Physion: Evaluating Physical Prediction from Vision in Humans and Machines

Daniel M. Bear[1,4,*], **Elias Wang**[2,4,*], **Damian Mrowca**[3,*], **Felix Binder**[5,*], **Hsiao-Yu Fish Tung**[1,7], **R.T. Pramod**[7], **Cameron Holdaway**[6], **Sirui Tao**[6], **Kevin Smith**[7], **Fan-Yun Sun**[3], **Li Fei-Fei**[3], **Nancy Kanwisher**[7], **Joshua B. Tenenbaum**[7], **Daniel L. K. Yamins**[1,3,4,**], **and Judith Fan**[6,**]

Department of Psychology[1], Electrical Engineering[2], and Computer Science[3], and
Wu Tsai Neurosciences Institute[4], Stanford, CA 94305
Department of Cognitive Science[5], and Psychology[6], UC San Diego, CA 92093
Department of Brain and Cognitive Sciences and CBMM[7], MIT, Cambridge, MA 02139

{dbear, eliwang, mrowca}@stanford.edu, fbinder@ucsd.edu

Abstract

While current vision algorithms excel at many challenging tasks, it is unclear how well they understand the physical dynamics of real-world environments. Here we introduce **Physion**, a dataset and benchmark for rigorously evaluating the ability to predict how physical scenarios will evolve over time. Our dataset features realistic simulations of a wide range of physical phenomena, including rigid and soft-body collisions, stable multi-object configurations, rolling, sliding, and projectile motion, thus providing a more comprehensive challenge than previous benchmarks. We used **Physion** to benchmark a suite of models varying in their architecture, learning objective, input-output structure, and training data. In parallel, we obtained precise measurements of human prediction behavior on the same set of scenarios, allowing us to directly evaluate how well any model could approximate human behavior. We found that vision algorithms that learn object-centric representations generally outperform those that do not, yet still fall far short of human performance. On the other hand, graph neural networks with direct access to physical state information both perform substantially better and make predictions that are more similar to those made by humans. These results suggest that extracting physical representations of scenes is the main bottleneck to achieving human-level and human-like physical understanding in vision algorithms. We have publicly released all data and code to facilitate the use of **Physion** to benchmark additional models in a fully reproducible manner, enabling systematic evaluation of progress towards vision algorithms that understand physical environments as robustly as people do.

1 Introduction

Vision algorithms that understand the physical dynamics of real-world environments are key to progress in AI. In many settings, it is critical to be able to anticipate when an object is about to roll into the road, fall off the table, or collapse under excess weight. Moreover, for robots and other autonomous systems to interact safely and effectively with their environment they must be able to accurately predict the physical consequences of their actions.

*/**Equal contribution

Preprint. Under review.

1.1 Establishing Common Standards for Evaluating Physical Understanding

Despite recent progress in computer vision and machine learning, it remains unclear whether any vision algorithms meet this bar of everyday physical understanding. This is because previously developed algorithms have been evaluated against disparate standards — some prioritizing accurate prediction of every detail of a scenario's dynamics and others that only require predictions about a specific type of event.

The first set of standards has generally been used to evaluate algorithms that operate on unstructured video inputs, such as in robotics [20]. These algorithms typically aim for fine-grained prediction of upcoming video frames or simulation of the trajectories of individual particles. However, only algorithms with near-perfect knowledge of the world's physical state – like Laplace's Demon – could hope to predict how a complete set of events will unfold. This explains why models of this kind have sufficed in less varied visual environments, but underfit on more diverse scenarios [17, 39]. Though recent efforts to scale these algorithms have led to improvements in the quality of predicted video outputs [65, 68], it remains to be seen whether their learned representations embody more general *physical* knowledge.

The second set of standards has been used to probe qualitative understanding of physical concepts, especially in cognitive and developmental psychology [4, 60, 15]. Much of this work has focused on measuring and modeling human judgments about discrete events, such as whether a tower of blocks will fall over or whether an object will reemerge from behind an occluder [10, 8, 5]. Findings from this literature suggest that humans simulate dynamics over more abstract representations of visual scenes to generate reliable predictions at the relevant level of granularity [49, 57]. However, existing models that instantiate such simulations typically require require structured input data (e.g., object segmentations) that may not be readily available in real-world situations [35, 32]. Moreover, the abstractions that are appropriate for one task may not work well in more general settings [64, 67, 43].

A key challenge in developing improved visual models of physical understanding is thus to establish common standards by which to evaluate them. Here we propose such a standard that both combines elements of previous approaches and goes beyond them: we require models to operate on highly varied and unstructured visual inputs to generate event-based predictions about a wide variety of physical phenomena. By contrast with prior efforts to evaluate vision algorithms, our proposed standard argues for the importance of considering a wider variety of physical scenarios and the ability to compare model predictions directly with human judgments. By contrast with prior efforts to model human physical understanding, our approach embraces the challenge of generating predictions about key events from realistic visual inputs.

1.2 Desiderata for a Generalized Physical Understanding Benchmark

We envision our generalized physical understanding benchmark as combining two key components: first, a dataset containing visually realistic and varied examples of a wide variety of physical phenomena; and second, a generic evaluation protocol that probes physical understanding in a way that is agnostic to model architecture and training regime.

Dataset. While there are several existing datasets that probe physical understanding to some extent, each of them fall short on at least one key dimension. Some datasets contain realistic visual scenes but do not adequately probe understanding of object dynamics [17]. Other datasets feature realistic scenarios with challenging object dynamics, but consider only a narrow set of physical phenomena, such as whether a tower of blocks will fall [29] or whether a viewed object's trajectory violates basic physical laws [49, 46, 57]. Other datasets featuring a greater diversity of physical phenomena are designed in simplified 2D environments that may not generalize to real-world 3D environments [6].

Evaluation protocol. In order to test a wide variety of models in a consistent manner, many commonly used evaluations will not suffice. For example, evaluations that query the exact trajectories of specific objects [9, 16] are not well posed for models that do not extract explicit object representations. Conversely, evaluations that depend on image matching or visual realism-based metrics [21, 69, 17, 68] are not straightforward to apply to models that do not re-render images. A more promising approach to measuring physical understanding in a model-agnostic manner may instead take inspiration from prior work investigating human physical prediction ability [10, 51, 8], which does not assume that the trajectories of all objects in a scene are represented with perfect fidelity.

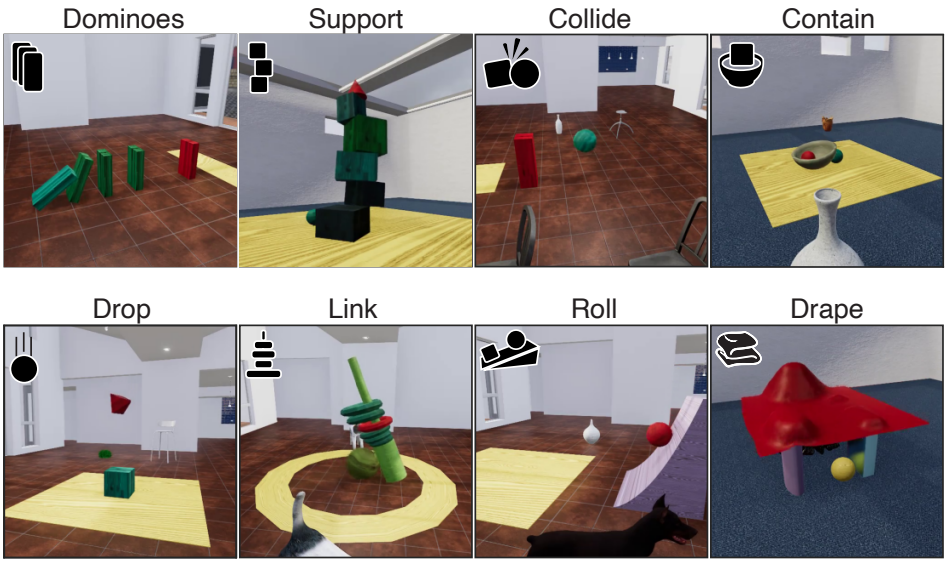

Figure 1: Example frames from the eight **Physion** scenarios. Red object is agent; yellow is patient.

1.3 Physion: A Dataset and Benchmark for Physical Understanding

In recognition of the above desiderata, we developed **Physion**, a new physical understanding dataset and benchmark. Our dataset contains a wide variety of visually realistic examples of familiar physical phenomena, including: collisions between multiple objects; object-object interactions such as support, containment, and attachment; projectile, rolling, and sliding motion that depends on object geometry; and the behavior of soft materials like cloth. For each of these eight scenario types (1), we operationalize physical understanding using the **object contact prediction (OCP) task**, which prompts agents to predict whether two cued objects will come into contact as a scene unfolds.

1.4 Using Physion to Benchmark Human and Model Physical Understanding

In addition to the dataset, we introduce a unified evaluation protocol for directly comparing model and human behavior. Approximating human physical understanding from vision is a natural target for AI systems for two key reasons: first, humans have already demonstrated their ability to competently navigate a wide variety of real-world physical environments; and second, it is important for AI systems to anticipate how humans understand their physical surroundings in order to co-exist safely with people in these environments. Towards this end, our paper conducts systematic comparison between humans and several state-of-the-art models on the same physical scenarios.

Our experiments feature a wide range of models that vary in their architecture, learning objective, input-output structure, and training regime. Specifically, we include vision models that make pixel-level predictions via fully convolutional architectures, [23, 1, 36, 21, 35, 70, 40, 41, 66, 30, 34, 54, 27]; those that either explicitly learn object-centric representations of scenes [64, 33, 19, 27, 50] or are encouraged to learn about objects via supervised training [56, 62]; and physics dynamics models that operate on object- or particle-graph representations provided as input [16, 9, 37, 8, 61, 52, 11, 42, 2, 57, 69, 47].

Models that perform physical simulation on a graph-like latent state are especially attractive candidates for approximating human prediction behavior, based on prior work that has found that *non*-machine learning algorithms that add noise to a hard-coded simulator accurately capture human judgments in several different physical scenarios [10, 51, 7, 13]. Consistent with these results, recurrent graph neural networks supervised on physical simulator states can learn to accurately predict full object trajectories [42, 37, 38, 53]. However, these models have not been tested for their ability to generalize across diverse, multi-object scenarios, and they require such detailed physical input and trajectory supervision that they have so far not been useful in cases where only realistic sensory observations are available.

Among models that take visual input, object-centric predictors in some cases make more accurate predictions than those that simulate scene dynamics in pixel space [64, 47, 19]; however, these comparisons have only been done in reduced environments with few distinct physical phenomena, so it is not known whether this result holds in more realistic settings. Indeed, models that make pixel-level predictions are standard in robotics applications [34, 68] due to the longstanding difficulty of inferring accurate object-centric representations from raw video data without supervision, despite recent progress [14, 64, 12].

1.5 Summary of Key Findings

By assessing many models on the same challenging physical understanding task, our experiments address previously unresolved questions concerning the roles of model architecture, dataset, and training protocols in achieving robust and human-like physical understanding. We found that no current vision algorithms achieve human-level performance in predicting the outcomes of **Physion** scenes. Vision algorithms encouraged to learn object-centric representations generally outperform those that do not, yet still fall far short of human performance. On the other hand, particle-based models with direct access to physical state information both perform substantially better and make predictions that are more similar to those made by humans. Taken together, these results suggest that extracting physical representations of visual scenes is the key bottleneck to achieving human-level and human-like physical understanding in vision algorithms.

1.6 Our Vision for Physion

Our initial public release of **Physion** includes large, labeled training and test datasets for each scenario, as well as code for for generating additional training data. As such, one potential way to use **Physion** is to train additional models directly on the OCP task for one or more of the scenarios, yielding, for example, a model that excels at predicting whether block towers will fall. However, the primary use case we have in mind for **Physion** is to test how well pretrained models transfer to challenging physical understanding tasks, analogous to how humans make predictions about **Physion** videos without extensive training on the OCP task. Towards this end, we have shared code to facilitate the use of the **Physion** test dataset to benchmark additional models in a fully reproducible manner, enabling systematic evaluation of progress towards vision algorithms that understand physical environments as robustly as people do.

