IntPhys: A Framework and Benchmark for Visual Intuitive Physics Reasoning

Ronan Riochet, Mario Ynocente Castro, Mathieu Bernard, Adam Lerer, Rob Fergus, Véronique Izard, Emmanuel Dupoux

IntPhys: A Benchmark for Visual Intuitive Physics Reasoning

Ronan Riochet
Ecole Normale Superieure
INRIA
ronan.riochet@inria.fr

Mario Ynocente Castro
ycmario@gmail.com

Mathieu Bernard
Ecole Normale Superieure
INRIA
mathieu.a.bernard@inria.fr

Adam Lerer
Facebook AI Research
alerer@fb.com

Rob Fergus
Facebook AI Research
robfergus@fb.com

Véronique Izard
Université Paris Descartes
CNRS
veronique.izard@parisdescartes.fr

Emmanuel Dupoux
CoML, ENS/CNRS/EHESS/INRIA/PSL Research University
emmanuel.dupoux@gmail.com

Abstract

In order to reach human performance on complex visual tasks, artificial systems need to incorporate a significant amount of understanding of the world in terms of macroscopic objects, movements, forces, etc. Inspired by work on intuitive physics in infants, we propose an evaluation benchmark which diagnoses how much a given system understands about physics by testing whether it can tell apart well matched videos of possible versus impossible events constructed with a game engine. The test requires systems to compute a physical plausibility score over an entire video. It is free of bias and can test a range of specific physical reasoning skills. We then describe two Deep Neural Networks systems aimed at learning intuitive physics in an unsupervised way, using only physically possible videos. The systems are trained with a future semantic mask prediction objective and tested on the possible versus impossible discrimination task. The analysis of their results compared to human data gives novel insights in the potentials and limitations of next frame prediction architectures.

1 Introduction

Despite impressive progress in machine vision on many tasks (face recognition [42], object recognition [24], [18], object segmentation [34], etc.), artificial systems are still far human performance when it comes to common sense reasoning about objects in the world or understanding of complex visual scenes. Indeed, even very young children have the ability to represent macroscopic objects and track their interactions through time and space. This ability develops at a fast pace: for instance, at 2-4 months, infants are able to parse visual inputs in terms of permanent, solid and spatio-temporally continuous objects ([20], [39]). At 6 months, they understand the notion of stability, support and causality ([38], [4], [2]). Between 8 and 10 months, they grasp the notions of gravity, inertia, and conservation of momentum in collision; between 10 and 12 months, shape constancy [43], and so on.

Reverse engineering the capacity to autonomously learn and exploit intuitive physical knowledge would help building more robust and adaptable real life applications (self-driving cars, workplace or household robots). Vice-versa, building models able of learning elementary physical reasoning would provide developmental scientists with predictive models for normal or pathological cognitive development in infants. Many roadblocks need to be overturned in order to achieve this objective. In

Preprint. Work in progress.

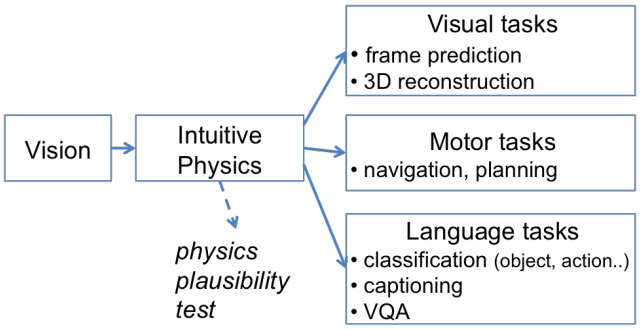

Figure 1: Popular end-to-end applications involving scene understanding and proposed evaluation method based on physical plausibility judgments.

this paper, we address one of them: the evaluation problem. How do we know that a given system has (or has not) learned a certain level of physical understanding?

One possibility could be to use end-to-end applications. As illustrated in Figure 1, one can distinguish three classes of machine vision tasks which require at least some understanding of the physical world. 'Visual' tasks aim at recovering high level structure from low level (pixel) information. For instance, the recovery of 3D structure from static or dynamic images (e.g., [8, 10]) or object tracking (e.g., [23, 7]). 'Motor' tasks aim at predicting the visual outcome of particular actions (e.g., [14]) or to plan an action in order to reach a given outcome (e.g. [32]). Language tasks requires the artificial system to translate input pixels into a verbal description, either through captioning [13] or visual question answering (VQA [45]). Obviously, depending on the complexity of the question, one touches the capacity to categorize objects based on their visual appearance (e.g., [37, 18]) to the capacity of classifying attributes or relations between objects (e.g., [22, 19]) or actions (e.g. [40]).

Using such end-to-end applications for the purpose of evaluation runs into two risks: (a) dataset bias (b) noisy measure. The first risk (also known as the Clever Hans problem [19]), is that real life application datasets often contain inherent statistical biases, which make it sometimes possible to achieve good performance with only minimal involvement in solving the problem at hand. The second risk is that the overall performance of a system is a complicated function of the performance of its parts; therefore, if a VQA system has a worse performance than another one, it could be, not because it better understands physics, but because it has a worse language model. The combined risk of both overestimating and underestimating the intuitive physical understanding of end-to-end systems could be alleviated by using a multi-task setup and show that an intermediate physical embedding derived for one type of task would help with another task.

Here, we take another route and propose an evaluation diagnostic which is completely independent of any model and end-to-end task. It is conceived as a set of "unit tests", which probe for specific aspects of intuitive physics independently of how the model has been constructed and what it is used for. We see three advantages of using such tests: (1) the tests provide directly interpretable results (as opposed to a composite score reflecting a black box performance), (2) as they are constructed in a careful counterbalanced way, they control for bias, (3) they enable for direct human-machine comparison. The proposed tests are based on the "violation of expectation" paradigm, whereby infants or animals are presented with real or virtual animated 3D scenes which may contain a physical impossibility. The measure is whether the organism displays a "surprise" reaction to the physical impossibility, which is taken to reflect a violation of it's internal predictions [3]. Similarly our *"physical plausibility test"* simply requires systems to output a scalar variable upon the presentation of a video clip, which we will call a 'plausibility score'. We expect the plausibility score to be lower for clips containing the violation of a physical principle than for matched clips with no violation. By varying the nature of the physical violation, one can probe different types of reasoning (laws regarding objects and their properties, laws regarding objects movement and interactions, etc.). Given that most vision systems are not specifically designed to output a plausibility score (but rather task-specific output), we provide a small development set to enable researchers to extract this single scalar (which could be relatively easy to do as many machine vision systems are based on probabilities or error minimization). Apart from this minor adjustment, the test can be administered to a large variety of models.

Table 1: List of the conceptual blocks of the Intuitive Physics Framework.

Block Name	Physical principles	Computational challenge
O1. Object permanence	Objects don't pop in and out of existence	Occlusion-resistant object tracking
O2. Shape constancy	Objects keep their shapes	Appearance-robust object tracking
O3. Spatio-temporal continuity	Trajectories of objects are continuous	Tracking/predicting object trajectories
O4. Energy / Momentum	Constant kinetic energy and momentum	Tracking/predicting object trajectories

The contribution of this paper is in two parts. The first part explains the logic of the intuitive physics tests and presents the IntPhys Benchmark containing several blocks, each of which testing for different aspects of macroscopic physics (object permanence, shape constancy, spatio-temporal continuity, etc). The second part describes two simple 'infant' models which attempt to pass the first block of IntPhys after a phase of self-supervised observation learning, on a training set containing on random videos of interacting objects, but which only contain physically possible examples. We compare the performance of this rather simple model to that of humans participants.

