84)	−0.54	0.027	0.126
T5	52.2	39.0 (−0.60)	64.8 (+0.74)	60.7 (+0.51)	52.5 (+0.02)	+2.23	0.026	0.870
T6	94.0	91.9 (−0.21)	81.9 (−1.22)	87.9 (−0.56)	92.2 (−0.18)	−0.76	0.012	0.227
T7	39.4	25.7 (−1.22)	42.5 (+0.20)	39.0 (−0.03)	11.9 (−2.21)	−4.17	0.105	<0.05
T8	73.1	56.0 (−1.00)	43.6 (−2.15	49.6 (−1.68)	44.9 (−2.16)	−6.28	0.295	<0.001
T9	92.0	83.5 (−0.55)	89.5 (−0.21)	83.2 (−0.49)	88.0 (−0.29)	−0.83	0.005	0.316
T10	89.1	89.6 (+0.06)	94.4 (+0.81)	92.0 (+0.48)	86.5 (−0.40)	−0.28	0.004	0.339

varied considerably across scenarios. Specifically, the standard deviation (SD) of β across scenarios was 3.40 in the manual mode and 2.78 in the automatic mode. Relative to the magnitude of β, the coefficient of variation (CV = SD/$|\beta|$) was 0.97 and 1.33, respectively, indicating substantial scenario-dependent variability. Furthermore, the relatively modest R^2 values suggest that the observed effects were influenced by random-seed variation and potential nonlinearity in the decoy effect.

Therefore, evaluating VLM captioning only under specific scenarios or decoy conditions may lead to misleading interpretations. A robust conclusion regarding model-wise trends can be drawn only by evaluating across the full set of 10 scenarios and the five decoy levels we defined. Indeed, in some scenarios, marked score reductions were observed only at specific numbers of decoys, and these cases are examined in detail below.

Detailed analysis of failures

First, attention is directed to the gray-shaded conditions in Table 2(a) (manual detection mode, scenario T3, 0, 3, and 4 decoy objects). Under these conditions, the effect size associated with the increase from 0 to 4 decoy objects (effect size = −3.16) was more than three times greater than that associated with the increase from 0 to 3 decoy objects (effect size = −0.92). Fig. 6 shows the representative examples of the generated outputs, each corresponding to the median score (the fifth-highest value) across 10 trials. This scenario involves a laboratory task in which a spatula is used to scoop a crystal and place it into a crystallizing dish, which is then covered with filter paper. Even in the 0 decoys condition, the covering action was omitted, leaving the essential aspect of creative tool use inadequately captured. In addition, as the number of decoy objects increased, the number of erroneous records tended to increase. In the condition with three decoy objects, a fictitious action, "Stir" was added in Step 1. This observation may indicate that the powder reagent, as a decoy, prompted an association with a crystal preparation step. Moreover, when an electronic balance was introduced as a decoy in the condition with four decoys, additional irrelevant steps involving its operation were added, confirming a substantial deviation from the GT caption.

Next, attention is turned to the gray-shaded conditions in Table 2(b) (automatic detection mode, scenario T5, one decoy). This scenario involves separating two samples (automatically detected as a copper sheet) using weighing paper and storing them in a sample tray (automatically detected as a plastic tray). Because this mode relies on automatic object detection, the results varied depending on the random seed of the VLM. For the weighing paper, five of the ten trials classified it as "plastic film," whereas the other five classified it as "adhesive film". Examples corresponding to the median score within each classification group are shown in Fig. 7. Since an "adhesive film" would be unsuitable as a separator because of its stickiness, this represents a functional misrecognition. Consequently, key steps related to creative tool use, such as "placing the film," were omitted and replaced with unrelated actions, such as "applying material from a bowl" or "pressing down". These misclassifications ultimately degraded the quality. These findings demonstrate that when the system relies on automatic object detection, recognition errors can misguide the captioning process, leading to failures in documenting creative tool use scenarios.