2 Methods

2.1 Benchmark Design

We used the ThreeDWorld simulator (TDW), a Unity3D-based environment [24], to create eight physical scenarios out of simple objects that incorporate diverse physical phenomena (Fig. 1):

1. **Dominoes** – sequences of collisions that depend on the arrangement and poses of objects
2. **Support** – stacks of objects that may fall over, depending on their shapes and arrangement
3. **Collide** – pairs of objects that may collide, depending on their placement and trajectories
4. **Contain** – container-like objects that may constrain other objects by virtue of their shapes
5. **Drop** – objects falling and bouncing under the force of gravity
6. **Link** – objects restricted in their motion because they are attached to other objects
7. **Roll** – objects that move across a surface either by rolling or sliding
8. **Drape** – cloth draping over other objects by virtue of their shape and the cloth's material.

In each scenario, contact between agent and patient serves as a non-verbal indicator of some physical higher-order variable – whether a tower fell over, a bowl contained a ball, a torus was attached to a post – whose prediction should require understanding of the relevant physical phenomena. Together, these scenarios cover much of the space of physical dynamics possible through simple rigid- and soft-body interactions; additional scenarios will be developed to include other material types (e.g., "squishy" objects, fluids) and complex interactions (e.g. multi-part, jointed objects.)

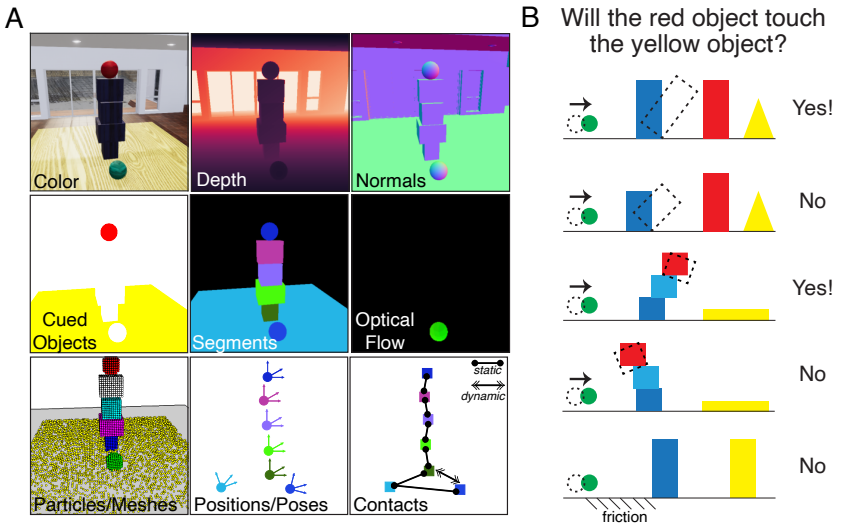

Figure 2: Stimulus attributes and task design. (**A**) Output of TDW for an example frame of a stimulus movie. (**B**) A schematic of the OCP task: humans and models must predict whether the *agent* object (red) will contact the *patient* (yellow), given the initial setup and the motion of the *probe* (green).

2.2 Stimulus Generation and Task Design

We constructed scenes out of basic "toy blocks" to avoid confounds from knowledge of object configurations that are common in the real world (e.g., cups typically appearing on tables); rather, accurate predictions should require judgments about objects' physical properties, relationships, and dynamics. To increase physical variability *within* each scenario, we identified multiple configurations of simulator parameters that lead to different types of physical dynamics. Configurations specify distributions of initial scene variables, such as the positions of objects; they also introduce substantial visual variation that does not affect the physical outcome of the scene, including variation in camera position and pose, object colors and textures, the choice of "distractor" object models that do not participate in scene dynamics, and the appearance of the background. Training and testing stimuli were generated by randomly sampling initial conditions and scene properties according to each configuration, then running the simulation until all objects came to rest. Additional stimuli can be generated by sampling further from our configurations or by creating new ones. Examples of stimuli from each scenario can be found in the Supplement.

Each stimulus is a 5-10 second movie rendered at 30 frames per second. For model training and evaluation we also supply the full output of the TDW simulation (Fig. 2A), which includes: 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 (Fig. 2B): 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. Models and people are asked to predict whether the agent and patient object will come into contact by the time all objects come to rest. We generated trials for human testing by sampling from scenario-specific configurations until we had 150 testing stimuli per scenario with an equal proportion of contact and no-contact outcomes.

2.3 Testing Humans on the Physics Prediction Benchmark

Participants. 800 participants (100 per scenario; 447 female, 343 male, 7 declined to state; all native English speakers) were recruited from Prolific and paid $4.00 for their participation. Each was shown all 150 stimuli from a single scenario. Data from 112 participants were excluded for not

meeting our preregistered inclusion criterion for accurate and consistent responses on attention-check trials (see Supplement). Our preregistered analysis plan is stored under version control in our GitHub repository. These studies were conducted in accordance with the UC San Diego and Stanford IRBs.

Task procedure. The structure of our task is shown in Fig. 3A. Each trial began with a fixation cross, which was shown for a randomly sampled time between 500ms and 1500ms. To indicate which of the objects shown was the agent and patient object, participants were then shown the first frame of the video for 2000ms. During this time, the agent and patient objects were overlaid in red and yellow respectively. The overlay flashed on and off with a frequency of 2Hz. After this, the first 1500ms of the stimulus were played. After 1500ms, the stimulus was removed and the response buttons were enabled. Participants proceeded to the next trial after they made a prediction by selecting either "YES" (the agent and patient would touch) or "NO" (they would not). The order of the buttons was randomized between participants. Before the main task, participants were familiarized with 10 trials that were presented similarly to the test trials, except (a) the full stimulus movie and accuracy feedback was presented after participants indicated their prediction, and (b) all trials were created from basic templates without occluding and distracting objects. Familiarization trials were always presented in the same order. After the test trials were completed, basic demographics were collected from participants. Finally, participants were informed of their overall accuracy.

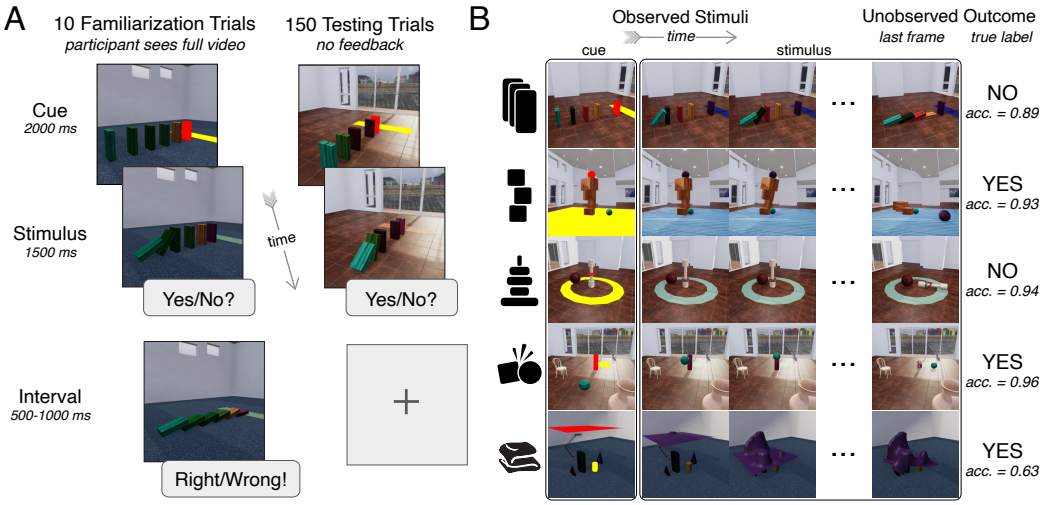

Figure 3: Human task. (A) Trial structure for the familiarization trials (*left*) and test trials (*right*) indicating the Cue, Stimulus, and Inter-trial periods. (B) Example stimuli (rows) including the last frame (not shown during the experiment). Last column indicates the outcome and human accuracy.

2.4 Benchmarking Computer Vision and Physical Dynamics Models

We developed a standard procedure for training machine learning models and evaluating any image- or physical state-computable algorithm on the benchmark. Let $\{X_t\}_{i=1}^{N_{test}}$ be the set of N_{test} testing stimuli for a single benchmark scenario, where $\{X_t\}_i$ denotes the ordered set of RGB images that constitutes the full movie of stimulus i and $\{X_{1:t_{vis}}\}$ the truncated movie shown to participants. Further let $\mathcal{O}_i := \{o_1, o_2, ..., o_{K_i}\}$ denote unique IDs for each of the K objects being simulated in this stimulus. Doing the OCP task can be formalized as making a binary contact prediction by applying to the testing stimuli a function $\mathcal{F}_\Theta : (\{X_{1:t_{vis}}\}, o_a, o_p) \mapsto P(contact)$, where o_a is the agent object, o_p is the patient, and $P(contact)$ is the predicted probability that they will come into contact. For people, feedback on only ten familiarization trials is sufficient to learn such a function. To adapt any image-computable model to the OCP task, we apply the following procedure. First, we assume that a model can be decomposed into a *visual encoder* that maps an input movie to a state-vector representation of each frame; a *dynamics predictor* that predicts unseen future states from the "observed" state vector; and a *task adaptor* that produces a trial-level response $P(contact)$ from the concatenation of the observed and predicted state vectors (Fig. 4). In general, models will include only a visual encoder and possibly a dynamics predictor in their original design; the task

adaptor is added and fit as part of our model evaluation pipeline, where it removes the need for the explicit trial-level cueing with superimposed object masks (see below.)

Testing, Readout Fitting, and Training sets. Each **Physion** scenario consists of three stimulus sets: *Testing*, *Readout Fitting*, and *Training*. The *Testing* stimuli are identical to the 150 trials per scenario shown to humans, except that the agent and patient objects are permanently colored red and yellow (Fig. 1) instead of being indicated by red and yellow masks on the first frame (Fig. 3). This difference allows models to be tested on RGB movie stimuli alone, without providing segmentation masks that most computer vision model architectures are not designed to handle as inputs. Each trial in the *Testing* sets includes the ground truth label of whether it ends in agent-patient contact and the responses of >100 human participants. We also provide the *Human Testing* stimuli with red and yellow cueing masks rather than permanently colored objects.

Each scenario's *Readout Fitting* set consists of 1000 stimuli generated from the same configurations as the *Testing* stimuli, such that the two sets have the same visual and physical statistics. The *Readout Fitting* stimuli are for fitting a OCP task-specific adaptor to each model. In designing **Physion**, we did not want to restrict testing only to models optimized directly to do the OCP prediction task. Thus, during evaluation we freeze the parameters of a pretrained model and fit a generalized linear model, the task adaptor, on various subsets of model features (see below). The *Readout Fitting* stimuli are the training set for this fitting procedure, with the ground truth object contact labels acting as supervision. This allows the task adaptor to generalize to the *Testing* stimuli.

Finally, each scenario's *Training* set includes 2000 movies generated from the same configurations as the *Testing* and *Readout Fitting* stimuli, but with no visual features indicating agent and patient objects. The purpose of the *Training* sets is to let models learn or fine-tune representations of physical dynamics in a way that is agnostic to any particular readout task: a model partly or entirely trained on a "non-physics" task like object categorization might nevertheless acquire a human-like representation of the physical world, which **Physion** should reveal via transfer learning. During training models see movie clips sampled from the entirety of each *Training* stimulus, not just the initial portion seen during readout fitting and testing, and they do not receive ground truth OCP labels.

The procedure for training a given model depends on its original architecture and optimization procedure. For models that take multi-frame inputs and include both a visual encoder and a dynamics predictor in their architecture, we train the full model end-to-end on the *Training* sets. For models that include only a visual encoder pretrained on another dataset and task (such as ImageNet), we add an RNN dynamics model that predicts future encoder outputs from the "observed" encoder outputs on an input frame sequence; the training loss is the mean squared error between each predicted output and the matching observed output, which optimizes the dynamics model. For these models, we train two versions: one in which the pretrained encoder parameters are fine-tuned and one in which they are frozen. See **Model Comparison** below and the Supplement for further details.

Model comparison. To get an overview of how current physical prediction algorithms compare to humans, we tested models from four classes (see Supplement for model details):

1. fully unsupervised, joint encoder-dynamics predictors trained only on the benchmark scenario data: **SVG** [18], **OP3** [64], **CSWM** [33];
2. encoder-dynamics models supervised on ground truth object data: **RPIN** [47];
3. visual encoders *pretrained with supervision on ImageNet* and extended with RNN dynamics predictors, which are trained in an *unsupervised* way on the benchmark scenario data: **pVGG-mlp/lstm** [56], **pDeIT-mlp/lstm** [62];
4. particle-relation graph neural network dynamics predictors that take the ground truth simulator state as input and have no visual encoder (i.e. assume perfect observability of physical dynamics): **GNS** [53], **GNS-RANSAC**, **DPI** [37].