2 IntPhys: a set of diagnostic tests for Intuitive Physics

IntPhys is a benchmark designed to address the evaluation challenges for intuitive physics in vision systems. It can be run on any of machine vision system (captioning and VQA systems, systems performing 3D reconstruction, tracking, planning, etc), be they engineered by hand or trained using statistical learning, the only requirement being that the tested system should output a scalar for each test video clip reflecting the plausibility of the clip as a whole. Such a score can be derived from prediction errors, or posterior probabilities, depending on the system.

The Benchmark consists of synthetic videos constructed with a python interfaced game engine (UnrealEngine 4), enabling both realistic physics and precise control. It consists in a *dev set* and a *test set* is constructed according to similar principles. The dev set is kept intentionally very small because its sole purpose is to verify the validity of the plausibility score and, importantly, *not* to fine tune the system on a possible vs impossible classification task.

We present four intuitive physics reasoning problems studied in this framework, as well as the design features of our test: minimal sets, parametric task difficulty, and evaluation metric.

2.1 A hierarchy of intuitive physics problems

Taking advantage of behavioral work on intuitive studies [1], we organize the tests into four blocks (see 1), each one corresponding to a core principle of intuitive physics, and each raising its particular machine vision challenge. The first two blocks are related to the conservation through time of intrinsic properties of objects. Object permanence (O1), corresponds to the fact that objects continuously exist through time and do not pop in or out of existence. This turns into the computational challenge of tracking objects through occlusion. The second block, shape constancy (O2) describes the tendency of rigid objects to preserve their shape through time. This principle is more challenging than the preceding one, because even rigid objects undergo a change in appearance due to other factors (illumination, distance, viewpoint, partial occlusion, etc.). The other two blocks (O3-4) relate to object's movement through time, and the conservation laws which govern these movements for rigid inanimate macroscopic objects. These principles map into progressively more challenging problems of trajectory prediction.

2.2 Minimal sets design

An important design principle of our evaluation framework relates to the organization of the possible and impossible movies in extremely well matched sets to avoid the Clever Hans problem. This is illustrated in Figure 2 for object permanence. We constructed matched sets comprising four movies, which contain an initial scene at time t_1 (either one or two objects), and a final scene at time t_2 (either one or two objects), separated by a potential occlusion by a screen which is raised and then lowered

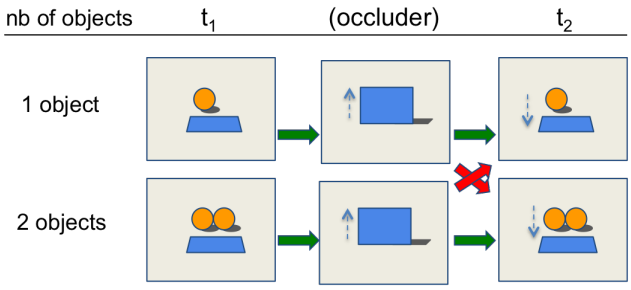

Figure 2: Illustration of the minimal sets design with object permanence. Schematic description of a static condition with one vs. two objects and one occluder. In the two possible movies (green arrows), the number of objects remains constant despite the occlusion. In the two impossible movies (red arrows),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

for a variable amount of time. At its maximal height, the screen completely occludes the objects so that it is impossible to know, in this frame, how many objects are behind the occluder.

The four movies are constructed by combining the two possible beginnings with the two possible endings, giving rise to two possible (1→1 and 2→2) and two impossible (1→2 and 2→1) movies. Importantly, across these 4 movies, the possible and impossible ones are made of the exact same frames, the only factor distinguishing them being the temporal coherence of these frames. Such a design is intended to make it difficult for algorithms to use cheap tricks to distinguish possible from impossible movies by focusing on low level details, but rather requires models to focus on higher level temporal dependencies between frames.

2.3 Parametric manipulation of task complexity

Our second design principle is that in each block, we will vary the stimulus complexity in a parametric fashion. In the case of the object permanence block, for instance, stimulus complexity can vary according to three dimensions. The first dimension is whether the change in number of objects occurs in plain view (*visible*) or hidden behind an occluder (*occluded*). A change in plain view is evidently easier to detect whereas a hidden change requires an element of short term memory in order to keep a trace of the object's through time. The second dimension is the complexity of the object's motion. Tracking an immobile object is easier than if the object has a complicated motion. The third dimension is the number of objects involved in the scene. This tests for the attentional capacity of the system as defined by the number of objects it can track simultaneously. Manipulating stimulus complexity is important to establish the limit of what a vision system can do, and where it will fail. For instance, humans are well known to fail when the number of objects to track simultaneously is greater than four [35].

2.4 The physical possibility metrics

Our evaluation metrics depend on the system's ability to compute a plausibility score $P(x)$ given a movie x. Because the test movies are structured in N matched k-uplets (in Figure 2, $k = 4$) of positive and negative movies $S_{i=1..N} = \{Pos_i^1..Pos_i^k, Imp_i^1..Imp_i^k\}$, we derive two different metrics. The *relative* error rate L_R computes a score within each set. It requires only that within a set, the positive movies are more plausible than the negative movies.

$$L_R = \frac{1}{N} \sum_i \mathbb{1}_{\sum_j P(Pos_i^j) < \sum_j P(Imp_i^j)} \tag{1}$$

The absolute error rate L_A requires that globally, the score of the positive movies is greater than the score of the negative movies. It is computed as:

$$L_A = 1 - AUC(\{i, j; P(Pos_i^j)\}, \{i, j; P(Imp_i^j)\}) \tag{2}$$

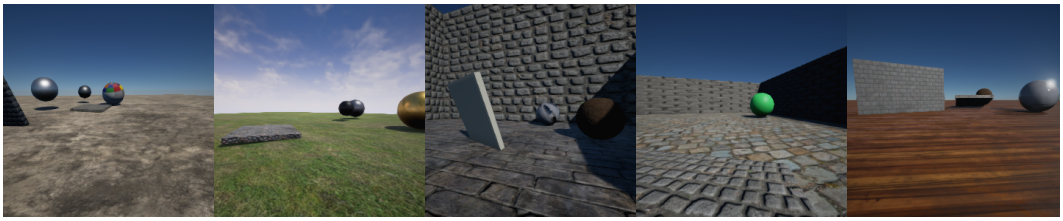

Figure 3: Examples of frames from the training set.

Where AUC is the Area Under the ROC Curve, which plots the true positive rate against the false positive rate at various threshold settings.

2.5 Implementation

Each of these four blocks consists in a set of videos constructed with Unreal Engine 4.0 (See Figure 3 for some examples), containing 18 types of movies (3 objects, 2 occlusions and 3 types of movements). A dev set block is instantiated by 20 different renderings of these 18 scenarios, objects positions, shapes, trajectories, resulting in 360 movies. A test set block is instantiated by 200 different renderings of these scenarios (for a total of 3600 movies) and uses different objects, textures, motions, etc. All of the objects and textures of the dev and test sets are present in the training set.

