Supplementary Materials: Compositional Physical Reasoning of Objects and Events from Videos

Zhenfang Chen Shilong Dong* Kexin Yi Yunzhu Li Mingyu Ding Antonio Torralba Joshua B. Tenenbaum Chuang Gan*

In this supplementary, we first provide more examples of the datasets in Section 1 and Section 2. We then provide more details on video generation in Section 3. We provide more details about symbolic program we learn in Section 4. We provide more details about the baselines in Section 5 and Section 6. We discuss how to evaluate our models on more diverse simulated scenes in Section 7. We study how to evaluate models on more diverse real scenes in Section 8. Finally, we discuss how to integrate the proposed PCR and large vision-language models in Section 9.

1 Examples from ComPhy

Here we provide more examples from ComPhy in Fig. 16. From these examples, we can see the following features of ComPhy. First, to answer the factual questions, models not only need to recognize objects' visual appearance attributes and events in the video but also identify their intrinsic physical properties from the given video set. Second, to answer counterfactual and predictive questions, models needs to predict objects' dynamics in counterfactual or future scenes, which can be severely affected by intrinsic physical properties. We also show some typical question and choice samples as well as their underlying reasoning program logic in Fig. 18 and Fig. 19.

2 Examples from Real-World Scenario

We also provide some examples captured from real-world scenarios in Fig. 17. Similarly to the procedure of answering questions for the synthetic data in ComPhy, the model needs first to answer the factual questions based on objects' visual attributes and intrinsic physical properties and then answer the counterfactual and predictive questions by predicting the related dynamics. In comparison to ComPhy, the real-world dataset exhibits two distinct characteristics. First, unlike objects with a single charge, magnetic monopoles do not exist in the natural world, which results in each magnetized object within a scene lacking a consistent magnetic label across different videos. This necessitates that models rigorously infer magnetic properties through interactions between objects, avoiding shortcuts based on strong coupling between objects and physical attributes. Second, the real-world dataset is manually collected, so it tends to be noisier, especially in more pronounced interaction instances, such as collisions, attraction, and repulsion between objects. These dynamic behaviors may even cause objects to temporarily leave the ground plane. As a result, robustness becomes a critical requirement for models trained on such datasets. In summary, the real-world dataset serves as a valuable complement to ComPhy, offering diverse challenges and enhancing model performance in handling complex and noisy scenarios.

- *Z. Chen and S. Dong contribute equally.*
- *Z. Chen is with MIT-IBM Watson AI lab. E-mail: zfchenzfc@gmail.com*
- *S. Dong is with New York University. E-mail: shilongdong00@gmail.com*
- *K. Yi is with Harvard University. E-mail: kyi@g.harvard.edu*
- *Y. Li is with UIUC. E-mail: yunzhuli@illinois.edu*
- *M. Ding is with UC Berkeley. E-mail: myding@berkeley.edu*
- *A. Torralba and J. B. Tenenbaum are with MIT. E-mail: {torralba, jbt}@mit.edu*
- *C. Gan is with MIT-IBM Watson AI lab and UMass Amherst. E-mail: ganchuang1990@gmail.com*

3 Video Generation

We provide more details for video generation. The generation of the videos in ComPhy can be decomposed into two steps. First, we adopt a physical engine Bullet [1] to simulate objects' motions and their interactions with each other. Since Bullet does not officially support the effect of electronic charges, we add external forces between charged objects, whose values are inversely proportional to the square of the objects' distance, to simulated Coulomb forces. We assign the *light* object with a mass value of 1 and assign *heavy* object with a mass value of 5. We manually make sure that each reference video at least contains an interaction (collision, charge, and mass) among objects to provide enough information for physical property inference. Each object should appear at least once in the reference videos. The simulated objects' motions are sent to Blender [2] to render high-quality image sequences.

4 Symbolic Program Details

The symbolic execution component first adopts a program parser to parse the query question into a functional program, containing a series of neural operations. The program parser is an attention-based seq2seq model [3], whose input is the word sequence in the question/choice and output is the sequence of neural operations. The symbolic executor then executes the operations on the predicted dynamic scene to get the answer to the question. We summarize all the symbolic operations in CPL in table 1. Compared with the previous benchmarks [4], [5], ComPhy has more operation on physical property identification, comparison and corresponding dynamic prediction. We show each symbolic operator in table 1.

5 Baseline Implementation Details

In this section, we provide more details for baselines in the experimental section. We implement baselines based on the publicly available source code. For multiple-choice questions, we independently concatenate the words of each option and the question as a binary classification question. Similar to CLEVRER [4], we use ResNet-50 [6] to extract visual feature sequences for **CNN+LSTM** and **MAC** and variants with reference videos. We evenly sample 25 frames for each target video and 10 frames for each reference video. For **HCRN**, we use the appearance feature from ResNet-101 [6] and the motion feature from ResNetXt-101 [7], [8] following the official implementation. For **ALOE**, we use MONet [9] to extract visual representation and sample 25 frames for each target video. For **ALOE (Ref)**, we sample 10 frames for each reference video and concatenate the reference frames and the target frames as visual representations. We train all the models until they are fully converged, select the best checkpoint on the validation set and finally test on the testing set.

6 Large Vision Language Models Details

In this section, we provide more details on how we utilize Large Vision Language Models, such as [10], [11], [12] to test their physical reasoning ability on ComPhy. For **ALPRO** [10], we fine-tune the model with both factual, counterfactual, and predictive questions in ComPhy's training set until they achieve satisfactory results on the validation set. We convert both open-ended and multiple-choice question formats to align the input of the model. For open-ended questions, we simply collect the answers to build the vocabulary dictionary. For multiple-choice questions, we assemble each choice with its question to form a new question and utilize the original True/False judgment as the answer. Due to the large variance between open-ended and multiple-choice questions' answer domains, we fine-tune the model separately on the two different types of questions. For **GPT-4V** [11] and **Gemini** [12], we leverage a zero-shot method to test their performance. We evenly sample 16 frames from each target video to form a sequence of frames to represent the original video in the test set and pair the sequence with related questions from the dataset. Then, we add an instructive prompt to guide the model in understanding the physical events that happened in the scenarios and answer the questions in a predefined format.

7 Evaluate Models on More Diverse Simulated Scenes

Goal of Our benchmark. We would like to clarify that the original goal of ComPhy is not to mimic complex real-world scenes, but rather to **provide a diagnostic testbed that isolates and evaluates the physical reasoning capabilities of AI models.** Simplicity in object design and scene setup allows for controlled physical interactions, making it easier to attribute model behavior to underlying reasoning mechanisms. However, we also agree that greater diversity can improve robustness evaluation and broaden the benchmark's applicability.

More Diverse Physical Simulated Scenes. To provide more diverse physical reasoning, we have significantly expanded the dataset to create a new version, ComPhy-DIV. This version introduces 13 distinct object categories—including items such as mugs, pots, chairs, and more—in contrast to the primitive shapes used in the original benchmark. In addition, we incorporate 9 varied backgrounds with realistic textures and lighting conditions, and increase the total number of possible question-answer pairs to 175. Note that there are only 3 primitive shapes in the same background in the original dataset. As shown in Figure 1, the new objects span a wider range of shapes and material properties. These enhancements allow for a richer set of physical interactions, enabling the simulation of complex, compositional events. Qualitative examples of these new scenes are presented in Fig. 2 and Fig. 10–12, which demonstrate diverse object movements, interactions, and backgrounds.

New Experimental Results on the Simulated Scenes. To evaluate the effectiveness of ComPhy-DIV, we conducted new experiments with both our proposed method and baseline models. Results are summarized in Table 2. Our model (PCR) continues to outperform baseline methods, indicating its superior reasoning ability even in the presence of increased visual and physical complexity. Notably, the overall performance of all models has declined compared to results on the original dataset (see Table 3 and Table 5 in the main paper), which confirms that the added diversity makes the benchmark more challenging and discriminative. Additionally, we conducted a human study following the same protocol used in the original ComPhy paper. Human participants achieved accuracies of 88.6% for factual questions, 73.7% for predictive questions, and 78.9% for counterfactual questions—substantially higher than those of AI models—demonstrating that despite increased complexity, humans remain robust and reliable at these reasoning tasks.