Training protocols. We tested models given three types of training (Fig. 4, left): *all*, training on all scenarios' training sets concurrently; *all-but*, training on all scenarios except the one the model would be tested on; and **only**, training on only the scenario type the model would be tested on. We consider the *all* protocol to be the best test of physical understanding, since it produces a model that is not specialized to a specific scenario. Differences between *all* and *all-but* or *only* indicate how well a model can generalize across scenarios or overfit to a single scenario, respectively.

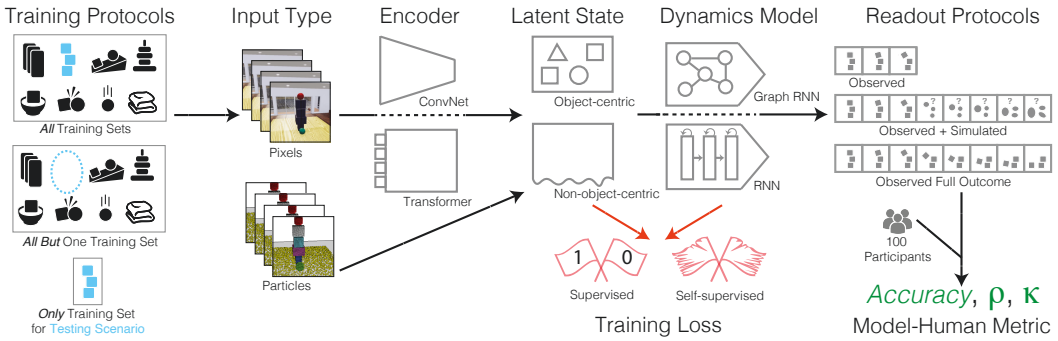

Figure 4: The model benchmarking pipeline including training, architecture, and readout variants.

Testing protocols. We fit logistic regression models as OCP task adaptors with three protocols (Fig. 4, right): *observed*, in which adaptors are fit only to the features produced by showing the human stimulus (first t_{vis} frames, equivalent to 1.5 seconds) to the model's visual encoder; *observed+simulated*, which uses the *observed* features concatenated with the "simulated" features output by the model's dynamics predictor; and *full*, which uses the features produced from showing the entire movie (not just the testing stimulus portion) to the visual encoder. Outputs from the *full* protocol cannot be directly compared to human data, since they represent a model's performance on a detection (rather than prediction) task; however, we use them to assess how well physical information is encoded in a model's visual features (see Experiments.) We compare a model's outputs to human responses on each scenario's testing stimuli with three standard metrics (Fig. 4, right): overall accuracy, Pearson correlation between model and average human responses across stimuli, and Cohen's κ, a measure of how much a model's binary predictions resemble a single human's, averaged across participants. For all three metrics, we assess how close models are to the "human zone" – the empirical distribution of each statistic across humans or human-human pairs.

3 Results and Discussion

Human behavior is reliable, with substantially above-chance performance. Human performance was substantially above chance across all eight scenarios (proportion correct = 0.71, t=27.5, $p<10^{-7}$, Fig. 5A), though there was variation in performance across scenarios (e.g., higher accuracy on **Roll** than **Link** or **Drape**). Moreover, the "human zones" for all metrics (raw performance, correlation-to-average, and Cohen's κ) were tight and far from chance (gray horizontal bars in Fig. 5A-E), showing that the human response patterns were highly reliable at our data collection scale and thus provide a strong empirical test for discriminating between models. Interestingly, each scenario included some stimuli on which the participant population scored significantly *below* chance (Fig. S1). Many of these "adversarial" stimuli had objects teetering on the brink of falling over or other unlikely events occurring after the observed portion of the movie. People may have accurately judged that most scenes *similar to* the observed stimulus would have one outcome, unaware that the other outcome actually occurred due to a physical fluke. This pattern of reliable errors is especially useful for comparing models with humans: if stimuli that fool people do not fool a model, it would suggest that the model draws on different information or uses a non-human strategy for making predictions.

Particle-based models approach human performance levels, with strong generalization. Models that received ground-truth TDW object particles as input and supervision (**GNS**, **GNS-RANSAC**, **DPI**) matched human accuracy on many scenarios, with the object-centric **DPI** reaching across-scenario human performance levels (Fig. 5A). These data are consistent with findings that probabilistic physical simulations can account for behavioral judgments on single scenarios that resemble ours [10, 51, 7, 13]. However, our results go beyond prior work in several ways. First, these three models are graph neural networks that *learn* to simulate physical scenes rather than assuming access to a "noisy" version of ground truth dynamics directly provided by the physics engine. Second, the models here performed well above chance when trained with the *all* and *all-but* protocols, not just when they were fit to single scenario types (*only*) as in the work where they were developed [37, 53] (Fig.

5A,E). These results imply that a single graph neural network can learn to make human-level physical predictions across a diverse set of physical scenarios.

Vision-based models substantially underperform humans, but object-related training may help. Particle input models have an enormous advantage over both humans and vision models: they operate on ground truth physical information that, in the real world, can never be observed directly, such as the 3D positions, poses, trajectories, and fine-scale shapes of all objects and their occluded surfaces. Whereas humans overcome these limits, none of the vision algorithms here came close to performing at human levels (Fig. 5A). Not all vision models were equally far off, though: among those whose encoders and dynamics simulators were fully unsupervised, **SVG**, a model with only convolutional latent states, performed nearly at chance levels; **OP3**, an object-centric model trained by rendering pixel-level future predictions (b=0.06, t=7.6, $p<10^{-11}$), performed marginally better; while **CSWM**, a model with contrastively-learned object-centric latent states, significantly outperformed both **SVG** and **OP3**. Interestingly, the *supervised* object-centric model **RPIN** was only more accurate than **CSWM** when trained with the *all-but* and *only* protocols, but not the *all* protocol (b=0.035, t=3.7, $p<10^{-3}$, Fig. 5A,E); further experiments are needed to test whether exactly matching the architectures of the two models would reveal a larger effect of ground truth supervision. Together, these results suggest that learning better object-centric representations from realistic, unlabeled video should be a core aim of visual prediction approaches.

The models with *ImageNet-pretrained* ConvNet encoders (**pVGG-mlp/lstm**) significantly outperformed the best fully TDW-trained models (**CSWM**, **RPIN**, b=0.015, t=2.9, p<0.01), and were themselves outperformed by models with ImageNet-pretrained Transformer encoders (**pDeIT-mlp/lstm**, b=0.067, t=16.5, $p<10^{-15}$). This suggests that (supervised) ImageNet pretraining and a better (and perhaps, more "object-aware"-attention driven) encoder architecture produce visual features that are better for physical prediction even *without* learning to explicitly simulate the future. Together these results highlight the importance of learning a "good" visual representation; vision algorithms may benefit from training their encoders on separate tasks and data before learning dynamics predictors.

Error-pattern consistency is strongly correlated with performance, but a substantial gap remains. A striking feature of our results is that error-pattern consistency as measured either by correlation-to-average human or Cohen's κ (Fig. 5B-C) is itself strongly correlated with absolute model performance. In other words, models that performed better on the prediction task also made errors that were more like those made by humans, strongly analogous to the situation with core visual object recognition [48]. This result suggests, albeit weakly, that human behavior has been highly optimized either directly for a prediction task like that measured in this paper, or for something highly correlated with it. However, none of the models fully reached the "human zone" in which their outputs would be statistically indistinguishable from a person's. This means that even the particle-based models can be improved to better match the judgments people make, including errors; prior work suggests that adding noise to these models could better recapitulate human mental "simulation" [10, 8, 58]. Consistent with this possibility, we found that the particle-based models' predictions were uncorrelated with human predictions on the "adversarial" stimuli, many of which would have opposite outcomes if their initial conditions were slightly different (Fig. S2). Adding noise to the models' forward dynamics might therefore mimic how humans make predictions about *probable* outcomes, rather than simulating dynamics so precisely that they capture even rare flukes.

What have vision-based models actually learned? Vision model predictions from the *observed+simulated* readout protocol were, overall, no better than predictions from the *observed* protocol (p=0.53, Fig. 5D). This implies that none of the visual dynamics models learned to "simulate" anything about the scenes that helped on the OCP task (though dynamics predictions during end-to-end training could have usefully shaped the encoder representations.) Rather, any above-chance performance for the vision models was likely due to having visual features that could discriminate some trial outcomes from cues in the initial movie segment. Understanding what makes these visual features useful is the subject of ongoing work: they could be an example of non-causal "shortcut learning" [26] or they could encode important physical properties like object position, shape, and contact relationships. The latter possibility is further supported by two observations. First, the *full* readout protocol yielded significantly higher accuracy for the vision models (b=0.094, t=12.0, $p<10^{-15}$, Fig. 5D), indicating that the learned visual features *are* useful for object contact *detection*. Thus, the best visual features carry some information about the observed objects' spatial relationships, and their relative failures in the *observed* protocol can be fairly said to be these models' lack of physical "understanding." Second, the ImageNet-pretrained models benefited the most from observing the

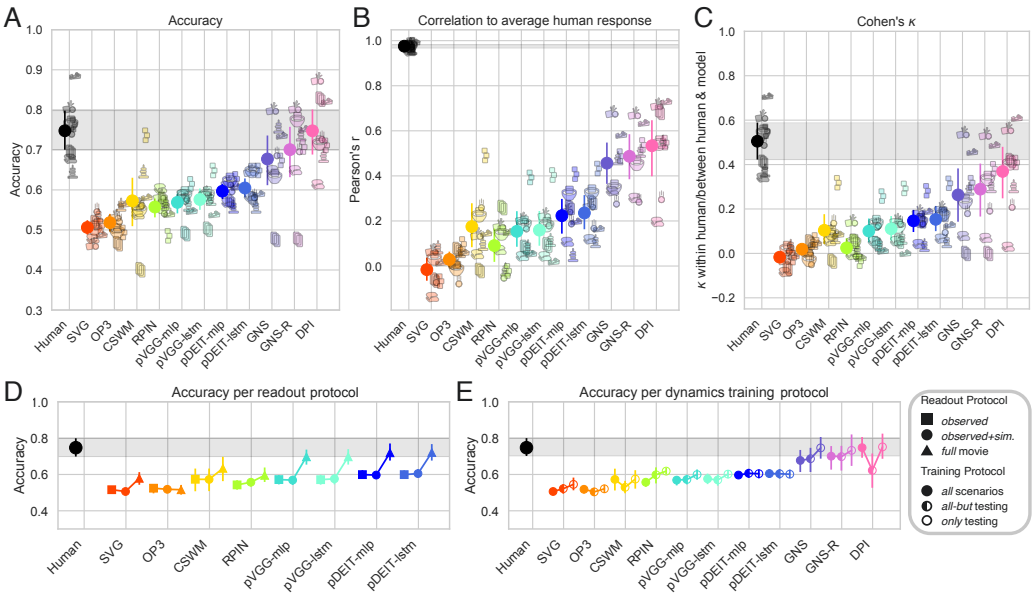

Figure 5: Comparisons between humans and models. First row: the *all*-scenarios trained, *observed+simulated*-readout task accuracy (**A**), Pearson correlation between model output and average human response (**B**), and Cohen's κ (**C**) for each model on each scenario, indicated by its icon. Black icons and the gray zones (2.5th-97.5th percentile) show human performance, mean correlation between split halves of participants, and mean human-human Cohen's κ, respectively. Second row: accuracy of models across the three readout (**D**) and training (**E**) protocols; note that particle-input models have only the *observed+simulated* readout protocol, as predictions are made based solely on whether two objects came within a threshold distance at the end of the predicted dynamics.

full movie, raising the possibility that their pretraining actually captured *more* physically-relevant information than object-centric learning on TDW. Untangling this will require finer-scale comparison between encoder architectures, training datasets, and various supervised and self-supervised losses.

Having sufficient variability across physical scenarios promotes strong generalization. Compared to models trained concurrently on *all* scenarios, vision-based models performed only slightly better when they were trained with the *only* protocol (b=0.21, t=4.4, $p<10^{-4}$), and not significantly worse when trained with the *all-but* protocol (b=0.009, t=1.9, p=0.057, Fig. 5E). Differences between protocols were larger for particle-based models, but nonetheless small relative to overall performance levels. These results strongly suggest that performance assessments are robust to the specific choices of scenarios we made. This makes sense because the diverse physical phenomena in our everyday environment result from a smaller set of underlying laws. Our results thus quantitatively support the qualitative picture in which an intuitive, approximate understanding of those laws gives rise to humans' outstanding ability to predict and generalize to previously unseen physical phenomena from an early age [60, 15, 5, 49]. However, we do find that models trained on any single scenario do not generalize well to most other scenarios (Fig. S5), suggesting that having substantial diversity of observations is critical for learning general physical forward predictors. It will be important, then, to develop additional testing scenarios that incorporate physical phenomena *not* covered here, such as "squishy" and fluid materials, the dynamics of jointed multi-part objects, and much larger ranges of mass, friction, density, and other physical parameters. We thus hope that our benchmark can be used to drive the development of algorithms with a more general, human-like ability to predict how key events will unfold and to anticipate the physical consequences of their own actions in the real world.

