Benchmarks for Physical Reasoning AI

Anonymous authors
Paper under double-blind review

Abstract

Physical reasoning is a crucial aspect in the development of general AI systems, given that human learning starts with interacting with the physical world before progressing to more complex concepts. Although researchers have studied and assessed the physical reasoning of AI approaches through various specific benchmarks, there is no comprehensive approach to evaluating and measuring progress. Therefore, we aim to offer an overview of existing benchmarks and their solution approaches and propose a unified perspective for measuring the physical reasoning capacity of AI systems. We select benchmarks that are designed to test algorithmic performance in physical reasoning tasks. While each of the selected benchmarks poses a unique challenge, their ensemble provides a comprehensive proving ground for an AI generalist agent with a measurable skill level for various physical reasoning concepts. This gives an advantage to such an ensemble of benchmarks over other holistic benchmarks that aim to simulate the real world by intertwining its complexity and many concepts. We group the presented set of physical reasoning benchmarks into subcategories so that more narrow generalist AI agents can be tested first on these clusters.

1 Introduction

Physical reasoning refers to the ability of an AI system to understand and reason about the physical properties and interactions of objects. While traditional machine learning models excel at recognizing patterns and making predictions based on large amounts of data, they often lack an inherent understanding of the underlying physical mechanisms governing those patterns. Physical reasoning aims to bridge this gap by incorporating physical laws, constraints, and intuitive understanding into the learning process.

In order to assess, contrast, and direct AI research efforts, benchmarks are indispensable. When starting a new study, it is essential to know the landscape of available benchmarks. Here we propose such an overview and analysis of benchmarks to support researchers interested in the problem of physical reasoning AI. We provide a comparison of physical reasoning benchmarks for testing deep learning approaches, identify clusters of benchmarks covering basic physical reasoning aspects, and discuss differences between benchmarks.

Reasoning about physical object interactions and physical causal effects may be one of the grounding properties of human-level intelligence that ensures generalization to other domains. The ability to adapt to an unknown task across a wide range of related tasks, known as a broad generalization, is gaining increasing attention within the AI research community (Milani et al., 2023; Malato et al., 2023). We propose to use a set of specialized benchmarks to test generalist physical reasoning AI architectures.

In this survey, we discuss 16 datasets and benchmarks (see Table 1) to train and evaluate the physical reasoning capabilities of AI agents. Most of the reviewed benchmarks focus on a certain aspect of physical reasoning, such as space (path finding, traversability, accessibilty, reachability), properties of object arrangements (stability, moveability, at/de-tachability, support, functional arrangements), object dynamics (how to push/drag/arrange objects), how to manipulate objects, recognition of degrees of freedom (DOF) (that are controllable), physical properties (deformability, breakability, decomposability), scene plausibility (occlusion, continuity of object existence).

One part of presented benchmarks provide only *passive* datasets without a possibility of interaction, while other benchmarks allow interaction with an environment.

The benchmarks we examine here involve a range of physical properties and features which are central to physical interactions amongst material objects, such as size, position, velocity, direction of movement, force and contact, mass, acceleration and gravity, and, in some cases even electrical charge. The observability of these properties and features is strongly affected by the perceptual modalities (e.g. vision, touch) that are available to an agent, and some features (such as gravity, or charge) are usually not represented according to their deeper physical concepts, but in intuitive ways via reactions of objects that can only be explained by introducing hidden interaction causes (called fields, charges, mass etc. in physics). Such representations in the form of empirical rules can be perfectly successful for providing an intuitive physical understanding whose development in humans and animals has itself been a subject of research (Melnik et al., 2018). Below, we provide a simple and pragmatic ordering into four groups arranged from most pervasive to least conspicuous (from the perspective of a human-like observer or machine learning algorithm):

1. **Global** environment variables affecting all objects (e.g. gravity)
2. **External** object variables visible from single frames (e.g. size, position)
3. **Temporal** object variables exposed through time evolution (e.g. velocity, direction of movement)
4. **Internal** object variables exposed only through object interaction (e.g. charge, mass)

Benchmarks which require (scientific and/or intuitive) understanding of phenomena related to other branches of physics, such as hydrodynamics, optics, or even quantum mechanics, are out of the scope of this survey. Likewise, pure question answering benchmarks that don't require any physical understanding are also out of the scope of this survey. Many robot movements, such as walking, climbing, jumping, or throwing, could benefit strongly from some intuitive physical understanding to be executable successfully and safely. However, we refrain from a deeper discussion of physical understanding for robots in this survey as well, since in robotics such understanding is frequently substituted by a suitable combination of embodiment and control algorithms to ensure that the robot's movements remain compatible with the situations for which the robot is designed.

In Section 2 we introduce details of each benchmark. After that, we provide a more rigorous taxonomy classification in Section 3, and finally a discussion in Section 4.

PHYRE (Bakhtin et al., 2019), Virtual Tools (Allen et al., 2020), Phy-Q (Xue et al., 2023), and OPEn (Gan et al., 2021) require agents to reason about physical scenarios and intervene in the scene to achieve a desired end goal (see Table 1).

CLEVRER (Yi et al., 2019), ComPhy (Chen et al., 2022), CRAFT (Ates et al., 2020), Interactive Language (Lynch et al., 2022), and CRIPP-VQA (Patel et al., 2022b) probe scene understanding using natural language and require systems to answer natural-language questions about images or videos (see Table 1).

CoPhy (Chen et al., 2022), SPACE (Duan et al., 2021) require future prediction (see Table 1).

Physical Bongard Problems (Weitnauer et al., 2023), CATER (Girdhar & Ramanan, 2019), IntPhys (Riochet et al., 2021), ComPhy (Chen et al., 2022), Physion (Bear et al., 2021), ShapeStacks (Groth et al., 2018) require classification prediction (see Table 1).

2 Physical Reasoning Benchmarks

In this section, we present individual benchmarks and assign tags capturing their defining properties in Table 1 to provide a pragmatic characterization. The tags concern either the type of problem to solve, the input data modality, or the evaluation metric of the benchmark, because we find those to be the most relevant aspects both for practitioners and for categorising physical reasoning tasks (see also Section 3).

The interaction tags SI , CI indicate that agents can influence an environment rather than just observe data, requiring agents with a capability to perform actions. Similarly, benchmarks involving natural language have their own tags since these require language understanding capabilities in agents. There are no specific

tags for visual data, however, since all benchmarks provide (at least) visual data as input to an agent. Classification tags indicate that agents have to solve traditional classification tasks. The future prediction tag, finally, indicates benchmarks that require explicit modeling of future states. These concepts are further discussed in the clusters we present in Section 3.

Detailed benchmark properties beyond these easy-access tags are presented in Tables 4 and 5, which provide information about physics concepts and reasoning tasks contained in each benchmark as well as links to their respective homepages. Table 2 presents the technical input and output formats required by individual benchmarks.

Benchmark		Task		
2.1	PHYRE	SI	BC	
2.2	Virtual Tools	SI	BC	
2.3	Phy-Q	CI		
2.4	Physical Bongard Problems	MC		
2.5	CRAFT	MC	BC	LP
2.6	ShapeStacks	BC		
2.7	SPACE	FP	BC	
2.8	CoPhy	FP		
2.9	IntPhys	BC		
2.10	CATER	MC		
2.11	CLEVRER	LP	LG	
2.12	ComPhy	MC	LP	LG
2.13	CRIPP-VQA	MC	LP	LG
2.14	Physion	MC		
2.15	Interactive Language	CI	LP	
2.16	OPEn	CI		

Table 1: Physical Reasoning benchmarks, category short-hands are described in the text. The Task column denotes what the intended purpose of the benchmark is. We found various benchmarks to be used for other purposes such as image segmentation or sequence prediction and added this to the respective benchmark description section in those cases.

SI **Single Interaction** benchmarks require from an agent a single environment interaction and are a special case of CI . While they are still considerably different from the other benchmark problems, they are usually less complex than instances from CI .

CI **Continued Interaction** benchmarks require from an agent ongoing interactions with an environment and represent a notably distinct group of problems regardless of what the actual benchmark task is. Since the performed actions directly influence the training data distribution, the trade-off between exploration and exploitation is one of the major sub-problems to solve here.

BC **Binary Classification** benchmarks require merely true/false discrimination. Instances of binary classification problems are stability prediction or collision detection.

MC **Multi-label Classification** benchmarks can come in the form of multiple-choice questionnaires or even natural language tasks with a limited and small set of tokens. Although the latter could be framed as natural language question answering, their solution can be effectively represented as an integer.

LP **Language Processing** benchmarks require the agent to understand natural language input.

LG **Language Generation** benchmarks require proper natural language answers that consist of multiple tokens. Thus, they usually constitute considerably more complex problems than MC or BC benchmarks because the solution space is much larger. The agent has to be capable of synthesizing natural language on top of understanding the physical reasoning task.

FP **Future Prediction** benchmarks require the agent to *explicitly* predict future frames or object properties, which requires more explicit physical reasoning than the other benchmark categories. While an imperfect model of the problem might be sufficient to perform well in most benchmarks, FP puts more emphasis on accurate prediction of the problem dynamics.

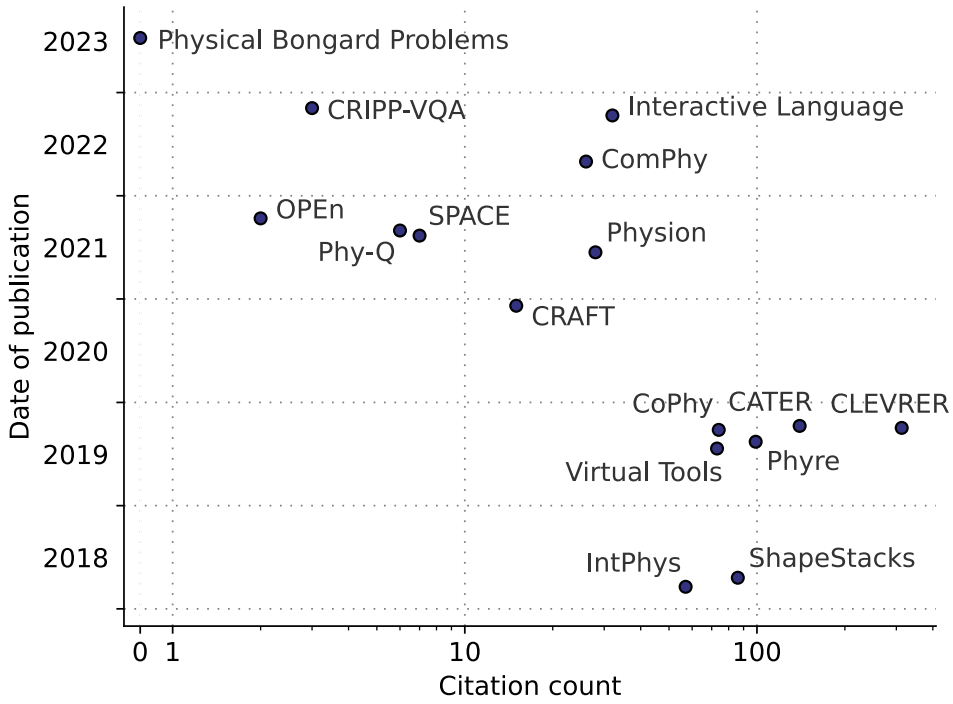


Figure 1: Publication date versus citations (symmetric log scale) for benchmarks. The date of publication is based on the first submission on arXiv or elsewhere. Citation count is a snapshot of Google Scholar information from 15 August 2023.

Figure 1 details the publication history and number of citations as a proxy measure for relevance in the field.

In the following, we describe the individual physical reasoning benchmarks in detail.

2.1 PHYRE

PHYRE (Bakhtin et al., 2019) studies physical understanding based on visual analysis of a scene. It requires an agent to generate a single action that can solve a presented puzzle. The benchmark is designed to encourage the development of learning algorithms that are sample-efficient and generalize well across seen and unseen puzzles.

Each of the puzzles in PHYRE is a deterministic 2D box-shaped environment following Newtonian physics. It is comprised of a goal and a set of rigid objects that are either static or dynamic, whereas only the latter can be influenced by gravity or collisions. To indicate their object category, dynamic objects can have various colors, and static objects are always black. The goal condition of a puzzle refers to two objects that should touch each other for at least 3 seconds and is not available to the agent. Observations are provided in the

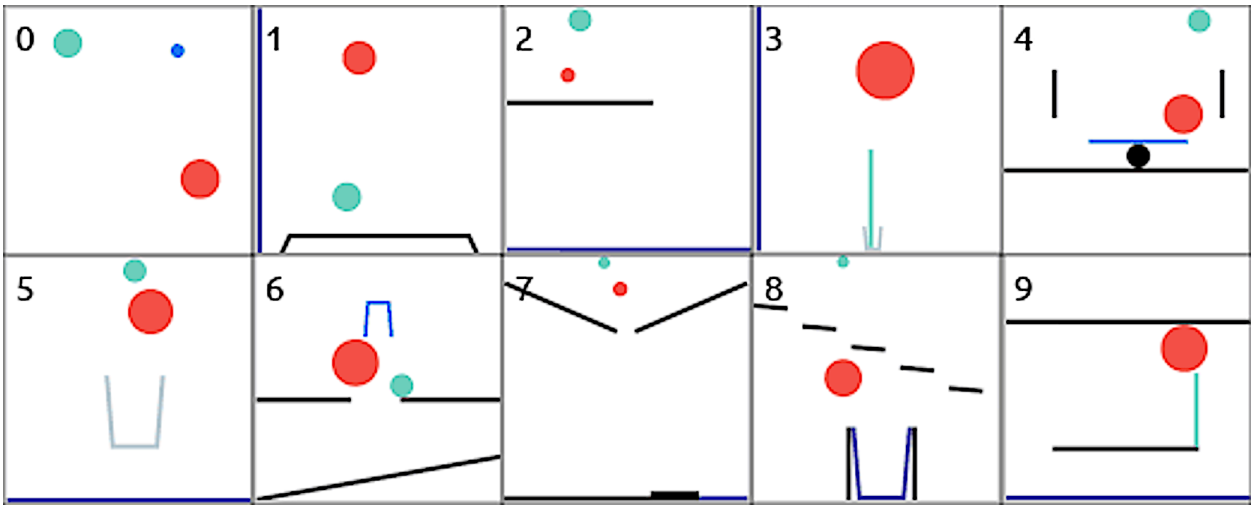


Figure 2: Ten templates of PHYRE-B (BALL) puzzles (Bakhtin et al., 2019). Task: Place the red action balls so that a green object ends up touching a blue surface. Image adapted from (Harter et al., 2020b).

form of images and valid actions are the initial placement of one or two red balls and the choice of their radii. Note that only a single action in the first frame is allowed; from there on the simulation unfolds without any interference by the agent.

PHYRE contains two benchmark tiers: *Phyre-B*, which requires choosing the position and radius of one red ball, Fig 2, and *Phyre-2B*, which requires choosing the same attributes for two balls. To limit the action search space, both tiers come with a predefined set of 10000 action candidates. Each tier contains 25 templates from which 100 concrete tasks per template are produced. While a template only defines the set of objects and the scene goal, tasks fill in the detailed object properties such as position and size. Since tasks from a single template are more similar than those from different templates, PHYRE differentiates between the within and the cross-template benchmark scenarios. Specifically, the cross-template scenario is intended to test the generalization capabilities of an agent beyond known object sets. At test time, PHYRE requires the agent to solve a task with as few trials as possible, whereas each attempt results in a binary reward that indicates whether the puzzle was solved or not. This provides the opportunity to adapt and improve the action of choice in case of a failed attempt. Two measures are taken to characterize the performance of an agent:

- The *success percentage* is the cumulative percentage of solved tasks as a function of the attempts per task.
- To put more emphasis on solving tasks with few attempts, the *AUCCESS* is a weighted average of the success percentages computed as $\sum_k w_k \cdot s_k / \sum_k w_k$ with $w_k = \log(k+1) - \log(k)$ and $k \in \{1, ..., 100\}$. Note that this results in an AUCCESS of less than 50% for agents which require more than 10 attempts on average to solve tasks.

The baseline solution to PHYRE is presented by (Bakhtin et al., 2019). The idea is to learn a critic-value function for state-action pairs, where the state is an embedding of the initial frame of an episode and the action is a specific choice of the red ball's position and radius. Then a grid search over a predefined set of 3-dimensional actions for a given state is performed and actions are ranked w.r.t. the value estimated by the *critic* network. The set of top ranked candidate actions by the *critic* network is provided for sampling until a trial is successful.

More advanced solutions for the PHYRE benchmark are proposed in (Girdhar et al., 2021; Qi et al., 2020; Wu et al., 2022; Harter et al., 2020a; Li et al., 2020; Ahmed et al., 2021; Li et al., 2022a; Rajani et al., 2020).