The purpose of the dev set released in IntPhys is to help in the selection of an appropriate plausibility score, and in the comparison of various architectures (hyper-parameters), but it should *not* serve to train the model's parameters (this should be done only with the training set). This is why the dev set is kept intentionally small. The test set has more statistical power and enables a fine grained evaluation of the results across the different movie subtypes. This benchmark along with video examples are available on the project page `http://www.intphys.com`.

3 Two 'infant' learning models

In this section, we present two learning systems which attempt to learn intuitive physics in an unsupervised/self-supervised observational setting. One can imagine an agent who only sees physical interactions between objects seen from a first-person perspective, but cannot move nor interact with them. Arguably, this is a much more empoverished learning situation than that faced by infants, who can to a limited extent, explore and interact with their environment even with the limited motor abilities of their first year of life. It is however interesting to establish how far one can get with such simplified inputs, which are easy to gather in abundant amounts in the real world with video cameras. In addition, this enables an easier comparison between models, because they all get the same training data. We only presented here the results of the model on the easiest block (O1) of the Benchmark.

In a setup like this, a rich source of learning information resides in the temporal dependencies between successive frames. Based on the literature on next frame prediction, we propose two neural network models, trained on a future frame objective. Our first model has a CNN encoder-decoder structure and the second is a conditional Generative Adversarial Network (GAN, [17]), with a similar structure as DCGAN [36]. For both model architectures, we investigate two different training procedures: in the first, we train models to predict short-future images with a prediction span of 5 frames; in the second, we predict long-future images with a prediction span of 35 frames.

Preliminary work with predictions at the pixel level revealed that our models failed at predicting convincing object motions, especially for small objects on a rich background. For this reason, we switched to computing predictions at a higher level, using object masks. We use the metadata provided in the benchmark training (see section 3.1) set to train a semantic mask Deep Neural Network (DNN). This DNN uses a resnet-18 pretrained on Imagenet to extract features from the image, from which a deconvolution network is trained to predict the semantic mask (distinguishing three types of entities: background, occluders and objects). We then use this mask as input to a prediction component which predicts future masks based on past ones.

To evaluate these models on our benchmark, our system needs to output a plausibility score for each movie. For this, we compute the prediction loss along the movie. Given past frames, a plausibility score for the frame f_t can be derived by comparing f_t with the prediction $\hat{f}_t$. Like in [16], we use the analogy with an agent running an internal simulation ("visual imagination"); here we assimilate a greater distance between prediction and observation with a lower plausibility. In subsection 3.3 we detail how we aggregate the scores of all frames into a plausibility score for the whole video.

3.1 Training set

We constructed a training set of videos constructed with the same software as the benchmark. However, object textures, dynamics as well as camera positions are sampled in a much richer (i.e. less controlled) distribution than videos in the test and dev sets. Since this is supposed to model unsupervised obervational learning, these videos are all physically possible. However, we do provide additional information which may help the learner: depth flow and instance segmentation masks.

This training set is composed of 15K videos of 100 frames, totalling 21 hours of videos (at rate 15 frames per second). Each video is delivered as stacks of raw image (288 x 288 pixels), totalling 157Gb of uncompressed data. We also release the source code for data generation, allowing users to generate a larger training set if desired.

3.2 Models

Through out the movie, our models take as input two frames (f_{i_1}, f_{i_2}) and predict a future frame f_{target}. The prediction span is independent from the model's architecture and depends only on the triplets $(f_{i_1}, f_{i_2}, f_{target})$ provided during the training phase. Our two architectures are trained either on a short term prediction task (5 frames in the future), or a long term prediction task (35 frames). Intuitively, short-term prediction will be more robust, but long-term prediction will allow the model to grasp long-term dependencies and deal with long occlusions.

CNN encoder-decoder We use a resnet-18 [18] pretrained on Imagenet [37] to extract features from input frames (f_{i_1}, f_{i_2}). A deconvolution network is trained to predict the semantic mask of future frame f_{target} conditioned to these features, using a L2 loss.

Generative Adversarial Network As a second baseline, we propose a conditional generative adversarial network (GAN, [29]) that takes as input predicted semantic masks from frames (f_{i_1}, f_{i_2}), and predicts the semantic mask of future frame f_{target}. In this setup, the discriminator has to distinguish between a mask predicted from f_{target} directly (*real*), and a mask predicted from past frames (f_{i_1}, f_{i_2}). Like in [11], our model combines a conditional approach with a similar structure as of DCGAN [36]. At test time, we derive a plausibility score by computing the conditioned discriminator's score for every conditioned frame. This is a novel approach based on the observation that the optimal discriminator D computes a score for x of

$$D(x) = \frac{P_{data}(x)}{P_G(x) + P_{data}(x)} \tag{3}$$

For non-physical events $\hat{x}$, $P_{data}(\hat{x}) = 0$; therefore, as long as $P_G(\hat{x}) > 0$, $D(\hat{x})$ should be 0 for non-physical events, and $D(x) > 0$ for physical events x. Note that this is a strong assumption, as there is no guarantee that the generator will ever have support at the part of the distribution corresponding to impossible videos.

All our models' architectures, as well as training procedures and samples of predicted semantic masks can be found in Supplementary Materials (Tables 3, 4, 5, 6 and Figure 6). The code is availabe on https://github.com/rronan/IntPhys-Baselines.

3.3 Video Plausibility Score

From forward models presented above, we can compute a plausibility score for every frame f_{target}, conditioned to previous frames (f_{i_1}, f_{i_2}). However, because the temporal positions of impossible

events are not given, we must decide of a score for a video, given the scores of all its conditioned frames. An impossible event can be characterized by the presence of one or more impossible frame(s), conditioned to previous frames. Hence, a natural approach to compute a video plausibility score is to take the minimum of all conditioned frames' scores:

$$\text{Plaus}(v) = \min_{(f_{i_1}, f_{i_2}, f_{target}) \in v} \text{Plaus}(f_{target} | f_{i_1}, f_{i_2}) \tag{4}$$

where v is the video, and $(f_{i_1}, f_{i_2}, f_{target})$ are all the frame triplets in v, as given in the training phase.

3.4 Results

Short-term prediction The first training procedure is a short-term prediction task; it takes as input frames f_{t-2}, f_t and predicts f_{t+5}, which we note $(f_{t-2}, f_t) \rightarrow f_{t+5}$ in the following. We train the two architectures presented above on short-term prediction task and evaluate them on the test set. For the relative classification task, CNN encoder-decoder has an error rate of 0.10 when impossible events are visible and 0.53 when they are occluded. The GAN has an error rate of 0.12 when visible and 0.48 when occluded. For the absolute classification task, CNN encoder-decoder has a L_A (see eq. 2) of 0.31 when impossible events are visible and 0.50 when they are occluded. The GAN has a L_A of 0.27 when visible and 0.51 when occluded. Results are detailed in Supplementary Materials (Tables 7, 8, 9, 10).
We observe that our short-term prediction models show good performances when the impossible events are visible, especially on the relative classifications task. However they perform poorly when the impossible events are occluded. This is easily explained by the fact that they have a prediction span of 5 frames, which is usually lower than the occlusion time. Hence, these models don't have enough "memory" to catch occluded impossible events.