Acknowledgments

D.M.B. is supported by a Wu Tsai Interdisciplinary Scholarship and is a Biogen Fellow of the Life Sciences Research Foundation. C.H. is supported by a Department of Defense National Defense Science and Engineering Graduate Fellowship. H.F.T., K.A.S, R.T.P., N.K., and J.B.T are supported by National Science Foundation Science Technology Center Award CCF-1231216 and Office of Naval Research Multidisciplinary University Research Initiative (ONR MURI) N00014-13-1-0333; K.A.S. and J.B.T. are supported by research grants from ONR, Honda, and Mitsubishi Electric. D.L.K.Y is supported by the McDonnell Foundation (Understanding Human Cognition Award Grant No. 220020469), the Simons Foundation (Collaboration on the Global Brain Grant No. 543061), the Sloan Foundation (Fellowship FG-2018-10963), the National Science Foundation (RI 1703161 and CAREER Award 1844724), and hardware donations from the NVIDIA Corporation. K.A.S., J.B.T., and D.L.K.Y. are supported by the DARPA Machine Common Sense program. J.E.F. is supported by NSF CAREER Award 2047191 and the ONR Science of Autonomy Program. This work was funded in part by the HAI-Google Cloud Credits Grant Program and the IBM-Watson AI Lab. We thank Seth Alter and Jeremy Schwartz for their help on working with the ThreeDWorld simulator.

Broader Impact

There are few aspects of everyday life that are not informed by our intuitive physical understanding of the world: moving and doing tasks around the home, operating motor vehicles, and keeping one's body out of harm's way are just a few of the broad behavioral categories that involve making predictions of how objects in the world will behave and respond to our actions. Although there may be ways for algorithms to safely and effectively perform specific tasks without general, human-like understanding of the physical world, this remains a wide open question in many of the areas where AI is rapidly being deployed: self-driving vehicles, robotics, and other systems that involve a "perceive-predict-act" feedback loop. As such, we think the **Physion** benchmark is an important step toward actually measuring whether a given algorithm *does* perceive visual scenes and make physical predictions the way people do. If it turns out that this is critical for achieving safe, high performance in some real-world domain, our benchmark (or its successors) could be used to screen for algorithms more likely to behave like people and to diagnose failures, e.g. by breaking them down into problems making predictions about particular physical phenomena. Moreover our results, though representing only an initial survey of existing algorithms, *do* suggest that models with more explicit physical representations of the world, including the grouping of scene elements into objects, are better equipped to make accurate predictions; they therefore begin to address longstanding questions in AI about whether some sort of "symbolic" representation, inspired by cognitive science, is necessary for an algorithm to accurately predict and generalize to new situations. Though such representations have fallen out of favor in large-scale visual categorization tasks, the fact that they outperform their less or non-symbolic counterparts on the **Physion** tasks raises the intriguing possibility that two broad types of understanding, "semantic" and "physical", may benefit from different algorithm architectures and learning principles. If this is the case, we should reevaluate popular claims that symbolic representations and "interpretable" algorithms are red herrings for making progress in AI.

References

[1] P. Agrawal, A. V. Nair, P. Abbeel, J. Malik, and S. Levine. Learning to poke by poking: Experiential learning of intuitive physics. In *Advances in Neural Information Processing Systems*, pages 5074–5082, 2016.

[2] A. Ajay, M. Bauza, J. Wu, N. Fazeli, J. B. Tenenbaum, A. Rodriguez, and L. P. Kaelbling. Combining physical simulators and object-based networks for control. In *International Conference on Robotics and Automation*, 2019.

[3] K. R. Allen, K. A. Smith, and J. 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. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1912341117.

[4] R. Baillargeon, E. S. Spelke, and S. Wasserman. Object permanence in five-month-old infants. *Cognition*, 20(3):191–208, 1985.

[5] R. Baillargeon, J. Li, Y. Gertner, and D. Wu. *How do infants reason about physical events?* Wiley-Blackwell, 2011.

[6] A. Bakhtin, L. van der Maaten, J. Johnson, L. Gustafson, and R. Girshick. Phyre: A new benchmark for physical reasoning. *Advances in Neural Information Processing Systems*, 32:5082–5093, 2019.

[7] C. J. Bates, P. W. Battaglia, I. Yildirim, and J. B. Tenenbaum. Humans predict liquid dynamics using probabilistic simulation. In *CogSci*, 2015.

[8] C. J. Bates, I. Yildirim, J. B. Tenenbaum, and P. W. Battaglia. Modeling human intuitions about liquid flow with particle-based simulation. *PLoS Computational Biology*, 2019.

[9] P. Battaglia, R. Pascanu, M. Lai, D. J. Rezende, et al. Interaction networks for learning about objects, relations and physics. In *Advances in Neural Information Processing Systems*, pages 4502–4510, 2016.

[10] P. W. Battaglia, J. B. Hamrick, and J. B. Tenenbaum. Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences*, 110(45):18327–18332, 2013.

[11] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

[12] D. M. Bear, C. Fan, D. Mrowca, Y. Li, S. Alter, A. Nayebi, J. Schwartz, L. Fei-Fei, J. Wu, J. B. Tenenbaum, et al. Learning physical graph representations from visual scenes. In *Advances in Neural Information Processing Systems*, 2020.

[13] W. Bi, A. D. Shah, K. W. Wong, B. Scholl, and I. Yildirim. Perception of soft materials relies on physics-based object representations: Behavioral and computational evidence. *bioRxiv*, 2021.

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

[15] S. Carey and F. Xu. Infants' knowledge of objects: Beyond object files and object tracking. *Cognition*, 80 (1-2):179–213, 2001.

[16] M. B. Chang, T. Ullman, A. Torralba, and J. B. Tenenbaum. A compositional object-based approach to learning physical dynamics. In *International Conference on Learning Representations*, 2017.

[17] S. Dasari, F. Ebert, S. Tian, S. Nair, B. Bucher, K. Schmeckpeper, S. Singh, S. Levine, and C. Finn. Robonet: Large-scale multi-robot learning. *arXiv preprint arXiv:1910.11215*, 2019.

[18] E. Denton and R. Fergus. Stochastic video generation with a learned prior. In *International Conference on Machine Learning*, pages 1174–1183. PMLR, 2018.

[19] D. Ding, F. Hill, A. Santoro, and M. Botvinick. Object-based attention for spatio-temporal reasoning: Outperforming neuro-symbolic models with flexible distributed architectures. *arXiv preprint arXiv:2012.08508*, 2020.

[20] F. Ebert, S. Dasari, A. X. Lee, S. Levine, and C. Finn. Robustness via retrying: Closed-loop robotic manipulation with self-supervised learning. In *Conference on Robot Learning*, pages 983–993. PMLR, 2018.

[21] C. Finn, I. Goodfellow, and S. Levine. Unsupervised learning for physical interaction through video prediction. In *Advances in Neural Information Processing Systems*, pages 64–72, 2016.

[22] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981. ISSN 0001-0782. doi: 10.1145/358669.358692. URL `https://doi.org/10.1145/358669.358692`.

[23] K. Fragkiadaki, P. Agrawal, S. Levine, and J. Malik. Learning visual predictive models of physics for playing billiards. In *International Conference on Learning Representations*, 2016.

[24] C. Gan, J. Schwartz, S. Alter, M. Schrimpf, J. Traer, J. De Freitas, J. Kubilius, A. Bhandwaldar, N. Haber, M. Sano, K. Kim, E. Wang, D. Mrowca, M. Lingelbach, A. Curtis, K. Feigelis, D. M. Bear, D. Gutfreund, D. Cox, J. J. DiCarlo, J. McDermott, J. B. Tenenbaum, and D. L. K. Yamins. ThreeDWorld: A Platform for Interactive Multi-Modal Physical Simulation. *arXiv preprint arXiv:2007.04954*, 2020.

[25] T. Gebru, J. Morgenstern, B. Vecchione, J. W. Vaughan, H. Wallach, H. Daumé III, and K. Crawford. Datasheets for datasets. *arXiv preprint arXiv:1803.09010*, 2018.

[26] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[27] R. Girdhar, L. Gustafson, A. Adcock, and L. van der Maaten. Forward prediction for physical reasoning. *arXiv preprint arXiv:2006.10734*, 2020.

[28] R. Girshick. Fast r-cnn. In *International Conference on Computer Vision*, pages 1440–1448, 2015.

[29] O. Groth, F. B. Fuchs, I. Posner, and A. Vedaldi. Shapestacks: Learning vision-based physical intuition for generalised object stacking. In *European Conference on Computer Vision*, pages 702–717, 2018.

[30] N. Haber, D. Mrowca, L. Fei-Fei, and D. L. Yamins. Learning to play with intrinsically-motivated self-aware agents. In *Advances in Neural Information Processing Systems*, 2018.

[31] H. Hecht and M. Bertamini. Understanding projectile acceleration. *Journal of Experimental Psychology: Human Perception and Performance*, 26(2):730–746, 2000. ISSN 1939-1277, 0096-1523. doi: 10.1037/0096-1523.26.2.730.

[32] M. Janner, S. Levine, W. T. Freeman, J. B. Tenenbaum, C. Finn, and J. Wu. Reasoning about physical interactions with object-oriented prediction and planning. In *International Conference on Learning Representations*, 2019.

[33] T. Kipf, E. van der Pol, and M. Welling. Contrastive learning of structured world models. In *International Conference on Learning Representations*, 2020.

[34] A. X. Lee, R. Zhang, F. Ebert, P. Abbeel, C. Finn, and S. Levine. Stochastic adversarial video prediction. *arXiv preprint arXiv:1804.01523*, 2018.

[35] A. Lerer, S. Gross, and R. Fergus. Learning physical intuition of block towers by example. In *International Conference on Machine Learning*, 2016.

[36] W. Li, S. Azimi, A. Leonardis, and M. Fritz. To fall or not to fall: A visual approach to physical stability prediction. *arXiv preprint arXiv:1604.00066*, 2016.

[37] Y. Li, J. Wu, R. Tedrake, J. B. Tenenbaum, and A. Torralba. Learning particle dynamics for manipulating rigid bodies, deformable objects, and fluids. In *International Conference on Learning Representations*, 2019.

[38] Y. Li, T. Lin, K. Yi, D. Bear, D. Yamins, J. Wu, J. Tenenbaum, and A. Torralba. Visual grounding of learned physical models. In *International Conference on Machine Learning*, pages 5927–5936. PMLR, 2020.

[39] A. Mandlekar, Y. Zhu, A. Garg, J. Booher, M. Spero, A. Tung, J. Gao, J. Emmons, A. Gupta, E. Orbay, et al. Roboturk: A crowdsourcing platform for robotic skill learning through imitation. In *Conference on Robot Learning*, pages 879–893. PMLR, 2018.

[40] R. Mottaghi, H. Bagherinezhad, M. Rastegari, and A. Farhadi. Newtonian scene understanding: Unfolding the dynamics of objects in static images. In *Conference on Computer Vision and Pattern Recognition*, pages 3521–3529, 2016.

[41] R. Mottaghi, M. Rastegari, A. Gupta, and A. Farhadi. "what happens if..." learning to predict the effect of forces in images. In *European Conference on Computer Vision*, pages 269–285. Springer, 2016.

[42] D. Mrowca, C. Zhuang, E. Wang, N. Haber, L. F. Fei-Fei, J. Tenenbaum, and D. L. Yamins. Flexible neural representation for physics prediction. In *Advances in Neural Information Processing Systems*, pages 8813–8824, 2018.

[43] S. Nair, S. Savarese, and C. Finn. Goal-aware prediction: Learning to model what matters. In *International Conference on Machine Learning*, pages 7207–7219. PMLR, 2020.

[44] R. S. Nickerson and M. J. Adams. Long-term memory for a common object. *Cognitive Psychology*, 11(3): 287–307, 1979. ISSN 00100285. doi: 10.1016/0010-0285(79)90013-6.