2.2 Virtual Tools

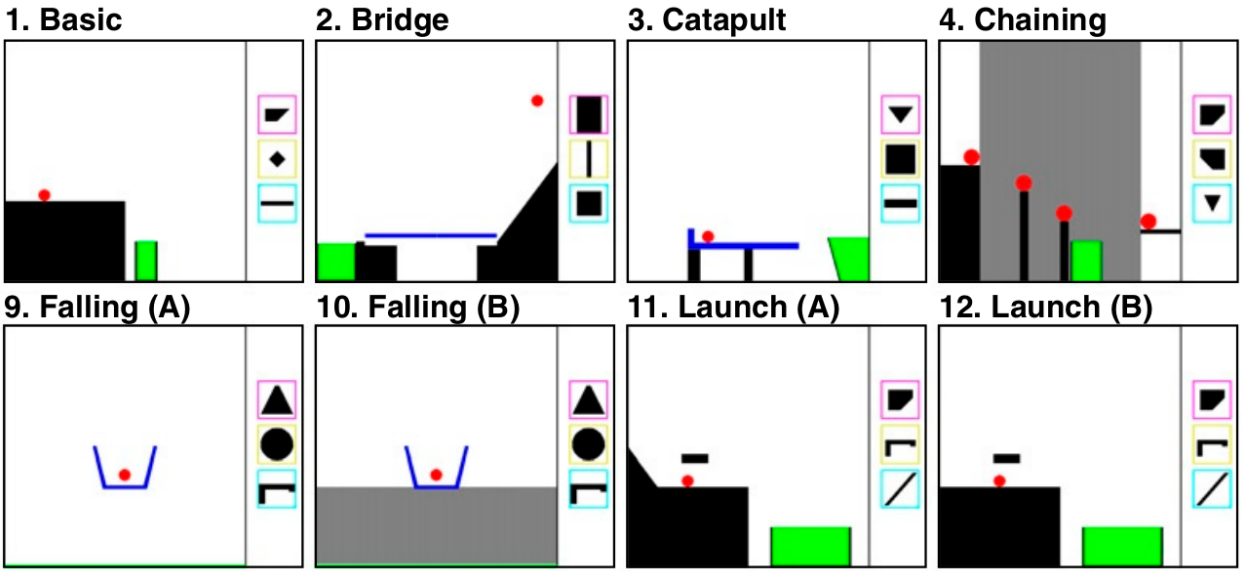


Figure 3: Twelve of the levels used in the Virtual Tools game. Players choose one of three tools (shown to the right of each level) to place in the scene to get a red object into the green goal area. Black objects (except tools) are fixed, while blue objects also move; gray regions are prohibited for tool placement. Levels denoted with A/B labels are matched pairs. Image adapted from Allen et al. (2020).

Virtual Tools (Allen et al., 2020) is a 2D puzzle (see Figure 3) where one of three objects (called tools) has to be selected and placed before rolling out the physics simulation. The goal is to select and place the tool so that a red ball ends up touching a green surface or volume. The benchmark consists of 30 levels (20 for training and 10 for testing) and embraces different physical concepts such as falling, launching, or bridging. 12 of the training levels are arranged in pairs, where one is a slight modification of the other. This allows for studying how learning is affected by perturbations of the setup.

As a baseline solution, the authors propose to sample random actions (sampling step) and run their ground truth simulation engine with some added noise (simulation step). The most promising simulated actions are executed as a solution with their ground truth simulation engine without noise. If this does not solve the puzzle, simulated and executed outcomes are accumulated using a Gaussian mixture model, and action probabilities are updated (update step). Sampling, simulation, and updating are iterated until the puzzle is solved.

While the *Virtual Tools* benchmark has been cited as much in cognitive science research as in machine learning research, no machine learning solutions have been proposed yet. Allen et al. (2022), however, propose a method and briefly mentions applying it on a modified version of *Virtual Tools*.

This benchmark is similar to the PHYRE puzzles, with the difference that *Virtual Tools* requires a selection of one of the three provided tools first.

2.3 Phy-Q

The Phy-Q benchmark (Xue et al., 2023) requires the interaction of an agent with the environment to solve 75 physical Angry Birds templates that cover 15 different physical concepts (see Figure 4). Similar to PHYRE (Bakhtin et al., 2019), for each template, a set of 100 different tasks with slightly different object configurations have been generated.

There are four types of objects in the game: birds, pigs, blocks and platforms. The agent can shoot birds from a given set with a slingshot. All objects except platforms can accrue damage when they are hit. Eventually

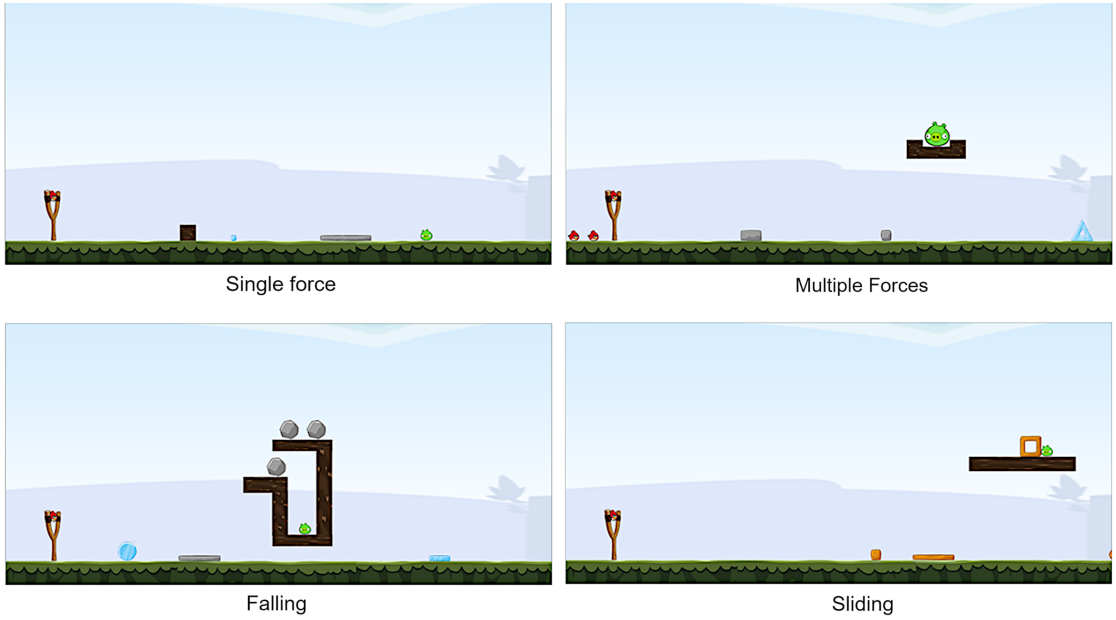


Figure 4: Four example tasks out of the 15 covered in Phy-Q. Single or multiple force means that one or several hits are required. The slingshot with birds is situated on the left of the frame. The goal of the agent is to hit all the green-colored pigs by shooting birds with the slingshot. The dark-brown objects are static platforms. The objects with other colors are dynamic and subject to the physics in the environments. Image adapted from Xue et al. (2023).

they get destroyed. Additionally, birds have different powers that can be activated during flight. The task in these Angry Birds scenarios is always to destroy all pigs in a scene with the provided set of birds. To achieve this goal, the agent has to provide as an action the relative release coordinates of the slingshot and the time point when to activate a bird's powers. In some scenarios, the agent has to shoot multiple birds in an order of its choosing. The observations available to the agent are screenshots of the environment and a symbolic representation of objects that contains polygons of object vertices and colormaps.

The benchmark evaluates local generalization within the same template and broad generalization across different templates of the same scenario. The authors define a Phy-Q score, inspired by human IQ scores, to evaluate the physical reasoning capability of an agent. It evaluates broad generalization across the different physical scenarios and relates it to human performance, so that a score of 0 represents random, and a score of 100 average human performance. In order to calculate these scores, the authors collect data on how well humans perform on their benchmarks.

The baseline solution uses a Deep Q-Network (DQN) (Mnih et al., 2015) agent. The reward is the 1 if the task is passed and 0 otherwise. The agent can choose from 180 possible discrete actions, each corresponding to the slingshot release degree at maximum stretch. The DQN agent learns a state representative with a convolutional neural network (CNN). In a modification of this baseline, a pre-trained ResNet-18 model was used for feature extraction from the first frame, followed by a multi-head attention module (MHDPA) (Zambaldi et al., 2018), followed by a DQN. Beyond the baseline solution, there are no other available solutions as of now.

Phy-Q is most similar to the Virtual Tools benchmark (see Chapter 2.2), although it supports continuous interaction. However, agents have to choose from a range of birds, which is somewhat equivalent to choosing from a set of tools in Virtual Tools.

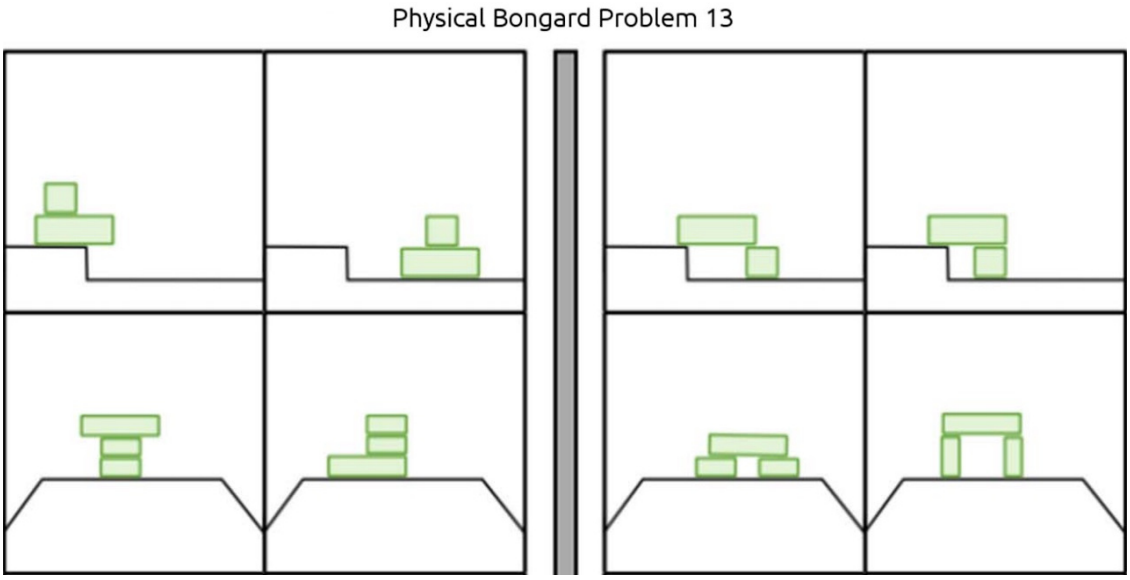


Figure 5: Example of a Physical Bongard Problem (number 13 in Weitnauer et al. (2023)). Solution: A circle is blocked vs. can be lifted. Image adapted from Weitnauer et al. (2023).

2.4 Physical Bongard Problems

The images in this benchmark contain snapshots of 2D physical scenes depicted from a side perspective. The scenes in Physical Bongard Problems (PBPs) (Weitnauer et al., 2023) contain arbitrary-shaped non-overlapping rigid objects which do not move at the time $t = t_0$ of the snapshot. There are 34 PBPs, each consisting of four scenes grouped on the left side of the image, and four on the right side. The task is to predict the concept that distinguishes the scenes on the left side from those on the right side of the PBP image. Here, a concept is an explicit description that explains the difference between the scenes on the left and on the right (for instance, circle blocked vs. liftable, see Fig 5).

In general, the solution of PBPs can be based on descriptions of the whole scene or parts of the scene at any point in time or on the reaction of objects to simple kinds of interaction, e.g., pushing. This focus on indicating physical understanding by coming up with an explicit, human-readable description distinguishes the approach from more implicit and black-box-like demonstrations of understanding in the form of successful acting.

The descriptions are constructed from searching a hypothesis space that encodes hypotheses as tuples [side, numbers, distances, sizes, shapes, stabilities] of a small number of hand-chosen features and object relations (such as scene side, number of its objects, inter-object distances, shapes, or stabilities). For example, the meaning of the hypothesis [left, 1-3, ?, small or large, ?, stable] is "all left scenes (and none of the right scenes) contain one to three objects that are small or large-sized and stable". The algorithm starts with all possible hypotheses and removes the incompatible ones for each scene. Finally, among the remaining hypotheses, the one with the shortest length is chosen as the solution. Thus the hypothesis space can be represented as a categorization space with many possible classes.

It is essential for solving PBPs to be able to predict and visualize the outcome of dynamic situations and interactions. The authors model this ability by giving the solver access to a physics engine (PE). It is used in two ways: First, to predict the unfolding of actions in the scenes. Second, the engine is used to estimate physical object features.

Beyond the original study, Lupyan & Zettersten (2021) and Weitnauer et al. (2014) test human performance on Physical Bongard Problems, which can serve as a baseline for future solutions. Only one computational approach has been proposed in Weitnauer et al. (2015).

2.5 CRAFT

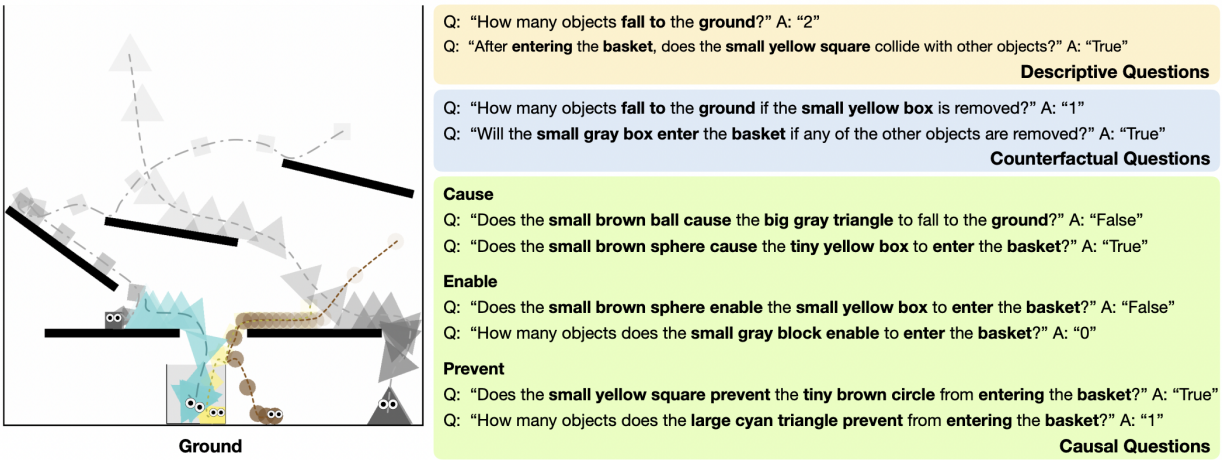


Figure 6: Example of CRAFT questions generated for a sample scene. Image from Ates et al. (2020).

The CRAFT dataset (Ates et al., 2020) is a question answering benchmark about physical interactions of objects presented in short videos. This dataset includes 57K videos and question pairs generated from 10K videos from 20 different two-dimensional scenes. An exemplary scene is shown in Figure 6. The figure also shows a sample of questions that are created for the scene. The types of questions in this benchmark are descriptive, counterfactual and explicitly causal.

The representation of simulation episodes involves different data structures: video frames and the causal graph of the events in the episode that is used to generate questions. The initial and the final states of the scene refer to object properties, including the object's color, position, shape, and velocity at the start/end of the simulation. This information is provided to enable a successful answering in the benchmark.

The first baseline solution uses R3D (Tran et al., 2018) with a pre-trained ResNet-18 CNN base to extract information from a down-sampled video version. The text information is extracted using an LSTM. Afterward, the textual and visual information is passed to a multilayer perception network (MLP) which makes the final decision. The second baseline solution also uses R3D (Tran et al., 2018) with a pre-trained ResNet-18 CNN base to extract information from a down-sampled video version. However, instead of processing the textual and visual information separately, it processes them simultaneously using a Memory, Attention, and Composition (MAC) model (Hudson & Manning, 2018). There are no available solutions, other than the baseline solutions, that cite the current CRAFT dataset paper (Ates et al., 2020).

The baseline approaches are evaluated in their multiple-choice accuracy within each question category and include heuristic models, LSTM, BERT, and LSTM-CNN. Video frames and text of the current question are given to the models as input.

CRAFT is most similar to the CATER (see Chapter 2.10) and CLEVRER benchmarks (see Chapter 2.11).

2.6 ShapeStacks

ShapeStacks (Groth et al., 2018) is a simulation-based dataset that contains 20,000 object-stacking scenarios. The diverse scenarios in this benchmark cover multiple object geometries, different complexity degrees for each structure, and various violations in the structure's stability (see Figure 7). A scenario is represented by 16 RGB images at the initial time step that shows the stacked objects from different camera angles. Every scenario carries a binary stability label and all images come with depth and segmentation maps. The segmentation maps annotate individual object instances, the object which violates the stability of the tower, the first object to fall during the collapse, and the base and top of the tower. While the actual benchmark

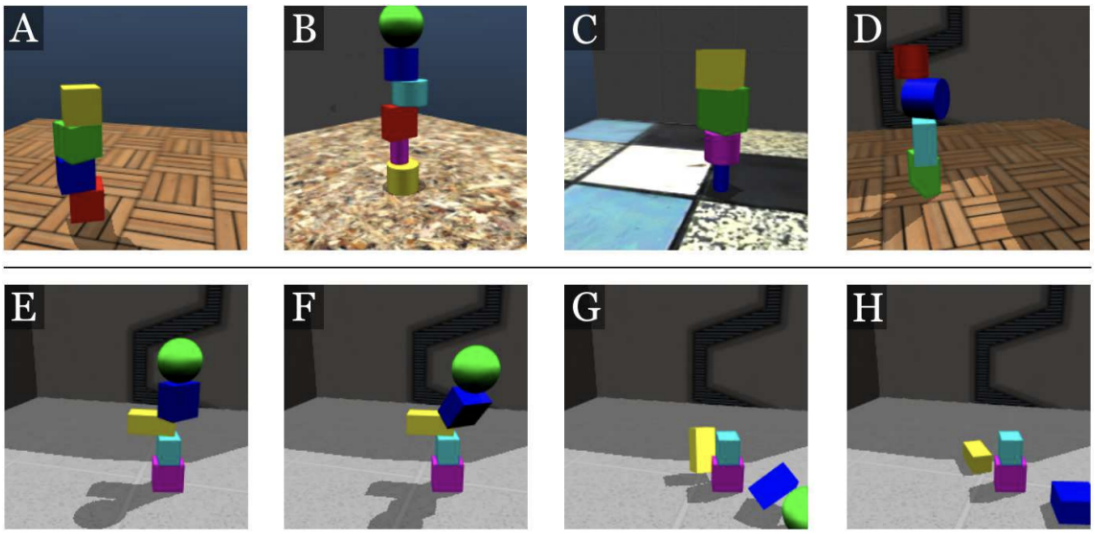


Figure 7: Different scenarios from the ShapeStacks data set. (A)-(D) initial stack setups:(A) stable, rectified tower of cubes, (B) stable tower where multiple objects counterbalance each other; some recorded images are cropped purposefully to include the difficulty of partial observability, (C) stable, but visually challenging scenario due to colors and textures, (D) violation of planar-surface principle. (E)-(H) show the simulation of an unstable, collapsing tower due to a center of mass violation. Image from Groth et al. (2018).

includes per scenario only the 16 images with accompanying depth and segmentation maps, the MuJoCo world definitions are also provided to enable the complete re-simulation of the stacking scenario.