Comparison with conventional evaluation metrics

We adopted a checklist-based self-evaluation scheme using a VLM in our pipeline and compared it with conventional evaluation metrics, namely, BLEU-4,[43] METEOR,[46,47] CIDEr-D,[48]

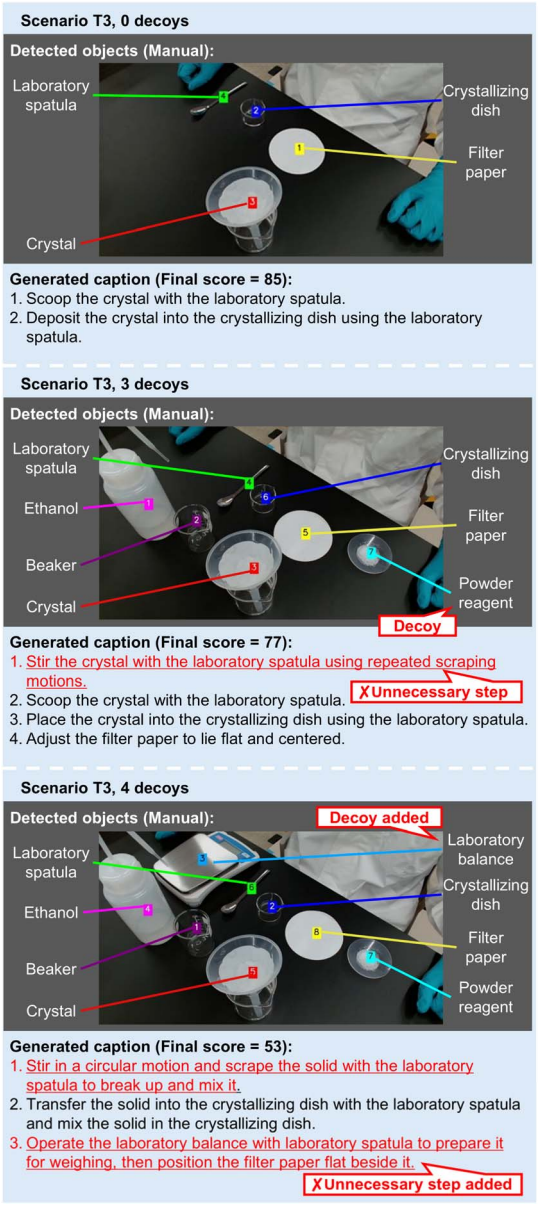

Fig. 6 Generated captions under the gray-shaded conditions in Table 2(a). Top: zero decoys, middle: three decoys; bottom: four decoys. (GT caption: 1. remove the crystal using a laboratory spatula and place it in the crystallizing dish. 2. Cover the crystallizing dish with filter paper).

