2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Learning Vision-Based Physics Intuition Models for Non-Disruptive Object Extraction

Sarthak Ahuja[1], Henny Admoni[2] and Aaron Steinfeld[3]

***Abstract*— Robots operating in human environments must be careful, when executing their manipulation skills, not to disturb nearby objects. This requires robots to reason about the effect of their manipulation choices by accounting for the support relationships among objects in the scene. Humans do this in part by visually assessing their surroundings and using physics intuition for how likely it is that a particular object can be safely manipulated (i.e., cause no disruption in the rest of the scene). Existing work has shown that deep convolutional neural networks can learn intuitive physics over images generated in simulation and determine the stability of a scene in the real world. In this paper, we extend these physics intuition models to the task of assessing safe object extraction by conditioning the visual images on specific objects in the scene. Our results, in both simulation and real-world settings, show that with our proposed method, physics intuition models can be used to inform a robot of which objects can be safely extracted and from which direction to extract them.**

I. Introduction

Robots operating in human environments need to perform a variety of dexterous manipulation tasks, such as procuring utensils from a large pile, grabbing a bottle from a stacked fridge, and fetching a book from a loaded shelf. In such environments, there are multiple objects near the robot's target of manipulation, and the cost of a failed attempt at extraction can be very high. Whether an object can be extracted or not is often non-obvious due to the complex support relationships between objects (Fig 1). Therefore, the ability of a robot to assess a scene visually, reason about which parts of the scene it can safely manipulate, and use this assessment to optimize its interactions is an important part of its autonomy.

Existing work in cognitive science [1], [2] has shown that humans are capable of visually assessing physical scenes quickly and inferring abstract properties such as stability using an internal intuitive physics engine that performs noisy and probabilistic simulations of a scene. More recently researchers have used computational machine learning models such as deep neural networks to approximate physics simulators and reason about the stability of a scene directly from visual inputs [3]–[5]. These are what we refer to as *physics intuition models* throughout the paper. These data-driven approaches greatly alleviate the need for explicit object modeling by using the richness of simulated passive observations to approximate the dynamics of complex scenes. Additionally, they allow for quick and accurate inferences during test time which are necessary for real-world assessment. In this work we propose to extend these models to go beyond gaining an intuition about *"Is this scene stable?"* and develop a binary judgment about *"Will the scene remain stable upon removing this object (in this direction)?"*.

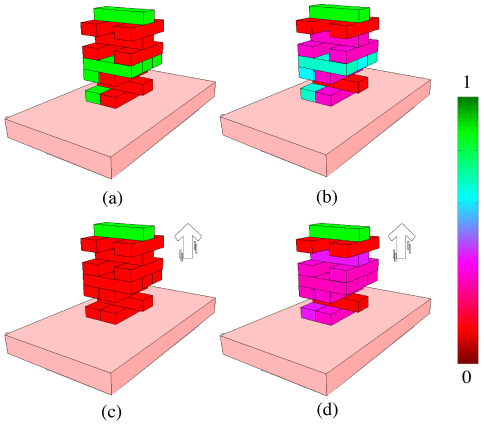

Fig. 1. (a) Ground truth values and (b) physics intuition model predictions for which blocks are safe to remove (green) and which are not (red), agnostic to extraction direction. (c) Ground truth and (d) model predictions for extraction in upward direction (depicted by white arrows).

While our system can predict the stability of a scene during extraction, it is not a complete solution. Any visual prediction will suffer uncertainty around unperceived features such as material properties and forces acting between objects (e.g., friction). To assess these, robots will need other interactions such as physical touch. Furthermore, they will need to account for their specific manipulation capabilities during these additional assessments. Despite that, visual assessment helps robots quickly filter out unsafe objects to extract and carefully pick only a few potentially safe candidates. This allows robots to make targeted choices for either additional non-visual assessments or direct manipulation. Hence, we posit that physics intuition models can make robot manipulation safe and efficient by making robot actions more targeted, yet safe at the same time.

Contributions: (1) We introduce a pipeline to implement physics intuition models for non-disruptive object extraction by adding a conditioning variable in the form of a mask on the target object during the training. (2) We demonstrate the effectiveness of our method in both simulation and real-world settings on a dataset of Jenga towers and Table Clutter configurations. (3) We analyze aggregation techniques to combine physics intuitions over multiple views to obtain a unified visual assessment.

[1]Carnegie Mellon University. sarthaka@cs.cmu.edu
[2]Carnegie Mellon University. henny@cmu.edu
[3]Carnegie Mellon University. steinfeld@cmu.edu

II. Related Work

A. Support-Order Prediction and Image Understanding

Traditionally, researchers have used explicit geometry and kinematics-based techniques to infer support-order among objects in the scene. [6] recreates real-world scenes in simulation by fitting primitive objects onto 3D point clouds computing support relationships on the obtained primitive shape arrangements. Along similar lines, there exist other explicit rule-based approaches for safe deconstruction of object piles [7], [8]. These methods require objects, their physical properties and range of support relationships to be known beforehand. This makes them difficult to scale across different domains without explicitly changing the underlying hand-crafted rules and carefully chosen thresholds.