The base task in ShapeStacks is to predict the stability of a scenario, although the data additionally contains information on the type of instability, i.e. whether a tower collapses due to center of mass violations or due to non-planar surfaces.

The two baseline solutions provided by Groth et al. (2018) use either AlexNet or Inception v4-based image discriminators along with training data augmentation to predict if a shape stack is stable. The benchmark score is computed as the percentage of correctly classified scenes from a test set that was withheld during training. The Inception v4-based discriminator performs best both on the artificial ShapeStacks scenes (Cubes only: 77.7% | Cubes, cylinders, spheres: 84.9%) as well as on real-world photographs of stacked cubes (Cubes: 74.7% | Cubes, cylinders, spheres: 66.3%).

The majority of publications that use ShapeStacks circumvent solving the stability classification problem by instead solving a related, but simpler prediction problem, such as frame or object mask prediction (Ye et al., 2019; ALIAS PARTH GOYAL et al., 2021; Qi et al., 2020; Ehrhardt et al., 2020; Singh et al., 2021; Engelcke et al., 2020a; Chang et al., 2022; Sauvalle & de La Fortelle, 2023a; Schmeckpeper et al., 2021; Sauvalle & de La Fortelle, 2023b; Jia et al., 2022b; Emami et al., 2022). Only a few works (Engelcke et al., 2020b; 2021; Fuchs et al., 2018) directly attack solving the stability classification problem. We argue that approaches from the second group need a more refined physics understanding and can be interpreted as superior to the first group in terms of physical reasoning capabilities.

CoPhy, Physion and SPACE (cf. Sec. 2.7 below) contain stacked object scenarios as well. CoPhy focuses on more counterfactual reasoning and requires stability prediction only as one of several abilities. Physion and SPACE in addition to stacking also cover multiple physical concepts that go beyond stability prediction alone.

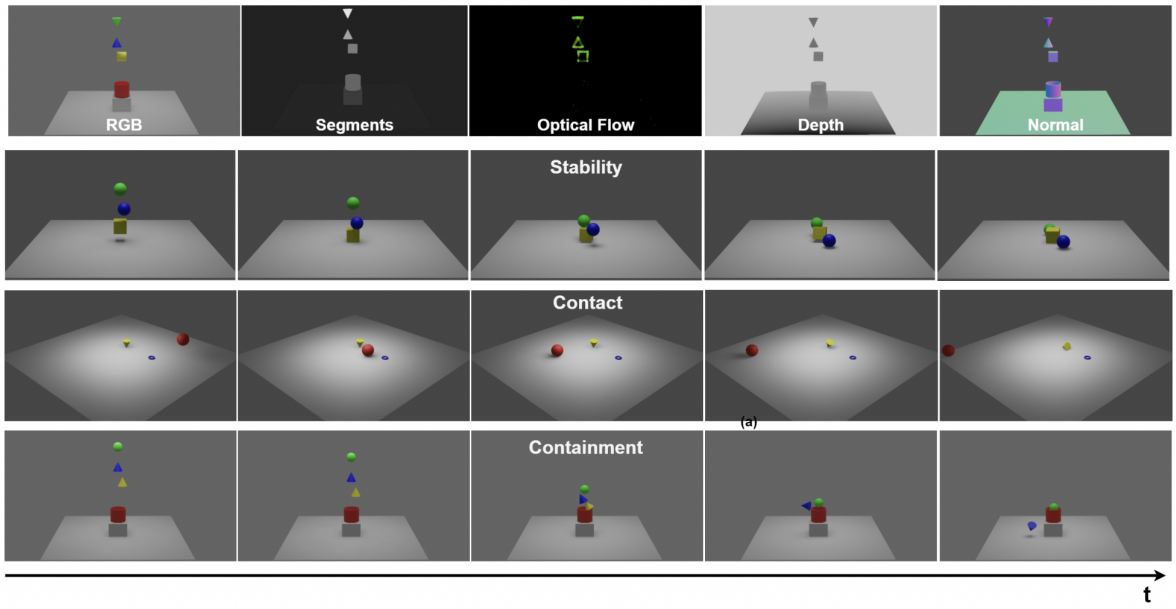


Figure 8: Example data from the SPACE benchmark. Top row: Visual data attributes for one example. The frame comprises RGB, object segmentation, optical flow, depth, and surface normal vector. Bottom three rows: Example frames from the three physical interactions. Image from Duan et al. (2021).

2.7 SPACE

SPACE, introduced by Duan et al. (2021), is based on a simulator for physical interactions and causal learning in 3D environments. The simulator is used to generate the SPACE dataset, a collection of 3D-rendered synthetic videos. The dataset comprises videos depicting three types of physical events: containment, stability, and contact. It contains 15,000 (5,000 per event type) unique videos of 3-second length. Each RGB frame is accompanied by maps for depth, segmentation, optical flow, and surface normals, as shown in Figure 8.

Objects are sampled from the set $O = \{\text{cylinder, cone, inverted cone, cube, torus, sphere, flipped cylinder}\}$. There are three different types of videos, which come with classification labels. These imply classification tasks, while the authors use their dataset only for future frame prediction. Containment videos show a container object (colored in red and sampled from the set $C = \{\text{wine glass, glass, mug, pot, box}\}$) below a scene object from the set O. The task is to predict whether the scene object is contained in the container object or not. Stability videos depict up to three objects from O which are stacked on top of each other, and the task is to predict whether the object configuration is stable or not. Contact videos contain up to three objects at varying locations in the scene and a sphere of constant size moving around the scene on a fixed trajectory. The task is to predict whether the objects are touched by the sphere or not.

While in principle a broad range of scene classification and understanding tasks are possible due to the procedurally generated nature of the SPACE dataset, the authors do not provide all of the necessary metadata and focus on video prediction or the recognition tasks described above. They show that pretraining with the synthetic SPACE dataset enables transfer learning to improve the classification of real-world actions depicted in the UCF101 action recognition dataset. UCF101 is a large collection of short human action videos covering 101 different action categories (Soomro et al., 2012). For their demonstration, they show that pretraining a model (PhyDNet, cf. below) on the SPACE dataset versus directly on the UCF101 dataset leads to improved performance when subsequently both pre-trained models are fine-tuned on the UCF101 dataset.

As of the time of writing, there are no further works that attempt to solve tasks on the SPACE datasets in the literature. The authors have however proposed an updated version SPACE+ (Duan et al., 2022) of their benchmark which quadruples the amount of videos and introduces some new object classes to better evaluate model generalization.

The SPACE dataset bears similarities to ShapeStacks, CoPhy and Physion in the sense that all of them contain stacking scenarios and ShapeStacks in particular provides object segmentation masks as well. Although in ShapeStacks the actual task is to predict stacking stability from a set of images, it is often used for frame or object mask prediction similar to SPACE.

2.8 CoPhy

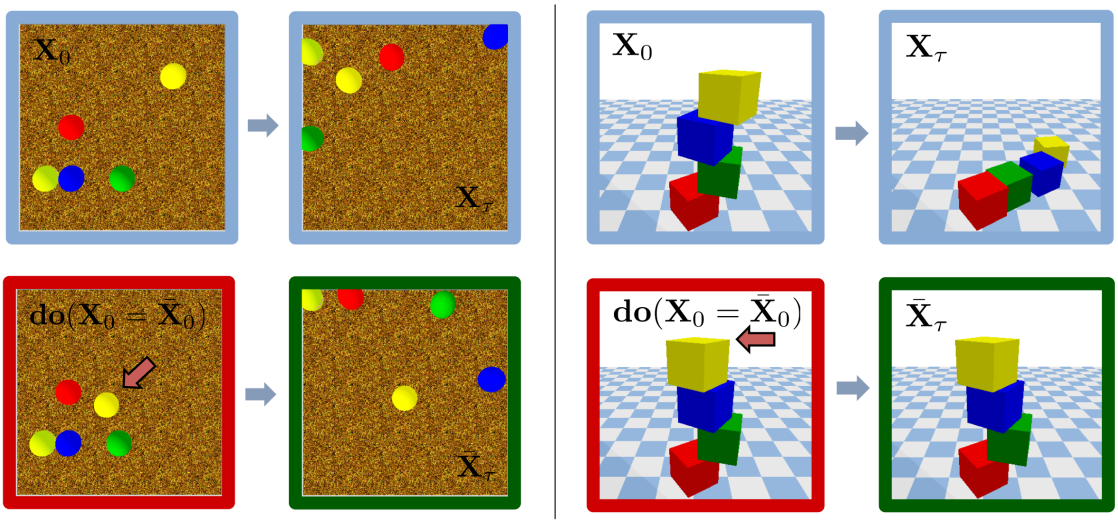


Figure 9: Exemplary CoPhy scenarios. Given an observed frame A $= X_0$ and a sequence of successor frames B $= X_{1:t}$, the question is how the outcome B would have changed if we changed X_0 to $\hat{X}_0$ by performing a do-intervention (e.g., changing the initial positions of objects in the scene). Image adapted from Baradel et al. (2020).

The Counterfactual Learning of Physical Dynamics (CoPhy) benchmark (Baradel et al., 2020) introduces a physical forecasting challenge that necessitates the use of counterfactual reasoning. In physical reasoning problems, a counterfactual setting refers to a hypothetical situation where a specific aspect of the real-world scenario is changed. CoPhy tests physical understanding by evaluating what would happen if the altered condition were true.

Specifically, given a video recording of objects moving in a first scenario (frames $X_{0:t}$ in Figure 9), the objective is to anticipate the potential outcome in a second scenario whose first frame ($\hat{X}_0$) differs from the first frame in the first scenario by subtle changes in the objects' positions. The key component of the benchmark is that there are hidden variables associated with the objects and the environment, including mass, friction, and gravity. These hidden variables or confounders are not discernible in the single altered frame ($\hat{X}_0$) of the second scenario, but they are observable in the video recording of the first scenario. Successfully predicting future outcomes for objects in the second scenario thus entails the estimation of the confounders from frames $X_{0:t}$ in the first scenario. For both scenarios, video recordings are provided for training of the prediction agent.

The observed sequence demonstrates the evolution of the dynamic system under the influence of laws of physics (gravity, friction, etc.) from its initial state to its final state. The counterfactual frame corresponds to the initial state after a so-called do-intervention, a visually observable change introduced to the initial physical setup, such as object displacement or removal.

All images for this benchmark have been rendered into the visual space (RGB, depth, and instance segmentation) at a resolution of 448 × 448 px. The inputs are the initial frames (RGB, depth, segmentation) of two slightly different scenes (such as object displacement or removal), a sequence of roll-out frames of the first scene, and 3D coordinates of all objects in all available frames. The task is to predict the final 3D coordinates of all objects in the second scene. The evaluation metric is an MSE final 3D coordinates of all objects in the second scene measured between the prediction and the ground truth roll-out in a simulation.

The proposed baseline solution model is CoPhyNet (Baradel et al., 2020), which predicts the position of each object after interacting with other blocks in the scene [1]. The inputs are RGB images of resolution 224×224. They first train an object detector to give the 3d location of each object. Then they use these estimations for training CoPhyNet. The CoPhyNet predicts the final 3D coordinates of all the objects after changing the scene or if the object's position will change. The experiments show that this model performs better than MLP and humans in predicting the object's position on unseen confounder combinations and on an unseen number of blocks and balls.

Two different works have so far proposed solutions to CoPhy (Li et al., 2022b; 2020). Additionally, Filtered-CoPhy (Janny et al., 2022) has been proposed as a new benchmark based on CoPhy. It is a modification of the original CoPhy with a different task definition. Instead of prediction of object coordinates, it requires future prediction directly in pixel space.

CoPhy is similar to the CRAFT (2.5), CLEVRER (2.11), ComPhy (2.12) and CRIPP-VQA (2.13) benchmarks, which also address counterfactual reasoning questions. These other benchmarks however do not focus as explicitly on counterfactuals and do not contain explicit interventions in their scenes. They also involve natural language tasks, which is an added complexity that is not present in CoPhy.

2.9 IntPhys

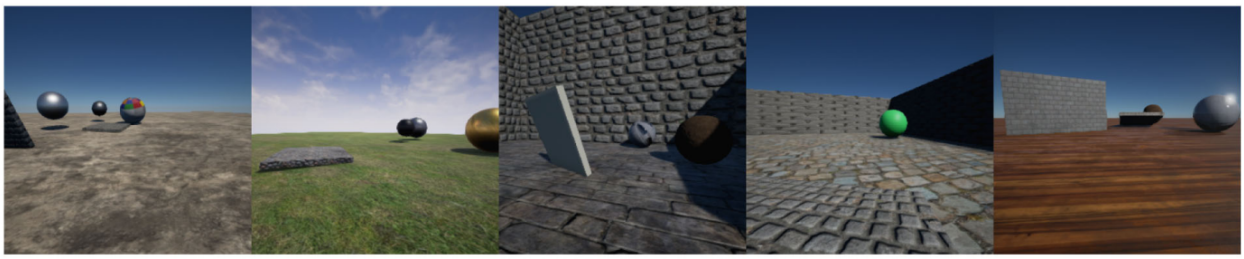


Figure 10: Exemplary screenshots from video clips used in IntPhys. Image from Riochet et al. (2021).

The IntPhys (Riochet et al., 2021)[2] benchmark evaluates physical reasoning for visual inputs based on the intuitive physics capabilities of infants. The benchmark is designed to measure the ability to recognize violations of three basic physical principles: Object permanence, shape consistency, and spatiotemporal continuity. Events that violate these principles (violation of expectation events – VoE) can already be detected by very young children. The benchmark consists of a set of short video clips of 7 seconds (see Figure 10) designed to present physical events that either obey the three principles (plausible scenarios) or violate at least one of them (implausible scenarios). The model is trained with videos that only show plausible scenarios and subsequently evaluated on a test set that also contains implausible ones. This evaluation is somewhat more abstract than directly using prediction error in some lower-level metric, which can be high due to limitations of such metrics to properly ignore all forms of irrelevant variance, even if the model responses might be well in line with a correct intuitive understanding of the physics, if such variance were absent.

The idea is that if the model has learned the laws of physics, it should attribute high probability to scenarios that are plausible so that low probability values can be used to identify scenarios that are implausible.

[1] https://github.com/fabienbaradel/cophy

[2] The author's original link https://intphys.com to this benchmark's website is outdated, please refer to Table 4 for an updated link.

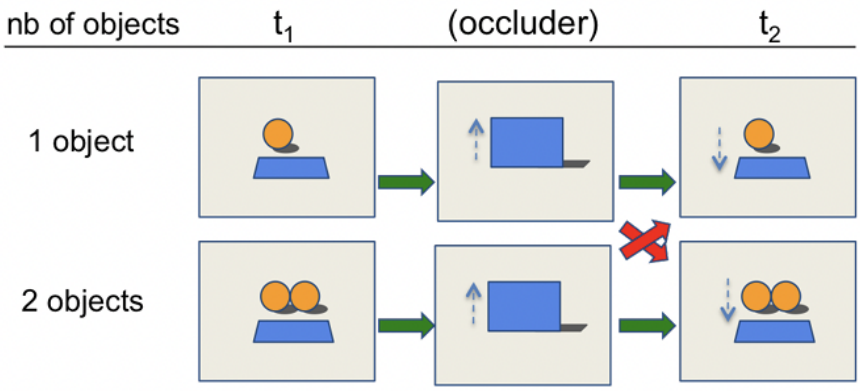


Figure 11: Illustration of the minimal sets design with object permanence in IntPhys. Schematic description of a static condition with one vs. two objects and one occluder. In the two possible movies (green arrows), the number of objects remains constant despite the occlusion. In the two impossible movies (red arrows), the number of objects changes (goes from 1 to 2 or from 2 to 1). Image from Riochet et al. (2021).

Therefore, the authors claim that a model that has only been trained on plausible scenarios should be able to generalize to other plausible scenarios but reject implausible ones. An illustration of this benchmark is shown in Figure 11.

The first baseline model is implemented through a ResNet that has been pre-trained on the Imagenet data and that subsequently is fine-tuned to become a classifier for the distinction of plausible versus implausible videos. One metric is the relative error rate which computes a score within each set that requires the plausible movies to be assigned a higher plausibility score than the implausible ones. The second metric is the absolute error rate which requires that globally the score of plausible videos is greater than the one of implausible videos.