[45] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, pages 8024–8035. Curran Associates, Inc., 2019. URL `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`.

[46] L. Piloto, A. Weinstein, A. Ahuja, M. Mirza, G. Wayne, D. Amos, C.-c. Hung, and M. Botvinick. Probing physics knowledge using tools from developmental psychology. *arXiv preprint arXiv:1804.01128*, 2018.

[47] H. Qi, X. Wang, D. Pathak, Y. Ma, and J. Malik. Learning long-term visual dynamics with region proposal interaction networks. In *International Conference on Learning Representations*, 2021.

[48] R. Rajalingham, E. B. Issa, P. Bashivan, K. Kar, K. Schmidt, and J. J. DiCarlo. Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. *Journal of Neuroscience*, 38(33):7255–7269, 2018.

[49] R. Riochet, M. Y. Castro, M. Bernard, A. Lerer, R. Fergus, V. Izard, and E. Dupoux. Intphys: A framework and benchmark for visual intuitive physics reasoning. *arXiv preprint arXiv:1803.07616*, 2018.

[50] R. Riochet, J. Sivic, I. Laptev, and E. Dupoux. Occlusion resistant learning of intuitive physics from videos. *arXiv:2005.00069 [cs, eess]*, 2020.

[51] A. N. Sanborn, V. K. Mansinghka, and T. L. Griffiths. Reconciling intuitive physics and Newtonian mechanics for colliding objects. *Psychological Review*, 120(2):411–437, 2013. ISSN 0033-295X. doi: http://dx.doi.org/10.1037/a0031912.

[52] A. Sanchez-Gonzalez, N. Heess, J. T. Springenberg, J. Merel, M. Riedmiller, R. Hadsell, and P. W. Battaglia. Graph networks as learnable physics engines for inference and control. In *International Conference on Machine Learning*, 2018.

[53] A. Sanchez-Gonzalez, J. Godwin, T. Pfaff, R. Ying, J. Leskovec, and P. Battaglia. Learning to simulate complex physics with graph networks. In *International Conference on Machine Learning*, pages 8459–8468. PMLR, 2020.

[54] K. Schmeckpeper, A. Xie, O. Rybkin, S. Tian, K. Daniilidis, S. Levine, and C. Finn. Learning predictive models from observation and interaction. In *European Conference on Computer Vision*, 2020.

[55] D. J. Simons and M. S. Ambinder. Change Blindness: Theory and Consequences. *Current Directions in Psychological Science*, 14(1):44–48, 2005. ISSN 0963-7214, 1467-8721. doi: 10.1111/j.0963-7214.2005.00332.x.

[56] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[57] K. Smith, L. Mei, S. Yao, J. Wu, E. Spelke, J. Tenenbaum, and T. Ullman. Modeling expectation violation in intuitive physics with coarse probabilistic object representations. *Advances in Neural Information Processing Systems*, 32:8985–8995, 2019.

[58] K. A. Smith and E. Vul. Sources of Uncertainty in Intuitive Physics. *Topics in Cognitive Science*, 5(1):185–199, 2013. ISSN 17568757. doi: 10.1111/tops.12009.

[59] K. A. Smith, P. W. Battaglia, and E. Vul. Different Physical Intuitions Exist Between Tasks, Not Domains. *Computational Brain & Behavior*, 1(2):101–118, 2018. ISSN 2522-0861, 2522-087X. doi: 10.1007/s42113-018-0007-3.

[60] E. S. Spelke. Principles of object perception. *Cognitive science*, 14(1):29–56, 1990.

[61] A. Tacchetti, H. F. Song, P. A. M. Mediano, V. F. Zambaldi, J. Kramár, N. C. Rabinowitz, T. Graepel, M. Botvinick, and P. W. Battaglia. Relational forward models for multi-agent learning. In *International Conference on Learning Representations*, 2019.

[62] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou. Training data-efficient image transformers & distillation through attention. *arXiv preprint arXiv:2012.12877*, 2020.

[63] T. D. Ullman, E. Spelke, P. Battaglia, and J. B. Tenenbaum. Mind Games: Game Engines as an Architecture for Intuitive Physics. *Trends in Cognitive Sciences*, 21(9):649–665, 2017. ISSN 1364-6613. doi: 10.1016/j.tics.2017.05.012.

[64] R. Veerapaneni, J. D. Co-Reyes, M. Chang, M. Janner, C. Finn, J. Wu, J. Tenenbaum, and S. Levine. Entity abstraction in visual model-based reinforcement learning. In *Conference on Robot Learning*, pages 1439–1456, 2020.

[65] R. Villegas, A. Pathak, H. Kannan, D. Erhan, Q. V. Le, and H. Lee. High fidelity video prediction with large stochastic recurrent neural networks. In *Advances in Neural Information Processing Systems*, 2019.

[66] C. Vondrick, H. Pirsiavash, and A. Torralba. Anticipating visual representations from unlabeled video. In *Conference on Computer Vision and Pattern Recognition*, pages 98–106, 2016.

[67] N. Watters, L. Matthey, M. Bosnjak, C. P. Burgess, and A. Lerchner. Cobra: Data-efficient model-based rl through unsupervised object discovery and curiosity-driven exploration. *arXiv preprint arXiv:1905.09275*, 2019.

[68] B. Wu, S. Nair, R. Martin-Martin, L. Fei-Fei, and C. Finn. Greedy hierarchical variational autoencoders for large-scale video prediction. *arXiv preprint arXiv:2103.04174*, 2021.

[69] Y. Ye, M. Singh, A. Gupta, and S. Tulsiani. Compositional video prediction. In *International Conference on Computer Vision*, pages 10353–10362, 2019.

[70] R. Zhang, J. Wu, C. Zhang, W. T. Freeman, and J. B. Tenenbaum. A comparative evaluation of approximate probabilistic simulation and deep neural networks as accounts of human physical scene understanding. In *CogSci*, 2016.

A Supplemental Material

A.1 Adversarial Stimuli and Model-Human Disagreement

Here we show the distribution of human accuracy on several of the scenarios, which reveals that people are *significantly below chance* on some of the stimuli. Upon investigation, many of these appear to have severe occlusion or are just on the verge of having the opposite trial outcome: a slight change to the initial physical configuration would lead to agent-patient (non)contact. Because **DPI** is not a vision model, it is insensitive to occlusion; and because it receives ground truth, high-resolution object positions and trajectories as inputs and supervision, it may be less susceptible to the "observation noise" that makes certain stimuli "adversarial" to humans. For these reasons, there may be an upper bound to how well particle-based models like **DPI** can match human responses. In addition, **DPI** and the other particle-based models are deterministic and always make binary predictions; this also limits how well they can match average human decisions, which are typically not 0 or 1. A model with probabilistic learned dynamics or decisions might thus, by averaging over samples, make decisions more like the average person [10].

We have attached 10 randomly sampled stimuli from each scenario at the end of the Supplement.

A.2 Across Scenario Generalization

In addition to the *all*, *all-but*, and *only* training protocols, we tested the "best" TDW-trained vision model (**CSWM**) and particle model (**DPI**) for their ability to generalize from any single scenario to any other scenario (Fig. S5). Generalization was fairly homogeneous across training sets for **CSWM**, but this may merely reflect poor overall performance. For **DPI**, clearer patterns emerged: some scenarios were hard to do well on unless they were in the training set (**Drape**, **Dominoes**, **Support**) whereas training on almost *any* scenario was sufficient to give good performance on **Drop**, **Link**, **Roll**, and especially **Collide**. However, no single scenario made for as strong a training set as combining **all** of them; **Drape** and **Support** came the closest, perhaps because they include many object-object interactions in every trial. Overall these data suggest that the eight scenarios cover many distinct physical phenomena, such that experience with any one is insufficient to learn a good prediction model; on the other hand, some phenomena (like object-object contact) may be so ubiquitous that the scenarios with more of them are simply better for efficiently learning about physics in general. The diversity of train-test "fingerprints" for even the most human-like model, combined with the fact that training on **all** scenarios gives the best across-the-board performance, implies that our chief desideratum for the **Physion** benchmark was a crucial choice: developing algorithms on only one or a few physical scenarios would not have produced nearly as general prediction models.

A.3 Model Performance Per Scenario

Table S1 shows model accuracies for every model in each of the eight scenarios, as compared to human performance. There is heterogeneity in performance across the scenarios, with some scenarios (e.g., **Roll**) that people find easy but for which no model approaches human performance, and other scenarios (e.g., **Link**) that people find difficult, but where model accuracy approaches or exceeds humans.

A.4 Model Details

Here we describe the four classes of model we test and provide implementation and training details for the representatives we selected. If not stated otherwise, models' visual encoder and/or dynamics predictor architectures were unchanged from their published implementations.

i. Unsupervised visual dynamics models. These are models explicitly designed to learn dynamical, predictive representations of the visual world without ground truth supervision on physical scene variables. We further divide them into two types: models with *image-like latent representations* and models with *object-like latent representations*. Our representative from the first type, SVG [18], uses a convolutional encoder $\mathcal{E}$ to predict a latent hidden state $\mathbf{p}$, then uses (a) an LSTM-based dynamics model based on the hidden state and a randomly sampled latent from a learned prior distribution to predict a future hidden state $\mathbf{q}$ and (b) a hidden-state-to-image decoder to predict a future frame of the

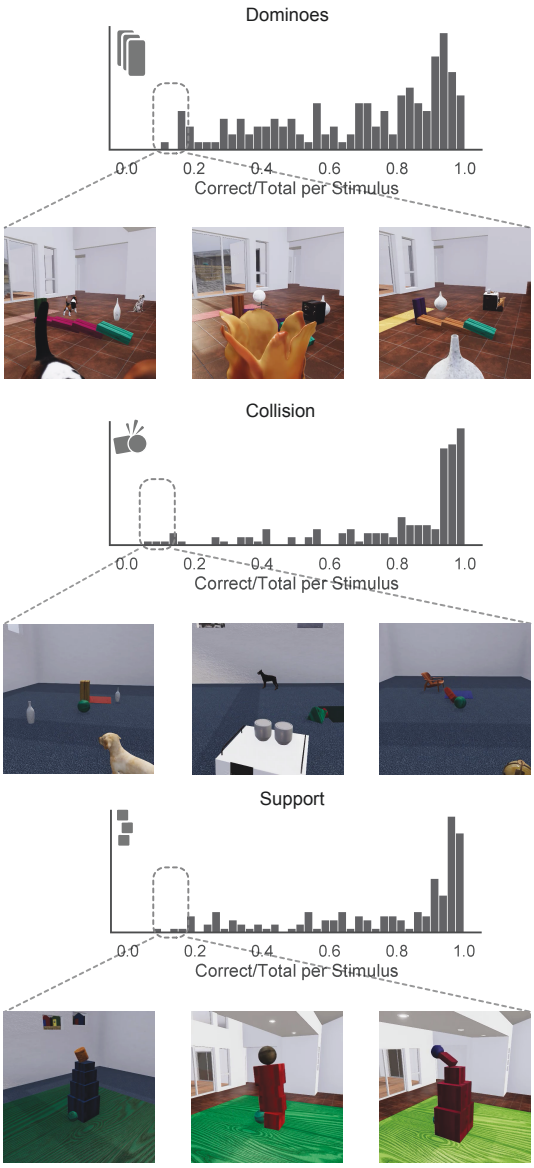

Figure S1: Examples of stimuli on which people performed significantly below chance. The top panel for each scenario shows the per-trial distribution of average human accuracy; sampling from the low end of this distribution gives the examples that are "adversarial" for physical prediction. In most cases, these trials are either impossible to get right on average because of occlusion or they are very close to having a different trial outcome: if the initial physical configuration had been just slightly different, the outcome would be the opposite.