Long-term prediction The second training procedure consists in a long-term prediction task: $(f_{t-5}, f_t) \rightarrow f_{t+35}$. For the relative classification task, CNN encoder-decoder has an error rate of 0.19 when impossible events are visible and 0.49 when they are occluded. The GAN has an error rate of 0.20 when visible and 0.43 when occluded. For the absolute classification task, CNN encoder-decoder has a L_A of 0.43 when impossible events are visible and 0.50 when they are occluded. The GAN has a L_A of 0.33 when visible and 0.50 when occluded. Results are detailed in Supplementary Materials (Tables 11, 12, 13, 14).
As expected, long-term models perform better than short-term models on occluded impossible events. Moreover, results on absolute classification task confirm that it is way more challenging than the relative classification task. Because some movies are more complex than others, the average score of each quadruplet of movies may vary a lot. It results in cases where one model returns a higher plausibility score to an impossible movie $M_{\{\text{imp, easy}\}}$ from an easy quadruplet than to a possible movie $M_{\{\text{pos, complex}\}}$ from a complex quadruplet.

Aggregated model To grasp short and long-term dependencies, we aggregate the scores of short-term and long-term models: $P_{agg}(v) = (P_{\text{short-term}}(v) + P_{\text{long-term}}(v))/2$. For the relative classification task, CNN encoder-decoder has an error rate of 0.14 when impossible events are visible and 0.51 when they are occluded. The GAN has an error rate of 0.12 when visible and 0.51 when occluded. For the absolute classification task, CNN encoder-decoder has a L_A of 0.35 when impossible events are visible and 50 when they are occluded. The GAN has a L_A of 0.27 when visible and 0.51 when occluded. Results are detailed in Supplementary Materials (Tables 15, 16, 17, 18, Figure 9).

3.5 Human Judgments

We presented the 3600 videos from the test set (Block O1) to human participants using Amazon Mechanical Turk. The experiment and human judgements results are detailed in Supplementary Materials, section 7. A mock example of the Amazon Mechanical Turk experiment is available on http://129.199.81.135/naive_physics_experiment/.

4 Discussion

We presented IntPhys, a benchmark for measuring intuitive physics in artificial vision systems inspired by research on conceptual development in infants. To pass the benchmark, a system is asked to return a plausibility score for each video clip. The system's performance is assessed by measuring its ability to discriminate possible from impossible videos illustrating several types of physical principles. Naive humans tested on the first block of the dataset show a generally good performance, despite the fact that they are given a more difficult binary choice (possible versus impossible), although some errors start when the number of objects is too large and when using one or two occlusion episodes, probably due to attentional overload. We presented two unsupervised learning models based on semantic masks, which learn from a training set only composed of physically plausible clips, and are tested on the same block as the humans.

The computational system generally performed poorly compared to humans but obtained above chance performance using a mask prediction task, with a very strong effect of the presence of occlusion. The relative success of the semantic mask prediction system compared to what we originally found with pixel-based systems indicates that operating at a more abstract level is a worthwhile pursuing strategy when it comes to modeling intuitive physics. Future work will explore alternative ways of constructing this abstract representation in particular instance masks and object detection bounding boxes. In addition, enriching the training through embedding the learner in an interactive version of the environment could add more information for the learning of the physics of macroscopic objects.

In brief, the systematic way of constructing the IntPhys Benchmark goes beyond a direct copy of developmental experiments, and open up the way to study the effect of attention and scene complexity through a direct comparison of humans and machines on the same tasks.

5 Related work

The modeling of intuitive physics has been addressed mostly through systems trained with some form of future prediction as a training objective. Some studies have investigated models for predicting the stability and forward modeling the dynamics of towers of blocks ([6, 25, 44, 26, 30, 27]). [6] proposes a model based on an intuitive physics engine, [25] and [26] follow a supervised approach using Convolutional Neural Networks (CNNs), [44] makes a comparison between simulation-based models and CNN-based models, [30] improves the predictions of a CNN model by providing it with a prediction of a generative model. In [28], authors propose different feature learning strategies (multi-scale architecture, adversarial training method, image gradient difference loss function) to predict future frames in raw videos.

Other models use more structured representation of objects to derive longer-term predictions. In [5] and [9], authors learn objects dynamics by modelling their pairwise interactions and predicting the resulting objects states representation (e.g. position / velocity / object intrinsic properties) . In [41], [15] and [12] authors combine factored latent object representations, object centric dynamic models and visual encoders. Each frame is parsed into a set of object state representations, which are used as input of a dynamic model. In [15] and [12], authors use a visual decoder to reconstruct the future frames, allowing the model to learn from raw (though synthetic) videos.

Regarding evaluation and benchmarks, apart from frame prediction datasets, which are not strictly speaking about intuitive physics, one can distinguish the Visual Newtonian Dynamics (VIND) dataset which includes more than 6000 videos with bounding boxes on key objects across frames, and annotated with a 3D plane which woud most closely fit the object trajectory [31]. There is also recent dataset proposed by a DeepMind team [33]. This last dataset seems very similar to ours. It is also inspired by the developmental literature and based on the violation of expectation principles and is structured around 3 blocks similar to our first 3 blocks (object permanence, shape constancy, continuity) and two other ones (solidity and containment). The number and characteristics of this dataset is not known at present. From the sample videos, two differences emerged: our dataset is better matched in terms of quadruplets of clips controlled at the level of the pixels, and our dataset has a factorial manipulation of scene and movement complexity. It would be interesting to explore the possibility to merge these two datasets, as well as add more blocks in order to increase the diversity and coverage of the physical phenomena.

References

[1] Baillargeon, R. and Carey, S. Core cognition and beyond. In Pauen, S. (ed.), *Early childhood development and later outcome*, chapter 3, pp. 33–65. Cambridge University Press, New York, 2012.

[2] Baillargeon, Renee and Hanko-Summers, Stephanie. Is the top object adequately supported by the bottom object? young infants' understanding of support relations. *Cognitive Development*, 5(1):29–53, 1990.

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

[4] Baillargeon, Renee, Needham, Amy, and DeVos, Julie. The development of young infants' intuitions about support. *Infant and Child Development*, 1(2):69–78, 1992.

[5] Battaglia, Peter, Pascanu, Razvan, Lai, Matthew, Jimenez Rezende, Danilo, and kavukcuoglu, koray. Interaction networks for learning about objects, relations and physics. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29*, pp. 4502–4510. Curran Associates, Inc., 2016. URL http://papers.nips.cc/paper/6418-interaction-networks-for-learning-about-objects-relations-and-physics.pdf.

[6] Battaglia, Peter W., Hamrick, Jessica B., and Tenenbaum, Joshua B. Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences of the United States of America*, 110(45), 2013. URL http://www.pnas.org/content/110/45/18327.

[7] Bertinetto, Luca, Valmadre, Jack, Henriques, João F, Vedaldi, Andrea, and Torr, Philip HS. Fully-convolutional siamese networks for object tracking. In *European Conference on Computer Vision*, pp. 850–865. Springer, 2016.

[8] Chang, Angel X, Funkhouser, Thomas, Guibas, Leonidas, Hanrahan, Pat, Huang, Qixing, Li, Zimo, Savarese, Silvio, Savva, Manolis, Song, Shuran, Su, Hao, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[9] Chang, Michael B, Ullman, Tomer, Torralba, Antonio, and Tenenbaum, Joshua B. A compositional object-based approach to learning physical dynamics. *arXiv preprint arXiv:1612.00341*, 2016.