The second baseline model works with semantic masks of input frames and predicted future frames. This work concludes that operating at a more abstract level is a worthwhile pursuing strategy when it comes to modeling intuitive physics.

While the IntPhys benchmark has been widely cited and regularly discussed, there are only two proposed solutions in the literature by Smith et al. (2019) and Nguyen et al. (2020). Beyond these, Du et al. (2020) propose an object recognition model and suggest using it for physical plausibility downstream tasks.

Beyond IntPhys, there are two additional VoE benchmarks that involve physical reasoning which are however very similar to IntPhys both in their covered concepts and their data format (Piloto et al., 2022; Dasgupta et al., 2021).

2.10 CATER

CATER Girdhar & Ramanan (2019) is a spatiotemporal reasoning benchmark that extends CLEVR (Johnson et al., 2017), which is based on static images, to the spatio-temporal domain. It provides blender-based rendering scripts to generate videos and associated classification tasks such as action recognition and target tracking. The videos contain simulated 3D objects that move on a 2D plane, as shown in Figure 12. For each video, objects have been sampled randomly from a small set of generic bodies (such as cubes, cones or cylinders) and come with a set of permitted atomic actions such as sliding and placing. These actions are assigned and applied randomly to each object throughout a video. The authors provide a pre-rendered set of 16500 videos that were created in this manner as their CATER dataset on which they perform their baseline experiments.

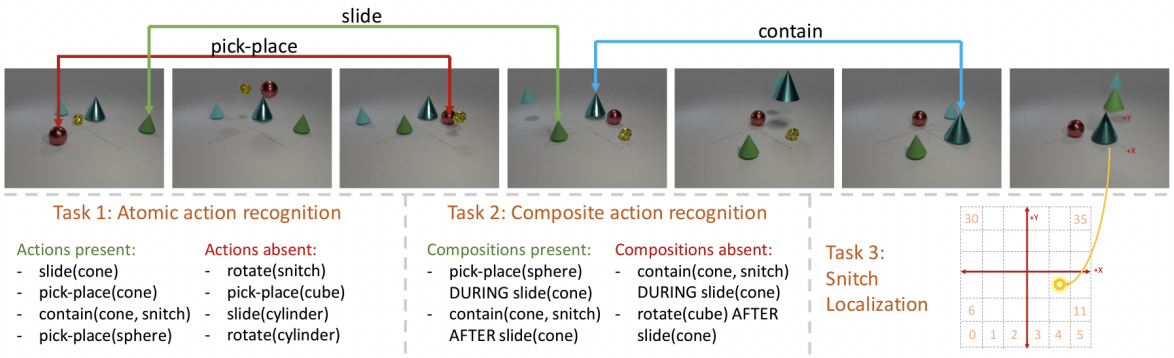


Figure 12: CATER dataset and tasks. Sampled frames from a random video from CATER. Some of the actions afforded by objects in the video are labeled on the top using arrows. Three tasks are considered: Task 1 requires a listing of all actions that are observable in the video. Task 2 requires identifying all observable compositions (i.e. temporal orders) of actions. Task 3 requires quantified spatial localization of a specific object (called snitch) that became covered by one or more cone(s) and, therefore, has disappeared from direct view, while still being dragged with the movements of the cone(s) enclosing it. This task tests agents for the high-level spatial-temporal understanding that is required in order to track an invisible object through the movements of its occluding container. Image from Girdhar & Ramanan (2019).

The benchmark comes with three predefined classification tasks: The first task (atomic action recognition) requires detecting all pairings between atomic actions and objects that have occurred within a video. The second task (compositional action recognition) requires to detect temporal compositions (only temporal pairs are considered) of object-action pairings, together with their temporal relation (before, during or after). The last and final task (snitch localization), only applicable to a subset of videos, requires locating a certain object, called snitch object, that has become covered and dragged around by one (or, through nesting, possibly several) of the cones (the only object type with this container property). The answer needs to specify the location only as a discrete grid cell to allow again a classifier approach to this last task.

The authors stress that they extend existing benchmarks beyond two frontiers: The tasks include detecting temporal relationships between actions, and the videos avoid scene or context bias, i.e., there is very little information about the task solution present in the background of video frames. Both aspects are meant to encourage solution approaches really capable of temporal reasoning rather than analyzing individual frames.

As baseline solutions, a few existing methods are adapted and compared. The base method used for all three tasks employs temporal CNNs (Wang et al., 2016; 2018), which process individual frames or short frame sequences. Results are aggregated by averaging over all frames of a video. As an alternative aggregation approach, the authors also employ an LSTM. In addition, they apply an object tracking method (Zhu et al., 2018) to locate the occluded snitch object. While the models work reasonably well for the simplest task of identifying atomic actions, performance breaks down to rather a mediocre accuracy for compositional action recognition. Snitch localization, finally, does not work well with any of the methods. In general, LSTM aggregation yields higher accuracy scores than aggregation by averages.

Various papers have addressed the tasks posed by CATER. Solutions for tasks one and two (atomic and compositional action recognition) have been proposed in Samel et al. (2022); Kim et al. (2022); Singh et al. (2022a). Solutions to the third task requiring hidden object localization have been proposed in Goyal et al. (2021); Ding et al. (2021a); Traub et al. (2022); Zhou et al. (2021); Zhang (2022); Harley et al. (2021); Faulkner & Zoran (2022); Castrejon et al. (2021); Sun et al. (2022); Luo et al. (2022). Beyond approaches to solve the proposed tasks, CATER has been used for object segmentation tasks in Kipf et al. (2021); Singh et al. (2022b); Bao et al. (2022); Frey et al. (2023) and unsupervised physical variable extraction in Kabra et al. (2021). Furthermore, some authors have created new datasets based on CATER, for object tracking tasks (see Van Hoorick et al. (2022); Shamsian et al. (2020)), for video generation from compositional action

graphs (see Bar et al. (2020)), for video generation based on images and text descriptions (see Hu et al. (2022); Xu et al. (2023)), and for dialogue systems concerning physical events (see Le et al. (2021; 2022)). All these additional datasets are approached by the respective authors specifically for particular tasks, but they nonetheless touch on various aspects of physical reasoning as well.

2.11 CLEVRER

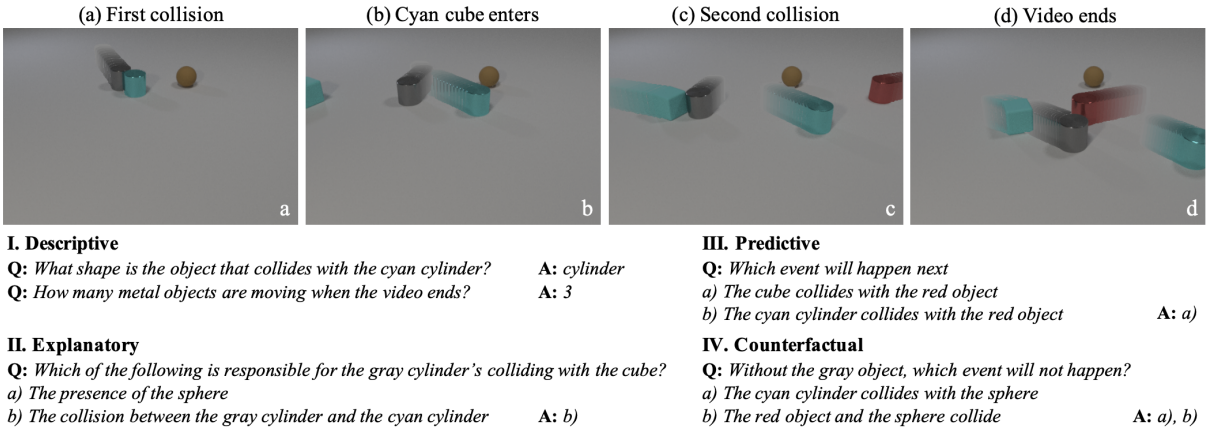


Figure 13: Examples of videos and questions in CLEVRER. They are designed to evaluate whether computational models can answer descriptive questions (I) about a video, explain the cause of events (II, explanatory), predict what will happen in the future (III, predictive), and imagine counterfactual scenarios (IV, counterfactual). In the four images (a–d), the blurred motion traces are only provided to reveal object motion to the human observer; they are absent in the input to the recognition system. Image from Yi et al. (2019).

Collision Events for Video Representation and Reasoning (CLEVRER) (Yi et al., 2019) is a benchmark that requires temporal and causal natural language question answering. Like CATER above, it extends CLEVR (Johnson et al., 2017) from images to short, artificially generated 5-second videos along with descriptive, predictive, explanatory, and counterfactual questions about their contents. Each video contains multiple instances of cubes, spheres, or cylinders that can have one of eight different colors and consist of either a shiny or a dull material. The combination of object geometry, material, and color is unique within a video, and all objects are placed on a white, planar surface. The objects are either sliding across the surface, resting, or colliding with each other (see Figure 13), with the videos generated such that often multiple cascading collisions occur. CLEVRER contains 10,000 training videos, 5,000 validation videos, and 5,000 test videos. They are accompanied by a total of 219,918 descriptive, 33,811 explanatory, 14,298 predictive, and 37,253 counterfactual questions, which were generated procedurally together with the videos. The training input consists of a video and a text question. Models treat each question as a multi-class classification problem over all possible answers. For example, in the case of descriptive questions, answers can be given in natural language, and in this case the correct answer is a single word, which can be specified with a model that outputs a softmax distribution over its vocabulary. Descriptive questions are evaluated by comparing the answered token to the ground truth, and the percentage of correct answers is reported. Multiple choice questions are evaluated based on two metrics: The per-option accuracy, which measures the correctness of the model regarding a single option across all questions, and the per-question accuracy, which measures the percentage of questions where all choices were correct.

In the three remaining categories, tasks are essentially binary classification problems. Answering the questions correctly is taken as an indicator for properly identifying, understanding, and predicting the dynamics of the video.

There are two baseline solutions. The first solution uses a convolutional LSTM to encode the video into a feature vector and then combines this feature vector with the input question using a MAC network (Shi

et al., 2015) to obtain a final prediction. The second baseline solution encodes the video information using a convolutional neural network (CNN) and encodes the input question using the average of the question's pre-trained word embeddings (Mikolov et al., 2013). These two embeddings are then passed to an MLP to make the final prediction.

Many subsequent studies have since provided solutions for the physical reasoning task on the CLEVRER benchmark. We categorize these works into two groups. The first group mainly uses graph convolutional network schemes as part of their models. These models use a trajectory extractor, graph networks, and a semantic parser and achieve state-of-the-art performances on CLEVRER without using ground-truth attributes and collision labels from simulations for training (Ding et al., 2021b; Chen et al., 2021; Lin et al., 2020; Wu et al., 2021; Chen et al., 2022; Le et al., 2021; Jia et al., 2022a; Zhong et al., 2022). The other group of approaches has hierarchical and dynamic inference neural network models. The key aspects of these models include self-attention and self-supervised learning (Ding et al., 2021a; Zhao et al., 2022; Xu et al., 2021; McDuff et al., 2022; Wu et al., 2022; Sautory et al., 2021; Patel et al., 2022a). Recent works by Zablotskaia et al. demonstrated good CLEVRER performances using an unsupervised model (Zablotskaia et al., 2020; 2021).

CLEVRER is most similar to the CATER (see Chapter 2.10), but also similar the CRAFT (see Chapter 2.5), IntPhys (see Chapter 2.9), and SPACE (see Chapter 2.7) benchmarks. Given a video input, these benchmarks require an agent to answer questions about object interactions. The primary difference is in what the agent is required to learn. CLEVRER and CRAFT are the most alike since they both ask the agent text-based descriptive (what happened in the video), explanatory/causal (why certain things happened), and counterfactual (what could have happened in the video) multiple-choice questions. However, CLEVRER asks predictive (what will happen in the video) questions. IntPhys requires the agent to predict whether the scenario is feasible or infeasible. CATER and SPACE require the agent to answer questions about actions that took place during the video. However, CATER requires the agent to recognize which actions took place during the video, and SPACE requires agents to answer only whether a particular action occurred in the video. CLEVRER is also similar to the ShapeStacks (see Chapter 2.6) and TRANCE (see Chapter 2.17 below) benchmarks since they require the agent to answer multiple choice questions about object interactions given visual input (video or images). However, CLEVRER receives video input, while TRANCE and ShapeStacks receive only single images as input.

2.12 ComPhy

ComPhy (Chen et al., 2022), shown in Figure 14, is a visual question answering benchmark for reasoning about hidden physical properties such as the charge and mass of objects. The authors speak of intrinsic as opposed to extrinsic properties, although we chose to call them internal as opposed to external variables in this survey (see also Section 1). In contrast to external, visible variables, internal variables are invisible and only become apparent through interaction between objects. This means internal variables are only observable over a time interval, rather than in any particular instant.

The benchmark consists of videos that show the temporal evolution of up to 5 simulated 3-dimensional objects that interact while moving on a plane. The objects have external properties of color, shape, and material and discrete internal properties of mass (light or heavy) and charge (negative, neutral, positive). Interaction between objects can take the form of attraction/repulsion or collision.

The videos come in atomic sets of 5, divided into 4 reference videos and one target video for few-shot learning. Reference videos are 2 seconds long and show object interaction but don't contain labels for physical properties, while the target video is 5 seconds long, comes with ground truth on physical properties (both internal and external variables), and only contains objects seen in at least one reference video. The actual train, validation, and test sets of the benchmark contain a few thousand atomic video sets, and while the physical properties of objects are consistent within atomic sets, they are assigned randomly between different atomic sets.

The task in ComPhy is to answer natural language questions either about facts in the target video, predictions about the evolution of the target video, or counterfactual questions concerning alternate outcomes if the internal properties of objects had been different. Factual questions are open-ended, while predictive and

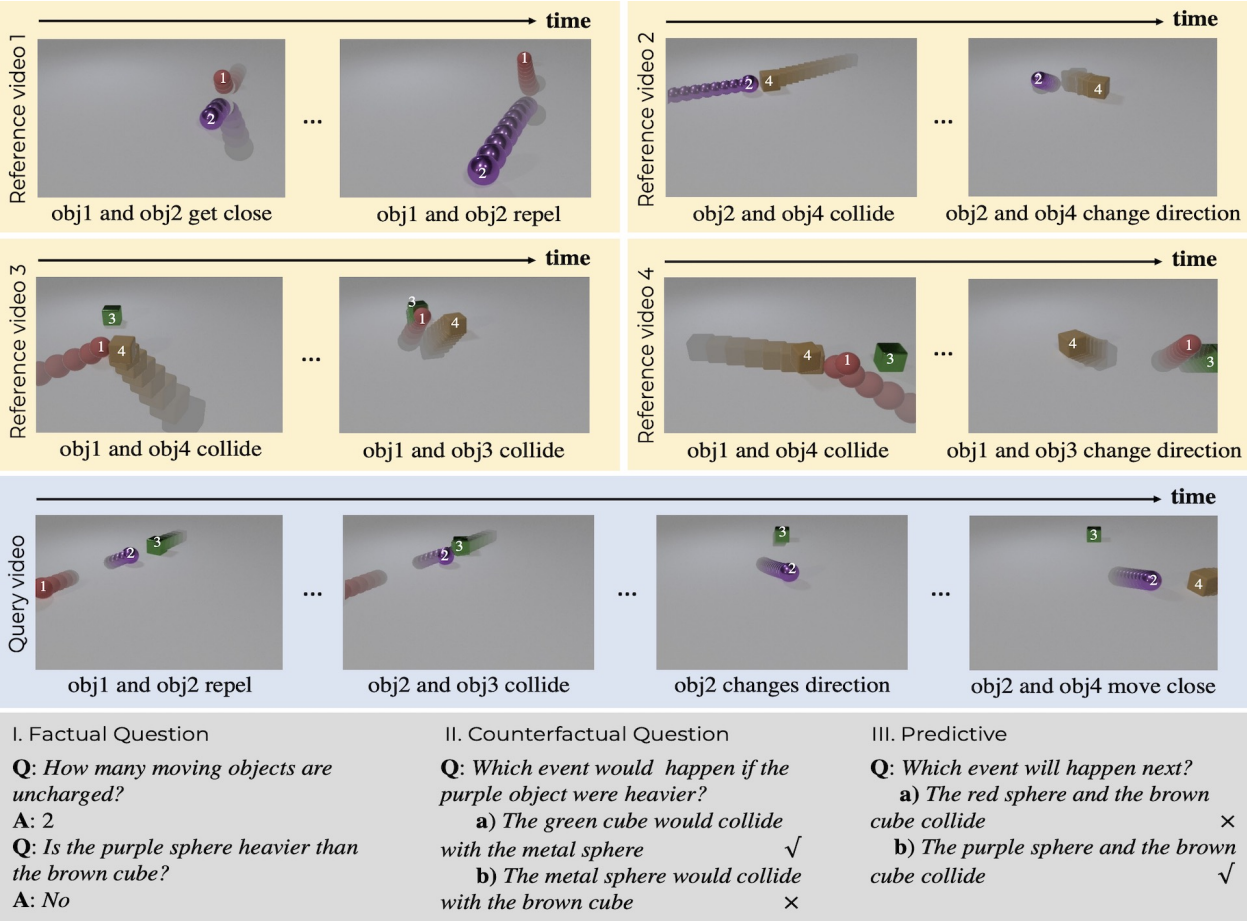


Figure 14: Sample target video, reference videos, and question-answer pairs from ComPhy. Image from Chen et al. (2022).

counterfactual questions need to be answered in multiple-choice, where several options can be simultaneously correct. This effectively amounts to multi-label classification.