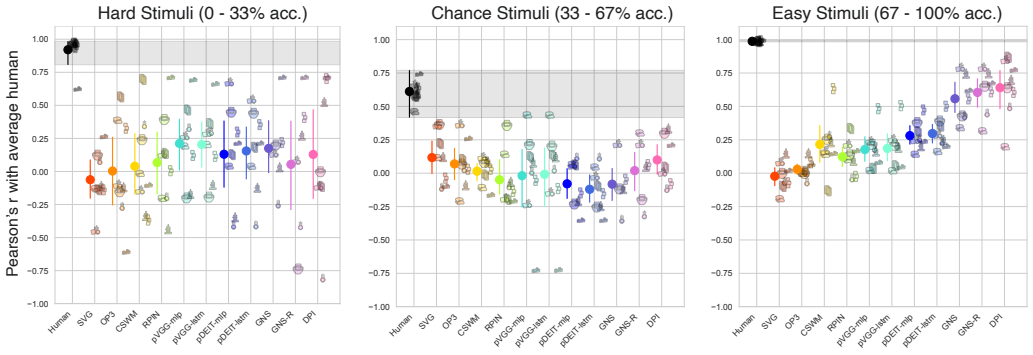

Figure S2: Pearson correlation between model or human responses and the average human response on stimulus subsets defined by average human accuracy. Hard (0 - 33% accuracy), "Chance" (33 - 67% accuracy), and Easy (67 - 100% accuracy) stimuli represent 10%, 22% and 68% of the total testing stimuli across all eight scenarios. Gray bars are the "human zones," defined as the 2.5th - 97.5th percentiles of the distribution the correlation between randomly split halves of the human participant pool. Error bars are the 2.5th - 97.5th percentiles of the bootstrapped across-scenario means.

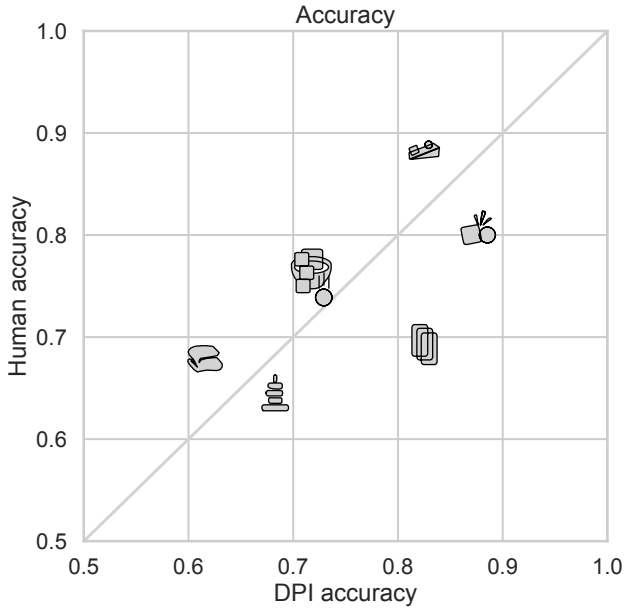

Figure S3: Human and **DPI** average accuracy across testing stimuli for each scenario. Scenarios below the diagonal indicate super-human performance, but the **DPI** model is fed ground truth physical inputs and so does not have to contend with occlusion or other limits of visual observation as humans do.

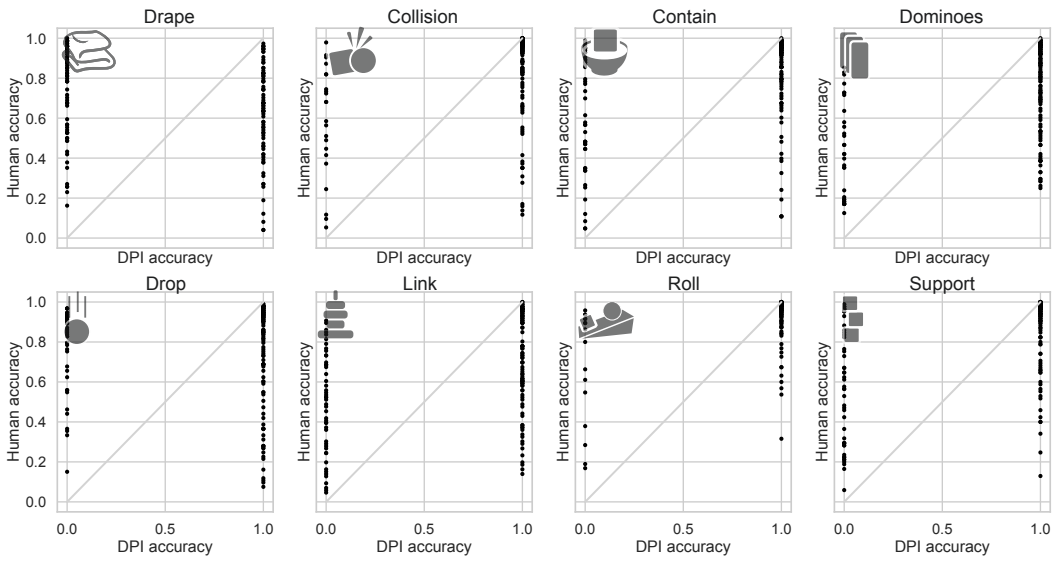

Figure S4: Human accuracy *versus* **DPI** accuracy per stimulus for each scenario. Each dot is one testing stimulus. Note that **DPI** makes predictions with the *observed+simulated* readout protocol only, and does so without a context adaptor: there is a fixed distance threshold that determines whether particles from the agent and patient object are in contact at the end of **DPI**'s learned simulation. As such, this model makes binary predictions, limiting how well correlated its outputs can be with the "average human" (real-valued "average predictions.") This hints that adding a probabilistic component to **DPI** and/or non-binarized readout model might lead to a better human-model match.

input movie, $\hat{X}_{t_{pred}}$. The model is trained by optimizing the variational lower bound. **SVG** is trained on movies from the benchmark; testing this model therefore tests whether physical understanding can emerge from a convolutional future prediction architecture, without imposing further constraints on the structure of the learned latent representation of scenes or dynamics.

Our representatives with object-like latent representations are **CSWM** and **OP3**. These models were designed under the hypothesis that physical understanding requires a decomposition of scenes into objects. We call these representations "object-like" rather than "object-centric" because the latent variables are not explicitly constrained to represent physical objects; they are merely encouraged to do so through the models' inductive biases and unsupervised learning signals. Specifically, both **CSWM** and and **OP3** use convolutional encoders $\mathcal{E}$ to predict *K-factor latent representations*,

$$\mathbf{p} := \mathbf{o}_1 \oplus \mathbf{o}_2 \oplus ... \oplus \mathbf{o}_K, \tag{1}$$

where each inferred *object vector* $\mathbf{o}_k \in \mathbb{R}^{t_{vis} \times P}$ is meant to encode information about one and only one object in the observed scene. The dynamics models for **CSWM** and **OP3** are *recurrent graph neural networks* that pass messages between the object vectors at each iteration of future prediction to produce a new set of predicted object vectors,

$$\mathcal{D}_{\theta_d} \equiv \mathcal{G}_{\theta_d}^{(t_{pred})} : \mathbf{p}[t_{vis}, :, :] \mapsto \hat{\mathbf{o}}_1 \oplus \hat{\mathbf{o}}_2 \oplus ... \oplus \hat{\mathbf{o}}_K \equiv \mathbf{q}, \tag{2}$$

where the graph neural network $\mathcal{G}$ is iterated t_{pred} times to produce as many estimates of the future object states. **OP3** learns the parameters $\theta_e \cup \theta_d$ by applying a *deconvolutional decoder* to render the future object states into a predicted future movie frame, which is used to compute an L2 loss with the actual future frame. **CSWM** instead learns these parameters with a contrastive hinge loss directly on the predicted object-like latent state q; see [33] for details. Thus, these models test whether physical understanding can emerge by predicting scene dynamics through a representation architecture with discrete latent factors, which *could* represent properties of individual objects in the scene but are not explicitly constrained to do so.

ii. Supervised visual-physical dynamics models. We next asked whether vision models with an *explicit object-centric representation*, rather than merely an "object-like" representation, would be

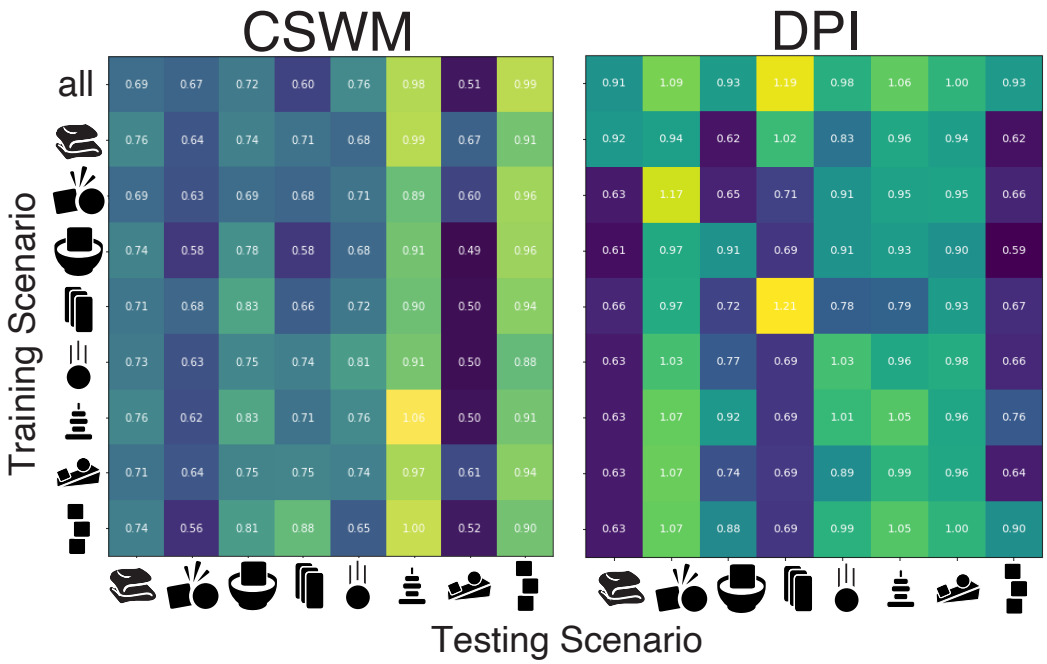

Figure S5: Performance on each scenario's testing set when **CSWM** (left) or **DPI** (right) were trained on each of the scenarios or all of them combined. Color and value for each cell indicate performance relative to the average human on that scenario. For **DPI**, training on any single scenario gave near-human performance on **Collide** and **Roll**, and training on most single scenarios gave near-human performance on **Drop** and **Link**. However, no single training scenario was suitable for generalization to all others, compared to training on all the scenarios. **Drape** and **Support** training appeared to yield the best generalization, perhaps because the ground truth dynamics of these scenarios include many soft and rigid object-object interactions at a wide range of velocities.

better suited for physical understanding. Our representative model from this class was **RPIN** [47]. Region Proposal Interaction Networks (**RPIN**) take a short sequence of N frames as inputs and output the future 2D object positions on the image. The sequence of frames is passed through an encoder network based on a R-CNN like object detection architecture [28] which uses RoIPooling to extract object-centric features from the images. A sequence of k object features is then forwarded to an interaction network [9] to resolve object and environment interactions and predict the future object features at the next time step. The future object features are then decoded to the individual 2D object locations on the image. To be able to estimate velocity and acceleration, we use 4 input images to the interaction network based physics predictor. In contrast to the unsupervised models in section **i**, supervision in the form of human annotated bounding boxes is required to train the RoIPooling based encoder and object location decoder. Thus this model is much more constrained than the models in **i** to represent scenes as a set of discrete objects whose positions change smoothly over time. Although it is not a realistic model of how humans *learn* about the physical world without ground truth supervision, success on our benchmark with **RPIN** where other models failed would strongly suggest that explicit, spatial object-centric representations are useful for intuitive physical understanding of scenes.

iii. Pretrained visual encoders. These visual encoders are optimized to perform a challenging vision task, such as object classification. Although these tasks are not directly related to intuitive physics, it is possible that machine learning models only solve them by learning some partial, implicit representation of the physical world. We tested two models, the standard Convolutional Neural Network VGG-19 and a newer model with a Transformer-based architecture, DeIT, both trained on the supervised ImageNet task. In our decomposition, these models consist only of pretrained encoders $\mathcal{E}_{\theta_e}$ that take t_{vis} independent movie frames as input and produce an output feature vector

$$\mathbf{p}_{1:t_{vis}} := \mathbf{v}_1 \oplus \mathbf{v}_2 \oplus ... \oplus \mathbf{v}_{t_{vis}}, \tag{3}$$

Table S1: Model and human accuracy for each of the eight different scenarios. Numbers indicate mean accuracy with bootstrapped 95% confidence intervals. Italicized values represent instances where the models perform reliably worse than people; bold values represent instances where the models perform reliably better.