8 Evaluate Models on More Diverse Real-World Scenes

Enhanced Diversity of Real Physical Scenes. To evaluate models on more diverse real physical scenes, we significantly expanded the variety and complexity of real-world scenes in our revised dataset, ComPhy-REAL. Specifically, we increased the object count from the original three to six distinct real-world objects, each varying significantly in shape and appearance, as illustrated in Figure 3. Additionally, we manually altered the surface colors of these objects by applying different paint colors, thus further diversifying

Type	Operation	Signature
Counterfact Operation	`Counterfactual_mass_heavy` Return all events after making the object heavy	$(object) \rightarrow events$
	`Counterfactual_mass_light` Return all events after making the object light	$(object) \rightarrow events$
	`Counterfactual_uncharged` Return all events after making the object uncharged	$(object) \rightarrow events$
	`Counterfactual_opposite_charged` Return all events after making the object oppositely charged	$(object) \rightarrow events$
Object Property Operations	`filter_heavy` select all the heavy objects	$(objects) \rightarrow objects$
	`filter_light` select all the light objects	$(objects) \rightarrow objects$
	`filter_charged` select all the charged objects	$(objects) \rightarrow objects$
	`filter_uncharged` select all the uncharged objects	$(objects) \rightarrow objects$
Object Appearance Operations	`Filter_static_attr` Select objects from the input list with the input static attribute	$(objects, attr) \rightarrow objects$
	`Filter_dynamic_attr` Selects objects in the input frame with the dynamic attribute	$(objects, attr, frame) \rightarrow objects$
Event Operations	`Filter_event` Select all events that involve the input objects	$(events, objects) \rightarrow events$
	`Get_col_partner` Return the collision partner of the input object	$(event, object) \rightarrow object$
	`Filter_before` Select all events before the target event	$(events, events) \rightarrow events$
	`Filter_after` Select all events after the target event	$(events, events) \rightarrow events$
	`Filter_order` Select the event at the specific time order	$(events, order) \rightarrow event$
	`Get_frame` Return the frame of the input event in the video	$(event) \rightarrow frame$
Others	`Unique` Return the only event/object in the input list	$(events/objects) \rightarrow event/object$
Input Operations	`Start` Returns the special "start" event	$() \rightarrow event$
	`end` Returns the special "end" event	$() \rightarrow event$
	`Objects` Returns all objects in the video	$() \rightarrow objects$
	`Events` Returns all events happening in the video	$() \rightarrow events$
	`UnseenEvents` Returns all future events happening in the video	$() \rightarrow events$
Output Operations	`Query_both_attribute` Returns the attributes of the input two objects	$(object, object) \rightarrow attr$
	`Query_direction` Returns the direction of the object at the input frame	$(object, frame) \rightarrow attr$
	`Is_heavier` Returns "yes" if *obj1* is heavier than *obj2*	$(obj1, obj2) \rightarrow bool$
	`Is_lighter` Returns "yes" if *obj1* is lighter than *obj2*	$(obj1, obj2) \rightarrow bool$
	`Query_attribute` Returns the attribute of the input objects like color	$(object) \rightarrow attr$
	`Count` Returns the number of the input objects/ events	$(objects) \rightarrow int$ $(events) \rightarrow int$
	`Exist` Returns "yes" if the input objects is not empty	$(objects) \rightarrow bool$
	`Belong_to` Returns "yes" if the input event belongs to the input event sets	$(event, events) \rightarrow bool$
	`Negate` Returns the negation of the input boolean	$(bool) \rightarrow bool$

TABLE 1: Symbolic operations of PCR on ComPhy. In this table, "order" denotes the chronological order of an event, e.g. "First" and "Last"; "static attribute" denotes object static concepts like "Red" and "Rubber" and "dynamic attribute" represents object dynamic concepts like "Moving".

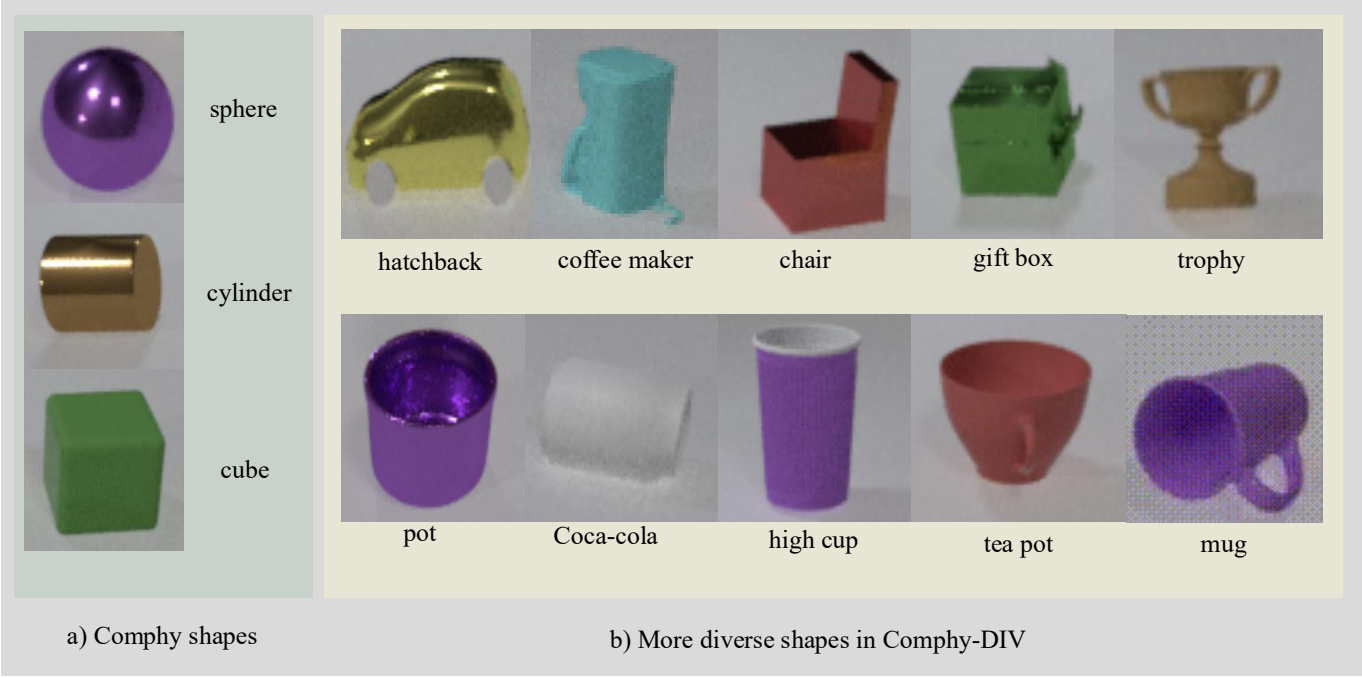


Fig. 1: Comparison of shape diversity between ComPhy and ComPhy-DIV. As shown in a), the three objects belong to the ComPhy dataset, whereas b) illustrates the ten newly added objects in ComPhy-DIV.

Methods	Factual	Predictive		Counterfactual	
		per opt.	per ques.	per opt.	per ques.
Random	1.8	50.1	22.9	48.1	24.0
Frequent	15.7	50.0	0.0	50.0	0.0
Blind-LSTM	43.2	50.3	25.0	49.2	23.2
CNN-LSTM [14]	49.6	52.8	29.9	55.7	29.7
HCRN [15]	51.5	**56.3**	**34.1**	51.9	30.1
MAC [16]	**51.7**	50.4	28.9	51.9	26.3
ALOE [17]	46.9	52.4	29.0	51.5	28.6
CNN-LSTM (Ref) [14]	49.7	51.4	23.3	55.6	30.5
MAC (Ref) [16]	50.6	51.9	33.3	50.8	25.2
ALOE (Ref) [17]	48.6	51.2	26.1	52.9	27.2
ALPRO [10]	47.1	51.8	28.9	52.6	28.4
GPT-4o-mini [11]	42.5	50.0	29.2	**58.8**	**30.7**
Gemini [12]	34.2	50.3	25.7	49.4	30.6
PCR (ours)	**68.4**	**58.3**	**34.9**	**60.3**	**32.8**
Human Performance	88.6	82.9	73.7	88.2	78.9

TABLE 2: Evaluation of physical reasoning on ComPhy-DIV. Human performance is based on sampled questions. See the text for more details. **Red** text and **blue** text indicate the first and second best results other than human performance.

their visual appearances. To enrich the visual context, we applied object matting techniques to place these objects onto nine different realistic backgrounds featuring varied textures and lighting conditions.

We acknowledge that collecting real-world data involves **substantial manual effort**, including carefully painting objects, initializing their positions and velocities, precisely segmenting objects from videos, and replacing backgrounds through matting. As a result of these efforts, our enhanced real-world dataset now comprises **123** distinct scene sets, yielding a total of **492** unique real-world videos. Figure 4 provides representative qualitative examples of these more diverse and realistic scenes, highlighting intricate physical interactions such as collisions and attraction events among multiple objects. Additional examples are presented in Figures 13–15.

New Experimental Results on Enhanced Real Scenes. To validate the increased complexity and diversity, we conducted extensive experiments using these newly collected real-world scenes. As reported in Table 3, our proposed model (PCR) consistently outperforms all baseline methods, demonstrating robustness and strong physical reasoning capabilities even when confronted with diverse and realistic data. Furthermore, we conducted an additional human evaluation study on this expanded dataset, revealing that human participants still achieve high accuracy, underscoring that although the dataset presents notable challenges for AI, it remains intuitive and manageable for humans.