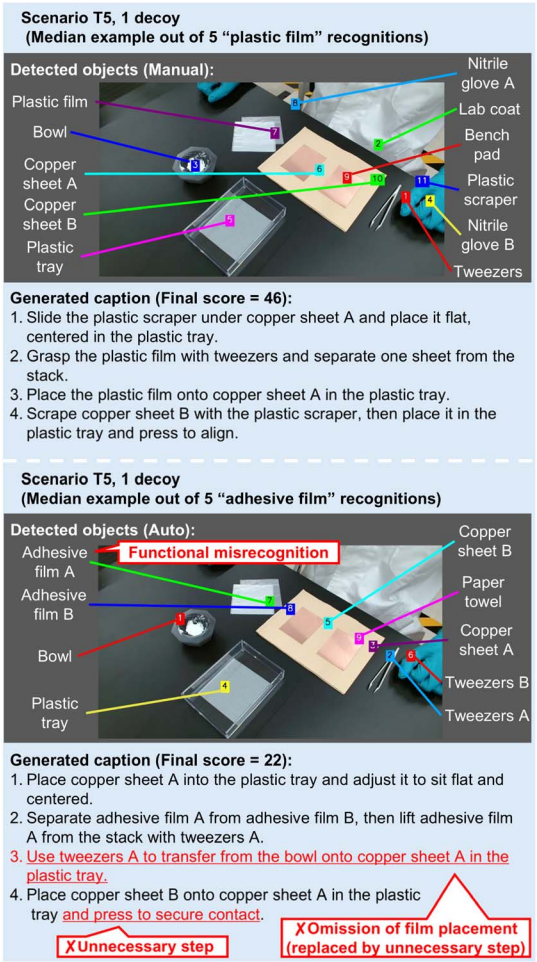

Fig. 7 Differences in generated captions based on automatic recognition results of the weighing paper under gray-shaded condition in Table 2(b). Top: when recognized as plastic film; bottom: when recognized as adhesive film. (GT caption: 1. place the metal sample on the weighing paper inside the sample storage case using tweezers. 2. Cover the metal sample with another sheet of weighing paper. 3. Place an additional metal sample on top of the weighing paper).

and BERTScore.[44] BLEU-4, METEOR, and CIDEr-D were computed using the Python library pycocoevalcap,[49] whereas BERTScore was computed as the F1 score using RoBERTa-large.[50]

Table 3 summarizes the results obtained by applying each metric to the Overall scores presented in Table 2. The proposed method (OUR) captured the degradation in video captioning performance associated with the increasing numbers of decoy objects more effectively than the conventional metrics. In particular, it produced a relatively small one-sided p-value compared to the conventional metrics for the test of a negative slope in the linear regression of score on decoy count. It also exhibited the largest effect size, indicating that it more clearly distinguished differences among conditions. Among the conventional metrics, BERTScore performed relatively well and detected a negative trend in the manual detection mode ($\beta < 0, p < 0.001$).

However, in automatic detection mode, none of the conventional metrics, including BERTScore, reliably detected a negative trend. One possible explanation is that the object names assigned by the VLM during automatic detection do not necessarily match the expressions used in the reference captions, rendering conventional metrics that emphasize lexical overlap inherently unstable. These results indicate that the proposed checklist-based method enables visualization of the breakdown of penalty factors and outperforms conventional metrics in detecting degradation in video captioning performance.

Table 3 Mean overall evaluation scores by the number of decoy objects for the proposed checklist-based evaluation (OUR) and conventional metrics. Values for 1–4 decoys are shown with Cohen's d relative to 0 decoys (in parentheses). β, R^2, and p denote the slope, coefficient of determination, and the one-sided p-value for testing a negative slope in the linear regression of score on decoy count. OUR corresponds to the Overall score reported in Table 2 and is reproduced here for comparison

	Number of decoy objects					Regression statistics		
Metric	0	1	2	3	4	β	R^2	$p(\beta < 0)$
(a) Manual detection mode								
OUR	81.8	80.2 (−0.11)	76.1 (−0.34)	74.1 (−0.42)	67.4 (−0.70)	−3.49	0.062	<0.001
BLEU-4 × 100	14.4	15.7 (+0.13)	15.5 (+0.11)	13.0 (−0.14)	13.1 (−0.13)	−0.53	0.006	<0.05
METEOR × 100	32.0	31.9 (−0.02)	31.2 (−0.17)	30.9 (−0.23)	30.4 (−0.34)	−0.42	0.015	<0.01
CIDEr-D	0.31	0.35 (+0.06)	0.38 (+0.12)	0.26 (−0.10)	0.21 (−0.20)	−0.030	0.006	<0.05
BERTscore × 100	59.5	59.0 (−0.06)	58.1 (−0.16)	57.1 (−0.29)	55.4 (−0.47)	−1.01	0.026	<0.001
(b) Auto detection mode								
OUR	73.7	66.8 (−0.28)	69.8 (−0.17)	68.8 (−0.21)	62.1 (−0.45)	−2.09	0.014	<0.01
BLEU-4 × 100	4.4	2.1 (−0.37)	4.5 (+0.02)	3.7 (−0.11)	2.6 (−0.28)	−0.19	0.002	0.163
METEOR × 100	18.9	18.3 (−0.11)	20.6 (+0.35)	19.6 (+0.15)	18.7 (−0.05)	+0.08	0.001	0.702
CIDEr-D	0.09	0.08 (−0.08)	0.10 (+0.02)	0.11 (+0.07)	0.04 (−0.25)	−0.01	0.002	0.178
BERTScore × 100	40.0	39.1 (−0.10)	42.6 (+0.27)	41.4 (+0.15)	39.1 (−0.10)	+0.05	0.000	0.571