As baseline solutions, the authors test adaptations of the models CNN-LSTM (Antol et al., 2015), HCRN (Le et al., 2020), MAC (Hudson & Manning, 2018) and ALOE (Ding et al., 2021a). They also provide a small study on human performance. Additionally, they provide their own solution approach called compositional physics learner. It consists of several modules which perform perception, property inference, dynamic prediction, and symbolic reasoning. This model achieves better performance than the baseline approaches but worse than the human testers.

The only work that has proposed an approach to address the ComPhy benchmark, as of now, is by Tang et al. (2023).

2.13 CRIPP-VQA

The Counterfactual Reasoning about Implicit Physical Properties Video Question Answering benchmark, in short CRIPP-VQA (Patel et al., 2022b), extends the field of classical video question answering into the domain of physical reasoning. It focuses on internal physical variables, specifically mass and friction.

The benchmark consists of 5000 videos, split into train, validation and test set, plus an additional 2000 videos to evaluate out-of-distribution generalization, where objects might have previously unseen physical properties. Each video is 5 seconds long and shows a 3d scene of computer-generated objects (cubes and

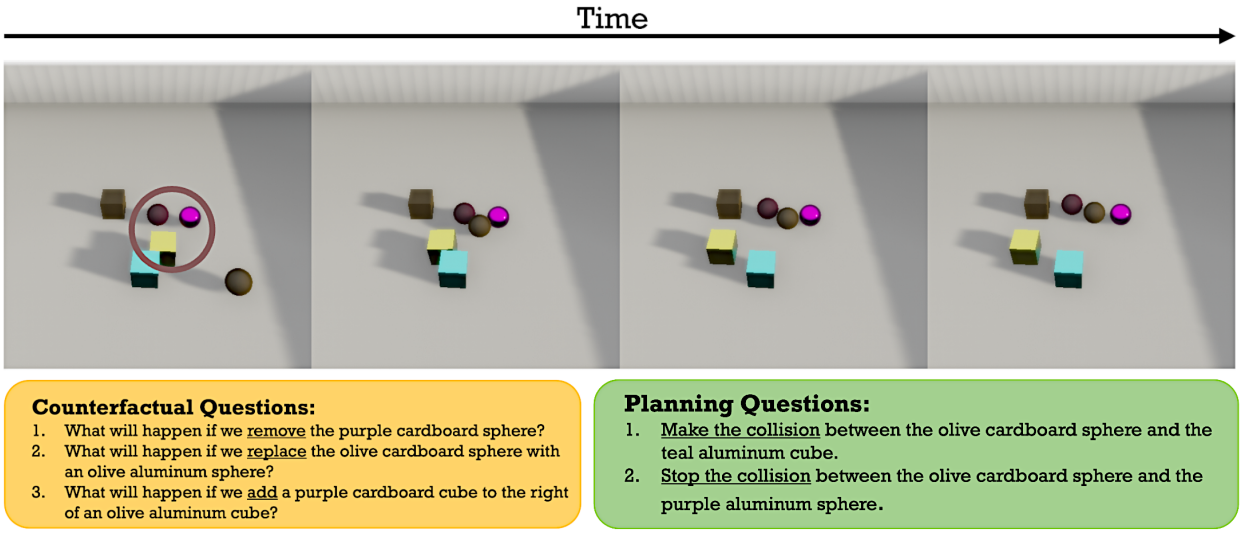


Figure 15: Stills from a video in CRIPP-VQA with exemplary counterfactual and planning questions. Image adapted from Patel et al. (2022b).

spheres) moving on a plane. Stills are shown in Figure 15. One object is initially in motion, or two in the out-of-distribution videos. This sets off a series of collisions which are affected by the friction and mass of each object.

Friction and mass of objects are assigned to unique combinations of external variables {Shape, Color, Texture}. This means mass and friction are important to extract but are not truly internal, as they can be deduced from individual frames once this mapping is known. This makes them effectively external. In the out-of-distribution videos, however, this assignment is removed and mass and friction become truly internal.

The videos are accompanied by about 100,000 natural language questions, each assigned to an individual video. Of these questions 45% are counterfactual, 44% descriptive and 11% concern planning. Descriptive questions ask facts about the scene, for instance about a count, material or color. Planning questions ask which action would have been necessary to instead obtain a certain different final state. The possible actions are to add, remove or replace objects. Both descriptive and planning questions require language answer tokens. The counterfactual questions ask how the final scene would have differed if the initial condition had been different and multiple natural language answers are provided together with the question. The task is to predict for each of the provided answers whether it is true of false, which means the counterfactual questions are essentially multilabel classification tasks.

For all categories and videos, however, success during evaluation is measured by accuracy in the answers given to questions. Either accuracy per option, in the counterfactual case, or accuracy per question.

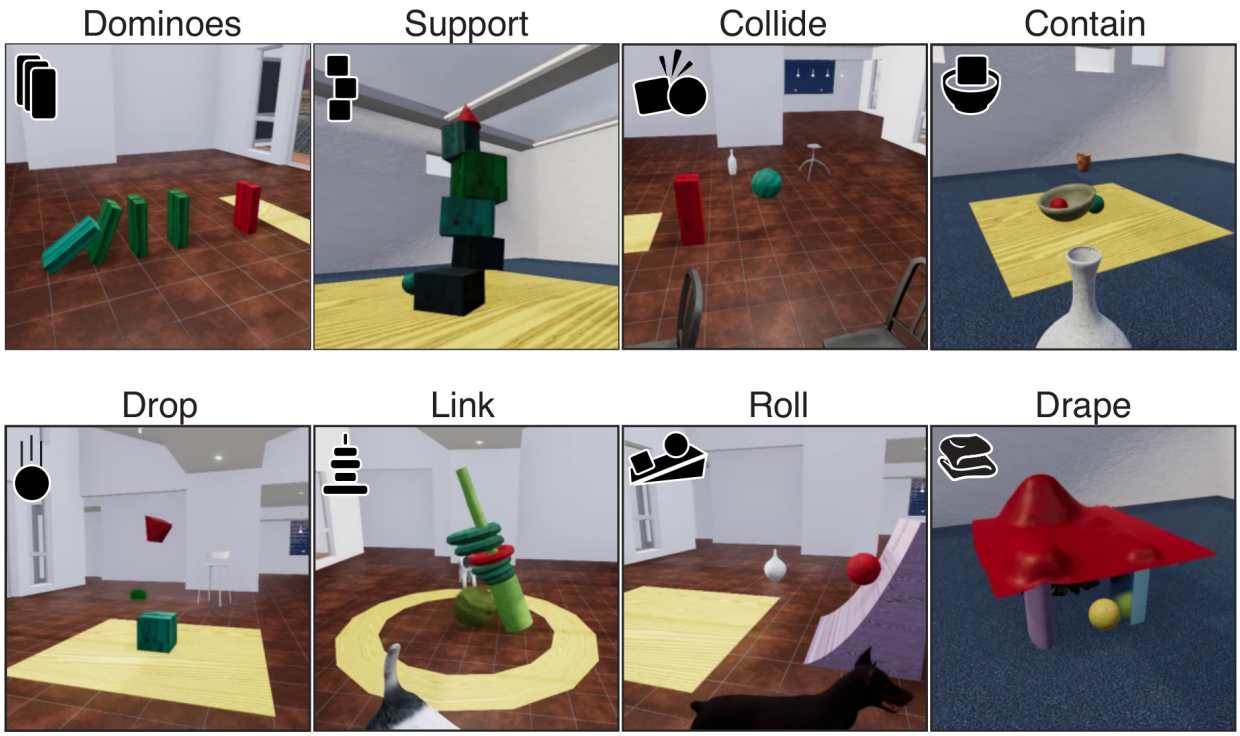
As baselines solutions, the authors test the MAC (Hudson & Manning, 2018) compositional VQA model, the HRCN VQA model (Le et al., 2020), and the Aloe visual reasoning model (Ding et al., 2021a), modified to work on Mask-RCNN features (He et al., 2017) and using BERT-generated word embeddings (Devlin et al., 2018). Results are varied, however the modified Aloe model generally performs best out of these three baselines. In the out-of-distribution case, the authors state that performance drops to close to random. In addition to these baseline models, the authors present results achieved by a small group of humans for comparison.

As of the time of writing, there are no further approaches proposed for CRIPP-VQA in the literature.

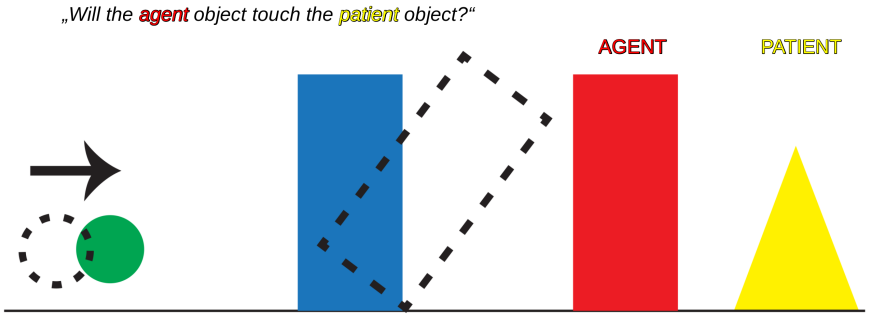
CRIPP-VQA is similar to CLEVRER (2.11) and ComPhy (2.12) since these depict similar 3d collision scenes on a plane and also involve natural language questions. However, as opposed to ComPhy, the part of CRIPP-VQA that is not out-of-distribution does not have truly internal variables and the questions types covered

in CRIPP-VQA do only partly overlap with both ComPhy and CLEVRER. Additionally, CRIPP-VQA has a focus on counterfactual questions, similar to CoPhy (2.8).

2.14 Physion



(a) Stills from the eight different physical concepts in the Physion benchmark.



(b) Simplified explanation of the Physion task.

Figure 16: Stills and task explanation of the Physion benchmark. The videos contain an agent (red) and patient (yellow), and possibly a probe (green) that moves and hence initiates the unfolding of the scene. Images are adapted from Bear et al. (2021).

Physion (Bear et al., 2021) contains a set of realistic 3d videos that are 5-10 seconds long and present 8 different physical interaction scenarios: dominoes, support, collide, contain, drop, link, roll and drape (see Figure 16a). The benchmark is evaluated on an object contact prediction (OCP) task, which asks whether two selected objects, called *agent* and *patient* (see Figure 16b) will touch throughout the video. A video ends after all objects come to rest.

The videos, also called stimuli, are rendered at 30 frames per second. The following data is supplied. 1.) visual data per frame: color image, depth map, surface normal vector map, object segmentation mask, and

optical flow map; 2.) physical state data per frame: object centroids, poses, velocities, surface meshes (which can be converted to particles), and the locations and normal vectors for object-object or object-environment collisions; 3.) stimulus-level labels and metadata: the model names, scales, and colors of each object; the intrinsic and extrinsic camera matrices; segmentation masks for the agent and patient object and object contact indicators; the times and vectors of any externally applied forces; and scenario-specific parameters, such as the number of blocks in a tower. All stimuli from all eight scenarios share a common OCP task structure. There is always one object designated the *agent*, and one object designated the *patient*, and most scenes have a *probe* object whose initial motion sets off a chain of physical events. Machine learning models and human test subjects are asked to predict whether the agent and patient object will have come into contact before or until the time all objects come to rest.

The Physion dataset consists of three different parts: *dynamics*, *readout*, and *test* sets. The *dynamics* training set contains full videos without *agent* and *patient* annotations or labels that indicate the outcome of the OCP task. It is meant to train representation or dynamics models from scratch. The *readout* training set presents only the first 1.5 seconds of videos and the corresponding OCP task labels to separately train the model for solving the OCP tasks with a frozen pre-trained dynamics model. Lastly, the *test* set contains only the initial 1.5 seconds of videos and OCP task labels and is meant to evaluate trained models.

While Physion is meant to benchmark machine learning models, the authors evaluate it on human volunteers to establish a baseline. Against this baseline, they compare a range of models which can be categorized into CNN-based vision models and graph-based dynamics models. Some of the vision models learn object-centric representations, while the graph-based dynamics models require a preexisting object-graph representation instead of raw visual input. Their experiments show that vision models perform worse than humans, although those with object-centric representations generally perform better than those without. The graph-based dynamics models can sometimes compete with humans. This leads to the authors concluding that the main bottleneck is learning object-centric representations from visual scenes.

Approaches to tackle the Physion benchmark have been proposed in Wu et al. (2022); Han et al. (2022); Nayebi et al. (2023). Physion videos are also used explicitly for the task of video prediction in Nayebi et al. (2023); Lu et al. (2023).

Physion++ is proposed in Tung et al. (2023) as a second, newer version of Physion. Its videos are based on the same physics engine as Physion, but the benchmark goes beyond Physion in that it focuses on internal variables and shows more object interactions in its videos. Models trained on Physion++ videos are thus expected to explicitly infer internal variables in order to solve its tasks. https://www.overleaf.com/project/63b6a3e02c72d209c1defb0c

2.15 Language Table

The Language Table dataset (Lynch et al., 2022) contains nearly 600,000 natural language-labeled trajectories of robotic arm moving blocks placed on a table according to language instruction (see Figure 17). An example instruction is: "Slide the yellow pentagon to the left side of the green star." This benchmark is intended for imitation learning for natural language-guided robotic manipulation. Each trajectory in the dataset records the state of a UFACTORY xArm6 robot with 6 joints and a video of the state of the table. The table is made of smooth wood and has 8 plastic blocks (4 colors and 6 shapes) placed on it. The dataset contains 413k trajectories gathered from real-world data and 181k simulated trajectories. The average episode length for real-world data is 9.9 minutes $\pm$ 5.6 seconds. The average episode length for simulated data is 36.8 seconds $\pm$ 15 seconds

Additionally, the authors release the *Language-Table environment*, which is a simulated environment closely matching the real-world setup used in the dataset. This environment is useful for evaluating potential solutions and hyper-parameter tuning. Using the Language Table environment they also create the Language Table benchmark. This benchmark computes automated metrics for 5 task families with a total of 696 unique task variations. The task families are as follows: block2block, block2abs, block2rel, block2blockrel, and separate. block2block tasks require the agent to push one block to another. block2abs tasks require the agent to push a block to an absolute location on the board (e.g. top-left, bottom-right, center, etc.). block2rel tasks require the agent to push a block in a relative offset direction (e.g. left, down, up and right,

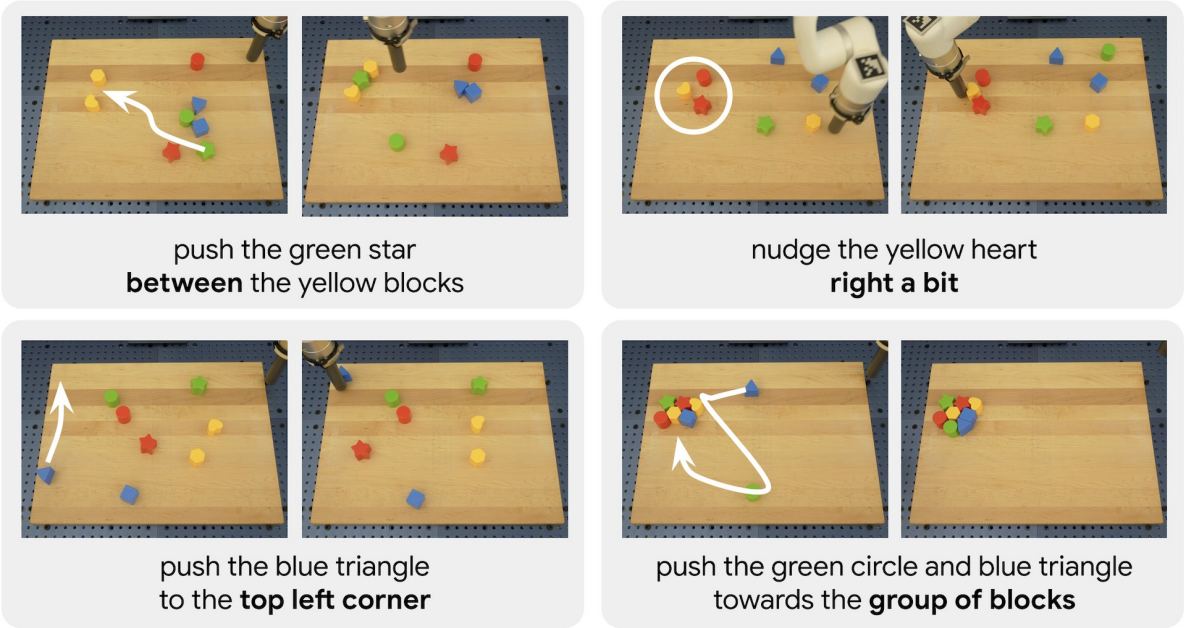


Figure 17: Interactive Language rollouts on a sample of the more than 87,000 crowdsourced natural language instructions with a wide variety of short-horizon open vocabulary behaviors. Image from Lynch et al. (2022).

etc.), block2blockrel tasks require the agent to push a block such that it is offset from the target block in a particular direction (left side, top right side, ... of block X). separate tasks require the agent to separate two blocks. For all tasks, success is a binary variable that is true if the distance between the source block and the target location/block is below a valid threshold.