[10] Choy, Christopher B, Xu, Danfei, Gwak, JunYoung, Chen, Kevin, and Savarese, Silvio. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. *arXiv preprint arXiv:1604.00449*, 2016.

[11] Denton, Emily L., Gross, Sam, and Fergus, Rob. Semi-supervised learning with context-conditional generative adversarial networks. *CoRR*, abs/1611.06430, 2016. URL http://arxiv.org/abs/1611.06430.

[12] Ehrhardt, Sebastien, Monszpart, Aron, Mitra, Niloy J, and Vedaldi, Andrea. Learning a physical long-term predictor. *arXiv preprint arXiv:1703.00247*, 2017.

[13] Farhadi, Ali, Hejrati, Mohsen, Sadeghi, Mohammad, Young, Peter, Rashtchian, Cyrus, Hockenmaier, Julia, and Forsyth, David. Every picture tells a story: Generating sentences from images. *Computer vision–ECCV 2010*, pp. 15–29, 2010.

[14] Finn, Chelsea, Goodfellow, Ian, and Levine, Sergey. Unsupervised learning for physical interaction through video prediction. In *Advances in Neural Information Processing Systems*, pp. 64–72, 2016.

[15] Fraccaro, Marco, Kamronn, Simon, Paquet, Ulrich, and Winther, Ole. A disentangled recognition and nonlinear dynamics model for unsupervised learning. *Advances in Neural Information Processing Systems 30, NIPS*, 2017.

[16] Fragkiadaki, Katerina, Agrawal, Pulkit, Levine, Sergey, and Malik, Jitendra. Learning visual predictive models of physics for playing billiards. *ICLR*, 2016. URL https://arxiv.org/abs/1511.07404.

[17] Goodfellow, Ian, Pouget-Abadie, Jean, Mirza, Mehdi, Xu, Bing, Warde-Farley, David, Ozair, Sherjil, Courville, Aaron, and Bengio, Yoshua. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[18] He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing, and Sun, Jian. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

[19] Johnson, Justin, Hariharan, Bharath, van der Maaten, Laurens, Fei-Fei, Li, Zitnick, C Lawrence, and Girshick, Ross. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. *arXiv preprint arXiv:1612.06890*, 2016.

[20] Kellman, Philip J and Spelke, Elizabeth S. Perception of partly occluded objects in infancy. *Cognitive psychology*, 15(4):483–524, 1983.

[21] Kingma, Diederik P. and Ba, Jimmy. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. URL http://arxiv.org/abs/1412.6980.

[22] Krishna, Ranjay, Zhu, Yuke, Groth, Oliver, Johnson, Justin, Hata, Kenji, Kravitz, Joshua, Chen, Stephanie, Kalantidis, Yannis, Li, Li-Jia, Shamma, David A, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.

[23] Kristan, Matej, Matas, Jiri, Leonardis, Aleš, Vojir, Tomas, Pflugfelder, Roman, Fernandez, Gustavo, Nebehay, Georg, Porikli, Fatih, and Čehovin, Luka. A novel performance evaluation methodology for single-target trackers, Jan 2016. URL http://arxiv.org/abs/1503.01313.

[24] Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 25*, pp. 1097–1105. Curran Associates, Inc., 2012. URL http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

[25] Lerer, Adam, Gross, Sam, and Fergus, Rob. Learning physical intuition of block towers by example. *International Conference on Machine Learning (ICML)*, 2016. URL http://arxiv.org/abs/1603.01312.

[26] Li, Wenbin, Leonardis, Ales, and Fritz, Mario. To fall or not to fall: A visual approach to physical stability prediction. *arXiv preprint*, 2016. URL https://arxiv.org/abs/1604.00066.

[27] Li, Wenbin, Leonardis, Aleš, and Fritz, Mario. Visual stability prediction and its application to manipulation. *arXiv preprint arXiv:1609.04861*, 2016.

[28] Mathieu, Michael, Couprie, Camille, and LeCun, Yann. Deep multi-scale video prediction beyond mean square error. *arXiv preprint arXiv:1511.05440*, 2015.

[29] Mirza, Mehdi and Osindero, Simon. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014. URL http://arxiv.org/abs/1411.1784.

[30] Mirza, Mehdi, Courville, Aaron, and Bengio, Yoshua. Generalizable features from unsupervised learning. *ICLR Workshop submission*, 2017. URL https://openreview.net/pdf?id=BynzZolYg.

[31] Mottaghi, Roozbeh, Rastegari, Mohammad, Gupta, Abhinav, and Farhadi, Ali. "what happens if..." learning to predict the effect of forces in images. *ECCV*, 2016. URL http://allenai.org/plato/forces/.

[32] Oh, Junhyuk, Guo, Xiaoxiao, Lee, Honglak, Lewis, Richard, and Singh, Satinder. Action-conditional video prediction using deep networks in atari games. *NIPS*, 2015. URL https://arxiv.org/abs/1507.08750.

[33] Piloto, Luis, Weinstein, Ari, TB, Dhruva, Ahuja, Arun, Mirza, Mehdi, Wayne, Greg, Amos, David, Hung, Chia-Chun, and Botvinick, Matt M. Probing physics knowledge using tools from developmental psychology. *CoRR*, abs/1804.01128, 2018. URL http://arxiv.org/abs/1804.01128.

[34] Pinheiro, Pedro O, Collobert, Ronan, and Dollar, Piotr. Learning to segment object candidates. In *Advances in Neural Information Processing Systems*, pp. 1990–1998, 2015.

[35] Pylyshyn, Zenon W and Storm, Ron W. Tracking multiple independent targets: Evidence for a parallel tracking mechanism. *Spatial vision*, 3(3):179–197, 1988.

[36] Radford, Alec, Metz, Luke, and Chintala, Soumith. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2015. URL http://arxiv.org/abs/1511.06434.

[37] Russakovsky, Olga, Deng, Jia, Su, Hao, Krause, Jonathan, Satheesh, Sanjeev, Ma, Sean, Huang, Zhiheng, Karpathy, Andrej, Khosla, Aditya, Bernstein, Michael, Berg, Alexander C., and Fei-Fei, Li. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. ISSN 1573-1405. doi: 10.1007/s11263-015-0816-y. URL http://dx.doi.org/10.1007/s11263-015-0816-y.

[38] Saxe, Rebecca and Carey, Susan. The perception of causality in infancy. *Acta psychologica*, 123(1):144–165, 2006.

[39] Spelke, Elizabeth S, Kestenbaum, Roberta, Simons, Daniel J, and Wein, Debra. Spatiotemporal continuity, smoothness of motion and object identity in infancy. *British Journal of Developmental Psychology*, 13(2): 113–142, 1995.

[40] Tapaswi, Makarand, Zhu, Yukun, Stiefelhagen, Rainer, Torralba, Antonio, Urtasun, Raquel, and Fidler, Sanja. Movieqa: Understanding stories in movies through question-answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4631–4640, 2016.

[41] Watters, N., Tacchetti, A., Weber, T., Pascanu, R., Battaglia, P., and Zoran, D. Visual Interaction Networks. *ArXiv e-prints*, June 2017.