Model	Dominoes	Support	Collide	Contain
Human	0.693	0.763	0.809	0.767
SVG	*0.538 [0.512, 0.565]*	*0.596 [0.574, 0.619]*	*0.597 [0.58, 0.612]*	*0.56 [0.545, 0.576]*
OP3	*0.47 [0.457, 0.485]*	*0.516 [0.504, 0.529]*	*0.511 [0.501, 0.522]*	*0.499 [0.488, 0.509]*
CSWM	*0.471 [0.432, 0.519]*	*0.691 [0.636, 0.748]*	*0.552 [0.528, 0.577]*	*0.557 [0.523, 0.593]*
RPIN	*0.625 [0.61, 0.641]*	*0.62 [0.591, 0.651]*	*0.645 [0.617, 0.674]*	*0.601 [0.576, 0.627]*
pVGG-mlp	0.601 [0.505, 0.7]	*0.669 [0.631, 0.708]*	*0.651 [0.608, 0.7]*	*0.638 [0.595, 0.684]*
pVGG-lstm	0.603 [0.513, 0.7]	*0.675 [0.641, 0.711]*	*0.651 [0.606, 0.699]*	*0.643 [0.599, 0.693]*
pDEIT-mlp	0.664 [0.572, 0.757]	*0.686 [0.636, 0.736]*	*0.677 [0.633, 0.721]*	*0.664 [0.645, 0.684]*
pDEIT-lstm	0.664 [0.572, 0.767]	*0.687 [0.637, 0.739]*	*0.681 [0.637, 0.727]*	*0.669 [0.654, 0.684]*
GNS	0.604 [0.477, 0.859]	*0.695 [0.674, 0.711]*	0.85 [0.804, 0.912]	*0.652 [0.62, 0.702]*
GNS-R	0.591 [0.477, 0.819]	*0.686 [0.619, 0.732]*	0.842 [0.808, 0.908]	0.683 [0.512, 0.776]
DPI	0.715 [0.477, 0.841]	*0.626 [0.477, 0.711]*	0.85 [0.725, 0.946]	*0.711 [0.698, 0.717]*
Model	Drop	Link	Roll	Drape
Human	0.744	0.643	0.883	0.678
SVG	*0.533 [0.52, 0.548]*	*0.544 [0.53, 0.558]*	*0.561 [0.545, 0.577]*	*0.545 [0.532, 0.559]*
OP3	*0.526 [0.512, 0.541]*	*0.545 [0.54, 0.551]*	*0.544 [0.529, 0.559]*	*0.548 [0.523, 0.57]*
CSWM	*0.577 [0.542, 0.613]*	0.627 [0.603, 0.649]	*0.609 [0.587, 0.632]*	*0.55 [0.496, 0.605]*
RPIN	*0.551 [0.538, 0.564]*	*0.597 [0.58, 0.614]*	*0.622 [0.604, 0.638]*	*0.596 [0.585, 0.608]*
pVGG-mlp	*0.606 [0.577, 0.639]*	0.614 [0.581, 0.649]	*0.573 [0.548, 0.6]*	*0.6 [0.572, 0.63]*
pVGG-lstm	*0.603 [0.572, 0.638]*	0.618 [0.583, 0.657]	*0.573 [0.546, 0.602]*	*0.599 [0.571, 0.629]*
pDEIT-mlp	*0.619 [0.589, 0.651]*	*0.59 [0.546, 0.633]*	*0.62 [0.601, 0.642]*	*0.608 [0.586, 0.631]*
pDEIT-lstm	*0.614 [0.582, 0.65]*	*0.592 [0.55, 0.639]*	*0.616 [0.597, 0.638]*	*0.608 [0.586, 0.633]*
GNS	*0.708 [0.69, 0.74]*	**0.73 [0.707, 0.756]**	*0.735 [0.718, 0.752]*	0.653 [0.598, 0.714]
GNS-R	*0.712 [0.7, 0.735]*	**0.725 [0.717, 0.737]**	*0.792 [0.752, 0.872]*	0.653 [0.598, 0.714]
DPI	0.755 [0.73, 0.77]	0.657 [0.615, 0.683]	*0.789 [0.769, 0.821]*	*0.556 [0.432, 0.623]*

where $\mathbf{v}_t$ is the vector of activations from the penultimate layer of the encoder on frame t. These were not designed to do explicit physical simulation and thus have no dynamics model $\mathcal{D}_{\theta_d}$. We therefore provide them with simple dynamics models that can be "rolled out" a variable number of time steps,

$$\mathcal{D}_{\theta_d} : \mathbf{p}_{1:t} \mapsto \mathbf{w}_{t+1}, \tag{4}$$

where $\mathcal{D}_{\theta_d}$ is a MLP for **pVGG/pDeIT-mlp** and a LSTM for **pVGG/pDeIT-lstm**, both with a single hidden layer. The encoder parameters θ_e are *frozen* and the dynamics model parameters θ_d are trained with an *unsupervised forward prediction* L2 loss on the unlabeled benchmark training datasets. Thus, dynamics training and evaluation of these models tests whether their pretrained representations contain latent information useful for physical understanding.

iv. Physical state-computable dynamics models. Finally, we consider several models that are not computer vision algorithms at all: rather than taking a movie of RGB frames $\{X_{1:t_{vis}}\}$ as input, they take (a subset of) the ground truth simulator state, $\{S_{1:t_{vis}}\}$ and make predictions about how it will evolve over time, supervised on the ground truth future states. The point of testing these non-visual models is to isolate two distinct challenges in physical understanding: (1) representing some of the physical structure of the world from visual observation (captured by encoding models $\mathcal{E}$) and (2) understanding how that structure behaves (captured by dynamics models $\mathcal{D}$). If models given the ground truth physical state – i.e., models that did not have to solve challenge (1) – matched human performance on our benchmark, we would conclude that the major objective for physical understanding research should be addressing the visual representation problem. On the other hand, if these pure dynamics models still did not match human performance, we would conclude that problem (2) remains open and would benefit from alternative proposals and tests of how people represent and use intuitive physical knowledge about scenes. Thus, comparing these physically explicit, supervised models with those in **i - iii** illustrates how to use our benchmark to diagnose key issues in machine physical understanding.

We consider two graph neural network architectures of this kind, DPI-Net (**DPI**) [37] and **GNS** [52]. Both models operate on a *particle graph representation* of scenes, which for our dataset is

Table S2: Table of open-source code used.

Name	URL	License
SVG[18]	https://github.com/edenton/svg	N/A
C-SWM[33]	https://github.com/tkipf/c-swm	MIT License
OP3[64]	https://github.com/jcoreyes/OP3	MIT License
RPIN [47]	https://github.com/HaozhiQi/RPIN	N/A
DeIT [62]	https://github.com/facebookresearch/deit	Apache License 2.0
VGG [56, 45]	https://github.com/pytorch/vision	BSD 3-Clause License
DPI-Net [37]	https://github.com/YunzhuLi/DPI-Net	N/A
TDW [24]	https://github.com/threedworld-mit/tdw	BSD 2-Clause License

constructed by taking the ground truth collider meshes of each object, converting each mesh vertex into a leaf-level graph node (i.e., particle), and connecting these particles *via* edges that represent physical connections. For GNS, edges are dynamically constructed by adding edges between 2 particles that have distance smaller than a threshold, δ. δ is set to 0.08 for all model variations. For **DPI**, aside from connecting particles with small enough distance, particles belonging to the same object is connected with an object-level root node. The root node can help propagate effect from far away particles within the same object. The DPI-Net run in our experiments differs from the original implementations in two ways: (1) we use relative particle positions, as opposed to absolute particle positions, to improve model generalization, as suggested in **GNS** [52]. (2) The original DPI-Net does not include any leaf-leaf edges between particles within an object. We find out excluding such edges leads to bad performance on objects with a large number of particles. To handle objects with diverse number of particles in our dataset, we include these within object edges that indicates close-by particles.

Both **DPI** and **GNS** explicitly represent each particle's 3D position and instantaneous velocity at each movie frame and make predictions about these node attributes' future values using a rolled out graph neural network, which at each iteration passes learned messages between particles that depend on their attributes and the presence or absence of an edge between them. The key difference between the two models is that DPI-Nets operate on graphs with 2-level hierarchy (, i.e., graph with leaf-level nodes and root-level nodes) while **GNS** operates on flat graphs with no hierarchy. We observe that **GNS** can make good prediction even without explicitly modeling the hierarchy explicitly, yet the objects tend to deform during long-term forward unrolling, due to error accumulation over time. These deformed objects can trigger the models to generate unreasonable predictions such as having all the particles scattering and floating in the free space. To solve the problem, we further include a model variation called GNS-RANSAC (**GNS-R**) that tries to enforce rigid objects to be rigid over time. During model forward unrolling for **GNS**, we run RANSAC [22] on top of each object to compute the 6-Dof rotation and translation matrix for the object and use the matrix to compute the updated positions for the object's particles.

A.5 Experimental Details

Experiments were run on Google Cloud Platform (GCP) across 80 GPUs (NVIDIA T4s & V100s) for two days. DPI-Nets and GNS are trained for 1.5M 2M iterations till converge using Adam optimizer with initial learning rate 1e-4. Experiments take around 2-5 days to train.

A.6 Links to access the dataset and its metadata.

A.7 Long-term preservation plan

A.8 License Information

All products created as part of this project is shared under the MIT license (including code *and* data), and this license has been uploaded to the Github repo where our code is stored and our data is referenced.

We used a number of third-party software packages, each of which typically has its own licensing provisions. Table S2 contains a list of these licenses for many of the packages used.

A.9 Datasheets for dataset

Here are our responses in reference to the Datasheets for Datasets [25] standards.

A.9.1 Motivation

- **For what purpose was the dataset created?** To measure adult human short-term physical future prediction abilities and compare these to predictions made by AI models.
- **Who created the dataset and on behalf of which entity?** The authors listed on this paper, including researchers from Stanford, UCSD, and MIT.
- **Who funded the creation of the dataset?** The various granting agencies supporting the above-named researchers, including both grants to the PIs as well as individual fellowships for graduate students and postdoctoral fellows involved with the project. A partial list of funders includes the NSF, NIH, DARPA, and the McDonnell Foundation.

A.9.2 Composition

- **What do the instances that comprise the dataset represent?** Each instance is a video of a simulated physical scene (e.g. a tower of blocks as it either collapses or remains steady), together with some metadata about that video, including map-structured metadata with depth maps, normal maps, object instance maps, &c, and information about object-object collisions at each timepoint.
- **How many instances are there in total?** The dynamics prediction model training dataset consists of 2000 examples for each of the 8 scenarios. The OCP readout fitting dataset consists of 1000 examples per each of the 8 scenarios. The test dataset (on which human responses were obtained) consists of 150 examples per scenario.
- **Does the dataset contain all possible instances or is it a sample of instances from a larger set?** Data is generated by a simulator; in a sense, the set of datapoints we created is an infinitesimally small subset of data that *could* have been generated. However, we are all here releasing all the examples we did actually generate.
- **What data does each instance consist of?** It consists of a video depicting a physical situation (e.g a tower of blocks falling over), together with simulator-generated metadata about the situation.
- **Is there a label or target associated with each instance?** For the training dataset, there are no labels. For both the OCP readout fitting dataset and the human testing dataset, there are binary labels describing whether the red object collided with the yellow zone during the duration of the trajectory.
- **Is any information missing from individual instances?** No.
- **Are relationships between individual instances made explicit?** Yes. All data is provided in a simple data structure that indicates which instances of data are connected with which instances of metadata.
- **Are there recommended data splits?** Yes, for each of the scenarios in the datasets, there are three splits: (a) a large training split for training physical prediction models from scratch; (b) a smaller readout-training set that is to be used for training the yes/no binary readout training as described in the paper, and (c) the test dataset on which human responses were obtained.
- **Are there any errors, sources of noise, or redundancies in the dataset?** Probably, but we don't know if any at the moment. As these are discovered, they will be fixed and versioned.
- **Is the dataset self-contained, or does it link to or otherwise rely on external resources?** It is self-contained.
- **Does the dataset contain data that might be considered confidential?** No.
- **Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety?** No.
- **Does the dataset relate to people?** No.

A.9.3 Collection Process

- **How was the data associated with each instance acquired? What mechanisms or procedures were used to collect the data? How was it verified?** Videos (for training, readout fitting, and human testing) were generated using the TDW simulation environment. Online crowdsourcing was used to obtain human judgements for each testing video. During the creation of the simulated videos, the researchers looked at the generated videos by eye to verify if the scenarios were correct (e.g. actually depicted the situations desired by our experimental design). Prior to running the actual data collection procedure for humans, we verified that the experimental websites were correct by having several of the researchers complete the experiment themselves.
- **Who was involved in the data collection process and how were they compensated?** PIs, students, and postdocs generated simulator-generated videos. Human responses were obtained via the Profilic platform, and subjects where compensated $4 for participation.
- **Over what timeframe was the data collected?** All simulator-generated scenarios were created during early May 2021. All human data was collected during approximately one week in May 2021.
- **Were any ethical review processes conducted?** All human data collection was approved by Stanford and UCSD IRBs.
- **Does the dataset relate to people?** No.