To solve the Language Table benchmark the authors propose a multi-stage process. They begin by using a pre-trained ResNet model (He et al., 2016) to extract visual features from the current input video frame and use the CLIP (Radford et al., 2021) to embed the natural language instruction into a visual latent space. Next, they train a language-attends-to-vision transformer (Vaswani et al., 2017) whose keys and values are based on the ResNet embeddings, and queries are based on the CLIP embeddings. The output of this transformer is considered to contain the current frame's state. Afterward, the last n results from the language-attends-to-vision transformer, gathered from the last n frames, are then given to a temporal transformer whose output is given to a ResNet MLP, which will predict the current frame's action.

A method to solve the Language Table tasks has been proposed by Driess et al. (2023). Furthermore, Xiao et al. (2022); Rana et al. (2023); Yang et al. (2023) have used the Language Table dataset for training their models, although they have solved different tasks such as robotic behaviour prediction.

2.16 OPEn

OPEn, introduced by Gan et al. (2021)[3], is a 3D interactive, open-ended physics environment (see Figure 18). It is comprised of a table framed by raised borders, with objects placed on top of it. Interactions happen through a rolling agent that observes visual data and can move on the table. The agent can execute actions to move in one of eight directions for a fixed distance and interact with the objects in the scene by colliding with them. *OPEn* consists of two modes. The first, called sandbox, provides randomly generated non-episodic environments which are used to learn state representations or even a physics model by interactive exploration without a specific task. The second, the evaluation suite, expresses tasks through reward functions of a reinforcement learning environment. The tasks are: to move towards a goal, move towards a preferred

[3]The author's original link http://open.csail.mit.edu to this benchmark's website is outdated, please refer to Table 4 for a link to the GitHub repository instead.

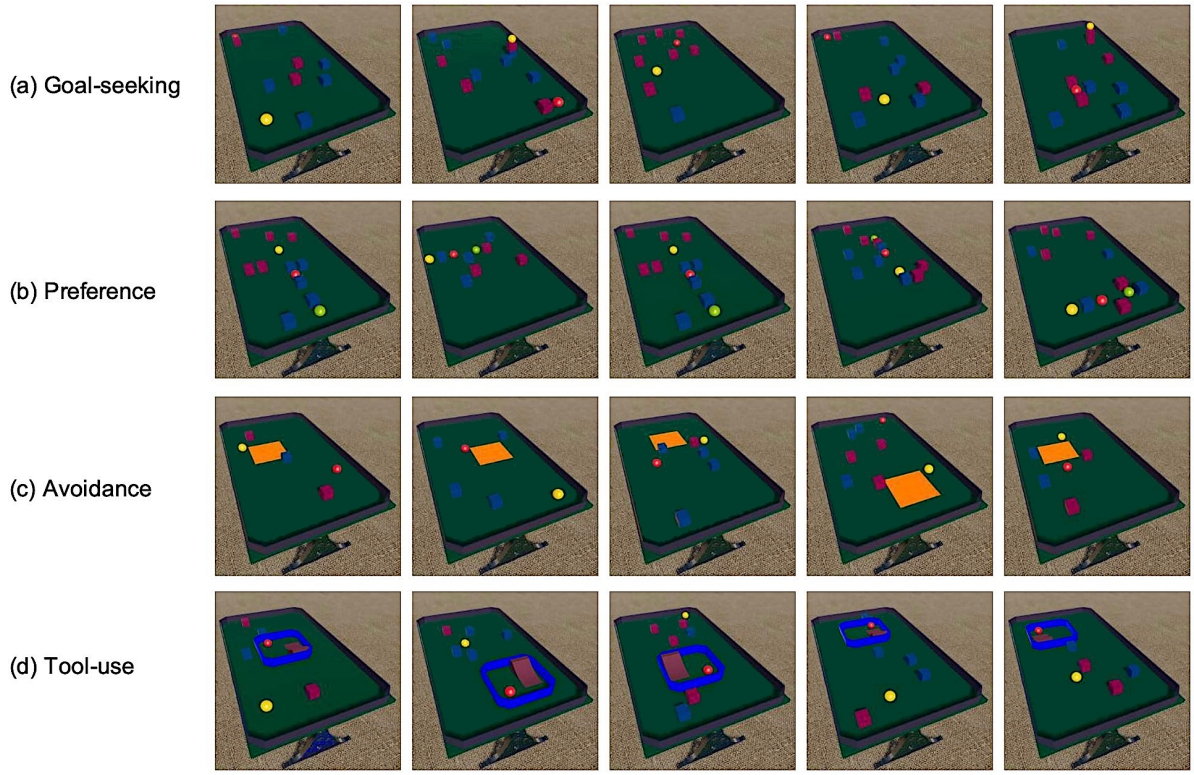


Figure 18: OPEn evaluation suite for 4 downstream physical reasoning tasks. Image from Gan et al. (2021).

object, avoid a region on the table, use a ramp to escape a region of the table, and seek a goal. The tasks are meant to assess models trained on the sandbox.

The authors test four baseline solutions on their task, i.e., they first run a sandbox and then the evaluation suite phase. The tested approaches are a vanilla PPO agent (Schulman et al., 2017), an intrinsic curiosity module (Pathak et al., 2017), random network distillation (Burda et al., 2018), and CURL (Laskin et al., 2020). Each of the latter three is also tested with RIDE rewards (Raileanu & Rocktäschel, 2020). The best-performing solution is CURL with RIDE rewards. However, the authors note that none of the baselines benefit meaningfully from the sandbox pretraining mode, which suggests that the used methods are unable to build rich, general world models in the absence of downstream tasks. The proposed baselines thus do not comprehensively solve the *OPEn* benchmark. Beyond these baselines, there are no proposed solutions to this benchmark as of the time of publication.

OPEn is similar to the *PHYRE* and *Virtual Tools* benchmarks in that it is interactive, but it is different in that it is 3-dimensional, requires a sequence of actions, and is more geared towards classical reinforcement learning, while *PHYRE* and *Virtual Tools* benchmarks only require a single action.

2.17 Other Benchmarks

Transformation Driven Visual Reasoning benchmark (TRANCE) (Hong et al., 2021) is considering two images, one for the initial state and the other for the final state. The aim is to infer the transformation of the objects between the two images. Transformation refers to any kind of change in the object. This new TRANCE benchmark evaluates how well different machines can understand the transformations. TRANCE is created based on the CLEVR dataset (Johnson et al., 2017), which depicts objects that are characterized by five types of attributes, including shape, size, color, material, and position; TRANCE adopts the same default values for the attributes of its objects as CLEVR. To start generating the benchmark starts with

randomly sampling a scene graph, similar to the initial step of CLEVR. In the second step, the questions and the answers are getting generated with a functional program based on the scene graph (the first step).

The *Watch & Move* benchmark (Netanyahu et al., 2022) is about one-shot imitation learning. An agent watches an "expert" rearrange 2d shapes and is then confronted with the same shapes in a different configuration. The agent needs to understand the goal of the expert demonstration and arrange its own shapes in a similar way. The authors propose to use graph-based equivalence mapping and inverse RL to achieve this.

3 Clusters

The ultimate goal in physical reasoning AI is to develop generalist, powerful physical reasoning agents. In this section, we propose four clusters of benchmarks, each representing a capability that we consider a critical part of a future generalist agent. Currently, there are only agents aiming to solve individual benchmarks. The next step in physical reasoning research will be to develop semi-generalist agents that perform well on multiple benchmarks within individual clusters. Eventually, a fully capable physical reasoning agent should be able to perform well across all clusters.

We propose to form these four clusters to probe the following four core capabilities: A general agent should (*i*) be interactive, i.e. able to explore the world by acting in it instead of just passively observing. It should also (*ii*) be able to recognise known physical concepts and categories. It should further (*iii*) be able to build models of world dynamics, to a point where it can extrapolate into the future and make predictions. Finally, agents should (*iv*) be capable of language in order to reason and communicate on abstract, yet semantically meaningful concepts rather than merely images, numbers or categories.

3.1 Interactive Benchmarks

From human and animal studies we know that active exploration of the world is an important prerequisite for building a solid understanding of the laws governing it. In that sense, interactive benchmarks provide the unique option of active hypothesis testing. The agent can – and has to – decide what to try next and how to fill the gaps in its world knowledge.

Interactive physical reasoning tasks are often formulated as reinforcement learning (RL) problem, since it caters to the sequential nature of physical processes and there is extensive prior work on balancing exploration and exploitation as well as dealing with uncertainty. We found that RL approaches are used in solution algorithms for all interactive benchmarks in our review.

Due to their interactive nature, interactive benchmarks usually have to rely on an internal simulator that dynamically reacts to inputs to generate data on the fly. It is up to the agent to collect useful and unbiased data, while non-interactive benchmarks may come with carefully balanced datasets. Since it is infeasible to test all possible input-output combinations of the simulator beforehand, it has to be as accurate and numerically stable as possible to achieve good extrapolation beyond what could be tested during development.[4]

While all interactive benchmarks listed in this survey (those carrying the tags SI or CI in Table 1) focus on achieving a specific final state of the simulation, other tasks such as question answering or counterfactual reasoning are conceivable and illustrate yet untapped potential of interactive physical reasoning tasks. To sum up, we see interactivity as an important requirement on the way to generalist physical agents and deem it to define one of the relevant clusters for semi-generalist agent development.

3.2 Concept Recognition Benchmarks

In physical reasoning, it is a difficult problem to learn explicit, basic physical concepts with a model that has no prior knowledge about physics. Agents and especially future generalist agents, however, benefit from explicit knowledge of physical concepts relevant to their reasoning task. Various current benchmarks

[4]An example for why inaccurate simulators are problematic are physics glitches that have been seen exploited by solution algorithms to circumvent or greatly simplify the proposed task.

Benchmark		Input Data	Target Data
2.1	PHYRE	**Images:** 256x256px 7-channel (each channel is distinct color)	3D or 6D action vectors for positions and radii of red ball(s)
2.2	Virtual Tools	**Images:** 600x600px RGB **Metadata:** Used tool, tool position, number of solution attempts	Integer representing the tool and 2D position vector
2.3	Phy-Q	**Images:** 480x640px RGB **Metadata:** Symbolic task representation in JSON format	Either integer $n \in \{0, ..., 179\}$ representing slingshot angle or 3D vector representing x/y coordinates of pulled back slingshot and the activation time of birds with special abilities
2.4	Physical Bongard Problems	**Images:** 102x102px RGB	Class label
2.5	CRAFT	**Images:** 256x256px RGB **Questions:** Natural language text	Class label
2.6	ShapeStacks	**Images:** 224x224px RGB, depth, object segmentation	Stability flag
2.7	SPACE	**Images:** 224x224px RGB, depth, object segmentation, surface normal, optical flow	**Video prediction:** RGB image of next frame **Scenarios:** True/false flag
2.8	CoPhy	**Images:** 448x448px RGB, depth, object segmentation	3D coordinates of objects
2.9	IntPhys	**Images:** 228x228px RGB, depth, object segmentation **Metadata:** Object positions, camera position, object IDs	Plausibility score between 0 and 1
2.10	CATER	**Images:** 320x240px RGB	Class label
2.11	CLEVRER	**Images:** 480x320px RGB **Questions:** Natural language text	**Descriptive questions:** Answer token **Other:** Class label
2.12	ComPhy	**Images:** 480x320px RGB **Questions:** Natural language text	**Factual questions:** Answer tokens **Other:** Class labels
2.13	CRIPP-VQA	**Images:** 512x512px RGB **Metadata:** Object locations, object velocities, object orientations, collision events **Questions:** Natural language text	**Counterfactual questions:** Boolean flags **Other:** Answer tokens
2.14	Physion	**Images:** 512x512px RGB, depth, object segmentation, surface normal, optical flow **Metadata:** Physical properties of objects, collision events, force vectors, model names of scene objects, additional segmentation masks, scenario-specific parameters (e.g. number of total objects in a stack)	Flag that indicates whether agent and target object touch or not
2.15	Language Table	**Images:** 180x320px RGB	Two 2D vectors representing gripper position and target, 512D instruction token vector
2.16	OPEn	**Images:** 168x168px RGB (can be configured to other resolutions)	Integer $n \in \{0, ..., 8\}$ representing the action label

Table 2: Input and target formats for all benchmarks. The *Input Data* column lists the kind and format of data a solution approach can make use of, the *Target Data* column lists training targets. For interactive benchmarks, the target data usually consists of actions which are input for a simulator and thus differs from the final performance measure. For non-interactive benchmarks, the target data equals the final performance measure or is easily mappable to it.

pose classification tasks, which effectively teach such concepts. By training on a physically meaningful classification target, an agent will learn to recognise the concept.

In the case of IntPhys (Section 2.9) for instance, the physical concept and classification target is physical plausibility, which requires object permanence, shape constancy and spatio-temporal continuity. The PBP benchmark (Section 2.4), on the other hand, requires identification of a concept that defines the difference between two physical scenes. In the case of CATER (Section 2.10), the classification target is temporal order of actions.

Physical scenarios can be decomposed into concepts on various levels of abstraction and granularity, as shown by these examples, which makes this cluster rather broad. Beyond IntPhys, PBPs and CATER it also contains CRAFT, ShapeStacks, SPACE, ComPhy and Physion. What they have in common, however, is that they require models to learn to recognise physical concepts. The remaining two benchmarks involving classification tasks, PHYRE and Virtual Tools, however, they are interactive benchmarks where the classification target concerns reaching a desirable state in the environment rather than recognising a physical concept.

In current benchmarks, classification or concept recognition happens for its own sake, and does not form part of more complex reasoning chains. In the future, however, this cluster might well extend to benchmarks that do not explicitly train models in a supervised manner on given concepts. Instead, benchmarks might expect models to learn concepts in an unsupervised way and as a part of more complex reasoning tasks. To evaluate concept understanding, they can then check whether concepts were explicitly learned during an evaluation phase.

3.3 World Model Benchmarks

World model benchmarks can be seen as a class of tasks that test the capability of an agent to use a world model to predict the consequences of different ways of acting in order to identify and compare feasible solution paths and their relative pro's and cons.

World models vary with respect to what they predict, e.g. forward models predict outcomes caused by inputs, inverse models predict inputs required to cause desired outputs. They also vary w.r.t. their domain and ranges (which can either be narrow or wide, discrete or continuous). Often, the modelled relationship is probabilistic, e.g., represented as a joint probability density of inputs and outputs. Depending on the entropy of this density, the model may provide very weak or very distinct predictions. This includes "model-free" approaches, where an agent starts with a "tabula rasa" that is shaped into a more or less predictive world model as a result of (supervised or unsupervised) learning, and on the other hand approaches that start with a very detailed world model that predicts deterministic and unique outcomes that are highly accurate within its domain of applicability. Models can also take other forms, such as, e.g. attention mechanisms, or value functions in reinforcement learning. Finally, adaptive models can also include predictions about the confidence (or confidence intervals) with (or within) which their predictions are valid.

Although a world model can be expected to boost performance in most physical reasoning tasks, it may be hard to obtain and the cost of obtaining and using it always has to be weighed off against its usefulness. For instance, tasks that contain high levels of branching (e.g. due to uncertainty or stochasticity) are usually no good candidates for world models as the model predictions are likely to quickly become unreliable. On the other hand, if the task dynamics exhibit only little or no branching at all, a world model is a promising tool to increase performance. One concrete example for this is by Ahmed et al. (2021), who demonstrate that adding the predictions of a learned dynamics model to their PHYRE solution algorithm notably increases performance compared to a variant without the dynamics model.

An ultimately generalist physical reasoning agent should be able to combine a set of world models, which complement each other w.r.t. different forms of reasoning, along with further factors, such as range, precision, reliability and computational effort in order to cope with the huge range of different physical contexts that need to be covered in real world situations. Moreover, the models with a clear focus on physics need to be aided by models that cover regularities "on top of" physics, such as mind states of other agents and how these are affected by factors that are in turn physical. This suggests architectures that arrange models in suitable hierarchies to avoid solving exhaustive and expensive searach problems, as can be seen e.g. in Ahmed et al.

(2021); Li et al. (2022a); Qi et al. (2020); Rajani et al. (2020). To avoid irrelevant simulation candidates in the first place, intuitive physics understanding could be leveraged as a guide for the precise world model so that the latter is used more economically. Whether being guided carefully or used for exhaustive search, we believe precise world models are an important and powerful tool in the arsenal of a physical reasoning agent. From the benchmarks we list in this survey, we see Phyre (2.1), Virtual Tools (2.2), Phy-q (2.3), and CRAFT (2.5) as core members of this cluster since their dynamics and visuals are simple enough to allow for learning a powerful world model and sequence data of the tasks is readily available. For benchmarks with higher visual fidelity and/or more interacting objects, learning a world model can be harder in practice and thus may be less likely to increase performance, although various solution approaches for ShapeStacks (2.6) and SPACE (2.7) use world models as well.

3.4 Language-related benchmarks

Benchmarks that comprise natural language processing or generation extend physical reasoning problems to the domain of natural language processing. These benchmarks require agents to harness the descriptive power of language to assess, describe and/or solve physical problems. In contrast to images, language can represent different aspects of a physical process in varying levels of abstraction and detail, potentially putting the focus on certain aspects while neglecting others. However, this comes at the cost of higher potential for ambiguity, redundancy, synonymy and variation. Another advantage of using natural language is an easier comparison of state-of-the-art agents to human physics understanding. On the other hand, adding language understanding and maybe even generation to the stack of problems makes natural language-based physical reasoning benchmarks a considerably harder problem class in general. Nonetheless, we see natural language physical reasoning benchmarks as an important cluster to master on the way to a generalist agent.