9 Discussion on Integrating PCR with LVLMs

We argue that it is quite promising to combine neuro-symbolic models like our PCR that learns neural modules for specific functions directly from the training question-answer pairs and the general capability of LVLMs. We think that LVLMs can at least help with the following aspects of the PCR framework, (1) improving the robustness of the language parsing capabilities; (2) enabling challenging commonsense reasoning that combines the outside knowledge from LVLMs and domain-specific knowledge; and (3) han-

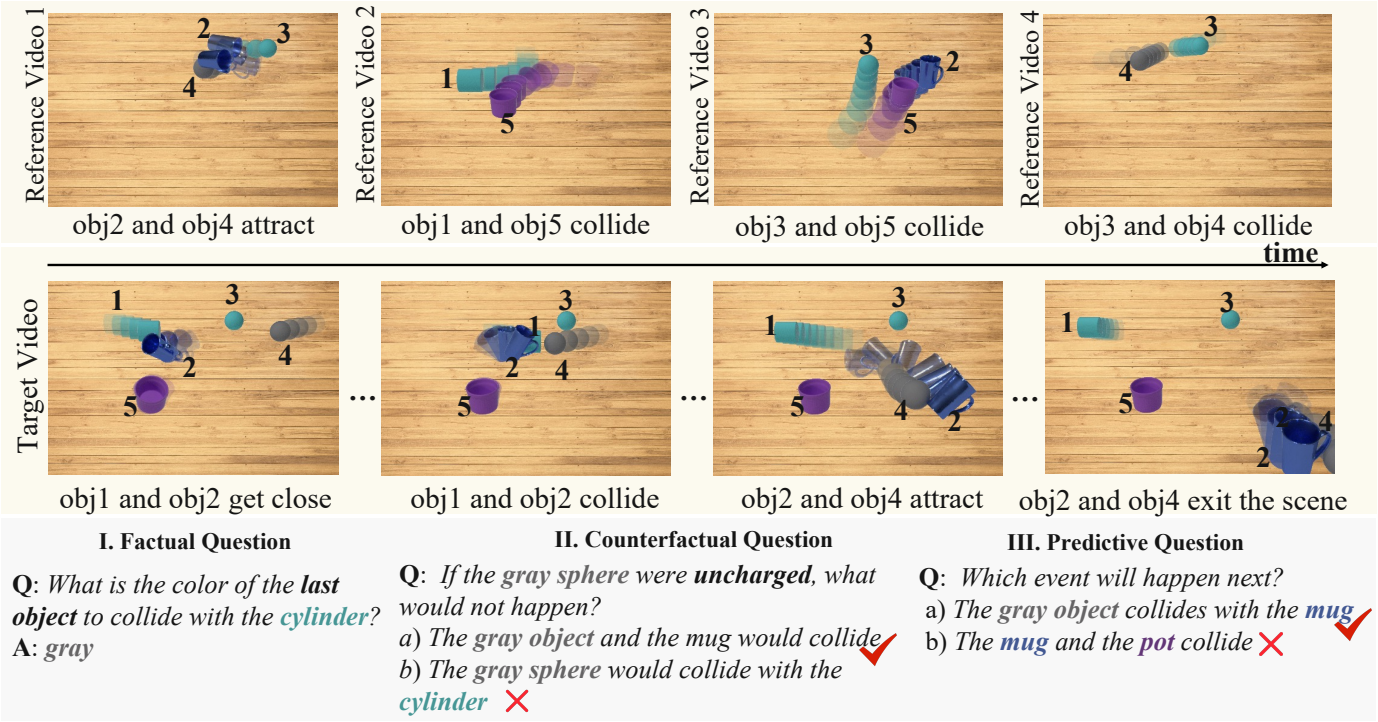


Fig. 2: Qualitative examples of more diverse scenes in the ComPhy-DIV. As shown in the figure, we have more diverse physical interactions between the blue mug and the sphere in the video. The image background is also more diverse with different textures and colors in contrast to the original ComPhy in [13].

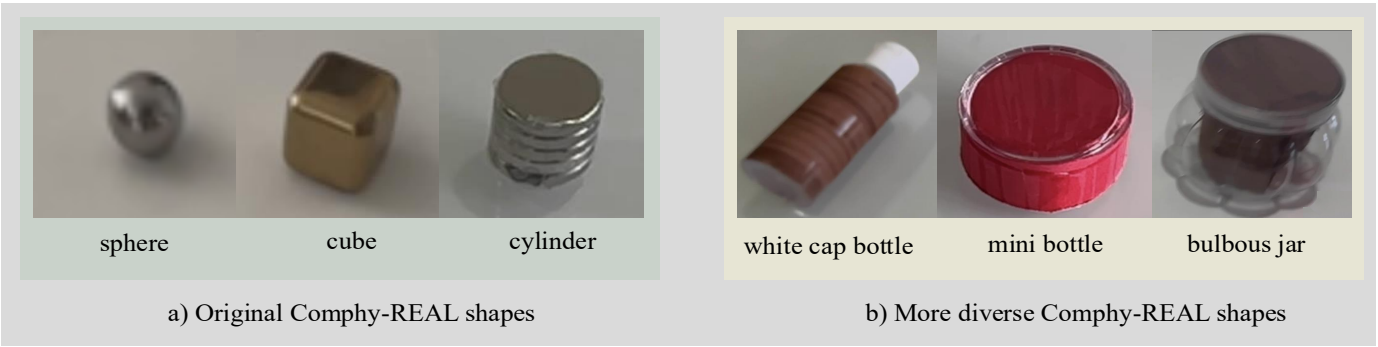


Fig. 3: Comparison of shape diversity between the original and extended ComPhy-REAL. As shown in a), the three objects belong to the original dataset, whereas we added three more diverse ones in b) to ComPhy-REAL.

dling new tasks by cooperating with pre-trained modules and learned modules.

(1). Improving Language Parsing Capabilities. To improve the AI systems' capability to handle understand the language query, we can replace the language parser [4], [18] with shallow two-layer Seq2seq LSTMs [19] with the LVLMs. One limitation for the previous shallow language parser is that it shows its limitations when transforming the language instructions with new format into execuable programs. And the capture the semantics of language is quite easy for LVLMs. Thus, we can use in-context learning to transform any language instructions into executable programs. To evaluate this capability by combining LVLMs and PCR, we first generate a new test set that contains much more diverse language instructions for the tasks in ComPhy. Specifically, we follow a generate-verify strategy to synthesize diverse language instructions. We first use `Qwen/Qwen2.5-72B-Instruct-AWQ` [20] to paraphrase the questions in ComPhy and generate questions with diverse formats but keeps the same meaning of the original questions. We then ask the LLM to verify that the revised new question has the same semantic meaning as the original ones and abandon those questions without the same meaning. Sample questions are shown in Table 4 and the results of using LLMs to parse the question can be seen in Table 5. From Table 5, we can see that LVLMs can parse the language instruction into the programs better much better than the original program parser [4], [18]. To provide a quantitative evaluation, we revsie the questions from the validation set of ComPhy and evaluate the perofrmance of the original PCR and PCR + LVLMs. To relieve the API cost, we use the Qwen(`Qwen/Qwen2.5-72B-Instruct-AWQ`) to serve

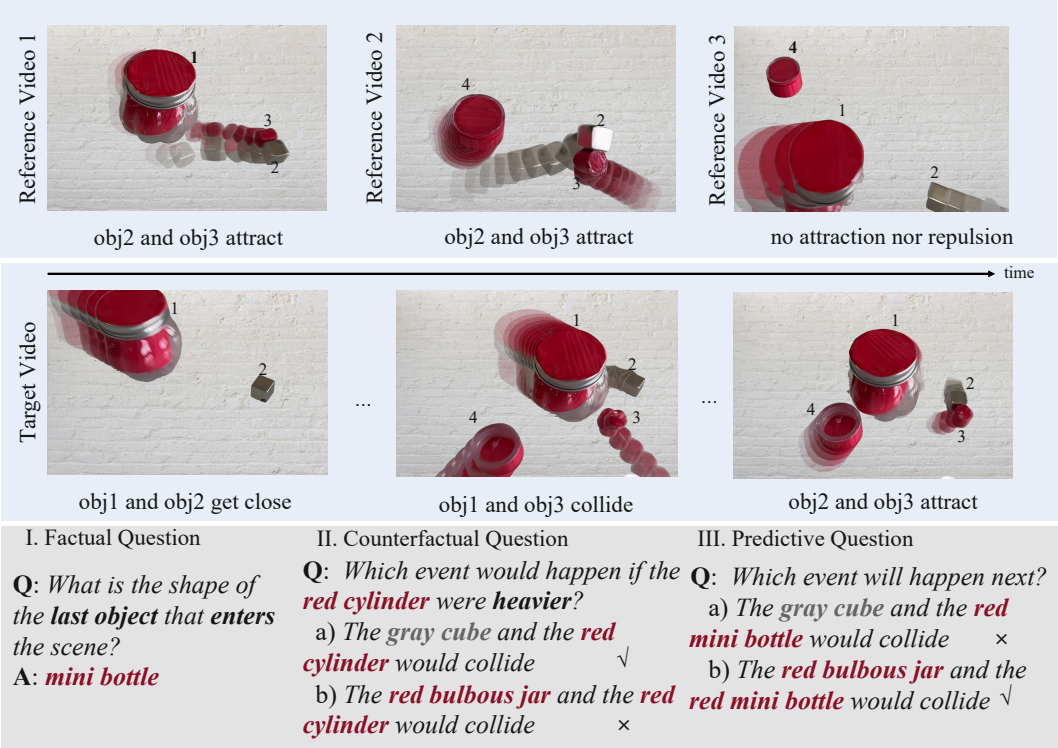


Fig. 4: Qualitative examples of more diverse scenes in the ComPhy-REAL. As shown in the figure, we have more diverse physical interactions between objects in the video. The image background is also more diverse with different textures and colors in contrast to the original ComPhy in [13].

as an alternative of LVLMs to parse the programs. The results are shown in Table 4. We found that although both models still work on this revised set. Combining Qwen for robust program parsing,it performs much better on all types of questions from the dataset.