A.9.4 Preprocessing, clearning and labelling.

- **Was any preprocessing/cleaning/labeling of the data done?** No. All our input data was simulator-generated (so we knew the labels exactly and could avoid any cleaning procedures). The comparison between model and human responses is made directly on the raw collected human judgements with no further preprocessing.

A.9.5 Uses.

- **Has the dataset been used for any tasks already?** Yes, the participants in the human experiments used the data for the single purposes for which it was designed: obtaining detailed characterization of human judgements about short-term physical prediction in simple scenes.
- **Is there a repository that links to any or all papers or systems that use the dataset?**. No other papers use the dataset yet.
- **What (other) tasks could the dataset be used for?** None.
- **Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses?** No.
- **Are there tasks for which the dataset should not be used?** The dataset can only be used to measure abilities of humans or models to make short-term forward predictions about simple physical scenarios.

A.9.6 Distribution.

- **Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created?** Yes it will be completely publicly available via a github repo and the links listed thereupon.
- **How will the dataset will be distributed?** It will be available on Github (where code for dataset generation will be available, and via links to the raw human data that will be listed on that Github repo, and which will refer to permanent Amazon S3 resources.
- **When will the dataset be distributed?** Immediately.
- **Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)?** The dataset and associated code will be licensed under the MIT license.
- **Have any third parties imposed IP-based or other restrictions on the data associated with the instances?** No.

- **Do any export controls or other regulatory restrictions apply to the dataset or to individual instances?** No.

A.9.7 Maintenance

- **Who is supporting/hosting/maintaining the dataset?** Code for dataset generation will be hosted in GitHub, via a publicly-accessible repo. The Github account with which this repo is associated is the institutional account for the CogTools lab (at UCSD).
- **How can the owner/curator/manager of the dataset be contacted?** The corresponding author of the paper can be contacted via email as described in the front page of the paper.
- **Is there an erratum?** Not yet, but there may be in the future.
- **Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances)?** Yes, we expect the dataset to be expanded over the next few months or so. Errors will be corrected as they are discovered on an ongoing basis. Updates will be communicated to users via notes on the commits to the Github repo.
- **Will older versions of the dataset continue to be supported/hosted/maintained?** If newer versions of the dataset are created, these will only be in additional to the existing data. Old versions will be maintained indefinitely.
- **If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so?** No. Making contributions to this dataset requires very substantial expertise in psychophysical experimental design, and we do not contemplate allowing third parties to (e.g.) add new examples of physical scenarios. Of course, the code for generating the data and for setting up crowd-sourced psychophysical collection is completely open source, so others could easily fork our repos and make their own versions of such benchmarks of they choose.

A.10 Structured metadata

We have not created structured metadata for our project in a format like that in schema.org or DCAT as yet, because we expect that through the review feedback process, the exact structure of what metadata we should provide may change a bit. We'd be happy to do this once review is complete. In the meantime, all of our data is available through our github repo, which provides a certain level of metadata bout the project that we think is appropriate for the review process.

A.11 Dataset identifier

Our project provides two types of resources: a dataset and a set of code for creating / analyzing the data. At the moment, we provide access to the code via the GitHub repo, and to the data via Amazon S3 links that are visible via the GitHub repo. We have not yet pushed out data into a standard data repository or created a DOI for it. This is because we expect the specifics of how the data is made available to develop a bit via the paper review process. Once this is complete, we will push the data into a standardized data repository and generate a DOI for it.

B Human experimental study preregistration

This analysis plan was prepared according to the suggested template for experimental study preregistration documents from the Open Science Framework.

B.1 Study information

Title: Human physics benchmarking

B.1.1 Research questions

Predicting the future outcome of physical scenarios is a paradigm case of using models to represent and reason about the world. Intuitive physics is central to intelligent behavior in physical environments. In this study, we aim to identify features of physical scenes that make correct human physical prediction

difficult. Additionally, we aim to collect data on which scenes are difficult for human participants to predict correctly in order to compare human participants against a range of computational models of physical scene prediction.

B.1.2 Hypotheses

We predict that scenes which (1) contain more elements, (2) contain distractor elements and (3) contains occluder elements are harder to correctly predict for human participants. Additionally (4), we predict that scenes that lead to more incorrect predictions also tend to have a longer reaction time (ie. people take longer to come up with an answer to difficult scenes).

B.2 Design Plan

B.2.1 Study design

We conducted 8 experiments, each testing physical judgments for different categories of physical scenarios.

Scenes are generated by sampling values of various physical parameters (e.g., number of physical elements, number of occluder objects, positional jitter, etc.) and generating a stimulus set containing >150 example scenes. From this set, 150 will be randomly sampled such that 50% of the chosen scenes are positive trials (ie. the red target object touches the yellow target zone) and 50% are negative trials. Additionally, we attempt to sample scenes such that the distribution of the other dimensions is roughly equal if possible. Stimuli will be manually checked to ensure that all scenes are usable, do not contain off screen elements, exhibits bugs in the physics engine, contain clipping objects, etc.

Manipulated variables As outlined above, participants are not assigned to any conditions. The manipulations consist of the stimuli with underlying parameters as well as the sampling of stimuli.

B.2.2 Study design: evaluation protocol

Sequence of events in a session 1. Consent form and study information 2. Task explanation 3. Familiarization trials – 10 shown 1. First frozen frame shown for 2000ms, with red/yellow segmentation map indicating agent/patient object flashing at 2Hz 2. Video is played for 1500ms, then hidden 3. Prediction is queried from subject (yes/no) 4. Full video is shown and feedback is given (correct/incorrect) 5. Participants can proceed after full video has played 5. Participants are informed that the main trial starts 6. 100 trials 1. Fixation cross is shown for random interval between 500ms and 1500ms 2. First frozen frame shown for 2000ms, with red/yellow segmentation map indicating agent/patient object flashing at 2Hz 3. Video is played for 1500ms, then hidden 4. Prediction is queried from subject (yes/no) 7. Demographics & Feedback * age * gender * education level * difficulty rating ("How difficult did you find this task?", 5 point Likert scale) 8. Participants are shown their rate of correct guesses 9. End of study

Each stimulus consists of a short video clip of a visual scene containing various objects physically interacting with each other. Each of these 150 trials began with a fixation cross, which was shown for a randomly sampled time between 500ms and 1500ms. To indicate which of the objects shown is the agent and patient object, participants were then shown the first frame of the video for 2000ms. During this time, the agent and patient objects were overlaid in red and yellow respectively. The overlay flashed on and off with a frequency of 2Hz. After this, the first 1500ms of the stimulus were played. After 1500ms, the stimulus is removed and the response buttons are enabled. The experiments moved to the next phase after the participants made a prediction by selecting either "YES" or "NO."

Participants first completed 10 familiarization trials before moving on to complete 150 test trials. During the familiarization phase, all participants were presented with the same sequence of stimuli and were provided with feedback indicating whether their prediction was correct and were shown the unabridged stimulus including the result of the trial. During the test phase, participants were presented with the same set of stimuli in a randomized sequence, and were not provided with accuracy feedback nor did they observe the subsequent video frames in the scenario.

B.2.3 Measured variables

We measure: * `response`: prediction (either yes/no) * `rt`: time taken to make prediction

After the trials, participants will be asked to provide: * age * gender * education level * difficulty rating ("How difficult did you find this task?", 5 point Likert scale) * free form feedback on the task

After the end of the study, participants will be told their overall accuracy and the corresponding percentile compared to other participants on the study.

B.3 Sampling Plan

B.3.1 Data collection procedure

Participants will be recruited from Prolific and compensated $4, which roughly corresponds to $12/hr. participants will not be rewarded for correct responses.

Participants are only allowed to take the task once. However, participants are able to take a version of the experiment with another scenario.

B.3.2 Sampling procedure

Data collection will be stopped after 100 participants have completed the experiment.

B.4 Analysis Plan

B.4.1 Data exclusion criteria

Data from an entire experimental session will be excluded if the responses: * contain a sequence of greater than 12 consecutive "yes" or 12 consecutive "no" answers (based on simulations run with p(yes)=0.5) * contain a sequence of at least 24 trials alternating "yes" and "no" responses * are correct for fewer than 4 out of 10 familiarization trials (i.e., 30% correct or lower) * the mean accuracy for that participant is below 3 standard deviations below the median accuracy across all participants for that scenario * the mean log-transformed response time for that participant is 3 standard deviations above the median log-transformed response time across all participants for that scenario

Excluded sessions will be flagged. Flagged sessions will not be included in the main analyses. We will also conduct our planned analyses with the flagged sessions included to investigate the extent to which the outcomes of the main analyses change when these sessions are included. Specifically, we will fit a statistical model to all sessions and estimate the effect of a session being flagged on accuracy.

B.4.2 Missing data

We will only include sessions that are complete (i.e., response collected for all trials) in our main analyses.

B.4.3 Planned analyses

Human accuracy across participants for each stimulus We will analyze accuracy for each stimulus by computing the proportion of correct responses across all participants who viewed that stimulus.

Human accuracy across stimuli for each participant We will analyze accuracy for each participant by computing the proportion of correct responses across all stimuli.

Human-human consistency for each stimulus We will estimate human-human consistency for each stimulus by computing the proportion of responses that match the modal response for that stimulus (whether that modal response is correct or incorrect).

Human-human consistency across stimuli (within scenario) We will analyze human-human consistency by computing the mean correlation between (binary) response vectors produced by each human participant across all stimuli within each scenario.

Human accuracy as a function of stimulus attributes We will conduct exploratory analyses of human accuracy as a function of various scenario-specific stimulus attributes that varied across trials. We will examine those stimulus attributes that varied across stimuli within each scenario and explore the relationship between each individual attribute and human accuracy, as well as beetween linear combinations of them and human accuracy.

Human accuracy by scenario We will fit human responses across all scenarios with a mixed-effects logistic regression model, including scenario as a fixed effect and participants and individual stimuli as random effects.

Other exploratory human behavioral analyses

- We will explore the relation of demographic variables on the performance of participants: how does age, gender, educational status and the the result of a one-trial spatial reasoning task relate to the overall accuracy of a subject?
- We will additionally explore any potential left/right or yes/no response biases.

Human-model comparisons We will compare human and model behavior in two ways: **absolute performance** and **response pattern.**

Absolute Performance We will compare the accuracy of each model to the mean accuracy of humans, for each scenario. To do this, we will first compute estimates of mean human accuracy for each scenario and construct 95% confidence intervals for each of these estimates. These confidence intervals will be constructed by bootstrapping: specifically, for an experiment with N participants, we will resample N participants with replacement and compute the proportion correct for that bootstrapped sample. We will take repeat this resampling procedure 1000 times to generate a sampling distribution for the mean proportion correct. The 2.5th and 97.5th percentile will be extracted from this sampling distribution to provide the lower and upper bounds of the 95% confidence interval.

For each model, we will then compare their proportion correct (a point estimate) to the human confidence interval.

Response Pattern We will compare the pattern of predictions generated by each model to the pattern of predictions generated by humans.

We will do this by using two standard inter-rater reliability metrics:

Correlation between average-human and model responses For each stimulus, we will compute the proportion of "hit" responses by humans. For each stimulus, we will extract the hit probability generated by models. For each scenario (i.e., domain), we will compute the root-mean-squared deviation between the human proportion-hit vector and the model probability-hit vector. To estimate variability across human samples, we will conduct bootstrap resampling (i.e., resampling data from individual participants with replacement), where for each bootstrap sample we will re-compute the correlation between the model probability-hit vector and the (bootstrapped) human proportion-hit vector.

Cohen's kappa

For each pair of human participants, we will compute Cohen's kappa between their responses across the 150 stimuli, yielding a distribution of pairwise human-human Cohen's kappa. The mutually exclusive categories used in calculating Cohen's kappa is whether each of the 150 responses was predicted to be positive or negative. For each model, we will compute Cohen's kappa between its response vector and every human participant, as well as every other model. A model's response pattern will be considered more similar to humans' insofar as the mean model-human Cohen's kappa (across humans) lies closer to the mean human-human Cohen's kappa (for all pairs of humans).