Another line of research tries to find the support relationships among objects in a scene directly from images through supervised learning or non-monotonic reasoning over hand-crafted features of individual objects in the scene [9], [10]. [11]–[13] build physically-plausible scene representations by modeling the world as cuboids and reasoning about the support structure and occluded regions. As opposed to these approaches, we directly learn features from data without assuming any predetermined form.

B. Vision-Based Intuitive Physics

Early work by [5] uses a feed-forward visual model to predict the stability and falling trajectories for simple block towers from images. [3], [4] use a similar model but use it for guiding block stacking. [3] samples candidate positions on the surface of an object and guides the construction of the tower by picking the candidate that leads to the highest stability score over the "hallucinated" scene from their learned physics intuition model. [4] do something similar, but they hallucinate sample candidate positions on the images themselves instead of sampling candidate positions in simulation. This makes exploiting the physics intuition models in the real world much more viable. However, they perform the training as well as candidate sampling on binary-valued foreground masks of the scene which limits the generalizability of the method to complex real-world scenes.

Our work differs from [3], [4] in that we try to learn physics intuition models that capture a notion of safe object extraction instead of stacking. Similar to [4] we try to sample candidate objects to remove by directly hallucinating object extractions in the images. But instead of training them on foreground masks of a scene, which would be infeasible given the visual complexity of cluttered scenes, we propose to add a separate conditioning variable in the form of a single object mask alongside the RGB images during training. Similar to [3], we use multiple views of the scene to make predictions but perform a more comprehensive analysis of various methods to aggregate these predictions in a significantly more occlusion sensitive setting.

There exists a plethora of ongoing research that aims to accurately model the physics dynamics of a scene from framing it as a future object state [14], [15], image frame [16], [17], or object trajectory [18], [19] problem. In our work, we focus on static scene analysis using images with no access to previous frames, object supervision beyond object masks and require prediction of a high level property (stability) of the system rather than the exact state of objects in the system. [18] is related to our work in the sense that it uses object masks to make predictions about the trajectory of an object in the image space. Our work in comparison makes predictions about the stability of the remaining scene rather than predicting the trajectory of the object-of-interest although using a similar object conditioning method.

III. Methodology

A. Overview

Learning physics intuition models is a supervised learning task. By changing the traditional image class labels to stability labels, we can learn a physics intuition model over a large number of images of scenes. The stability labels (*stable* or *unstable*) for these images are obtained by running actual simulations of a scenario in a physics engine. The objective is to learn a mapping f that, given an image I of the initial configuration of a scene S (consisting of n objects defined as $\{s_1, s_2 \dots s_n\}$), can provide the stability prediction $P(S)$, which is a probability value between 0 (unstable) and 1 (stable).

$$f : I(S) \rightarrow P(S) \tag{1}$$

1) Target Object Conditioning: We note that in the above formulation, the model is unable to naturally provide inferences about individual objects in the scene, i.e., it is unable to answer "*What is the stability of the scene after removing object s_i?*". One way to answer this question in the current setting is to remove the corresponding object from the scene and get the inference from the same function mapping f.

$$f : I(S \setminus s_i) \rightarrow P(S \setminus s_i) \tag{2}$$

Computing $I(S \setminus s_i)$ at test-time is non-trivial as it requires a robot to hallucinate the removal of an object from an image. As a simple solution to resolve this issue and adapt these models for object extraction tasks, we propose to generate only stable scenes to train our physics intuition models and, in each scene, remove a block and obtain the stability label for the resulting configuration. The corresponding segmentation masks of object s_i in image I can be defined as $\phi(s_i)$. These masks can be added to the above mapping function, thereby conditioning the obtained probability value on the object (i) to which the mask corresponds.

$$f : I(S|\phi(s_i)) \rightarrow P(S \setminus s_i) \tag{3}$$

We can obtain the segmentation masks over target objects from a separate object segmentation method [20] at test time for real-world evaluation. During training, we obtain these masks directly from the simulator.

2) Aggregation Over Multiple Views: We may obtain different predictions for the same scenario from different camera angles because of occlusion from objects in the scene and the inability of a single 2D image to capture all of relevant 3D information in the scene. For camera angle k,

$$f : I_k(S|\phi_k(s_i)) \rightarrow P_k(S \setminus s_i) \quad (4)$$

Therefore, it is important to account for multiple views of a scenario and obtain an accurate assessment of the scene in order to generate a single prediction. A common choice for capturing this mapping f has been deep convolutional neural networks [3]–[5], which consist of a feature extractor module (multiple convolution layers, CNN) followed by a classifier module (multiple fully connected layers, FNN). We explore two ways of performing aggregation over K views of a scenario in the context of these deep convolutional neural networks.