(2). Enabling New Commonsense Reasoning Capabilities. By cooperating the PCR with LVLMs, we are able to answer questions that requires commonsense knowledge that does not exist in the original PCR's training set. For example, as shown in Figure 5, when we ask the model PCR +LVLMs the question, *"If you stacked the gray object on the first object gets out of the video, would the structure be stable?"*, the LVLM (specially, GPT4-o in this example) is able to write a program in Python (Figure 6) that calls the reasoning modules (`get_color` and `filter_out`) in PCR and (`llm_query`) from LVLMs to handle the problem and provides the correct answer with explanation (*"No, it will not be stable to stack a cube on a sphere. The cube will not have a flat surface to rest on and will likely roll off the sphere."*). Note that either PCR or GPT4-o alone is not able to solve this task. PCR can not transform such an out-of-domain question query into executable python program (Figure 6) and does not have the commonsense to know the outcome of stacking a cube on a sphere. When adopting GPT4-o alone, we can not distinguish the fine-grained details in the video and might miss the frame where the first object that gets out of the scene from only a few frames.

(3). Handling New Tasks with Modules beyond PCR and ComPhy. Another benefit of LVLMs is that it can be used as a controller to control both the modules in PCR and other modules that are learned from other datasets and tasks. As shown in Figure 7, Figure 8 and Figure 9, we show how we can achieve the goal of fine-grained video editing by combining PCR and LVLMs. The LVLM first parses the question into an executable python program (Figure 8) that calls neural modules from PCR (`get_color`) to identify the target object and adopts the existing diffusion model module [21] (`edit_objects`) to perform fine-grained edits.

REFERENCES

[1] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning," http://pybullet.org, 2016–2021. 1

[2] B. O. Community, "Blender - a 3d modelling and rendering package," Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. [Online]. Available: http://www.blender.org 1

[3] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *ICLR*, 2015. 2

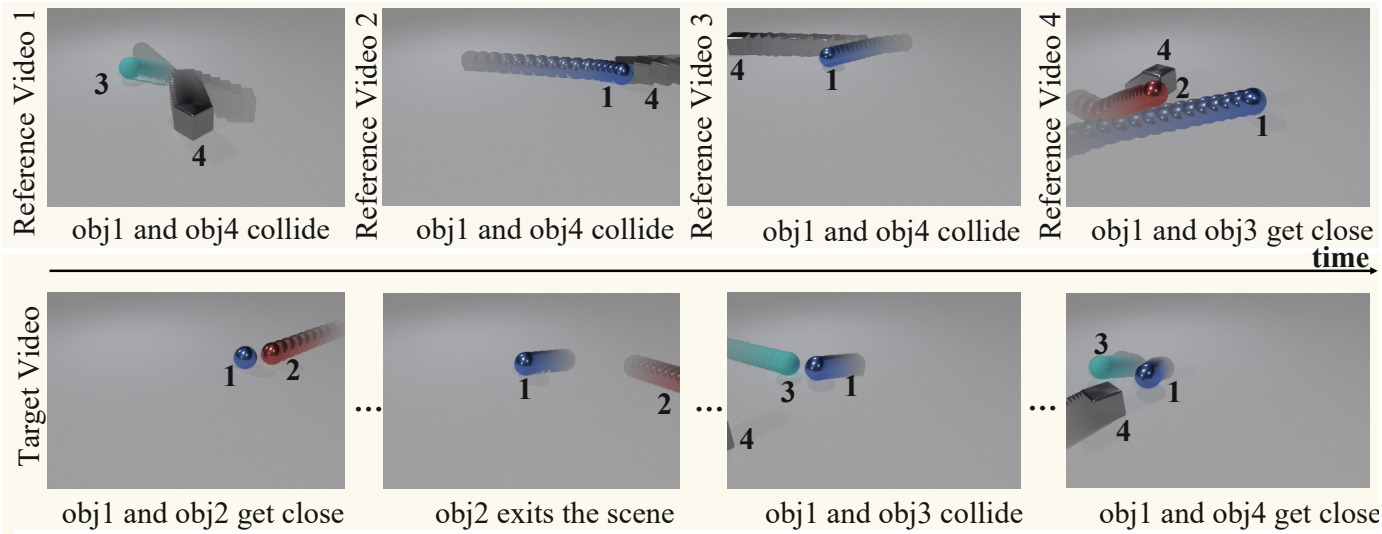
[4] K. Yi, C. Gan, Y. Li, P. Kohli, J. Wu, A. Torralba, and J. B. Tenenbaum, "Clevrer: Collision events for video representation and reasoning," in *International Conference on Learning Representations*, 2020. 2, 5, 8

[5] T. Ates, M. S. Atesoglu, C. Yigit, I. Kesen, M. Kobas, E. Erdem, A. Erdem, T. Goksun, and D. Yuret, "Craft: A benchmark for causal reasoning about forces and interactions," *arXiv preprint arXiv:2012.04293*, 2020. 2

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016. 2

[7] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *CVPR*, 2017. 2

[8] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?" in *CVPR*, 2018. 2



Question: If you stacked the gray object on the first object that gets out of the video, would the structure be stable?

Answer: No, it will not be stable to stack a cube on a sphere. The cube will not have a flat surface to rest on and will likely roll off the sphere.

Fig. 5: An example of combining the strength of PCR and LVLMs to enable new commonsense reasoning capabilities. The LVLM is able to write a program in Python (Figure 6) that calls the reasoning modules (`get_color` and `filter_out`) in PCR and (`llm_query`) from LVLMs to handle the problem and provides the correct answer with explanation.

```python
def execute_command(video, possible_answers, query, ImagePatch,
            VideoSegment, llm_query, bool_to_yesno, distance, best_image_match):
    video_segment = VideoSegment(video)
    num_objects = video_segment.count_objects()
    # Find the first object that gets out of the scene
    out_list = []
    for idx in range(num_objects):
        out_frm = video_segment.filter_out(idx)
        if out_frm is not None:
            out_list.append([idx, out_frm])
    if len(out_list) == 0:
        return "There is no object that exits the scene"
    out_list = sorted(out_list, key=lambda x: x[1])
    first_out_idx = out_list[0][0]
    # Find the gray object
    gray_obj_idx = None
    for idx in range(num_objects):
        color = video_segment.get_color(idx)
        if color == 'gray':
            gray_obj_idx = idx
            break
    if gray_obj_idx is None:
        return "There is no gray object in the video"
    # Get shapes of the gray object and the first object that gets out
    gray_shape = video_segment.get_shape(gray_obj_idx)
    first_out_shape = video_segment.get_shape(first_out_idx)
    # Use llm_query to determine stability
    answer = llm_query(f"Will it be stable to stack a {gray_shape} on
                a {first_out_shape}?")
    return answer
```

Fig. 6: The program that the LVLM generates to handle the query in Figure 5. The program first calls the modules (`get_color` and `filter_out`) in PCR to identify the object 4 and the object 2 in the video. The program that calls the `get_shape` module in PCR to get the objects' shape and finally sends the LVLM a question based on the shape to identify the stability of the structure and gives the explanation (the **answer** in Figure 5).

Categories	Methods	Factual	Predictive		Counterfactual	
			per opt.	per ques.	per opt.	per ques.
Bias analysis models	Random	7.6	50.0	25.0	50.9	20.8
	Frequent	41.7	53.6	28.7	50.0	23.9
	Blind-LSTM	50.6	61.5	46.0	51.9	32.2
video question answering models	CNN-LSTM [14]	55.6	64.2	47.3	50.9	33.3
	HCRN [15]	51.9	62.5	53.5	50.9	32.1
Compositional reasoning models	MAC [16]	58.9	60.9	**57.1**	**52.8**	**35.8**
	ALOE [17]	60.8	60.6	42.4	47.1	28.7
Models with Reference Videos	CNN-LSTM (Ref) [14]	49.0	**64.3**	41.3	50.0	26.3
	MAC (Ref) [16]	56.4	56.2	46.4	51.4	34.9
	ALOE (Ref) [17]	**61.6**	61.4	42.8	51.6	32.1
Large Vision Language Models	ALPRO [10]	50.9	55.3	39.2	49.7	29.2
	GPT-4o-mini [11]	42.6	49.6	23.2	47.5	26.0
	Gemini [12]	32.5	57.7	23.1	52.1	29.8
	PCR(ours)	**63.5**	**70.4**	**62.7**	**54.6**	**36.5**
	Human Performance	90.0	95.0	90.0	94.4	88.9

TABLE 3: Evaluation of physical reasoning on ComPhy-REAL. Human performance is based on sampled questions. See the text for more details. **Red** text and **blue** text indicate the first and the second best results other than human performance.

Original Question 1	If the cyan sphere were heavier, what would not happen?
Revised Question 1	What would not occur if the cyan sphere had more weight? If the cyan sphere had more mass, what outcome would be impossible? What would not occur if the cyan sphere were to have a greater weight? What would not occur if the cyan sphere had more weight?
Original Question 2	What will happen next?
Revised Question 2	What is the next event that will take place? What is likely to happen next? What is the next event that will occur? What is expected to happen next?

TABLE 4: Examples of revised questions that preserve the original semantics while exhibiting greater linguistic diversity and flexibility. These variations challenge language parsers [4], [18] by introducing textual patterns not encountered during training.

Question	What would be the outcome if the sphere had a greater mass?
PCR's parser	all_events, objects, sphere, filter_shape, unique, counterfact_uncharged, filter_counterfact, belong_to, not
LVLMs	all_events, objects, sphere, filter_shape, unique, counterfact_heavier, filter_counterfact, belong_to
Question	What color is the metal sphere that remains stationary at the start of the video?
PCR's parser	objects, metal, filter_material, sphere, filter_shape, filter_end, query_frame, filter_stationary, query_color
LVLMs	objects, metal, filter_material, sphere, filter_shape, filter_start, query_frame, filter_stationary, query_color

TABLE 5: Comparison of parsing results between PCR 's program parser and LVLMs. PCR 's parser fails on the revised questions due to distribution shift from its original training set, whereas LVLMs succeed thanks to superior generalization. Key operators are highlighted in red.

