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

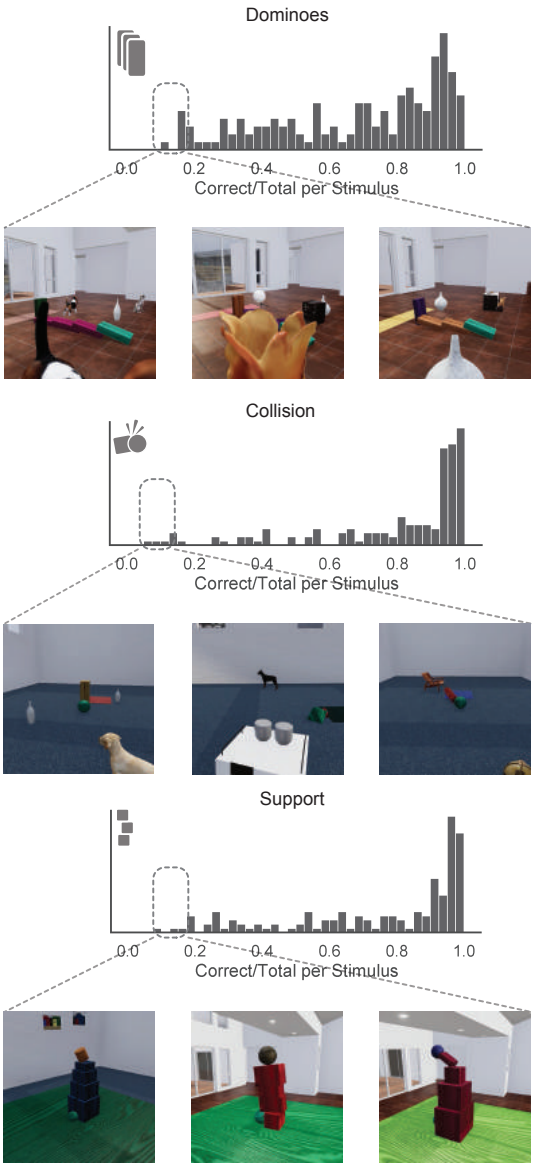


Figure S1: Examples of stimuli on which people performed significantly below chance. The top panel for each scenario shows the per-trial distribution of average human accuracy; sampling from the low end of this distribution gives the examples that are "adversarial" for physical prediction. In most cases, 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.

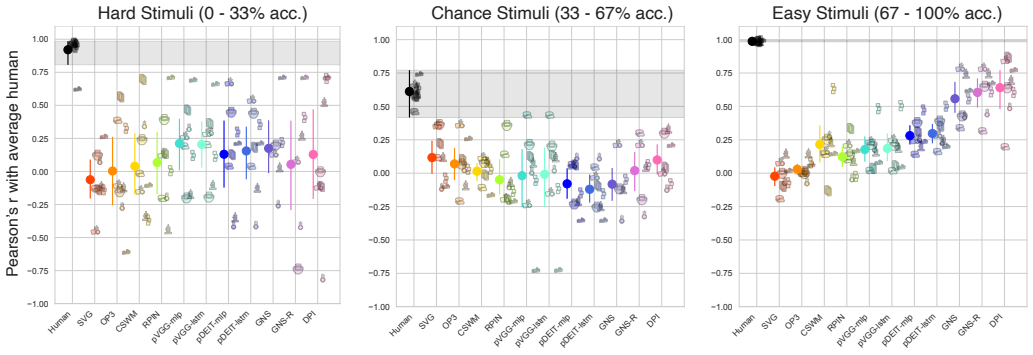


Figure S2: Pearson correlation between model or human responses and the average human response on stimulus subsets defined by average human accuracy. Hard (0 - 33% accuracy), "Chance" (33 - 67% accuracy), and Easy (67 - 100% accuracy) stimuli represent 10%, 22% and 68% of the total testing stimuli across 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.