[42] Wright, John, Yang, Allen Y, Ganesh, Arvind, Sastry, S Shankar, and Ma, Yi. Robust face recognition via sparse representation. *IEEE transactions on pattern analysis and machine intelligence*, 31(2):210–227, 2009.

[43] Xu, Fei and Carey, Susan. Infants' metaphysics: The case of numerical identity. *Cognitive psychology*, 30 (2):111–153, 1996.

[44] Zhang, Renqiao, Wu, Jiajun, Zhang, Chengkai, Freeman, William T., and Tenenbaum, Joshua B. A comparative evaluation of approximate probabilistic simulation and deep neural networks as accounts of human physical scene understanding. *CogSci*, 2016. URL http://blocks.csail.mit.edu/.

[45] Zitnick, C Lawrence and Parikh, Devi. Bringing semantics into focus using visual abstraction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3009–3016, 2013.

Supplementary Material

6 IntPhys Dataset

6.1 The training set

The training set has been constructed using Unreal Engine 4.0; it contains a large variety of objects interacting one with another, occluders, textures, etc. It is composed of 15K videos of possible events (around 7 seconds each at 15fps), totalling 21 hours of videos. Each video is delivered as stacks of raw image (288 x 288 pixels), totalling 157Gb of uncompressed data. We also release the source code for data generation, allowing users to generate a larger training set if desired.

Even though the spirit of IntPhys is the unsupervised learning of intuitive physics, we do provide additional information which may help the learner. The first one is the depth field for each image. This is not unreasonable, given that in infants, stereo vision and motion cues could provide an approximation of this information. We also deliver object instance segmentation masks. Given that this information it is probably not available as such in infants, we provide it only in the training set, not in the test set, for pretraining purposes.

6.2 The dev and test sets

This section describes the dev and test sets for block O1 (object permanence). The design of these dev and test sets follow the general structure of matched sets described in section 3.2. As for parametric complexity, we vary the number of objects (1, 2 or 3), the presence and absence of occluder(s) and the complexity of the movement (static, dynamic 1 and dynamic 2). In the static case, the objects do not move; in the dynamic 1 case, they bounce or roll from left to right or right to left. In both these types of events, one occluder may be present on the scene, and objects may sometimes pop into existence (a $0 \rightarrow 1$ event), or disappear suddenly ($1 \rightarrow 0$) - these impossible events occur behind the occluder when it is present, or in full view otherwise. In dynamic 2 events (illustrated in Figure 4), two occluders are present and the existence of objects may change twice. For example, one object may be present on the scene at first, then disappear after going behind the first occluder, later reappearing behind the second occluder ($1 \rightarrow 0 \rightarrow 1$). Dynamic 2 events were designed to prevent systems from detecting inconsistencies merely by comparing the number of objects visible at the beginning and at the end of the movie. Matched sets contain four videos: two possible events ($0 \rightarrow 0 \rightarrow 0$ and $1 \rightarrow 1 \rightarrow 1$) and two impossible events ($0 \rightarrow 1 \rightarrow 0$ and $1 \rightarrow 0 \rightarrow 1$).

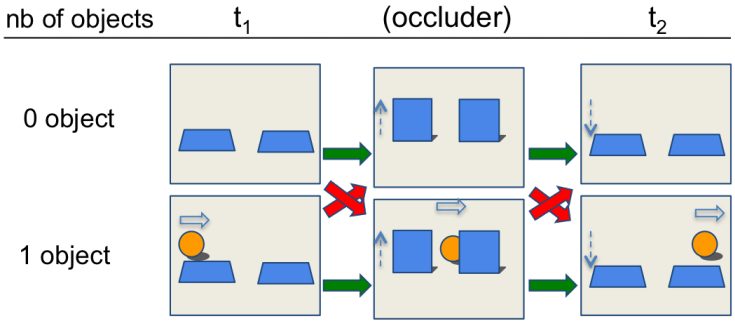

Figure 4: Illustration of the 'dynamic 2' condition. In the two possible movies (green arrows), the number of objects remains constant despite the occlusion. In the two impossible movies (red arrows), the number of objects changes temporarily (goes from 0 to 1 to 0 or from 1 to 0 to 1).

In total, the Block O1 test set contains 18 types of movies (3 objects, 2 occlusions and 3 types of movements). The dev set is instantiated by 20 different renderings of these 18 scenarios, objects positions, shapes, trajectories, resulting in 360 movies. The test set is instantiated by 200 different renderings of these scenarios (for a total of 3600 movies) and uses different objects, textures, motions, etc. All of the objects and textures of the dev and test sets are present in the training set.

The purpose of the dev set released in IntPhys V1.0 is to help in the selection of an appropriate plausibility score, and in the comparison of various architectures (hyper-parameters), but it should *not* serve to train the model's parameters (this should be done only with the training set). This is why the dev set is kept intentionally small. The test set has more statistical power and enables a fine grained evaluation of the results across the different movie subtypes. This benchmark along with video examples are available on the project page www.intphys.com.

6.3 Evaluation software

For each movie, the model should issue a scalar plausibility score. This number together with the movie ID is then fed to the evaluation software which outputs two tables of results, one for the absolute score and the other for the relative score.

The evaluation software is provided for the dev set, but not the test set. For evaluating on the test set, participants are invited to submit their system and results (see www.intphys.com) and their results will be registered and time-stamped on the website leaderboard.

7 Human Judgement - Experiment

We presented the 3600 videos from the test set (Block O1) to human participants using Amazon Mechanical Turk. Participants were first presented 8 examples of possible scenes from the training set, some simple, some more complex. They were told that some of the test movies were incorrect or corrupted, in that they showed events that could not possibly take place in the real world (without specifying how). Participants were each presented with 40 randomly selected videos, and labeled them as POSSIBLE or IMPOSSIBLE. They completed the task in about 7 minutes, and were paid $1. A response was counted as an error when a possible movie was classified as impossible or vice versa. A total of 346 persons participated, but for 99 of them the data were discarded because they failed to respond 100% correctly to the easiest condition, i.e., static, one object, visible. A mock sample of the AMT test is available on http://129.199.81.135/naive_physics_experiment/.

Table 2: Average error rate on plausibility judgments collected in humans using MTurk for the IntPhys(Block O1) test set. * this datapoint has been "forced" to be zero by our inclusion criterion.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	*Avg.*	1 obj.	2 obj.	3 obj.	*Avg.*
Static	0.00*	0.09	0.07	*0.06*	0.22	0.20	0.20	*0.21*
Dynamic (1 violation)	0.13	0.08	0.11	*0.11*	0.24	0.21	0.26	*0.24*
Dynamic (2 violations)	0.08	0.05	0.09	*0.07*	0.26	0.26	0.37	*0.29*
Avg.	*0.07*	*0.07*	*0.09*	*0.08*	*0.24*	*0.22*	*0.28*	*0.25*

The average error rates were computed across condition, number of objects and visibility for each remaining participant and are shown in Table 2. The overall error rate was rather low (16.5%), but, in general, observers missed violations more often when the scene was occluded. There was an increase in error going from static to dynamic 1 and from dynamic 1 to dynamic 2, but this pattern was only consistently observed in the occluded condition. For visible scenario, the dynamic 1 appeared more difficult than the dynamic 2. This was probably due to the fact that when objects are visible, the dynamic 2 impossible scenarios contain two local discontinuities and are therefore easier to spot than when one discontinuity only is present. When the discontinuities occurred behind the occluder, the pattern of difficulties was reversed, presumably because participants started using heuristics, such as checking that the number of objects at the beginning is the same as at the end, and therefore missed the intermediate disappearance of an object.