- **Pre-training**: As first proposed in [21], we can modify our model architecture during training to compute the feature representations over all available views (regardless of their order) and use view pooling to get an aggregated representation of the scene. This representation can be passed onto the classifier module to make a single prediction. See Figure 2.

$$f : \{I_0(S|\phi_0(s_i))...I_K(S|\phi_k(s_i))\} \rightarrow P(S \setminus s_i) \quad (5)$$

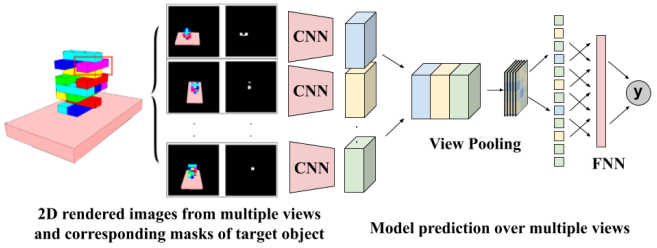

Fig. 2. Pre-training multi-view Aggregation: A single inference y is made on all views of the scenario. Target block is highlighted by the red box.

- **Post-training**: An alternative is to use a function mapping g that combines the predictions obtained over multiple views from the existing single view model, trained on all the views at once, using an aggregation method Ψ (in our case, we evaluate mean, median, mode, maximum and minimum). See Figure 3.

$$g : \Psi(\{P_0(S \setminus s_i)...P_k(S \setminus s_i)\}) \rightarrow P(S \setminus s_i) \quad (6)$$

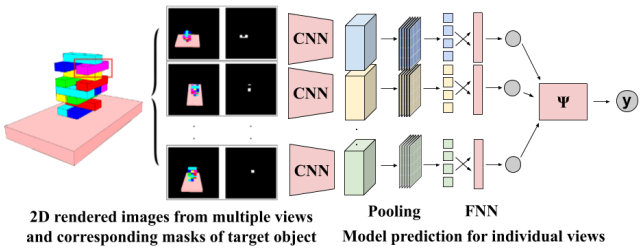

Fig. 3. Post-training multi-view Aggregation: Inferences for each view are made on the scenario and are aggregated to provide a single inference y.

3) Predicting the Extraction Direction: Currently our formulation only accounts for predicting whether a particular object can be safely removed from the scene and does not account for a robot's skill, i.e., our inference function f is **skill-agnostic**. In order to obtain a **skill-specific** model, we must account for the robot's skill during the generation of the stability labels. For object extraction, we parameterize a robot's skill as a set of discrete extraction directions (from the perspective of the robot). We define 5 discrete skills as *[Extract Up (UP), Extract Forward (FW), Extract Backward (BK), Extract Left (LF), Extract Right (RT)]*. For learning the skill-specific models, we reformulate our problem from being a logistic regression problem with a single label to one with 5 labels.

$$f : \{I_0(S|\phi_0(s_i))...I_K(S|\phi_k(s_i))\} \rightarrow \begin{bmatrix} P^{UP}(S \setminus s_i) \\ P^{FW}(S \setminus s_i) \\ P^{BK}(S \setminus s_i) \\ P^{LF}(S \setminus s_i) \\ P^{RT}(S \setminus s_i) \end{bmatrix} \quad (7)$$

For each extraction direction, we move the target object (canceling out all forces) by 0.2 m assuming a maximum acceleration of 0.1 m/sec^2. We determine these values as being reasonable for a robot to complete a clean and careful extraction after securing an object.

B. Dataset Generation

Since collecting data in the real-world is an expensive process, a common approach is to use domain knowledge and synthesize data in simulation. Publicly available image datasets of stable and unstable scenes [3], [5] generated in simulation are limited to block towers as they primarily focus on stacking tasks and do not feature some of the characteristic features of a cluttered scene, such as objects being supported by multiple objects, objects supporting each other along the plane, etc. Therefore, to evaluate our approach in a more principled way, we propose a taxonomy of cluttered scenes and choose a scene type from each proposed category to generate data and report results.

1) Taxonomy: We propose a categorization of cluttered scenes based on two factors that we believe affect the learning capacity of physics intuition models:

- **Homogeneous Structure**: An inherent homogeneous structure aids the learning process while diverse inter-object support-relationships in the scene make the learning process difficult. This can be understood by observing that in the presence of inherent pattern, information learned from one part of the scene can be extended to parts that exhibit a similar pattern.
- **Tractability**: Learning over scenarios with a large number of individual objects requires the physics intuition model to capture a larger uncertainty around the object interactions as well as handle the increased occlusions. This has been shown to be true in the case of block stacking tasks [4], and we expect this relationship to extend to object extraction tasks as well.

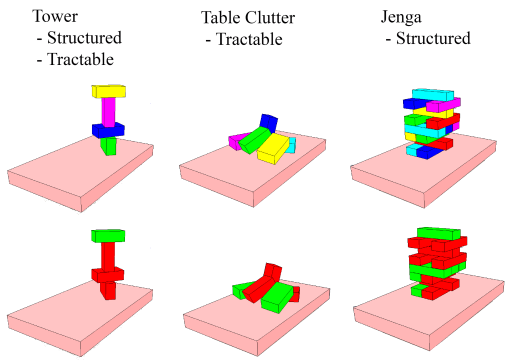

Fig. 4. Dataset visualization For skill-agnostic models. Top row depicts sample scenes and the bottom row visualizes the ground truth - **green** indicates blocks that can be removed and **red** indicates otherwise.