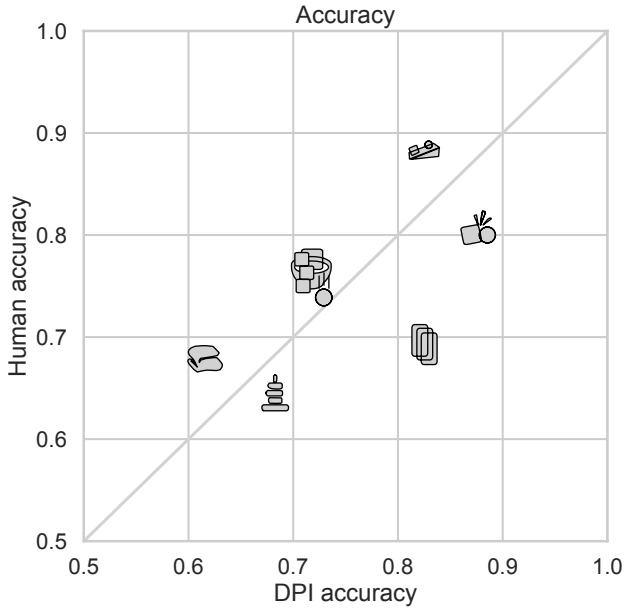


Figure S3: Human and **DPI** average accuracy across testing stimuli for each scenario. Scenarios below the diagonal indicate super-human performance, but the **DPI** model is fed ground truth physical inputs and so does not have to contend with occlusion or other limits of visual observation as humans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

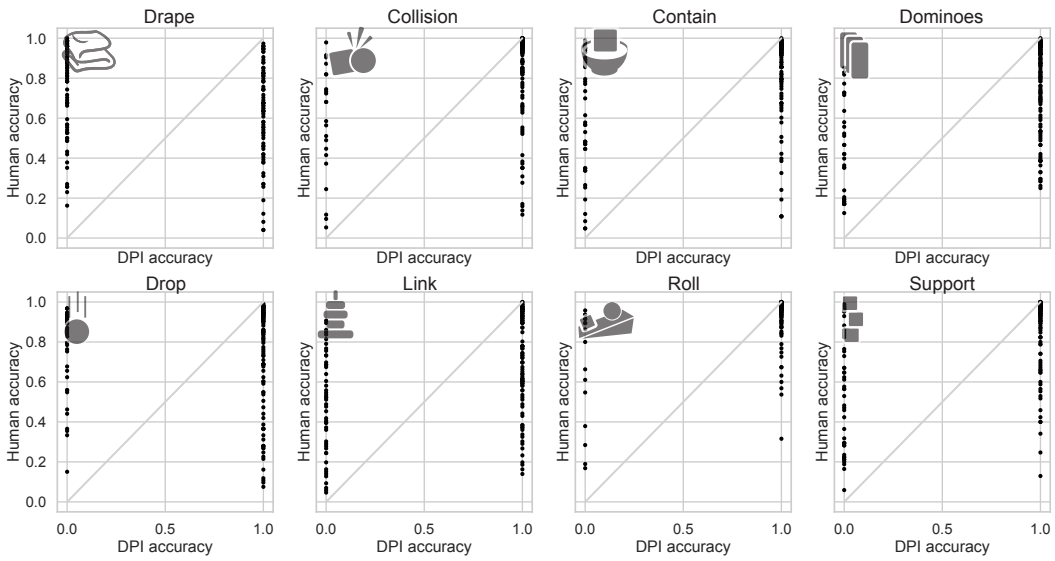


Figure S4: Human accuracy *versus* **DPI** accuracy per stimulus for each scenario. Each dot is one testing stimulus. Note that **DPI** makes predictions with the *observed+simulated* readout protocol only, and does so without a context adaptor: there is a fixed distance threshold that determines whether 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