These results suggest that human participants are not responding according to the gold standard laws of physics due to limitations in attentional capacity - and this, even though the number of objects to track is below the theoretical limit of 4 objects. The performance of human observers can thus serve as a reference besides ground truth, especially for systems intended to model human perception.

8 Models and training procedure

8.1 Detailed models

See Tables 3, 4, 5, 6 for models' architectures, and Figure 6 for samples of predicted semantic masks. The code is available on https://github.com/rronan/IntPhys-Baselines.

Table 3: Mask predictor (9747011 parameters). BN stands for batch-normalization.

Input frame 3 x 64 x 64
7 first layers of resnet-18 (pretrained, frozen weights)
Reshape 1 x 8192
FC 8192 $\rightarrow$ 128
FC 128 $\rightarrow$ 8192
Reshape 128 x 8 x 8
UpSamplingNearest(2), 3 x 3 Conv. 128 - 1 str., BN, ReLU
UpSamplingNearest(2), 3 x 3 Conv. 64 - 1 str., BN, ReLU
UpSamplingNearest(2), 3 x 3 Conv. 3 - 1 str., BN, ReLU
3 sigmoid Target mask

Table 4: CNN for forward prediction (13941315 parameters). BN stands for batch-normalization.

Input frames 2 x 3 x 64 x 64
7 first layers of resnet-18 (pretrained, frozen weights) applied to each frame
Reshape 1 x 16384
FC 16384 $\rightarrow$ 512
FC 512 $\rightarrow$ 8192
Reshape 128 x 8 x 8
UpSamplingNearest(2), 3 x 3 Conv. 128 - 1 str., BN, ReLU
UpSamplingNearest(2), 3 x 3 Conv. 64 - 1 str., BN, ReLU
UpSamplingNearest(2), 3 x 3 Conv. 3 - 1 str., BN, ReLU
3 sigmoid Target mask

8.2 Training Procedure

We separate 10% of the training dataset to control the overfitting of our forward predictions. All our models are trained using Adam [21]. For the CNN encoder-decoder we use Adam's default parameters and stop the training after one epoch. For the GAN, we use the same parameters as in [36]: we set the generator's learning rate to $8e-4$ and discriminator's learning rate to $2e-4$. On the short-term prediction task, we train the GAN for 1 epoch; on the long-term prediction task we train it for 5 epochs. Learning rate decays are set to 0 and $beta1$ is set to 0.5 for both generator and discriminator.

9 Detailed baseline results

Table 5: Generator G (14729347 parameters). SFConv stands for spatial full convolution and BN stands for batch-normalization.

Input masks 2 x 3 x 64 x 64	Noise $\in \mathbf{R}^{100}$ $\sim$ Unif$(-1, 1)$
4 x 4 conv 64 - 2 str., BN, ReLU	
4 x 4 conv 128 - 2 str., BN, ReLU	
4 x 4 conv 256 - 2 str., BN, ReLU	
4 x 4 conv 512 - 2 str., BN, ReLU	
4 x 4 conv 512, BN, ReLU	
stack input and noise	
4 x 4 SFConv. 512 - 2 str., BN, ReLU	
4 x 4 SFConv. 256 - 2 str., BN, ReLU	
4 x 4 SFConv. 128 - 2 str., BN, ReLU	
4 x 4 SFConv. 64 - 2 str., BN, ReLU	
4 x 4 SFConv. 3 - 2 str., BN, ReLU	
3 sigmoid Target mask	

Table 6: Discriminator D (7629698 parameters). BN stands for batch-normalization.

history 2 x 3 x 64 x 64	input 3 x 64 x 64
Reshape 3 x 3 x 64 x 64	
4 x 4 convolution 512 - 2 strides, BN, LeakyReLU	
4 x 4 convolution 254 - 2 strides, BN, LeakyReLU	
4 x 4 convolution 128 - 2 strides, BN, LeakyReLU	
4 x 4 convolution 64 - 2 strides, BN, LeakyReLU	
4 x 4 convolution 5 - 2 strides, BN, LeakyReLU	
fully-connected layer	
1 sigmoid	

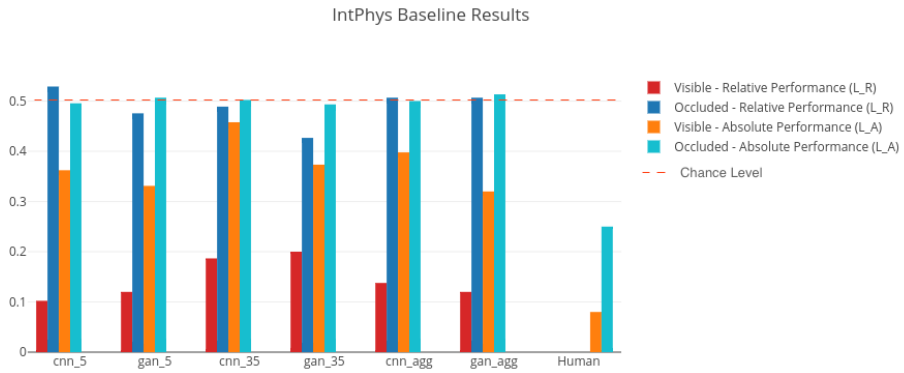

Figure 5: Results of our baselines in cases where the impossible event occurs in the open (*visible*) or behind an occluder (*occluded*). Y-axis represents the losses L_R (see Equation 1) for the relative performance and L_A (see Equation 2) for the absolute performance.

Table 7: Detailed relative classification scores for the CNN encoder-decoder with prediction span of 5.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	Total	1 obj.	2 obj.	3 obj.	Total
Static	0.04	0.04	0.00	0.03	0.44	0.48	0.72	0.55
Dynamic (1 violation)	0.00	0.20	0.44	0.21	0.56	0.48	0.44	0.49
Dynamic (2 violations)	0.00	0.04	0.16	0.07	0.60	0.48	0.56	0.55
Total	0.01	0.09	0.20	0.10	0.53	0.48	0.57	0.53

Table 8: Detailed absolute classification scores for the CNN encoder-decoder with prediction span of 5.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	Total	1 obj.	2 obj.	3 obj.	Total
Static	0.16	0.12	0.11	0.13	0.50	0.50	0.50	0.50
Dynamic (1 violation)	0.33	0.40	0.49	0.41	0.50	0.50	0.50	0.50
Dynamic (2 violations)	0.33	0.39	0.46	0.40	0.50	0.50	0.49	0.50
Total	0.27	0.30	0.36	0.31	0.50	0.50	0.50	0.50

Table 9: Detailed relative classification scores for the GAN with prediction span of 5.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	Total	1 obj.	2 obj.	3 obj.	Total
Static	0.00	0.08	0.00	0.03	0.44	0.60	0.40	0.48
Dynamic (1 violation)	0.00	0.16	0.36	0.17	0.40	0.52	0.52	0.48
Dynamic (2 violations)	0.00	0.20	0.28	0.16	0.44	0.44	0.52	0.47
Total	0.00	0.15	0.21	0.12	0.43	0.52	0.48	0.48