TABLE I
DATASET STATISTICS FOR SKILL-AGNOSTIC MODELS

	Total	Stable Scenarios	Unstable Scenarios
Clutter	6873	5483	1390
Jenga	13053	5473	7580

Figure 4 illustrates the resulting categories and the specific representative scene types. Table I summarizes the dataset statistics. We do not consider the category where a cluttered scene can be both intractable and lacking homogeneous structure as it becomes increasingly impractical to care about disruption of individual objects. Here one must reason about the effect of manipulating a collection of objects instead and we leave this for future work. In the following subsection, we go into more detail about the selected scene types and their corresponding data generation methods.

2) Synthetic Data Generation: To generate data corresponding to each scene type, we use the V-REP simulation platform [22]. In order to aid the network in identifying individual objects, similar to previous work [3], [5], we generate randomly colored objects. The dimensions of the base plane upon which a scenario is generated is constant across our data generation pipeline. We generate scenes using rigid homogeneous cuboids across all of the scene types to aid the generation of stable structures. Evaluating the method on more complex shapes will be addressed in future work. The specific details about the methods used to generate data for each scene type are below.

- **Tower**: For each sample, we simulate a tower of 1 to 4 objects on a uniformly sampled base location on the table. We simulate 500 towers for each height. While the size of each block is kept constant, we uniformly sample the orientation of the sides on which each block will rest and the orientation around the table normal. This scene type is an example of a configuration which displays both tractability and inherent structure and serves as a sanity check for our approach (we expect only the top block to be predicted as extractable).
- **Table Clutter**: For each sample, we simulate a tabletop cluttered scene with 2 to 5 objects (500 arrangements for each). The size of the additional objects is kept constant and their positions are sampled uniformly across the table. Blocks are generated at a small distance above the table with a normally sampled orientation. They are allowed to fall freely in the simulator and a sample is saved from all of the blocks that remain at rest on the table.
- **Jenga Tower**: For each sample, we generate a stable Jenga tower at the center of the table with the tower height ranging between 5 and 8 (250 towers each). We ensure that each row is supported by either ≥ 2 blocks from below or by one block placed along the center. The size of the blocks is based on the standard Jenga block dimensions and is constant throughout the scene.

3) Obtaining Ground Truth Labels: From each scene, we can obtain multiple scenarios based on the number of objects in the scene (where a *scenario* comprises of the sampled scene and one of the target objects). To obtain the ground-truth label for each scenario, we run the target object extraction in our simulator. We enable surface friction and gravity during the simulations. For the skill-agnostic case, we simply delete the target object from the scene. For the skill-specific case, we remove the target object using the 5 discrete skills described in the previous section (Section III-A.3). We then step through the simulation for a fixed number of steps and record the position and velocities for the remaining objects. This change is used to label a scenario as stable or unstable according to empirically determined threshold. To account for the class imbalance that results from the stochastic data generation pipeline (Table I), we perform up-sampling and ensure a equal stable-unstable ratio in our test sample scenarios. We use 1096 and 1516 test samples for Clutter and Jenga scenarios respectively.

Images and object masks of individual blocks from the scene are captured from 8 uniformly spaced camera angles. The images are rendered in color at a resolution of 224x224. To avoid visual ambiguity in the extraction direction, we choose camera angles that uniquely identify the table in each view. An alternate approach would have been to choose camera views from only one side of the table. In the next subsection we describe our training methodology in detail.

C. Training

We chose AlexNet's [23] feature extractor across our experiments as it consistently gave a reasonable performance on our dataset while requiring a limited number of parameters. Our classifier consists of a single 256 hidden unit fully connected layer. We use the Multi-View Convolutional Neural Networks (MVCNN) architecture proposed in [21] to extend the AlexNet architecture to multi-view inputs of the scene. Similar to [3], we optimize the standard logistic regression loss function for the binary classification task for the skill-agnostic models:

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n}\left[y_i \log\left(p_i\right) + \left(1-y_i\right)\log\left(1-p_i\right)\right] \quad (8)$$

Here n is the batch size, p_i is the output of the logit function at the end of the network for input image x_i, and y_i is the ground truth label. For the skill-specific models, we train only on a subset of scenarios that are classified as stable by the skill-agnostic model. This is because the scenarios that are classified as unstable by the skill-agnostic model will always be labeled as unstable by the skill-specific model. The loss function used for the skill-specific models is the same as above, with the only difference being that it comprises the multi-dimensional loss. We evaluate our model on a validation set during training over 80 epochs with a batch size of 32 and keep the model with the best accuracy (skill-agnostic) or lowest Hamming loss (skill-specific). We use the Adam optimizer [24] to train our models with a learning rate of 0.0001.

For both skill-specific and skill-agnostic cases we train two models: one trained over all single-view images of the scene and the other trained over multiple views of the scene using MVCNN. We evaluate both models on our test set for each scene type. For skill-specific models we use a single validation dataset combining the scenarios marked as positive in the skill-agnostic setting across the the validation and test set. We also use multiple aggregation techniques on predictions obtained over multiple camera views of the scene. For the purposes of this paper, a model that uses only a single view of the scene to make a prediction is called **single-view** and a model that uses multiple views is called **multi-view**.