Statistical performance evaluation of other VLMs

Previous analyses focused on GPT-5 as the primary VLM. Here, we applied the same evaluation protocol to other prominent VLMs, namely, LLaMA-4 (Llama 4 Maverick 17B-128E), Gemini-3 (gemini-3.1-pro-preview), and Claude-4 (claude-haiku-4.5), and report their quantitative trends. The aim of this analysis is not to assess model superiority, but to evaluate whether the proposed experimental pipeline and the CREOLab dataset are broadly applicable to other VLMs.

The results are summarized in Table 4. For each model, we report the values corresponding to the "Overall" metric in Table 2. The linear regression coefficient β is negative across all models, indicating, as with the GPT-5 results, that decoys generally have an adverse effect on captioning. Notably, the slope is statistically significant for LLaMA-4 and Claude-4. By contrast, for Gemini-3, the absolute value of β and the associated effect size are relatively small, and the p-value is not sufficiently low, suggesting that the impact of decoy quantity is limited within the 10 test scenarios and may not be fully captured by this dataset.

Overall, the proposed experimental pipeline demonstrates consistent applicability across multiple VLMs and effectively reveals challenges posed by decoys. At the same time, although the CREOLab dataset exposes such challenges across models, the results indicate that improving evaluation robustness remains an open challenge, potentially achievable through dataset expansion with additional high-difficulty scenarios.

Future challenges in the recording

Evaluation experiments conducted in CREOLab revealed that even state-of-the-art VLMs are often misled by the presence of objects in a scene. This misinterpretation leads to the omission of critical actions or the insertion of fictitious, redundant procedural steps, making it difficult to document instances of creative tool use accurately. To address these limitations, four primary directions for improvement are proposed.

(1) Although this study evaluated VLMs in a relatively simple video captioning pipeline design, there remains a marked potential for refinement through architectural exploration. One promising strategy is the sequential segment captioning

Table 4 Mean overall evaluation scores by the number of decoy objects across multiple VLMs. Values for 1–4 decoys are shown with Cohen's d relative to 0 decoys (in parentheses). β, R^2, and p denote the slope, coefficient of determination, and the one-sided p-value for testing a negative slope in the linear regression of score on decoy count

	Number of decoy objects					Regression statistics		
VLM	0	1	2	3	4	β	R^2	$p(\beta < 0)$
(a) Manual detection mode								
LLaMA-4	59.5	56.6 (−0.15)	56.0 (−0.17)	53.0 (−0.34)	53.4 (−0.33)	−1.58	0.013	<0.01
Gemini-3	84.7	82.2 (−0.17)	81.9 (−0.19)	82.6 (−0.14)	80.4 (−0.26)	−0.82	0.005	0.054
Claude-4	58.2	57.3 (−0.05)	54.3 (−0.21)	52.4 (−0.30)	48.7 (−0.48)	−2.41	0.027	<0.001
(b) Auto detection mode								
LLaMA-4	52.0	48.1 (−0.23)	45.9 (−0.35)	41.7 (−0.65)	43.0 (−0.57)	−2.44	0.037	<0.001
Gemini-3	80.5	79.3 (−0.07)	82.4 (+0.12)	77.0 (−0.20)	77.8 (−0.15)	−0.77	0.004	0.082
Claude-4	46.3	40.8 (−0.28)	38.8 (−0.43)	37.6 (−0.51)	35.6 (−0.58)	−2.46	0.038	<0.001

approach,[51] in which the latent semantics of one segment serve as contextual conditioning for subsequent segments. Furthermore, integrating a reflection mechanism[52,53] that can detect and correct missing procedural steps could improve accuracy. In particular, existing video-protocol alignment frameworks[29–31] could serve as verification agents that prompt caption regeneration when inconsistencies with the video are detected.