[9] C. P. Burgess, L. Matthey, N. Watters, R. Kabra, I. Higgins, M. Botvinick, and A. Lerchner, "Monet: Unsupervised scene decomposition and representation," *arXiv preprint arXiv:1901.11390*, 2019. 2

[10] D. Li, J. Li, H. Li, J. C. Niebles, and S. C. Hoi, "Align and prompt: Video-and-language pre-training with entity prompts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4953–4963. 2, 4, 8

[11] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv*, 2023. 2, 4, 8

[12] G. Team, R. Anil, S. Borgeaud, Y. Wu, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth *et al.*, "Gemini: a family of highly capable multimodal models," *arXiv*, 2023. 2, 4, 8

[13] Z. Chen, K. Yi, Y. Li, M. Ding, A. Torralba, J. B. Tenenbaum, and C. Gan, "Comphy: Compositional physical reasoning of objects and events from videos," in *International Conference on Learning Representations*. 5, 6

[14] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh, "Vqa: Visual question answering," in *ICCV*, 2015. 4, 8

[15] T. M. Le, V. Le, S. Venkatesh, and T. Tran, "Hierarchical conditional relation networks for video question answering," in *CVPR*, 2020. 4, 8

[16] D. A. Hudson and C. D. Manning, "Compositional attention networks for machine reasoning," in *ICLR*, 2018. 4, 8

[17] D. Ding, F. Hill, A. Santoro, and M. Botvinick, "Attention over learned object embeddings enables complex visual reasoning," *arXiv*, 2020. 4, 8

[18] Z. Chen, J. Mao, J. Wu, K.-Y. K. Wong, J. B. Tenenbaum, and C. Gan, "Grounding physical concepts of objects and events through dynamic visual reasoning," in *International Conference on Learning Representations*, 2021. 5, 8

[19] S. Hochreiter and J. Schmidhuber, "Long short-term memory,"

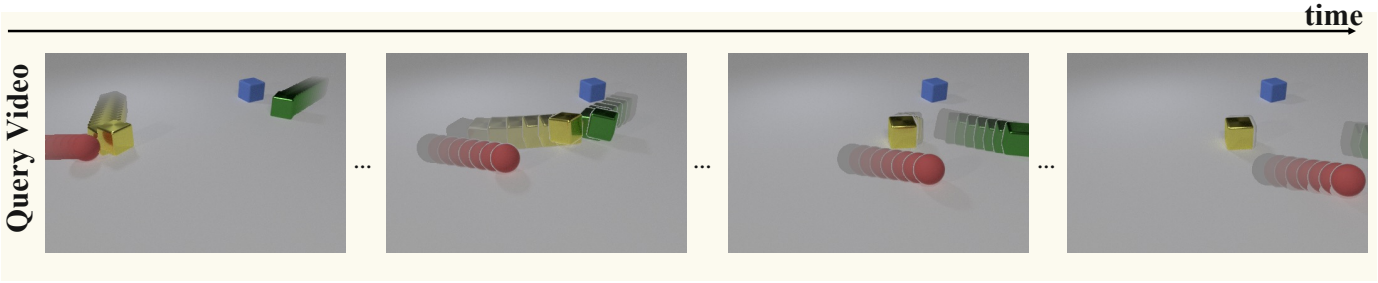
Methods	Factual	Predictive		Counterfactual	
		per opt.	per ques.	per opt.	per ques.
PCR	51.9	60.6	44.2	57.4	41.5
PCR+LVLMs	69.7	73.5	54.0	76.3	60.9

TABLE 6: Performance Comparison of PCR and PCR+LVLMs on revised questions, where PCR fails under a distribution shift from its training set, whereas PCR+LVLMs succeed thanks to superior generalization.

Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997. 5

[20] A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, H. Lin, J. Yang, J. Tu, J. Zhang, J. Yang, J. Yang, J. Zhou, J. Lin, K. Dang, K. Lu, K. Bao, K. Yang, L. Yu, M. Li, M. Xue, P. Zhang, Q. Zhu, R. Men, R. Lin, T. Li, T. Xia, X. Ren, X. Ren, Y. Fan, Y. Su, Y. Zhang, Y. Wan, Y. Liu, Z. Cui, Z. Zhang, and Z. Qiu, "Qwen2.5 technical report," *arXiv*, 2024. 5

[21] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *CVPR*, 2022. 6, 10



Instruction: Could you please replace the red object in the video with a golf ball?

Fig. 7: An input video and a language instruction to ask AI models to edit the target video.

```python
def execute_command(video) -> list:
    video_segment = VideoSegment(video)
    num_objects = video_segment.count_objects()
    red_obj_idx = None
    for idx in range(num_objects):
        color = video_segment.get_color(idx)
        if color == 'red':
            red_obj_idx = idx
            break
    if red_obj_idx is not None:
        edited_images = video_segment.edit_objects(red_obj_idx, "golf ball")
        return edited_images
    return []
```

Fig. 8: Program generated by the LVLM to address the query in Figure 7. It first uses the `get_color` module to locate the red object, then applies the Stable Diffusion–based `edit_objects` module to transform that region into a *golf ball*. Integrating PCR with LVLMs empowers the model to incorporate new modules for new tasks such as video editing.

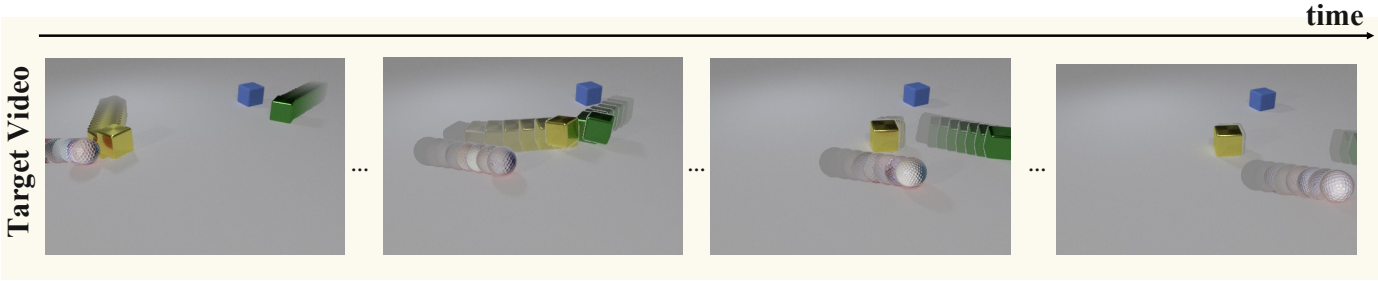


Fig. 9: The output video of replacing the red object with a **golf ball** by calling the new stable diffusion module (`edit_object`) [21] to edit the target object region.

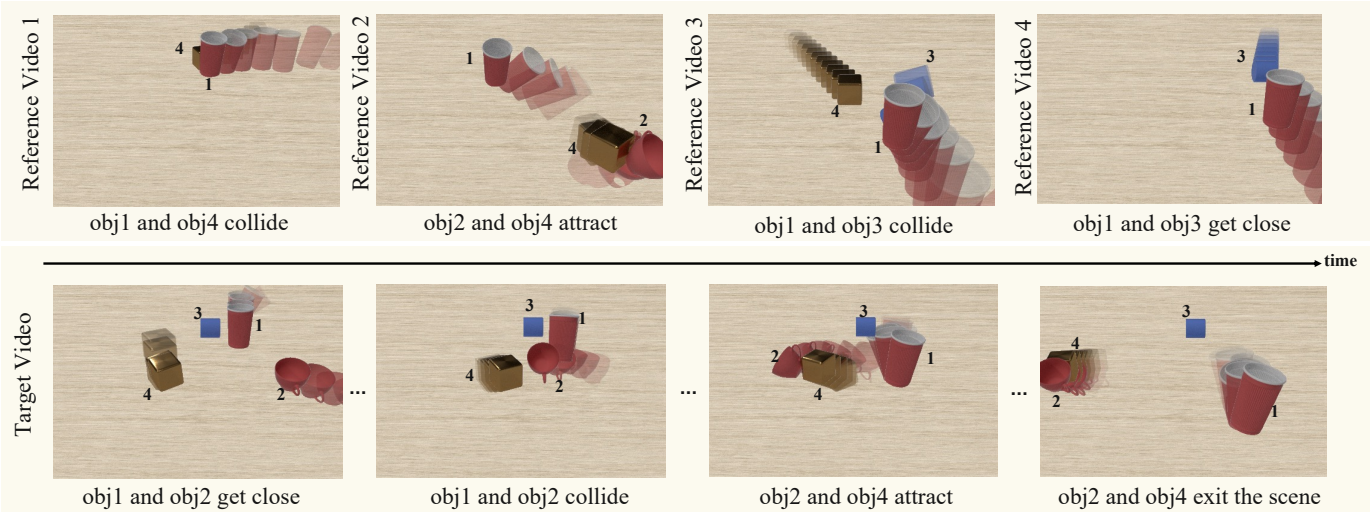


Fig. 10: More qualitative examples of more diverse scenes in the ComPhy-DIV.