IV. Evaluation

We evaluate our claim that our proposed approach can achieve a high performance over our test set by comparing it to chance (label every block as extractable). We also evaluate our claim that aggregating inferences over multiple views of a scene improves performance as compared to a single view by comparing the multi-view models to the best performance obtained from any single-view model on our test set. For multi-view models, we performed both pre-training and post-training aggregation. For pre-training aggregation, we use the MVCNN architecture proposed in [21]. For post-training, we evaluated five aggregation techniques: mean, median, mode, max, and min. We consistently saw better performance with mean aggregation and, in the interest of space, we only report on this post-training aggregation method.

A. Metrics

Since we are dealing with a binary prediction problem, we use the following two metrics to evaluate our approach on different scene types and across different methods:

- **Balanced Accuracy (ACC)**: This metric measures the average recall obtained on each class.
- **Macro Precision (PRE)**: This metric measures the average precision obtained on each class.

Even though we balance our test dataset initially, the weighted versions of accuracy and precision are used to account for the class imbalance that emerges when curating data for the skill-specific models.

TABLE II
Simulation Test Set Evaluation for Skill-Agnostic Models (Percentage Accuracy and Precision)

Type	Multi-View (MVCNN)		Multi-View (Mean)		Single-View		Chance
	ACC	**PRE**	**ACC**	**PRE**	**ACC**	**PRE**	**ACC/PRE**
Clutter	**70.89**	75.22	70.26	**78.60**	68.61	72.51	50.00
Jenga	**94.79**	**94.79**	91.16	91.83	86.21	86.52	50.00

B. Simulation Results

For both the skill-agnostic and skill-specific models, we observe that all of the models perform perfectly on the Tower scene type. This is expected because of the low complexity of the scene type and served as a sanity test for our method. For the two other scene types, our models demonstrate significantly higher accuracy and precision compared to chance in the skill-agnostic (Table II) and skill-specific (Table III) settings and across both multi-view and single-view models.

Multi-view methods perform better on both metrics compared to single-view methods for the skill-agnostic case but do not yield much advantage for the skill-specific case. We believe this is because a single view of the scene conveys latent information about the application of particular skills and bolsters the prediction capability of single-view models. For example, classification on whether an occluded object can be extracted from the left often did not require multiple views as this judgment can be made based on whether there are objects to its left which are not occluded.

In the skill-agnostic setup the advantage gained from using multiple views is lower for the Clutter scenarios as compared to Jenga. This may indicate that for a limited number of objects, one can identify a particular camera angle that can make accurate predictions even when the scene lacks any homogeneous structure. But using multiple-views is definitely advantageous in scenarios with a large number of objects and an inherent homogeneous structure.

For the Clutter scene type, extraction in the upward direction yields particularly low performance across the models. This is largely due to the resulting label imbalance after pruning using the skill-agnostic model. After pruning, there remain negligible scenarios where a block cannot be removed in the upward direction and the model is unable to capture this during the training phase.

Finally, we do not observe a clear winner among different aggregation methods for multiple-views. For Jenga towers, a correct prediction for a particular arrangement can often be reached only from a few corresponding selected angles (while many angles remain ambiguous), explaining why MVCNN proves to be a marginally better choice of skill-agnostic model. On the contrary, for Clutter scenes, a correct prediction for a particular arrangement can be reached from multiple camera angles (while a few remain ambiguous), and taking the mean prediction proves to be sufficient. Selected predictions across different scenes for both skill-specific and skill-agnostic models are shown in Figures 5 and 6.

TABLE III

SIMULATION VALIDATION SET EVALUATION FOR SKILL-SPECIFIC MODELS (PERCENTAGE ACCURACY AND PRECISION)

Type	Model	Up		Forward		Backward		Left		Right		Average	
		ACC	PRE	ACC	PRE	ACC	PRE	ACC	PRE	ACC	PRE	ACC	PRE
Clutter	Multi-View (MVCNN)	**50.00**	**46.68**	**91.18**	**90.42**	**90.86**	**91.60**	**90.77**	80.85	**92.52**	**91.90**	**83.26**	**82.29**
	Multi-View (Mean)	**50.00**	**46.68**	88.04	88.04	90.80	90.80	87.91	87.95	90.51	91.03	81.45	80.90
	Single-View	**50.00**	**46.68**	86.33	86.33	89.66	89.51	88.26	**88.44**	88.36	89.61	80.52	80.11
	Chance	**50.00**	**46.68**	50.00	20.75	50.00	22.82	50.00	23.65	50.00	19.29	50.00	26.64
Jenga	Multi-View (MVCNN)	99.53	99.74	85.33	85.87	86.17	88.30	**89.32**	90.15	86.27	88.16	89.33	90.44
	Multi-View (Mean)	**100.00**	**100.00**	86.83	**88.65**	86.21	**90.04**	87.75	**91.00**	88.36	**91.27**	89.83	**92.19**
	Single-View	**100.00**	**100.00**	**87.38**	87.95	**87.75**	89.59	88.35	89.04	**89.68**	90.25	**90.63**	91.37
	Chance	50.00	17.77	50.00	20.93	50.00	19.10	50.00	20.93	50.00	20.27	50.00	19.80