(2) Improving environmental conditions for video capture can further reduce captioning errors. For instance, attaching QR codes to objects[31] may help mitigate misrecognition-related inaccuracies. Developing an extended version of the CREOLab dataset that integrates such measures could also be explored in the future.

(3) Although the proposed dataset focuses only on video information, incorporating nonvisual modalities such as tactile[54] and auditory cues[55,56] could help mitigate object-knowledge bias in automated captioning. More broadly, such an extension would also make the ELN database valuable beyond serving merely as a repository of human operations. For instance, automatically generated captions could function as annotations for training robot foundation models.[57,58] In addition, human operation records enriched with nonvisual modality information could support the training of vision-tactile-language-action models.[59] This type of multimodal experimental record could provide a foundation for future robot-driven automated experimentation and SDLs.

(4). The object-knowledge bias revealed by the proposed dataset may be intrinsic to machine-learning-based VLM approaches. By contrast, when experiments are performed using captioning systems grounded in nonmachine-learning methods, such as Bayesian approaches exemplified by Bayesian networks,[60] the presence or extent of such bias may substantially differ.

Conclusions

Herein, we introduced the CREOLab dataset to systematically quantify the challenge of object-knowledge bias in VLMs during procedural captioning of scientific experiments. The dataset was designed according to two principles: (1) employing creative tool-use scenarios that are not easily interpreted based on object knowledge alone, and (2) introducing decoy objects in a stepwise manner to induce bias. We constructed a dataset comprising 65 videos organized into 13 scenarios, including 3 for prompt development and 10 for testing, together with an automated evaluation pipeline, and we experimentally demonstrated that this combination can quantitatively reveal the effects of object-knowledge bias in VLMs.

Nevertheless, the dataset does not yet encompass the full diversity of experimental scenarios. For example, it does not include complex, specialized procedures, such as preprocessing in materials analysis or operations involving multiple experimental instruments. Consequently, even if a model performs well across the 13 scenarios in CREOLab, this should not be interpreted as evidence of comprehensive video captioning capability for all scientific experiments. In future work, we plan to expand CREOLab to include a broader array of scenarios. Furthermore, user-specific scientific experiment scenarios could be evaluated if datasets are generated following the format established in this study. While careful attention is essential to confidentiality, we hope that such scenarios can be shared within the scientific community whenever feasible. By sharing these challenging scenarios and continuously expanding the dataset, more robust evaluations will become possible, facilitating improvements in captioning technologies. Moreover, these advancements can lay the groundwork for future data-driven laboratories by integrating reliable captioning technologies into ELNs and providing foundational laboratory records for autonomous robotic experimentation and SDLs.

Author contributions

Shigeaki Goto: conceptualization, methodology, software, investigation, formal analysis, resources, writing, original draft. He designed and implemented the experiment pipeline, conducted the experiments, performed the studies, and coordinated the overall manuscript. Tatsuki Hasebe: conceptualization, methodology, investigation, resources, writing – original draft. He designed dataset scenarios, conducted user hearings for data collection and validation, and co-authored specific sections of the manuscript.

Conflicts of interest

This study is a system evaluation study. The Experimental Ethics Review Committee of Toyota Central R&D Laboratories, Inc. reviewed the research and determined that a formal ethics review was not required (August 2025, application number 25-22). The authors declare no competing financial or nonfinancial interests.

Data availability

The CREOLab dataset, experimental logs, and code with prompts are publicly available on Zenodo (https://doi.org/10.5281/zenodo.19702197).

Acknowledgements

We would like to express our gratitude to Yuya Harada, Tamami Kamiya, Tomoko Kubota, Kengo Mimura, Keitaro Ohashi, Yohei Suzuki and Yu Takashima for their valuable discussions and insightful comments on this study regarding scientific experiment procedures.

References

1 S. G. Higgins, A. A. Nogiwa-Valdez and M. M. Stevens, *Nat. Protoc.*, 2022, **17**, 179–189.
2 H. K. Machina and D. J. Wild, *J. Lab. Autom.*, 2013, **18**, 264–268.
3 K. R. Scroggie, K. J. Burrell-Sander, P. J. Rutledge and A. Motion, *Digital Discovery*, 2023, **2**, 1188–1196.