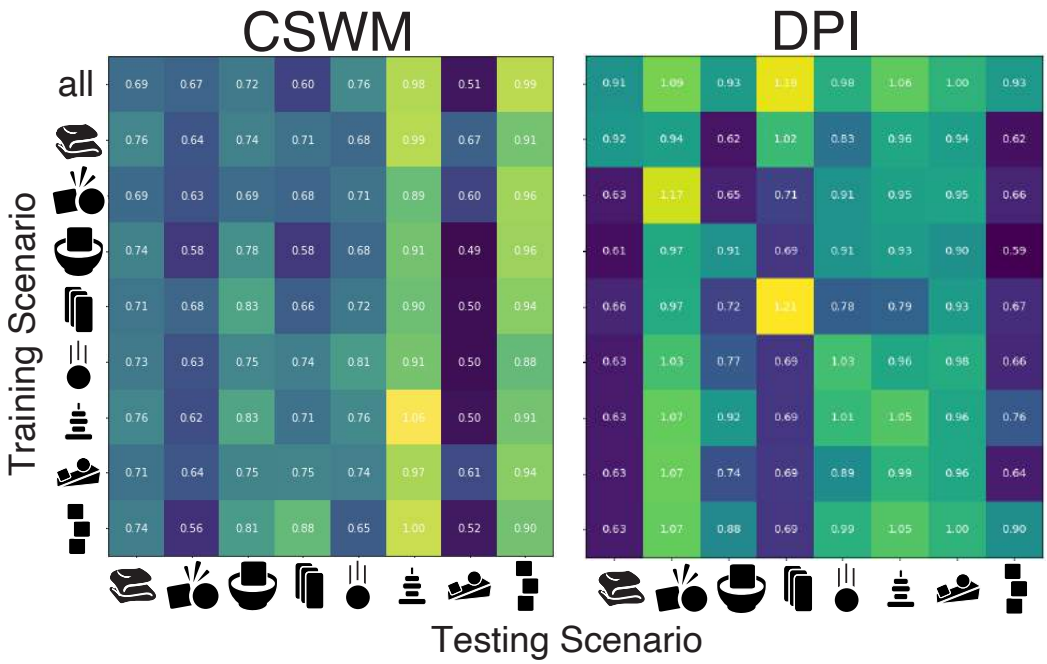


Figure S5: Performance on each scenario's testing set when **CSWM** (left) or **DPI** (right) were trained on each of the scenarios or all of them combined. Color and value for each cell indicate performance relative to the average human on that scenario. For **DPI**, training on any single scenario 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).