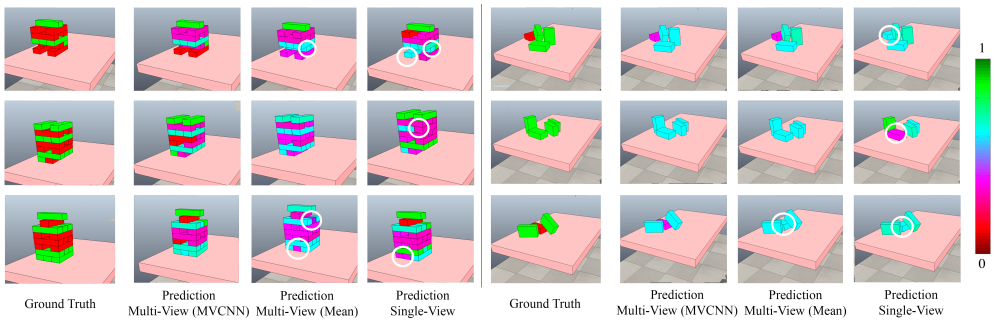

Fig. 5. Visualizing predictions made by multi-view skill-agnostic models in simulation. Single-view models often misclassify objects (white circles) that are partially occluded or where the object's pose is not clearly identifiable. For certain blocks, merely taking the mean of the prediction across multiple views may not be sufficient, as there may only be a few camera angles that are able to make accurate predictions.

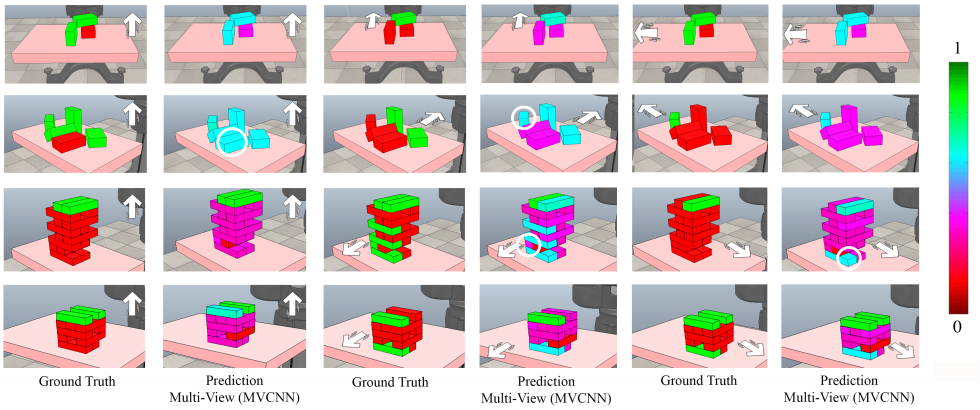

Fig. 6. Visualizing predictions made by multi-view skill-specific models in simulation. Which objects can be safely removed depends on which skill that we try to remove them with (white arrow). We observe that skill-specific models make few misclassifications (white circle) and are able to accurately predict this skill-specific notion of object-extraction. Note that classifications are made on blocks that are marked extractable by the skill-agnostic models.

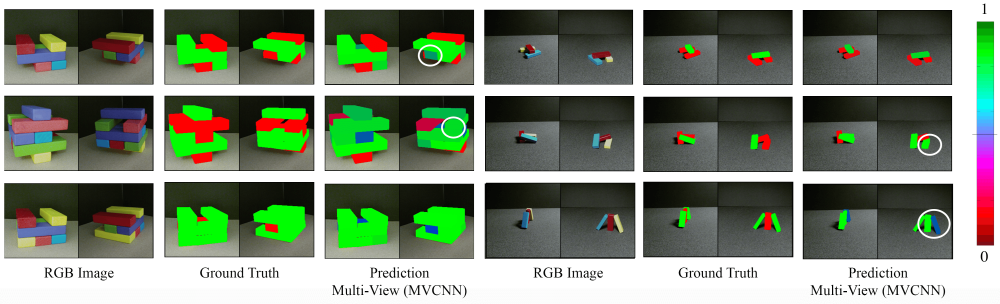

Fig. 7. Visualizing predictions made by skill-agnostic models on real image inputs. For each scene, images taken from two different camera views are passed to our trained physics intuition model and the probabilistic predictions are recorded. The misclassifications are highlighted with a white circle.

TABLE IV
REAL WORLD EVALUATION FOR SKILL-AGNOSTIC MODELS
(PERCENTAGE ACCURACY AND PRECISION)

Type	Multi-View (MVCNN)		Multi-View (Mean)	
	ACC	PRE	ACC	PRE
Clutter	**67.86**	68.72	57.14	**76.92**
Jenga	**74.07**	**78.27**	51.65	52.85

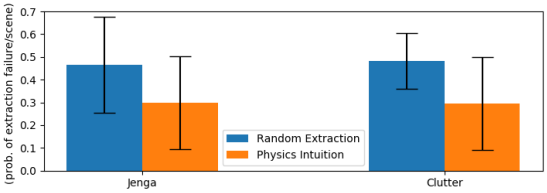

Fig. 8. Probability of extraction failure, with and without physics intuition.