4 S. Herres-Pawlis, F. Bach, I. J. Bruno, S. J. Chalk, N. Jung, J. C. Liermann, L. R. McEwen, S. Neumann, C. Steinbeck, M. Razum and O. Koepler, *Angew. Chem., Int. Ed.*, 2022, **61**, e202203038.
5 D. A. Leins, S. B. Haase, M. Eslami, J. Schrier and J. T. Freeman, *Digital Discovery*, 2023, **2**, 12–27.
6 C. L. Bird, C. Willoughby and J. G. Frey, *Chem. Soc. Rev.*, 2013, **42**, 8157–8175.
7 M. Jalali, Y. Luo, L. Caulfield, E. Sauter, A. Nefedov and C. Wöll, *Mater. Today Commun.*, 2024, **40**, 109801.
8 R. Buonsanti, *Chem. Mater.*, 2023, **35**, 805–806.
9 G. Tom, S. P. Schmid, S. G. Baird, Y. Cao, K. Darvish, H. Hao, S. Lo, S. Pablo-García, E. M. Rajaonson, M. Skreta, N. Yoshikawa, S. Corapi, G. D. Akkoc, F. Strieth-Kalthoff, M. Seifrid and A. Aspuru-Guzik, *Chem. Rev.*, 2024, **124**, 9633–9732.
10 M. Abolhasani and E. Kumacheva, *Nat. Synth.*, 2023, **2**, 483–492.
11 R. Sasaki, M. Fujinami and H. Nakai, *Digital Discovery*, 2024, **3**, 2458–2464.
12 H. Hu, K. Cheng, Z. Li, J. Chen and H. Hu, *Pattern Recognit. Lett.*, 2020, **130**, 267–274.
13 G. Gabrieli, I. Espejo Morales, D. Christofidellis, M. Graziani, A. Giovannini, F. Zipoli, A. Thakkar, A. Foncubierta, M. Manica and P. W. Ruch, *Digital Discovery*, 2025, **4**, 393–402.
14 S. Ji, W. Xu, M. Yang and K. Yu, *IEEE Trans. Pattern Anal. Mach. Intell.*, 2013, **35**, 221–231.
15 G. Ding, F. Sener and A. Yao, *IEEE Trans. Pattern Anal. Mach. Intell.*, 2023, **46**, 1011–1030.
16 J. Leeming, *Nat. Careers*, 2025, DOI: 10.1038/d41586-025-02719-z, https://www.nature.com/articles/d41586-025-02719-z.
17 M. Xu, P. Huang, W. Yu, S. Liu, X. Zhang, Y. Niu, T. Zhang, F. Xia, J. Tan and D. Zhao, *arXiv*, 2023, preprint arXiv:2310.13065v1, DOI: 10.48550/arxiv.2310.13065v1.
18 T. Fitzgerald, A. Goel and A. Thomaz, *Front. Robot. AI*, 2021, **8**, 674292.
19 J. Vaishnavi and V. Narmatha, *Multimed. Tool. Appl.*, 2025, **84**, 947–978.
20 B. Xin, N. Xu, Y. Zhai, T. Zhang, Z. Lu, J. Liu, W. Nie, X. Li and A.-A. Liu, *Multimed. Syst.*, 2023, **29**, 3781–3804.
21 A. Yang, A. Nagrani, P. H. Seo, A. Miech, J. Pont-Tuset, I. Laptev, J. Sivic and C. Schmid, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 10714–10726.
22 Z. Liu and R. Song, *Appl. Sci.*, 2025, **15**, 4990.
23 C. Fu, Y. Dai, Y. Luo, L. Li, S. Ren, R. Zhang, Z. Wang, C. Zhou, Y. Shen, M. Zhang, P. Chen, Y. Li, S. Lin, S. Zhao, K. Li, T. Xu, X. Zheng, E. Chen, C. Shan, R. He and X. Sun, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025, pp. 24108–24118.
24 H. Wei, Y. Yuan, X. Lan, W. Ke and L. Ma, *arXiv*, 2025, preprint arXiv:2504.05040v2, DOI: 10.48550/arxiv.2504.05040v2.