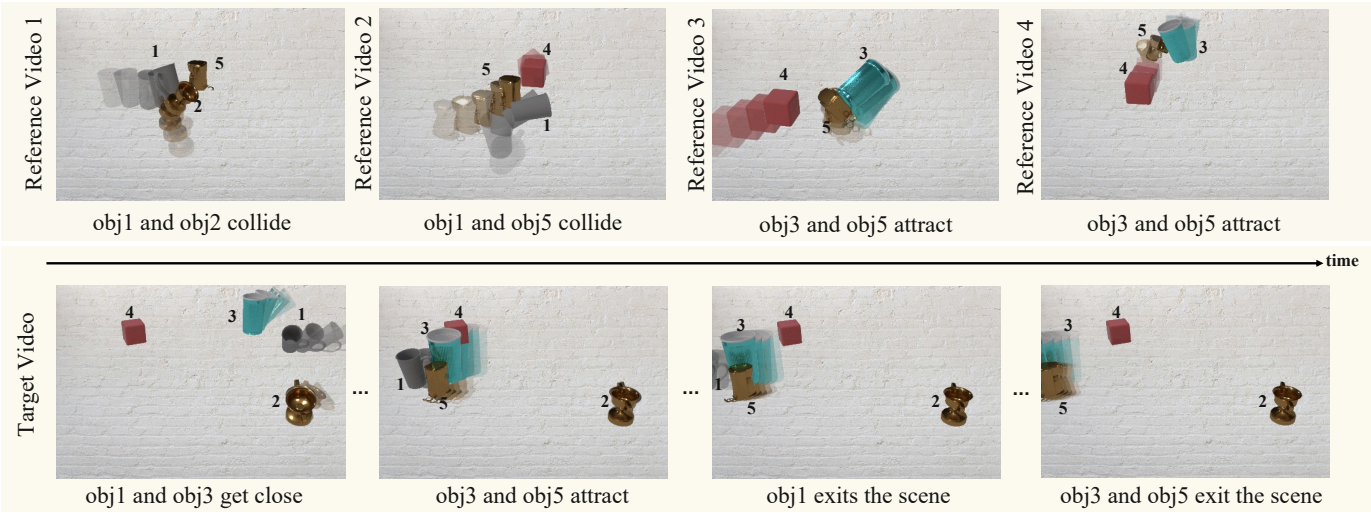


Fig. 11: More qualitative examples of more diverse scenes in the ComPhy-DIV.

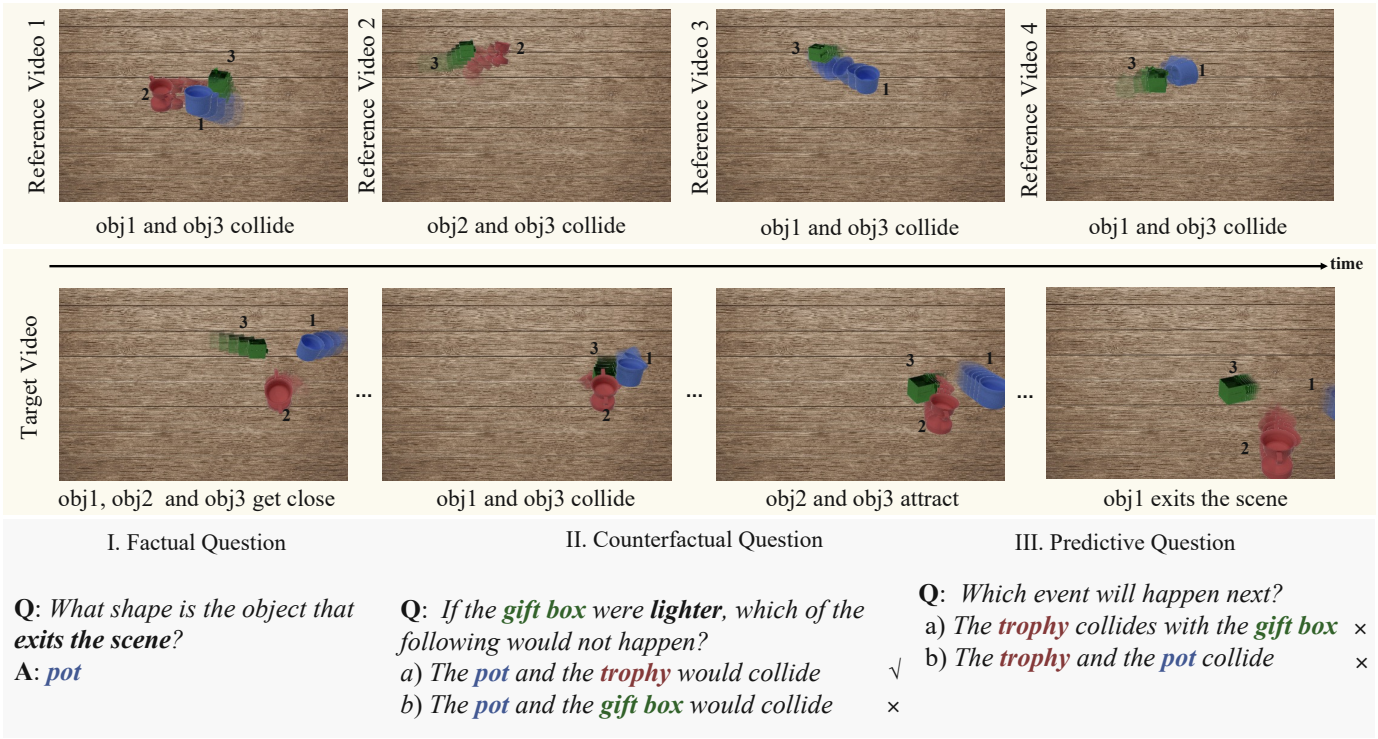


Fig. 12: More qualitative examples of more diverse scenes in the ComPhy-DIV.

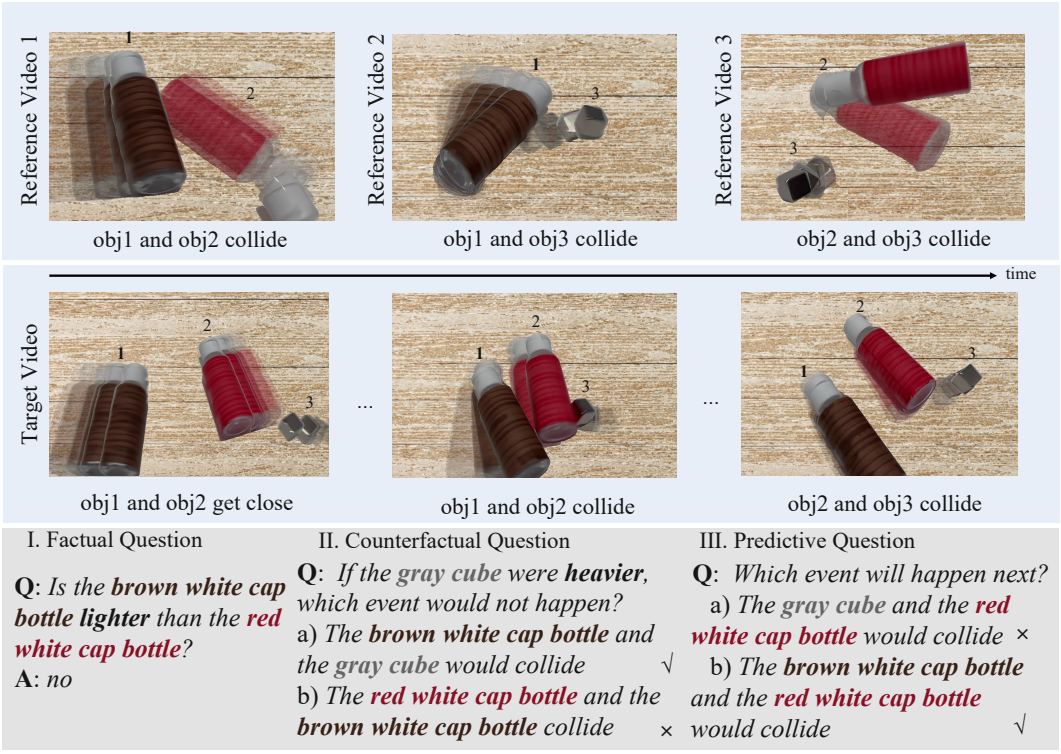


Fig. 13: More qualitative examples of more diverse scenes in the ComPhy-REAL.