Benchmark		Visual Input	Question Format	Output Format	Evaluation Criteria
2.5	CRAFT	video	raw text	single-label P(all_Token|Q)	single-CA
2.6	ShapeStacks	image-IS	constant	binary	multi-CA
2.7	SPACE**	image-IS	constant in sub-task	multi-label	multi-CA
2.10	CATER*	video	constant in sub-task	single-label, multi-label	single-CA, multi-CA
2.11	CLEVRER	video	raw text	multi-label P(all_Token|Q)	single-CA, multi-CA
2.13	CRIPP-VQA	video	raw text	multi-label P(all Token|Q)	single-CA, multi-CA
2.17	TRANCE	image-IS, image-FS	constant	sequence of transformations	Seq Acc, Dist

Table 3: Comparison of all benchmarks which require the agent to answer questions about a given visual input (i.e., video or images). Section 3.4 provides a legend for this table.

We create Table 3 as a way to compare the language-related benchmarks listed within this survey. Of these benchmarks, we consider CRAFT 2.5, CLEVRER 2.11, and CRIPP-VQA 2.13 as the core members because they are the only benchmarks which require agents to answer arbitrary questions provided via raw text. This requires agents to possess language understanding ability to solve these benchmarks. Additionally, the ShapeStacks 2.6, SPACE 2.7, CATER 2.10, and TRANCE 2.17 benchmarks also require agents to answer questions based on visual input. However, we do not consider these benchmarks to be core members because the questions to be answered are implicit within a sub-task, meaning that an understanding of natural language is not strictly required to solve these benchmarks.

Below we provide a description of all terms contained in Table 3.

The benchmarks column contains the names of the benchmarks, with an asterisk ***** denoting that the benchmark's tasks are independent and a double asterisk ****** indicating that question answering is a potential alternate focus of the benchmark.

The visual input format column specifies the type of visual input given to the model. **image-IS** denotes that only an image of the initial state is given and **image-FS** denotes that only an image of the final state is given.

The question input format column specifies how the questions are presented to the models during evaluation. The notation **constant in sub-task** implies that the question is implicit and given within the context of the current task, treating each task as a separate dataset. On the other hand, **constant** indicates that the question remains constant and implicit across all test examples.

The output format column outlines the expected format of the models' predictions. **multi-label P(all_Tokens | Q)** represents a multi-label probability distribution over all possible answers conditioned on the current question and input, allowing for multiple correct answers. Similarly, **single-label P(all_Tokens | Q)** represents a single-label probability distribution over all possible answers conditioned on the current question and input, permitting only one correct answer. **multi-label** represents a multi-label probability distribution over a small set of possible answers conditioned on the current question and input, allowing for multiple correct answers. **single-label** represents a single-label probability distribution over a small set of possible answers conditioned on the current question and input, permitting only one correct answer.

The evaluation criteria column outlines the metrics used to assess the performance of the models. **multi-CA** represents multi-label classification accuracy. **single-CA** represents single-label classification accuracy. **binary-CA** denotes binary classification accuracy. Given a sequence of transformations, the **Seq Acc** evaluation criteria outputs a binary label describing whether the sequence describes a valid solution. **Dist** quantifies the distance between the true final state and the final state described by the predicted transformations.

Benchmark		Website	Concepts	Variables	Scene
2.1	PHYRE	Link	Collision, Falling	G, E, T	2D simplistic
2.2	Virtual Tools	Link	Collision, Falling	G, E, T	2D simplistic
2.3	Phy-Q	Link	Collision, Falling	G, E, T	2D realistic
2.4	Physical Bongard Problems	Link	Containment, Falling, Collision	G, E	2D simplistic
2.5	CRAFT	Link	Collision, Falling	G, E, T	2D simplistic
2.6	ShapeStacks	Link	Falling, Stacking	G, E	3D realistic
2.7	SPACE	Link	Collision, Falling, Occlusion, Containment	G, E, T	3D simplistic
2.8	CoPhy	Link	Collision, Falling, Stacking	G, E, T, I	3D realistic
2.9	IntPhys	Link	Occlusion	G, E, T	3D realistic
2.10	CATER	Link	Occlusion, Containment, Lifting	G, E, T	3D simplistic
2.11	CLEVRER	Link	Collision	G, E, T	3D simplistic
2.12	ComPhy	Link	Collision, Attraction / Repulsion	G, E, T, I	3D simplistic
2.13	CRIPP-VQA	Link	Collision	G, E, T	3D simplistic
2.14	Physion	Link	Collision, Falling, Stacking, Containment, Draping	G, E, T	3D realistic
2.15	Language Table	Link	Collision	G, E	3D simplistic
2.16	OPEn	Link	Collision	G, E, T	3D simplistic

Table 4: Benchmarks categorised according to their concepts, physical variable types, and scene composition. G, E, T, and I stand for global, extrinsic, temporal, and intrinsic physical variables (see Section 1). If internal variables are not explicitly stated this means that relevant internal variables are encoded through (and therefore become) external variables. Physical concepts are to some extent a question of abstraction (e.g. bouncing implies collision and deformation) so the concepts column here summarizes general and somewhat orthogonal base concepts.

Benchmark		Agent	Reasoning Task
2.1	PHYRE	Interactive	Make objects touch
2.2	Virtual Tools	Interactive	Make objects touch
2.3	Phy-Q	Interactive	Hit objects with slingshot
2.4	Physical Bongard Problems	Passive	Recognize conceptual differences (D)
2.5	CRAFT	Passive	Answer questions (D, E, C)
2.6	ShapeStacks	Passive	Stability prediction (P)
2.7	SPACE	Passive	Recognize containment or interaction (D), or future frame prediction (P)
2.8	CoPhy	Passive	Predict future state from changed initial state (P, C)
2.9	IntPhys	Passive	Judge physical feasibility (D)
2.10	CATER	Passive	Recognize compositions of object movements (D)
2.11	CLEVRER	Passive	Answer questions (D, P, E, C)
2.12	ComPhy	Passive	Answer questions (D, P, C)
2.13	CRIPP-VQA	Passive	Answer questions (D, C), Planning
2.14	Physion	Passive	Predict object contact (P)
2.15	Language Table	Interactive	Push objects to absolute or relative positions
2.16	OPEn	Interactive	Push objects to relative positions

Table 5: Benchmarks categorised according to their reasoning tasks. D, P, E, C stand for the common categories of descriptive, predictive, explanatory and counterfactual tasks, defined for instance in Yi et al. (2019).

4 Discussion

In this work we have compiled a comprehensive collection of physical reasoning benchmarks that encompass various input and output formats across different task domains. Each benchmark is accompanied by a detailed description, including information about input-output formats, task nature, similarities to other benchmarks, and solution approaches. None of the physical reasoning benchmarks we have encountered so far encompasses all types of physical reasoning, and none of them pose challenges in all available task dimensions (see Table 1). We propose the utilization of clusters to address these challenges, employing semi-generalist agents that can serve as a stepping stone towards the development of a truly generalist physical reasoning agent.

Creating benchmarks that do not offer shortcuts for machine learning approaches can pose a challenge. To address this, we propose a clear and formal description of the physical phenomena that should be tested. Additionally, we provide a comprehensive and distinct breakdown of all the components comprising a physical reasoning benchmark in Section 1. This approach aims to facilitate both the detection of possible shortcuts and a better understanding of the benchmark's underlying structure.

Regardless of whether language processing or interaction is involved, an orthogonal classification can be made based on the visual complexity of the benchmarks. Based on available solutions for the presented benchmarks, we conjecture that higher visual fidelity tends to correlate with higher benchmark difficulty if all other variables are kept fixed. However, depending on the task even a simplistic 2D task environment can already bring state-of-the-art agents to their limits. We argue that a generalist agent should solve both visually simple as well as complex tasks and that visually challenging benchmarks are necessary to achieve this goal. However, we perceive the visual complexity to be more or less orthogonal to the physical reasoning difficulty. Thus, one way to obtain a capable generalist agent could be a greater focus on curriculum learning w.r.t. not only the task difficulty but also the visuals. While we did not find any benchmark that provides this feature, we believe that smoothly ramping up input along with task complexity would be helpful in the creation of a generalist physical reasoning agent.

Drawing from our comprehensive benchmark comparison, we offer a taxonomy of physical reasoning benchmarks by clustering existing works and explaining descriptive properties of each cluster. In addition, we aim to shed light on existing gaps in the current benchmark landscape. One such gap is the often overlooked concept of internal physical variables, which is only explicitly implemented in the ComPhy and Physion++

benchmarks within our collection. While it is important to prioritize mastery of fundamental concepts, we believe that the advanced notion of internal physical variables deserves greater attention and should be incorporated into state-of-the-art physical reasoning benchmarks.

As the field continues to evolve, we anticipate the inclusion of new candidates to expand this collection of benchmarks. By regularly incorporating emerging benchmarks, we can ensure that our evaluation framework remains up-to-date and comprehensive.

The majority of datasets we present in this context are generated through simulation. While these datasets may offer a certain level of detail in the generated samples, they still fall short in terms of the complexity and noise found in real-world data. Notably, several benchmark papers have indicated or demonstrated that state-of-the-art models often struggle when confronted with real-world data. Related to this idea is the concept of domain randomization. In cases where an accurate mapping of the real world is challenging, domain randomization involves creating multiple instances of simulated domains with the hope that their combined characteristics encapsulate those of the real world. A well-known example of domain randomization is the OpenAI Rubik's Cube paper (Akkaya et al., 2019), which demonstrates how this technique can be applied effectively.

It is essential to acknowledge that there still exists a substantial amount of unexplored terrain. In light of this, it is imperative for future benchmark authors to clearly define the physical concepts encompassed within their benchmarks, identify the types of physical variables that are relevant, and address the possibility of interactions with the environment. Future benchmarks can contribute to the continued advancement of our understanding of physical reasoning and pave the way for more comprehensive assessments of cognitive abilities across AI agents.

References

Eltayeb Ahmed, Anton Bakhtin, Laurens van der Maaten, and Rohit Girdhar. Physical reasoning using dynamics-aware models. *arXiv preprint arXiv:2102.10336*, 2021.

Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, et al. Solving rubik's cube with a robot hand. *arXiv preprint arXiv:1910.07113*, 2019.

Anirudh Goyal ALIAS PARTH GOYAL, Aniket Didolkar, Nan Rosemary Ke, Charles Blundell, Philippe Beaudoin, Nicolas Heess, Michael C Mozer, and Yoshua Bengio. Neural production systems. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 25673–25687. Curran Associates, Inc., 2021. URL `https://proceedings.neurips.cc/paper/2021/file/d785bf9067f8af9e078b93cf26de2b54-Paper.pdf`.

Kelsey R Allen, Kevin A Smith, and Joshua B Tenenbaum. Rapid trial-and-error learning with simulation supports flexible tool use and physical reasoning. *Proceedings of the National Academy of Sciences*, 117 (47):29302–29310, 2020.

Kelsey R Allen, Yulia Rubanova, Tatiana Lopez-Guevara, William Whitney, Alvaro Sanchez-Gonzalez, Peter Battaglia, and Tobias Pfaff. Learning rigid dynamics with face interaction graph networks. *arXiv preprint arXiv:2212.03574*, 2022.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

Tayfun Ates, M Samil Atesoglu, Cagatay Yigit, Ilker Kesen, Mert Kobas, Erkut Erdem, Aykut Erdem, Tilbe Goksun, and Deniz Yuret. Craft: A benchmark for causal reasoning about forces and interactions. *arXiv preprint arXiv:2012.04293*, 2020.

Anton Bakhtin, Laurens van der Maaten, Justin Johnson, Laura Gustafson, and Ross Girshick. Phyre: A new benchmark for physical reasoning. *Advances in Neural Information Processing Systems*, 32, 2019.

Zhipeng Bao, Pavel Tokmakov, Allan Jabri, Yu-Xiong Wang, Adrien Gaidon, and Martial Hebert. Discovering objects that can move. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11789–11798, 2022.

Amir Bar, Roei Herzig, Xiaolong Wang, Anna Rohrbach, Gal Chechik, Trevor Darrell, and Amir Globerson. Compositional video synthesis with action graphs. *arXiv preprint arXiv:2006.15327*, 2020.

Fabien Baradel, Natalia Neverova, Julien Mille, Greg Mori, and Christian Wolf. Cophy: Counterfactual learning of physical dynamics. In *International Conference on Learning Representations*, 2020.

Daniel M Bear, Elias Wang, Damian Mrowca, Felix J Binder, Hsiao-Yu Fish Tung, RT Pramod, Cameron Holdaway, Sirui Tao, Kevin Smith, Fan-Yun Sun, et al. Physion: Evaluating physical prediction from vision in humans and machines. *arXiv preprint arXiv:2106.08261*, 2021.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018.

Lluis Castrejon, Nicolas Ballas, and Aaron Courville. Inferno: Inferring object-centric 3d scene representations without supervision. 2021.

Michael Chang, Thomas L. Griffiths, and Sergey Levine. Object representations as fixed points: Training iterative refinement algorithms with implicit differentiation, 2022. URL `https://arxiv.org/abs/2207.00787`.

Zhenfang Chen, Jiayuan Mao, Jiajun Wu, Kwan-Yee Kenneth Wong, Joshua B Tenenbaum, and Chuang Gan. Grounding physical concepts of objects and events through dynamic visual reasoning. *arXiv preprint arXiv:2103.16564*, 2021.

Zhenfang Chen, Kexin Yi, Yunzhu Li, Mingyu Ding, Antonio Torralba, Joshua B Tenenbaum, and Chuang Gan. Comphy: Compositional physical reasoning of objects and events from videos. *arXiv preprint arXiv:2205.01089*, 2022.

Arijit Dasgupta, Jiafei Duan, Marcelo H Ang Jr, Yi Lin, Su-hua Wang, Renée Baillargeon, and Cheston Tan. A benchmark for modeling violation-of-expectation in physical reasoning across event categories. *arXiv preprint arXiv:2111.08826*, 2021.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

David Ding, Felix Hill, Adam Santoro, Malcolm Reynolds, and Matt Botvinick. Attention over learned object embeddings enables complex visual reasoning. *Advances in neural information processing systems*, 34:9112–9124, 2021a.

Mingyu Ding, Zhenfang Chen, Tao Du, Ping Luo, Josh Tenenbaum, and Chuang Gan. Dynamic visual reasoning by learning differentiable physics models from video and language. *Advances In Neural Information Processing Systems*, 34:887–899, 2021b.

Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

Yilun Du, Kevin Smith, Tomer Ulman, Joshua Tenenbaum, and Jiajun Wu. Unsupervised discovery of 3d physical objects from video. *arXiv preprint arXiv:2007.12348*, 2020.

Jiafei Duan, Samson Yu Bai Jian, and Cheston Tan. SPACE: A simulator for physical interactions and causal learning in 3d environments. *CoRR*, abs/2108.06180, 2021. URL `https://arxiv.org/abs/2108.06180`.

Jiafei Duan, Samson Yu, Soujanya Poria, Bihan Wen, and Cheston Tan. Pip: Physical interaction prediction via mental simulation with span selection. In *European Conference on Computer Vision*, pp. 405–421. Springer, 2022.

Sebastien Ehrhardt, Oliver Groth, Aron Monszpart, Martin Engelcke, Ingmar Posner, Niloy Mitra, and Andrea Vedaldi. Relate: Physically plausible multi-object scene synthesis using structured latent spaces. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 11202–11213. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/806beafe154032a5b818e97b4420ad98-Paper.pdf.

Patrick Emami, Pan He, Sanjay Ranka, and Anand Rangarajan. Towards improving the generation quality of autoregressive slot vaes, 2022. URL https://arxiv.org/abs/2206.01370.

Martin Engelcke, Oiwi Parker Jones, and Ingmar Posner. Reconstruction bottlenecks in object-centric generative models. *CoRR*, abs/2007.06245, 2020a. URL https://arxiv.org/abs/2007.06245.

Martin Engelcke, Adam R. Kosiorek, Oiwi Parker Jones, and Ingmar Posner. Genesis: Generative scene inference and sampling with object-centric latent representations. In *International Conference on Learning Representations*, 2020b. URL https://openreview.net/forum?id=BkxfaTVFwH.

Martin Engelcke, Oiwi Parker Jones, and Ingmar Posner. Genesis-v2: Inferring unordered object representations without iterative refinement. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 8085–8094. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/43ec517d68b6edd3015b3edc9a11367b-Paper.pdf.

Ryan Faulkner and Daniel Zoran. Solving reasoning tasks with a slot transformer. *arXiv preprint arXiv:2210.11394*, 2022.

Markus Frey, Christian F Doeller, and Caswell Barry. Probing neural representations of scene perception in a hippocampally dependent task using artificial neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2113–2121, 2023.

Fabian B. Fuchs, Oliver Groth, Adam R. Kosiorek, Alex Bewley, Markus Wulfmeier, Andrea Vedaldi, and Ingmar Posner. Scrutinizing and de-biasing intuitive physics with neural stethoscopes, 2018. URL https://arxiv.org/abs/1806.05502.

Chuang Gan, Abhishek Bhandwaldar, Antonio Torralba, Joshua B Tenenbaum, and Phillip Isola. Open: An open-ended physics environment for learning without a task. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5878–5885. IEEE, 2021.

Rohit Girdhar and Deva Ramanan. Cater: A diagnostic dataset for compositional actions and temporal reasoning. *arXiv preprint arXiv:1910.04744*, 2019.

Rohit Girdhar, Laura Gustafson, Aaron Adcock, and Laurens van der Maaten. Forward prediction for physical reasoning. In *ICML Workshop*, 2021.