25 Z. Wang, A. Blume, S. Li, G. Liu, J. Cho, Z. Tang, M. Bansal and H. Ji, *Adv. Neural Inf. Process. Syst.*, 2023, **36**, 20729–20749.
26 N. Shvetsova, A. Nagrani, B. Schiele, H. Kuehne and C. Rupprecht, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, CVPR, 2025, pp. 29050–29059.
27 W. Ma, K. Li, Z. Jiang, M. Meshry, Q. Liu, H. Wang, C. Häne and A. Yuille, *Proceedings of the European Conference on Computer Vision*, 2024, pp. 254–269.
28 T. Yagi, M. Ohashi, Y. Huang, R. Furuta, S. Adachi, T. Mitsuyama and Y. Sato, *Int. J. Comput. Vis.*, 2025, **133**, 7352–7367.
29 T. Nishimura, K. Sakoda, A. Hashimoto, Y. Ushiku, N. Tanaka, F. Ono, H. Kameko and S. Mori, *IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2021, pp. 3129–3133.
30 J. Cui, Z. Gong, B. Jia, S. Huang, Z. Zheng, J. Ma and Y. Zhu, *Adv. Neural Inf. Process. Syst.*, 2023, vol. 36.
31 T. Nishimoto, T. Nishimura, K. Yamamoto, K. Shirai, H. Kameko and Y. Haneji, *2025 IEEE International Conference on Image Processing*, ICIP, 2025, pp. 695–700.
32 J. Chen, Y. Jia, Z. Wu, J. Yang, J. Chen, X. Hei, J. Xie, Y. Cai and Q. Li, *arXiv*, 2025, preprint arXiv:2507.09693, DOI: 10.48550/arxiv.2507.09693.
33 I. Qasim, A. Horsch and D. K. Prasad, *arXiv*, 2023, preprint arXiv:2311.02538v1, DOI: 10.48550/arxiv.2311.02538v1.
34 L. Zhou, Y. Zhou, J. J. Corso, R. Socher and C. Xiong, *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2018, **35**, 8739–8748.
35 A. Rohrbach, M. Rohrbach and B. Schiele, *Pattern Recogn.*, 2015, **2015**, 209–221.
36 M. Abdar, M. Kollati, S. Kuraparthi, F. Pourpanah, D. McDuff, M. Ghavamzadeh, S. Yan, A. Mohamed, A. Khosravi, E. Cambria and F. Porikli, *IEEE Trans. Pattern Anal. Mach. Intell.*, 2024, 1–20.
37 S. H. Chou, J. J. Little and L. Sigal, *Comput. Vis. Image Underst.*, 2024, **247**, 104064.
38 Y. Niu, W. Guo, L. Chen, X. Lin and S.-F. Chang, *International Conference on Learning Representations*, ICLR, 2024, https://openreview.net/forum?id=abL5LJNZ49.
39 S. Agrawal and S. Ochoa, NVIDIA Technical Blog, *Vision Language Model Prompt Engineering Guide for Image and Video Understanding*, 2025.
40 Z. Shou, D. Wang and S.-F. Chang, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1049–1058.
41 M. M. Islam, N. Ho, X. Yang, T. Nagarajan, L. Torresani and G. Bertasius, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, CVPR, 2024, pp. 18198–18208.
42 S. Chu, S. Seo and B. Han, *Proceedings of the 42nd International Conference on Machine Learning*, PMLR, 2025, 267, pp. 10801–10817.
43 K. Papineni, S. Roukos, T. Ward and W. J. Zhu, *Proceedings of the 40th Annual Meeting of the*, Association for Computational Linguistics, 2002, pp. 311–318.