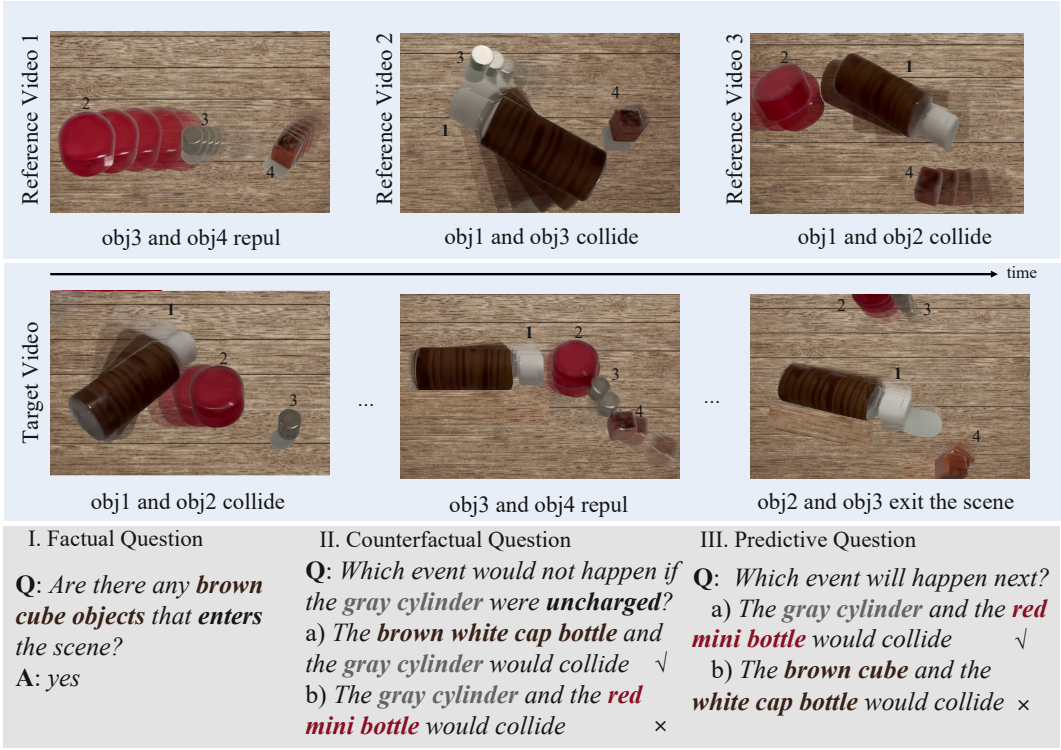


Fig. 14: More qualitative examples of more diverse scenes in the ComPhy-REAL.

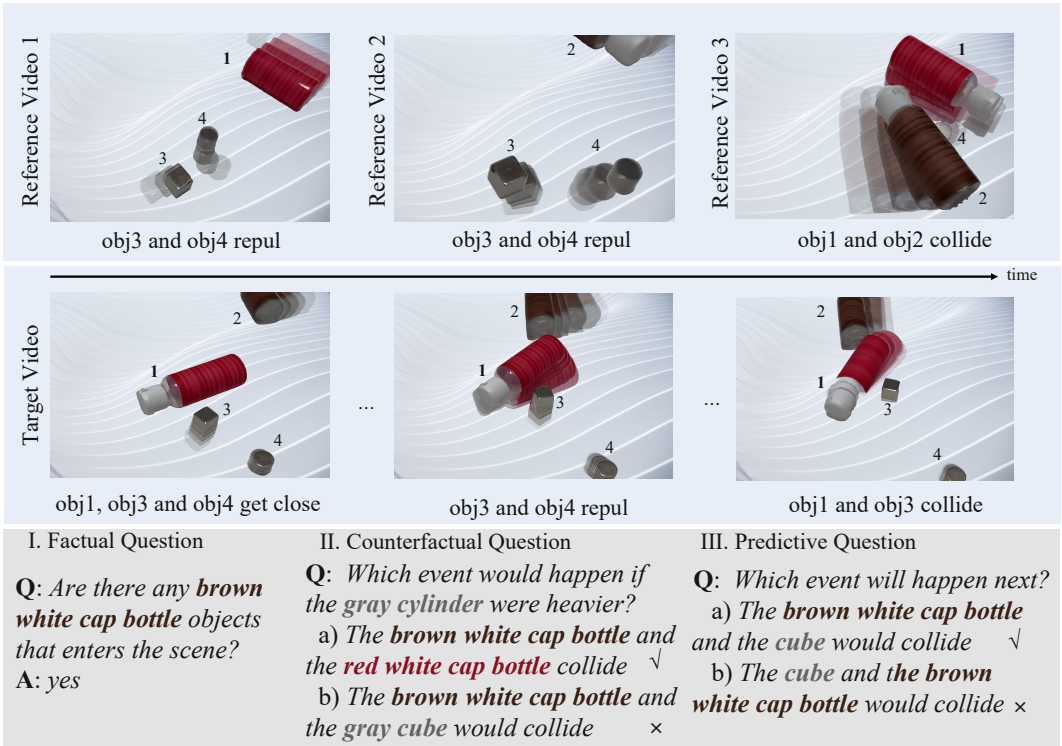
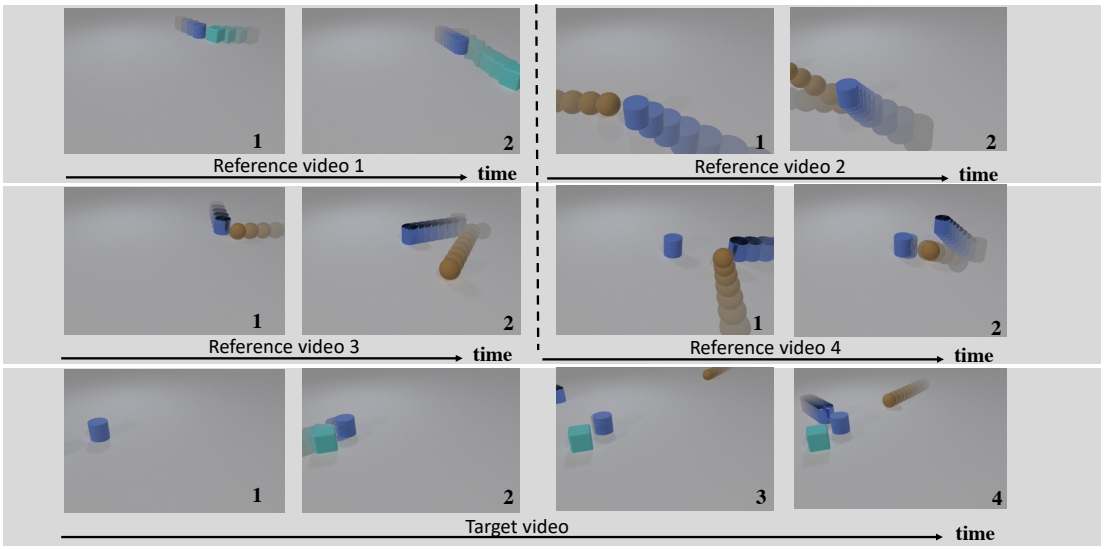


Fig. 15: More qualitative examples of more diverse scenes in the ComPhy-REAL.



I. Factual

Q1: Is the cyan cube heavier than the rubber cylinder? A: No.
Q2: Are there any blue cylinders that enter the scene? **A**:Yes.

II. Counterfactual

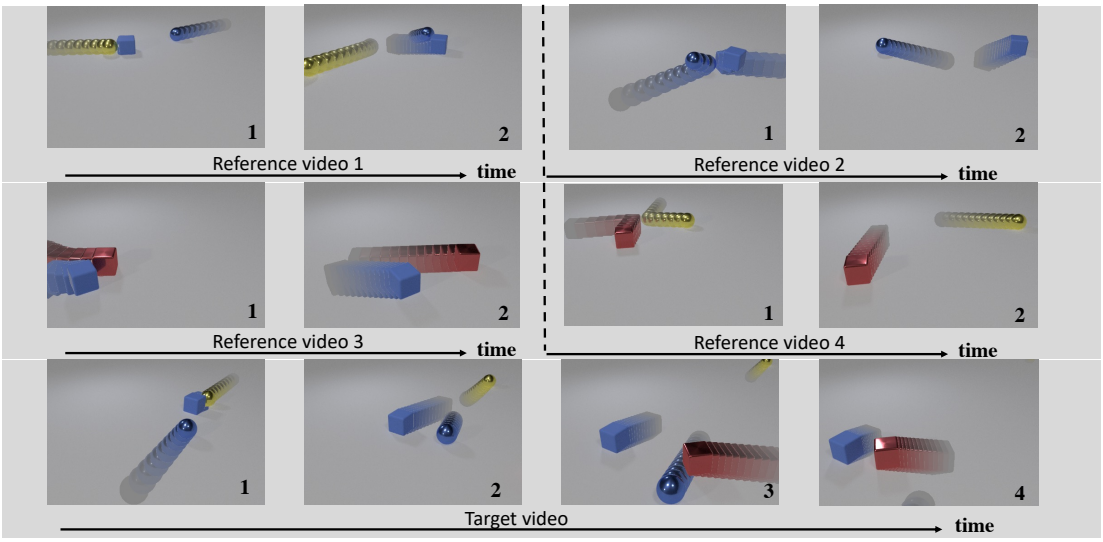
Q3: If the rubber cylinder were lighter, which of the following would happen?

a) The cube would collide with the rubber cylinder √
b) The rubber cylinder and the sphere would collide √
c) The metal object would collide with the sphere ×

III. Predictive

Q4: What will happen next?

a) The rubber cylinder and the metal object collide √
b) The rubber cylinder and the sphere collide √
c) The cube collides with the sphere ×



I. Factual

Q1: What are the colors of the two objects that are charged? **A1**: Yellow and blue.
Q2: Are there any metal cubes that enter the scene? **A2**: No.
Q3: What is the direction of the blue cube when the video ends? **A3**: Left.

II. Counterfactual

Q3: If the blue sphere were oppositely charged, what would happen?