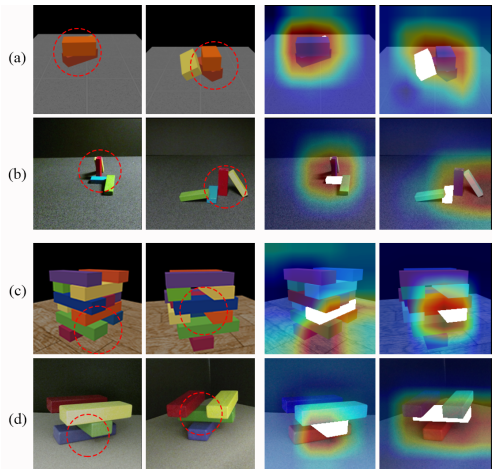

Fig. 9. Class activation maps for Clutter (a,b) and Jenga (c,d) scenes. Target object is highlighted in white. Red circles in the original images (left) highlight the region corresponding to the activation maps (right).

C. Real World Evaluation

To test transfer of our model to the real world, we conduct a preliminary evaluation of our multi-view skill-agnostic models on a small dataset of real image inputs. Our dataset consists of two images taken from different angles for 9 configurations of Jenga and Clutter each. We render a small dataset, similar to as explained in section III-B, in MuJoCo [25] to obtain more realistic images and train our model in the same manner as explained in section III-C. We extensively perform domain randomization [26] by varying the block color and table textures during training to make the sim-to-real transfer tractable.

Despite the drop in performance due to the implicit sim-to-real gap, we see that our models show promising results on real-world images (Table IV and Figure 7). As mean aggregation suffers due to the limited number of views in our experiments, we use the MVCNN model for our further analysis. We observe that our models are sensitive to lighting conditions and sometimes mistake internal shadows for blocks, leading to misclassifications in the Jenga scenes (middle row left, 7). Common failure cases for the Clutter scenes involve arrangements of blocks that are not abundantly present in our training dataset, such as blocks leaning up against each other (bottom row right, 7).

We further evaluate the average per-scene probability of failure (selecting an unsafe block) when selecting a block to extract using the MVCNN physics intuition model. We compare it against the average probability of failure when extracting any random block from a scene (chance) and summarize the results in Fig. 8. Even with an imperfect sim-to-real transfer, we observe that visual assessment can reduce the chance of failure in both Jenga and Clutter scenes.

D. Analysis

In this section, we analyze the models trained in simulation to gain further insight into their internal properties and understand their limitations to inform future work.

1) Visualization: To inspect which properties of the scene our models focus on when making predictions, we visualize the learned discriminative image regions from the CNN layer of the models. As proposed in [27], introducing a Global Average Pooling layer between the last convolution layer and the final fully connected layer of our model allows us to back-project the weights from the fully-connected layer and obtain Class Activation Maps (CAMs).

We inspect these discriminative regions in our test set for the skill-agnostic models. Figure 9 displays visualizations across two views of the Jenga and Clutter scenes both in the simulated and real image settings. The regions that contribute to the models' predictions are proximal to the target object. We can also see that for Jenga scenes that have a clear view of the objects in the target object row, the network focuses on that row, while in other scenes, it focuses on the blocks below it. This may be because having an alternate supporting block in the row is critical for the stability of the tower as is having supporting blocks underneath to counter the weight. Since it is difficult for a network to infer both these properties from a single view alone, multi-view models may have an edge over single view models. In the table Clutter scenario, we see that the network focuses at the contact region of the target object and all the object that it is supporting or being supported by. This may be because having contact with another block and the arrangement of nearby non-target objects are important factors that determine a scene's stability.

2) Generalizability: We evaluate the performance of skill-agnostic multi-view (MVCNN) models when trained on scenes consisting of different numbers of blocks than the testing scenes. Furthermore, we evaluate the sensitivity of our model by introducing small random noise in block sizes in the Clutter scenes and the absolute position of blocks in the Jenga scenes. Results from this experiment are summarized in Table V. We observe that models trained on a larger number of blocks extend well to scenes with fewer blocks but not vice versa. Additionally, training on noisy scenes captures a better generalizable physics intuition that extends well to samples with a more homogeneous structure.

TABLE V

GENERALIZABILITY BETWEEN SCENES (ACCURACY AND PRECISION)

Scene	Training	Testing	ACC	PRE
	Num. Blocks	*Num. Blocks*		
Clutter	2, 3	4	83.03	77.14
Clutter	2, 4	3	86.82	77.11
Clutter	3, 4	2	**91.80**	**79.09**
	Tower Height	*Tower Height*		
Jenga	5, 6	7	85.19	80.05
Jenga	5, 7	6	86.81	**94.07**
Jenga	6, 7	5	**91.40**	93.06
	Noise	*Noise*		
Jenga	Yes	No	**98.82**	**98.82**
Jenga	No	Yes	92.10	93.18
Clutter	Yes	No	**89.35**	**89.59**
Clutter	No	Yes	77.29	82.83

V. CONCLUSION

Existing research has shown that robots can use vision-based physics intuition models to predict a scene's stability directly from images and exploit this reasoning to create stable stacks of objects. Here, we demonstrated how robots could use similar visual assessment to perform the inverse process of predicting which objects can be extracted safely from a configuration and in which direction, hence, effectively reducing the probability of a robot disrupting the scene.