44 T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger and Y. Artzi, *International Conference on Learning Representations*, ICLR, 2020, https://openreview.net/forum?id=SkeHuCVFDr.
45 M. Hanna and O. Bojar, *Proceedings of the Sixth Conference on Machine Translation (WMT 2021)*, 2021, pp. 507–517.
46 S. Banerjee and A. Lavie, *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, 2005, pp. 65–72.
47 M. Denkowski and A. Lavie, *Proceedings of the Ninth Workshop on Statistical Machine Translation*, 2014, pp. 376–380.
48 R. Vedantam, C. L. Zitnick and D. Parikh, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4566–4575.
49 Pycocoevalcap, GitHub repository, 2020, available at: https://github.com/salaniz/pycocoevalcap.
50 Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer and V. Stoyanov, *arXiv*, 2019, preprint arXiv:1907.11692, DOI: 10.48550/arxiv.1907.11692.
51 X. Zhou, A. Arnab, S. Buch, S. Yan, A. Myers, X. Xiong, A. Nagrani and C. Schmid, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, PP. 18243–18252.
52 M. Renze and E. Guven, *arXiv*, 2024, preprint arXiv:2405.06682v3, DOI: 10.48550/arxiv.2405.06682v3.
53 D. Zhang, J. Lei, J. Li, X. Wang, Y. Liu, Z. Yang, J. Li, W. Wang, S. Yang, J. Wu, P. Ye, W. Ouyang and D. Zhou, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 9050–9061.
54 V. Belcamino, N. M. D. Le, Q. K. Luu, A. Carfì, V. A. Ho and F. Mastrogiovanni, *arXiv*, 2025, preprint arXiv:2505.08657v1, DOI: 10.48550/arxiv.2505.08657v1.
55 M. B. Shaikh, D. Chai, S. M. S. Islam and N. Akhtar, *IEEE International Conference on Visual Communications and Image Processing*, 2022, PP. 1–5.
56 J.-H. Kim and C. S. Won, *Appl. Sci.*, 2024, **14**, 1190.
57 N. Agarwal, A. Ali, M. Bala, Y. Balaji, E. Barker, T. Cai, P. Chattopadhyay, Y. Chen, Y. Cui, Y. Ding, D. Dworakowski, J. Fan, M. Fenzi, F. Ferroni, S. Fidler, D. Fox, S. Ge, Y. Ge, J. Gu, S. Gururani, E. He, J. Huang, J. Huffman, P. Jannaty, J. Jin, S. W. Kim, G. Klár, G. Lam, S. Lan, L. Leal-Taixe, A. Li, Z. Li, C.-H. Lin, T.-Y. Lin, H. Ling, M.-Y. Liu, X. Liu, A. Luo, Q. Ma, H. Mao, K. Mo, A. Mousavian, S. Nah, S. Niverty, D. Page, D. Paschalidou, Z. Patel, L. Pavao, M. Ramezanali, F. Reda, X. Ren, V. R. N. Sabavat, E. Schmerling, S. Shi, B. Stefaniak, S. Tang, L. Tchapmi, P. Tredak, W.-C. Tseng, J. Varghese, H. Wang, H. Wang, H. Wang, T.-C. Wang, F. Wei, X. Wei, J. Z. Wu, J. Xu, W. Yang, L. Yen-Chen, X. Zeng, Y. Zeng, J. Zhang, Q. Zhang, Y. Zhang, Q. Zhao and A. Zolkowski, *arXiv*, 2025, preprint arXiv:2501.03575v3, DOI: 10.48550/arxiv.2501.03575.
58 Q. Li, Y. Deng, Y. Liang, L. Luo, L. Zhou, C. Yao, L. Zeng, Z. Feng, H. Liang, S. Xu, Y. Zhang, X. Chen, H. Chen, L. Sun, D. Chen, J. Yang and B. Guo, *arXiv*, 2025, preprint arXiv:2510.21571v1, DOI: 10.48550/arxiv.2510.21571.
59 Z. Cheng, Y. Zhang, W. Zhang, H. Li, K. Wang, L. Song and H. Zhang, *arXiv*, 2025, preprint arXiv:2508.08706v2, DOI: 10.48550/arxiv.2508.08706v2.
60 S. Nie, M. Zheng and Q. Ji, *IEEE Signal Process. Mag.*, 2018, **35**, 101–111.