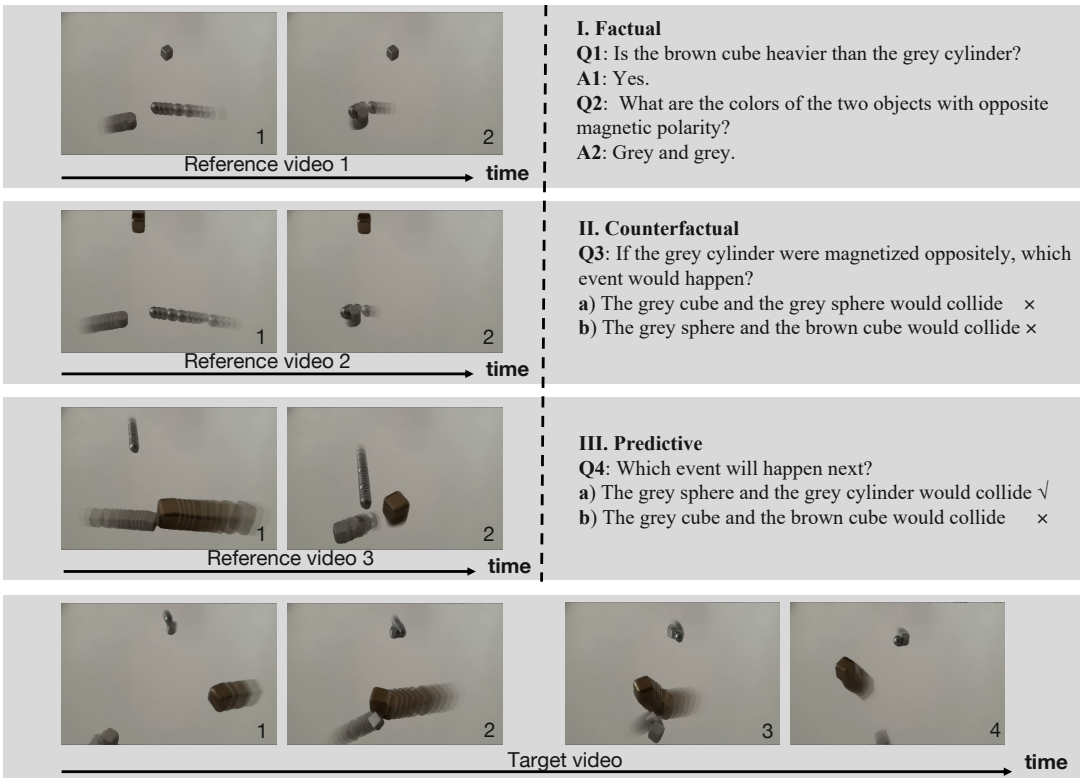
a) The yellow sphere and the rubber cube would collide √
b) The yellow object and the blue sphere would collide √
c) The blue cube and the metal cube would collide ×
d) The yellow object and the red object would collide ×

III. Predictive

Q4: Which event will happen next?

a) The blue cube and the red cube collide √
b) The blue sphere collides with the metal cube ×

Fig. 16: Sample target video, reference videos and question-answer pairs from ComPhy.



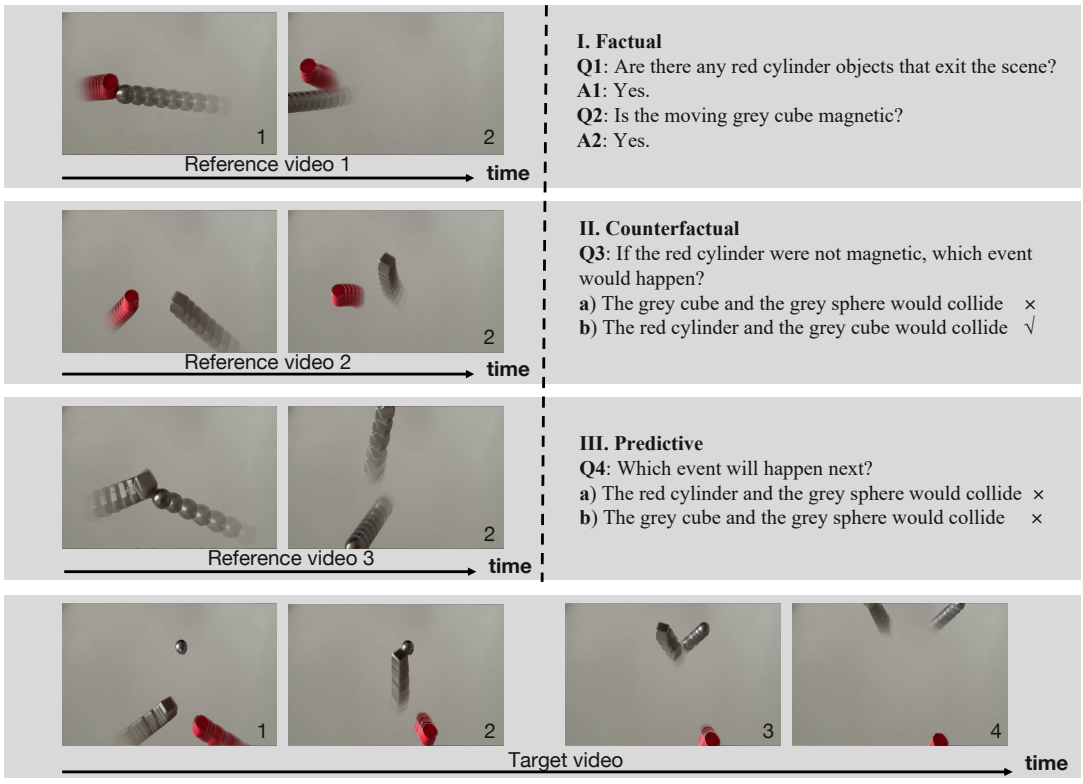
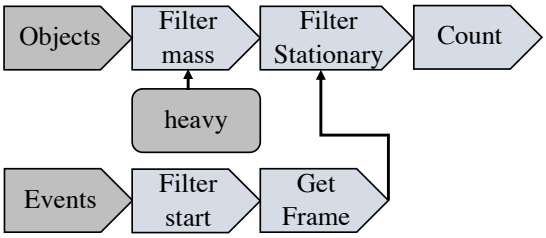
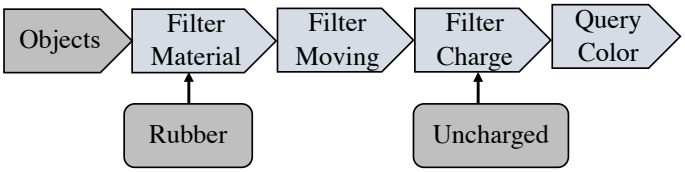


Fig. 17: Sample target video, reference videos and question-answer pairs from real-world dataset.

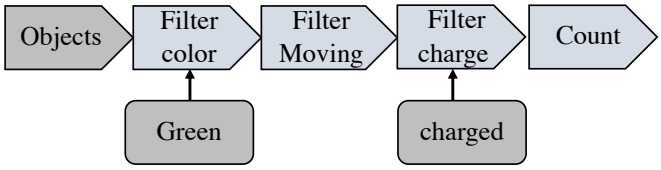
Q1: How many heavy stationary objects are there when the video begins?



Q2: What color is the moving rubber object that is uncharged?



Q3: How many moving green objects are charged?



Q4: What shape is the moving metal object that is light?

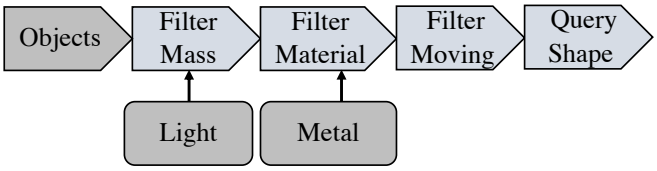
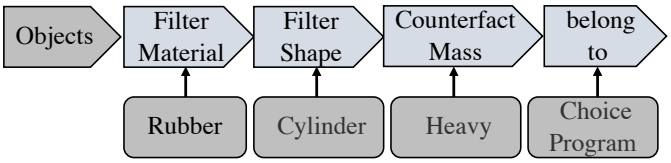
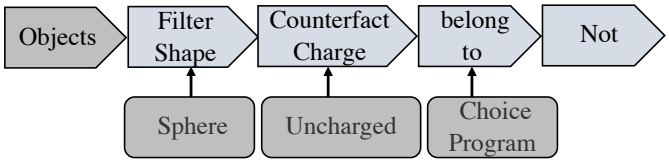


Fig. 18: Sample of factual questions and their underlying functional programs in ComPhy.

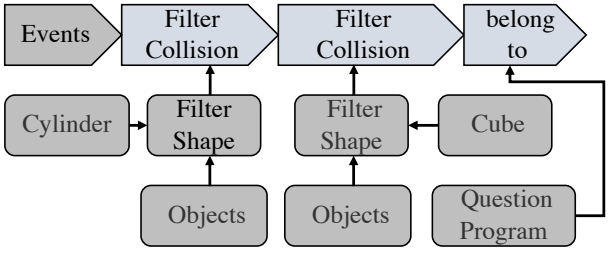
Q1: If the rubber cylinder were heavier, which of the following would happen?



Q2: Which of the following would not happen if the sphere were uncharged?



C1: The cylinder and the cube would collide



C2: The blue object and the metal object would collide

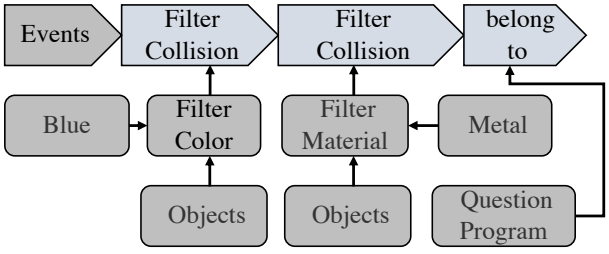


Fig. 19: Sample of counterfactual questions, choice options and their underlying functional programs in ComPhy.