Table 10: Detailed absolute classification scores for the GAN with prediction span of 5.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	Total	1 obj.	2 obj.	3 obj.	Total
Static	0.05	0.13	0.09	0.09	0.50	0.52	0.52	0.52
Dynamic (1 violation)	0.30	0.42	0.45	0.39	0.50	0.52	0.47	0.50
Dynamic (2 violations)	0.22	0.39	0.42	0.34	0.50	0.50	0.51	0.50
Total	0.19	0.31	0.32	0.27	0.50	0.52	0.50	0.51

Table 11: Detailed relative classification scores for the CNN encoder-decoder with prediction span of 35.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	Total	1 obj.	2 obj.	3 obj.	Total
Static	0.04	0.16	0.00	0.07	0.60	0.48	0.60	0.56
Dynamic (1 violation)	0.00	0.36	0.28	0.21	0.44	0.40	0.52	0.45
Dynamic (2 violations)	0.00	0.40	0.44	0.28	0.40	0.56	0.40	0.45
Total	0.01	0.31	0.24	0.19	0.48	0.48	0.51	0.49

Table 12: Detailed absolute classification scores for the CNN encoder-decoder with prediction span of 35.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	Total	1 obj.	2 obj.	3 obj.	Total
Static	0.31	0.40	0.37	0.36	0.51	0.50	0.48	0.50
Dynamic (1 violation)	0.39	0.48	0.48	0.45	0.50	0.50	0.50	0.50
Dynamic (2 violations)	0.40	0.48	0.50	0.46	0.50	0.49	0.50	0.50
Total	0.37	0.46	0.45	0.43	0.50	0.50	0.50	0.50

Table 13: Detailed relative classification scores for the GAN with prediction span of 35.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	Total	1 obj.	2 obj.	3 obj.	Total
Static	0.00	0.12	0.00	0.04	0.48	0.48	0.60	0.52
Dynamic (1 violation)	0.00	0.36	0.48	0.28	0.36	0.36	0.44	0.39
Dynamic (2 violations)	0.00	0.44	0.40	0.28	0.24	0.52	0.36	0.37
Total	0.00	0.31	0.29	0.20	0.36	0.45	0.47	0.43

Table 14: Detailed absolute classification scores for the GAN with prediction span of 35.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	Total	1 obj.	2 obj.	3 obj.	Total
Static	0.07	0.23	0.18	0.16	0.50	0.50	0.52	0.51
Dynamic (1 violation)	0.31	0.43	0.45	0.40	0.47	0.49	0.50	0.49
Dynamic (2 violations)	0.37	0.48	0.47	0.44	0.49	0.49	0.49	0.49
Total	0.25	0.38	0.37	0.33	0.49	0.50	0.50	0.50

Table 15: Detailed relative classification scores for the aggregation of CNN models with prediction spans of 5 and 35.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	Total	1 obj.	2 obj.	3 obj.	Total
Static	0.00	0.04	0.00	0.01	0.64	0.48	0.60	0.57
Dynamic (1 violation)	0.00	0.20	0.44	0.21	0.52	0.52	0.36	0.47
Dynamic (2 violations)	0.00	0.32	0.24	0.19	0.48	0.44	0.52	0.48
Total	0.00	0.19	0.23	0.14	0.55	0.48	0.49	0.51

Table 16: Detailed absolute classification scores for the aggregation of CNN models with prediction spans of 5 and 35.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	Total	1 obj.	2 obj.	3 obj.	Total
Static	0.18	0.20	0.16	0.18	0.51	0.50	0.49	0.50
Dynamic (1 violation)	0.35	0.44	0.49	0.43	0.50	0.50	0.50	0.50
Dynamic (2 violations)	0.38	0.43	0.47	0.43	0.50	0.49	0.50	0.50
Total	0.30	0.36	0.37	0.35	0.50	0.50	0.49	0.50

Table 17: Detailed relative classification scores for the aggregation of GAN models with prediction spans of 5 and 35.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	Total	1 obj.	2 obj.	3 obj.	Total
Static	0.00	0.12	0.00	0.04	0.52	0.60	0.48	0.53
Dynamic (1 violation)	0.00	0.16	0.32	0.16	0.48	0.52	0.56	0.52
Dynamic (2 violations)	0.00	0.20	0.28	0.16	0.44	0.44	0.52	0.47
Total	0.00	0.16	0.20	0.12	0.48	0.52	0.52	0.51

Table 18: Detailed absolute classification scores for the aggregation of GAN models with prediction spans of 5 and 35.

	Visible				Occluded			
Type of scene	1 obj.	2 obj.	3 obj.	Total	1 obj.	2 obj.	3 obj.	Total
Static	0.01	0.17	0.07	0.08	0.51	0.52	0.52	0.52
Dynamic (1 violation)	0.30	0.42	0.44	0.39	0.50	0.52	0.48	0.50
Dynamic (2 violations)	0.22	0.39	0.42	0.34	0.50	0.51	0.51	0.50
Total	0.17	0.33	0.31	0.27	0.50	0.51	0.50	0.51

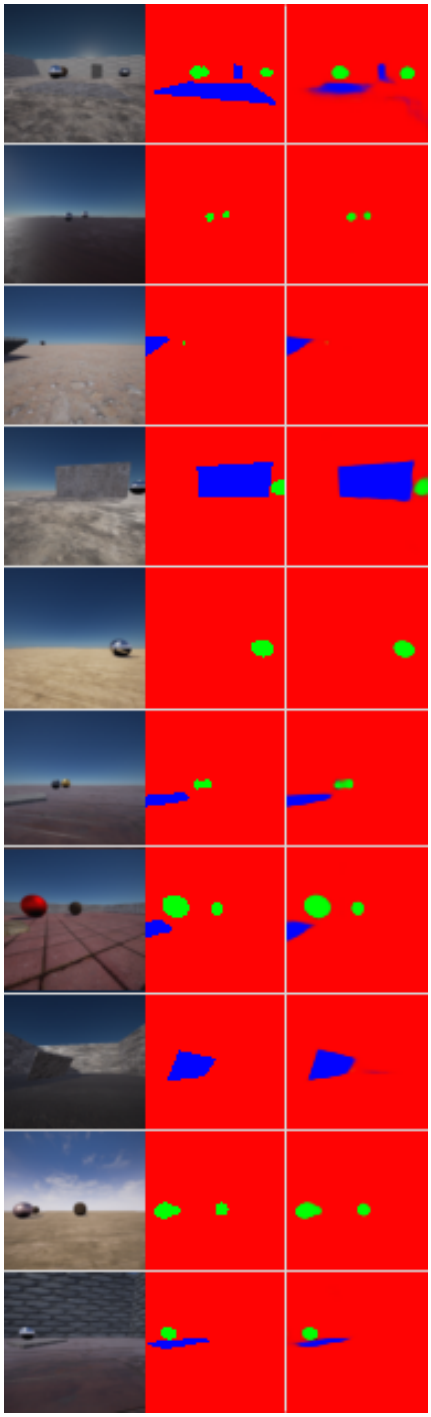

Figure 6: Output examples of our semantic mask predictor. From left to right: input image, ground truth semantic mask, predicted semantic mask.