T = 0 | T = 50 | T = 100 | T = 150 | T = 200 | T = 250 | T = 300

label = True
acc = 0.850

label = False
acc = 0.860

label = False
acc = 0.950

label = False
acc = 0.940

label = True
acc = 0.310

label = False
acc = 0.620

label = True
acc = 0.560

label = True
acc = 0.370

label = False
acc = 0.960

label = True
acc = 0.260

T = 0 | T = 25 | T = 50 | T = 75 | T = 100 | T = 125 | T = 150

label = True acc = 0.980

label = True acc = 0.990

label = True acc = 0.970

label = False acc = 0.170

label = True acc = 0.250

label = False acc = 0.970

label = True acc = 0.530

label = False acc = 0.970

label = True acc = 0.940

label = True acc = 0.980

T = 0 | T = 25 | T = 50 | T = 75 | T = 100 | T = 125 | T = 150

label = False
acc = 0.723

label = False
acc = 0.950

label = True
acc = 0.485

label = False
acc = 0.960

label = False
acc = 0.980

label = False
acc = 0.723

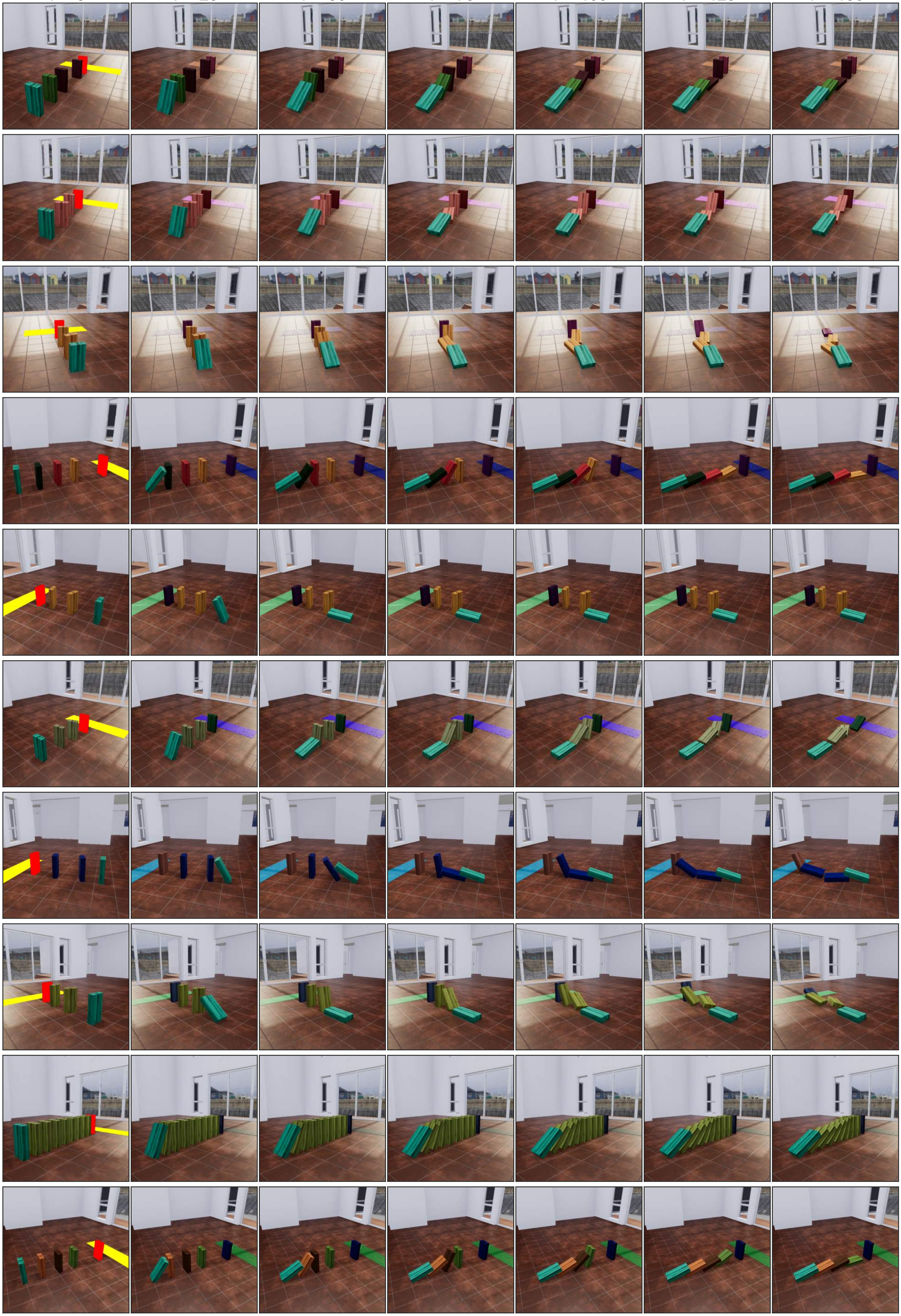
label = True
acc = 0.475

label = True
acc = 0.792

label = False
acc = 0.990

label = False
acc = 0.970

T = 0 | T = 25 | T = 50 | T = 75 | T = 100 | T = 125 | T = 150



T = 0 | T = 25 | T = 50 | T = 75 | T = 100 | T = 125 | T = 150

label = True
acc = 0.360

label = False
acc = 0.530

label = True
acc = 0.370

label = False
acc = 0.910

label = True
acc = 0.960

label = True
acc = 0.900

label = False
acc = 0.820

label = False
acc = 0.710

label = True
acc = 0.280

label = True
acc = 0.430

T = 0 | T = 50 | T = 100 | T = 150 | T = 200 | T = 250 | T = 300

label = True
acc = 0.430

label = False
acc = 0.940

label = False
acc = 0.880

label = False
acc = 0.580

label = True
acc = 0.720

label = False
acc = 0.420

label = True
acc = 0.930

label = True
acc = 0.300

label = False
acc = 0.700

label = True
acc = 0.960

T = 0 | T = 25 | T = 50 | T = 75 | T = 100 | T = 125 | T = 150

label = False
acc = 0.960

label = False
acc = 0.910

label = False
acc = 0.840

label = False
acc = 0.940

label = False
acc = 0.960

label = True
acc = 0.920

label = True
acc = 0.970

label = True
acc = 0.970

label = False
acc = 0.960

label = False
acc = 0.960

T = 0 | T = 50 | T = 100 | T = 150 | T = 200 | T = 250 | T = 300

label = True acc = 0.941

label = False acc = 0.343

label = True acc = 0.706

label = True acc = 0.980

label = False acc = 0.843

label = True acc = 0.931

label = True acc = 0.304

label = False acc = 0.618

label = False acc = 0.196

label = True acc = 0.686