We extended existing physics intuition training methods by conditioning the images on specific objects using an object mask alongside the image of the scene. We showed that aggregating multiple views can increase the model's performance for assessing both, structured and unstructured object arrangements. Future work will explore how a robot can actively select multiple views by explicitly accounting for the prediction uncertainty in the available views.

In analyzing the discriminative image regions found by the model, we observed that discriminative regions correlated with regions that were critical to the stability of the scene, such as objects being directly supported by the target object or regions where alternate support must be present in order to avoid disruption. This analysis suggests that the system has learned meaningful intuitive physics features of the scenes.

ACKNOWLEDGMENT

This work has been supported in part by the Office of Naval Research (ONR N00014-18-1-2503). We thank Allan Wang, Elizabeth Carter, Michael Lee and Samantha Reig for their helpful comments on the manuscript.

REFERENCES

[1] J. R. Kubricht, K. J. Holyoak, and H. Lu, "Intuitive physics: Current research and controversies," *Trends in cognitive sciences*, vol. 21, no. 10, pp. 749–759, 2017.

[2] P. W. Battaglia, J. B. Hamrick, and J. B. Tenenbaum, "Simulation as an engine of physical scene understanding," *Proceedings of the National Academy of Sciences*, vol. 110, no. 45, pp. 18 327–18 332, 2013.

[3] O. Groth, F. B. Fuchs, I. Posner, and A. Vedaldi, "Shapestacks: Learning vision-based physical intuition for generalised object stacking," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 702–717.

[4] W. Li, A. Leonardis, J. Bohg, and M. Fritz, "Learning manipulation under physics constraints with visual perception," *arXiv preprint arXiv:1904.09860*, 2019.

[5] A. Lerer, S. Gross, and R. Fergus, "Learning physical intuition of block towers by example," *arXiv preprint arXiv:1603.01312*, 2016.

[6] R. Kartmann, F. Paus, M. Grotz, and T. Asfour, "Extraction of physically plausible support relations to predict and validate manipulation action effects," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3991–3998, 2018.

[7] O. Ornan and A. Degani, "Toward autonomous disassembling of randomly piled objects with minimal perturbation," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 4983–4989.

[8] S. Kimura, T. Watanabe, and Y. Aiyama, "Force based manipulation of jenga blocks," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 4287–4292.

[9] H. Riley and M. Sridharan, "Non-monotonic logical reasoning and deep learning for explainable visual question answering," in *Proceedings of the 6th International Conference on Human-Agent Interaction*, 2018, pp. 11–19.

[10] S. Panda, A. A. Hafez, and C. Jawahar, "Learning support order for manipulation in clutter," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 809–815.

[11] A. Gupta, A. A. Efros, and M. Hebert, "Blocks world revisited: Image understanding using qualitative geometry and mechanics," in *European Conference on Computer Vision*. Springer, 2010, pp. 482–496.

[12] T. Shao, A. Monszpart, Y. Zheng, B. Koo, W. Xu, K. Zhou, and N. J. Mitra, "Imagining the unseen: Stability-based cuboid arrangements for scene understanding," *ACM Trans. on Graphics*, vol. 33, no. 6, 2014.

[13] Z. Jia, A. C. Gallagher, A. Saxena, and T. Chen, "3d reasoning from blocks to stability," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 5, pp. 905–918, 2014.

[14] K. Fragkiadaki, P. Agrawal, S. Levine, and J. Malik, "Learning visual predictive models of physics for playing billiards," *arXiv preprint arXiv:1511.07404*, 2015.

[15] M. B. Chang, T. Ullman, A. Torralba, and J. B. Tenenbaum, "A compositional object-based approach to learning physical dynamics," *arXiv preprint arXiv:1612.00341*, 2016.

[16] Y. Ye, M. Singh, A. Gupta, and S. Tulsiani, "Compositional video prediction," 2019.

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

[18] R. Mottaghi, M. Rastegari, A. Gupta, and A. Farhadi, ""what happens if..." learning to predict the effect of forces in images," in *European conference on computer vision*. Springer, 2016, pp. 269–285.

[19] K. M. Kitani, D.-A. Huang, and W.-C. Ma, "Activity forecasting: An invitation to predictive perception," in *Group and Crowd Behavior for Computer Vision*. Elsevier, 2017, pp. 273–294.

[20] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[21] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3d shape recognition," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 945–953.

[22] M. F. E. Rohmer, S. P. N. Singh, "V-rep: a versatile and scalable robot simulation framework," in *Proc. of The International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[24] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[25] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 5026–5033.

[26] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ Int. Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 23–30.

[27] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2921–2929.