Anirudh Goyal, Aniket Didolkar, Alex Lamb, Kartikeya Badola, Nan Rosemary Ke, Nasim Rahaman, Jonathan Binas, Charles Blundell, Michael Mozer, and Yoshua Bengio. Coordination among neural modules through a shared global workspace. *arXiv preprint arXiv:2103.01197*, 2021.

Oliver Groth, Fabian B Fuchs, Ingmar Posner, and Andrea Vedaldi. Shapestacks: Learning vision-based physical intuition for generalised object stacking. In *Computer Vision–ECCV 2018: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part I*, pp. 724–739, 2018.

Jiaqi Han, Wenbing Huang, Hengbo Ma, Jiachen Li, Josh Tenenbaum, and Chuang Gan. Learning physical dynamics with subequivariant graph neural networks. *Advances in Neural Information Processing Systems*, 35:26256–26268, 2022.

Adam W Harley, Yiming Zuo, Jing Wen, Ayush Mangal, Shubhankar Potdar, Ritwick Chaudhry, and Katerina Fragkiadaki. Track, check, repeat: An em approach to unsupervised tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16581–16591, 2021.

Augustin Harter, Andrew Melnik, Gaurav Kumar, Dhruv Agarwal, Animesh Garg, and Helge Ritter. Solving physics puzzles by reasoning about paths. *arXiv preprint arXiv:2011.07357*, 2020a.

Augustin Harter, Andrew Melnik, Gaurav Kumar, Dhruv Agarwal, Animesh Garg, and Helge Ritter. Solving physics puzzles by reasoning about paths. *arXiv preprint arXiv:2011.07357*, 2020b.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

Xin Hong, Yanyan Lan, Liang Pang, Jiafeng Guo, and Xueqi Cheng. Transformation driven visual reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6903–6912, 2021.

Yaosi Hu, Chong Luo, and Zhenzhong Chen. Make it move: controllable image-to-video generation with text descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18219–18228, 2022.

Drew A Hudson and Christopher D Manning. Compositional attention networks for machine reasoning. *arXiv preprint arXiv:1803.03067*, 2018.

Steeven Janny, Fabien Baradel, Natalia Neverova, Madiha Nadri, Greg Mori, and Christian Wolf. Filtered-cophy: Unsupervised learning of counterfactual physics in pixel space. *arXiv preprint arXiv:2202.00368*, 2022.

Baoxiong Jia, Ting Lei, Song-Chun Zhu, and Siyuan Huang. Egotaskqa: Understanding human tasks in egocentric videos. *arXiv preprint arXiv:2210.03929*, 2022a.

Baoxiong Jia, Yu Liu, and Siyuan Huang. Improving object-centric learning with query optimization, 2022b. URL `https://arxiv.org/abs/2210.08990`.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2901–2910, 2017.

Rishabh Kabra, Daniel Zoran, Goker Erdogan, Loic Matthey, Antonia Creswell, Matt Botvinick, Alexander Lerchner, and Chris Burgess. Simone: View-invariant, temporally-abstracted object representations via unsupervised video decomposition. *Advances in Neural Information Processing Systems*, 34:20146–20159, 2021.

Kiyoon Kim, Shreyank N Gowda, Oisin Mac Aodha, and Laura Sevilla-Lara. Capturing temporal information in a single frame: Channel sampling strategies for action recognition. *arXiv preprint arXiv:2201.10394*, 2022.

Thomas Kipf, Gamaleldin F Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional object-centric learning from video. *arXiv preprint arXiv:2111.12594*, 2021.

Michael Laskin, Aravind Srinivas, and Pieter Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning. In *International Conference on Machine Learning*, pp. 5639–5650. PMLR, 2020.

Hung Le, Chinnadhurai Sankar, Seungwhan Moon, Ahmad Beirami, Alborz Geramifard, and Satwik Kottur. Dvd: A diagnostic dataset for multi-step reasoning in video grounded dialogue. *arXiv preprint arXiv:2101.00151*, 2021.

Hung Le, Nancy F Chen, and Steven CH Hoi. Multimodal dialogue state tracking. *arXiv preprint arXiv:2206.07898*, 2022.

Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. Hierarchical conditional relation networks for video question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9972–9981, 2020.

Minne Li, Mengyue Yang, Furui Liu, Xu Chen, Zhitang Chen, and Jun Wang. Causal world models by unsupervised deconfounding of physical dynamics, 2020. URL `https://arxiv.org/abs/2012.14228`.

Shiqian Li, Kewen Wu, Chi Zhang, and Yixin Zhu. On the learning mechanisms in physical reasoning, 2022a. URL `https://arxiv.org/abs/2210.02075`.

Zongzhao Li, Xiangyu Zhu, Zhen Lei, and Zhaoxiang Zhang. Deconfounding physical dynamics with global causal relation and confounder transmission for counterfactual prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 1536–1545, 2022b.

Zhixuan Lin, Yi-Fu Wu, Skand Peri, Bofeng Fu, Jindong Jiang, and Sungjin Ahn. Improving generative imagination in object-centric world models. In *International Conference on Machine Learning*, pp. 6140–6149. PMLR, 2020.

Haoyu Lu, Guoxing Yang, Nanyi Fei, Yuqi Huo, Zhiwu Lu, Ping Luo, and Mingyu Ding. Vdt: An empirical study on video diffusion with transformers. *arXiv preprint arXiv:2305.13311*, 2023.

Calvin Luo, Ting Chen, Boqing Gong, and Chen Sun. Towards a unified neural architecture for visual recognition and reasoning. 2022.

Gary Lupyan and Martin Zettersten. Does vocabulary help structure the mind? In *Minnesota Symposia on Child Psychology: Human Communication: Origins, Mechanisms, and Functions*, volume 40, pp. 160–199. Wiley Online Library, 2021.

Corey Lynch, Ayzaan Wahid, Jonathan Tompson, Tianli Ding, James Betker, Robert Baruch, Travis Armstrong, and Pete Florence. Interactive language: Talking to robots in real time. *arXiv preprint arXiv:2210.06407*, 2022.

Federico Malato, Florian Leopold, Ville Hautamaki, and Andrew Melnik. Behavioral cloning via search in embedded demonstration dataset. *arXiv preprint arXiv:2306.09082*, 2023.

Daniel McDuff, Yale Song, Jiyoung Lee, Vibhav Vineet, Sai Vemprala, Nicholas Alexander Gyde, Hadi Salman, Shuang Ma, Kwanghoon Sohn, and Ashish Kapoor. Causalcity: Complex simulations with agency for causal discovery and reasoning. In *Conference on Causal Learning and Reasoning*, pp. 559–575. PMLR, 2022.

Andrew Melnik, Felix Schüler, Constantin A Rothkopf, and Peter König. The world as an external memory: the price of saccades in a sensorimotor task. *Frontiers in behavioral neuroscience*, 12:253, 2018.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

Stephanie Milani, Anssi Kanervisto, Karolis Ramanauskas, Sander Schulhoff, Brandon Houghton, Sharada Mohanty, Byron Galbraith, Ke Chen, Yan Song, Tianze Zhou, et al. Towards solving fuzzy tasks with human feedback: A retrospective of the minerl basalt 2022 competition. *arXiv preprint arXiv:2303.13512*, 2023.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

Aran Nayebi, Rishi Rajalingham, Mehrdad Jazayeri, and Guangyu Robert Yang. Neural foundations of mental simulation: Future prediction of latent representations on dynamic scenes. *arXiv preprint arXiv:2305.11772*, 2023.

Aviv Netanyahu, Tianmin Shu, Joshua Tenenbaum, and Pulkit Agrawal. Discovering generalizable spatial goal representations via graph-based active reward learning. In *International Conference on Machine Learning*, pp. 16480–16495. PMLR, 2022.

Hung Nguyen, Jay Patravali, Fuxin Li, and Alan Fern. Learning intuitive physics by explaining surprise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 374–375, 2020.

Maitreya Patel, Tejas Gokhale, Chitta Baral, and Yezhou Yang. Benchmarking counterfactual reasoning abilities about implicit physical properties. In *NeurIPS 2022 Workshop on Neuro Causal and Symbolic AI (nCSI)*, 2022a.

Maitreya Patel, Tejas Gokhale, Chitta Baral, and Yezhou Yang. CRIPP-VQA: Counterfactual reasoning about implicit physical properties via video question answering. *arXiv preprint arXiv:2211.03779*, 2022b.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pp. 2778–2787. PMLR, 2017.

Luis S Piloto, Ari Weinstein, Peter Battaglia, and Matthew Botvinick. Intuitive physics learning in a deep-learning model inspired by developmental psychology. *Nature human behaviour*, 6(9):1257–1267, 2022.

Haozhi Qi, Xiaolong Wang, Deepak Pathak, Yi Ma, and Jitendra Malik. Learning long-term visual dynamics with region proposal interaction networks, 2020. URL `https://arxiv.org/abs/2008.02265`.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Roberta Raileanu and Tim Rocktäschel. Ride: Rewarding impact-driven exploration for procedurally-generated environments. *arXiv preprint arXiv:2002.12292*, 2020.

Nazneen Fatema Rajani, Rui Zhang, Yi Chern Tan, Stephan Zheng, Jeremy Weiss, Aadit Vyas, Abhijit Gupta, Caiming Xiong, Richard Socher, and Dragomir Radev. ESPRIT: Explaining solutions to physical reasoning tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7906–7917, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.706. URL `https://aclanthology.org/2020.acl-main.706`.

Krishan Rana, Andrew Melnik, and Niko Sünderhauf. Contrastive language, action, and state pre-training for robot learning. *arXiv preprint arXiv:2304.10782*, 2023.

Ronan Riochet, Mario Ynocente Castro, Mathieu Bernard, Adam Lerer, Rob Fergus, Véronique Izard, and Emmanuel Dupoux. Intphys 2019: A benchmark for visual intuitive physics understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5016–5025, 2021.

Karan Samel, Zelin Zhao, Binghong Chen, Shuang Li, Dharmashankar Subramanian, Irfan Essa, and Le Song. Learning temporal rules from noisy timeseries data. *arXiv preprint arXiv:2202.05403*, 2022.

Theophile Sautory, Nuri Cingillioglu, and Alessandra Russo. Hyster: A hybrid spatio-temporal event reasoner. *arXiv preprint arXiv:2101.06644*, 2021.

Bruno Sauvalle and Arnaud de La Fortelle. Unsupervised multi-object segmentation using attention and soft-argmax. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 3267–3276, January 2023a.

Bruno Sauvalle and Arnaud de La Fortelle. Autoencoder-based background reconstruction and foreground segmentation with background noise estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 3244–3255, January 2023b.

Karl Schmeckpeper, Georgios Georgakis, and Kostas Daniilidis. Object-centric video prediction without annotation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13604–13610, 2021. doi: 10.1109/ICRA48506.2021.9561541.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Aviv Shamsian, Ofri Kleinfeld, Amir Globerson, and Gal Chechik. Learning object permanence from video. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pp. 35–50. Springer, 2020.

Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28, 2015.

Akash Singh, Tom De Schepper, Kevin Mets, Peter Hellinckx, José Oramas, and Steven Latré. Deep set conditioned latent representations for action recognition. *arXiv preprint arXiv:2212.11030*, 2022a.

Gautam Singh, Fei Deng, and Sungjin Ahn. Illiterate dall-e learns to compose, 2021. URL `https://arxiv.org/abs/2110.11405`.

Gautam Singh, Yi-Fu Wu, and Sungjin Ahn. Simple unsupervised object-centric learning for complex and naturalistic videos. *Advances in Neural Information Processing Systems*, 35:18181–18196, 2022b.

Kevin Smith, Lingjie Mei, Shunyu Yao, Jiajun Wu, Elizabeth Spelke, Josh Tenenbaum, and Tomer Ullman. Modeling expectation violation in intuitive physics with coarse probabilistic object representations. *Advances in neural information processing systems*, 32, 2019.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild, 2012.

Chen Sun, Calvin Luo, Xingyi Zhou, Anurag Arnab, and Cordelia Schmid. Towards learning implicit symbolic representation for visual reasoning. 2022.

Qu Tang, Xiangyu Zhu, Zhen Lei, and Zhaoxiang Zhang. Intrinsic physical concepts discovery with object-centric predictive models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23252–23261, 2023.

Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6450–6459, 2018. doi: 10.1109/CVPR.2018.00675.

Manuel Traub, Sebastian Otte, Tobias Menge, Matthias Karlbauer, Jannik Thuemmel, and Martin V Butz. Learning what and where: Disentangling location and identity tracking without supervision. In *The Eleventh International Conference on Learning Representations*, 2022.

Hsiao-Yu Tung, Mingyu Ding, Zhenfang Chen, Daniel Bear, Chuang Gan, Joshua B Tenenbaum, Daniel LK Yamins, Judith E Fan, and Kevin A Smith. Physion++: Evaluating physical scene understanding that requires online inference of different physical properties. *arXiv preprint arXiv:2306.15668*, 2023.

Basile Van Hoorick, Purva Tendulkar, Didac Suris, Dennis Park, Simon Stent, and Carl Vondrick. Revealing occlusions with 4d neural fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3011–3021, 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pp. 20–36. Springer, 2016.

Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7794–7803, 2018.

Erik Weitnauer, Paulo Carvalho, Robert Goldstone, and Helge Ritter. Similarity-based ordering of instances for efficient concept learning. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 36, 2014.

Erik Weitnauer, David Landy, Robert L Goldstone, and Helge J Ritter. A computational model for learning structured concepts from physical scenes. In *CogSci*, 2015.

Erik Weitnauer, Robert L Goldstone, and Helge Ritter. Perception and simulation during concept learning. *Psychological Review*, 2023.

Bo Wu, Shoubin Yu, Zhenfang Chen, Joshua B Tenenbaum, and Chuang Gan. Star: A benchmark for situated reasoning in real-world videos. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

Ziyi Wu, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg. Slotformer: Unsupervised visual dynamics simulation with object-centric models, 2022. URL `https://arxiv.org/abs/2210.05861`.

Ted Xiao, Harris Chan, Pierre Sermanet, Ayzaan Wahid, Anthony Brohan, Karol Hausman, Sergey Levine, and Jonathan Tompson. Robotic skill acquisition via instruction augmentation with vision-language models. *arXiv preprint arXiv:2211.11736*, 2022.

Li Xu, He Huang, and Jun Liu. Sutd-trafficqa: A question answering benchmark and an efficient network for video reasoning over traffic events. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9878–9888, 2021.

Yucheng Xu, Nanbo Li, Arushi Goel, Zijian Guo, Zonghai Yao, Hamidreza Kasaei, Mohammadreze Kasaei, and Zhibin Li. Controllable video generation by learning the underlying dynamical system with neural ode. *arXiv preprint arXiv:2303.05323*, 2023.

Cheng Xue, Vimukthini Pinto, Chathura Gamage, Ekaterina Nikonova, Peng Zhang, and Jochen Renz. Phy-q as a measure for physical reasoning intelligence. *Nature Machine Intelligence*, 5(1):83–93, 2023.

Mengjiao Yang, Yilun Du, Bo Dai, Dale Schuurmans, Joshua B Tenenbaum, and Pieter Abbeel. Probabilistic adaptation of text-to-video models. *arXiv preprint arXiv:2306.01872*, 2023.

Yufei Ye, Maneesh Singh, Abhinav Gupta, and Shubham Tulsiani. Compositional video prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10353–10362, 2019.

Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. Clevrer: Collision events for video representation and reasoning. *arXiv preprint arXiv:1910.01442*, 2019.

Polina Zablotskaia, Edoardo A Dominici, Leonid Sigal, and Andreas M Lehrmann. Unsupervised video decomposition using spatio-temporal iterative inference. *arXiv preprint arXiv:2006.14727*, 2020.

Polina Zablotskaia, Edoardo A Dominici, Leonid Sigal, and Andreas M Lehrmann. Provide: a probabilistic framework for unsupervised video decomposition. In *Uncertainty in Artificial Intelligence*, pp. 2019–2028. PMLR, 2021.

Vinicius Zambaldi, David Raposo, Adam Santoro, Victor Bapst, Yujia Li, Igor Babuschkin, Karl Tuyls, David Reichert, Timothy Lillicrap, Edward Lockhart, et al. Relational deep reinforcement learning. *arXiv preprint arXiv:1806.01830*, 2018.

Shiwen Zhang. Tfcnet: Temporal fully connected networks for static unbiased temporal reasoning. *arXiv preprint arXiv:2203.05928*, 2022.

Rong Zhao, Zheyu Yang, Hao Zheng, Yujie Wu, Faqiang Liu, Zhenzhi Wu, Lukai Li, Feng Chen, Seng Song, Jun Zhu, et al. A framework for the general design and computation of hybrid neural networks. *Nature communications*, 13(1):3427, 2022.

Yaoyao Zhong, Junbin Xiao, Wei Ji, Yicong Li, Weihong Deng, and Tat-Seng Chua. Video question answering: Datasets, algorithms and challenges. *arXiv preprint arXiv:2203.01225*, 2022.

Honglu Zhou, Asim Kadav, Farley Lai, Alexandru Niculescu-Mizil, Martin Renqiang Min, Mubbasir Kapadia, and Hans Peter Graf. Hopper: Multi-hop transformer for spatiotemporal reasoning. *arXiv preprint arXiv:2103.10574*, 2021.

Zheng Zhu, Qiang Wang, Bo Li, Wei Wu, Junjie Yan, and Weiming Hu. Distractor-aware siamese networks for visual object tracking. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 101–117, 2018.