Appendix – Physion++: Evaluating Physical Scene Understanding that Requires Online Inference of Different Physical Properties

Hsiao-Yu Tung[1,2*] **Mingyu Ding**[1,3*] **Zhenfang Chen**[4] **Daniel M. Bear**[2] **Chuang Gan**[4,5]
Joshua B. Tenenbaum[1] **Daniel L. K. Yamins**[2] **Judith Fan**[2] **Kevin A. Smith**[1]
[1]MIT [2]Stanford [3]UC Berkeley [4]MIT-IBM Watson AI Lab [5]UMass Amherst
{sfish0101, kevin.smith3}@gmail.com myding@berkeley.edu jefan@stanford.edu

A Project Page and Dataset Release

We have released the Physion++ dataset at our project page. All three subsets (training set [2], readout fitting set [3], and testing set [4]) and the human data [5] are publicly available. We also post a list of all test filenames in the test set (192 entries per physical property). The dataset is organized as follows.

```
# Download and unzip the three data files
-- Training_data
   ...
-- Readout_data
   ...
-- Testing_data
    -- Folder organized by physical properties  # folders named with copy0 and copy1
    indicate two matched trials (e.g., the same video location from copy0 will have a
    different property/outcome in copy1 but the same initial conditions)
        -- Subfolder organized by scenarios and objects  # e.g. bouncy_platform-
    use_blocker_with_hole=0-target=cube
            -- [id].json  # object id and instance information over time
            -- [id].pkl  # meta information of the physical event and the video
            -- [id]_image.mp4  # raw input, RGB video
            -- [id]_seg.mp4  # segmentation masks of the video
            -- [id]_map.png  # indicating the yellow and red objects
    # Below are lists of test filenames and ids (192 entries per physical property)
    -- Mass.csv
    -- Friction.csv
    -- Elasticity.csv
    -- Deformability.csv
```

Listing 1: Organization of the dataset.

The .json file contains the object id and segmentation mask of each instance over time. The .pkl file contains the meta information of the scene, including 1) the dynamic friction, static friction, initial position, initial rotation, mass, elasticity, color, and mesh of each object; 2) the camera matrix and projection matrix for each frame; 3) angular velocities, positions, rotations, and velocities of all objects over time; 4) the collision events, including the object ids, relative velocities, time and states; 5) the trial seed used to generate the video, the label for the video, and 'start_frame_for_prediction', which is a timestamp indicating the part of the video before the timestamp is visible, and the part after the timestamp is required to be predicted.

*Equal contribution. This work was done when Mingyu was at MIT.
[2]Training data: https://physion-v2.s3.amazonaws.com/train_data.zip
[3]Readout data: https://physion-v2.s3.amazonaws.com/readout_data.zip
[4]Testing data: https://physion-v2.s3.amazonaws.com/test_data.zip
[5]Human data: https://physion-v2.s3.amazonaws.com/human_data.zip

37th Conference on Neural Information Processing Systems (NeurIPS 2023) Track on Datasets and Benchmarks.

Table 1: Table of open-source code used.

Name	URL	License
MCVD [2]	https://github.com/voletiv/mcvd-pytorch	MIT License
ALOE [3]	deepmind/object_attention_for_reasoning	Apache License 2.0
ResNet [4]	https://github.com/pytorch/vision	BSD 3-Clause License
DeiT [5]	https://github.com/facebookresearch/deit	Apache License 2.0
VGGNet [6]	https://github.com/pytorch/vision	BSD 3-Clause License
DPI-Net [7]	https://github.com/YunzhuLi/DPI-Net	N/A
TDW [1]	https://github.com/threedworld-mit/tdw	BSD 2-Clause License
SlotFormer [8]	https://github.com/pairlab/SlotFormer	MIT License

Author statement. We confirm that we bear all responsibility in case of any violation of rights during the collection of the data or other work, and will take appropriate action when needed, e.g. to remove data with such issues.

Hosting, licensing, and maintenance plan. We host the dataset on Amazon AWS. We ensure access to the data and will provide the necessary maintenance. All products created as part of this project is shared under the MIT license. We used a number of third-party software packages, each of which typically has its own licensing provisions. Only TDW [1] was used in the creation of the dataset; all others were models used for assessment. Table 1 contains a list of these licenses for many of the packages used.

B Scenario Details

Mass-Dominoes. This scenario starts with an inference phase, where a set of "dominoes" (equally sized cuboids standing long end up) are placed approximately in a row with semi-random spacing and orientations. One of the dominoes is visually marked with a different texture, indicating a different material. At the start of the video, a domino at one end starts to fall as if it had been pushed over, starting a sequence of the dominoes being pushed over. The video continues past the point where the textured domino is hit by or hits one of the other dominoes, providing information about the textured dominoes' mass. A transition phase is then required (as all relevant objects have toppled) and the dominoes are reset. In the reset scene, there is a mat on the floor, and one of the dominoes is indicated as the 'red' object while the mat is the 'yellow' object. The textured domino is placed in the chain so that its mass will influence the chain of dominoes: e.g., if it is too heavy it would not topple when another domino strikes it, but if not it will continue the sequence of collisions and cause the red domino to land on the yellow mat.

Mass-Waterpush. The inference phase begins with an object at rest and a stream of water shooting towards the object as if out of a hose. This stream may move the object, and thus give information about its mass. The transition phase then occurs where the object is moved to another location (and placed upright if it has tipped over) and marked as 'red'. In addition, a new object is added falling from midair, marked as 'yellow'. The yellow object might be positioned above the red object, in which case, depending on the masses, the water might knock the red object out of the way or fail to move it. Or the yellow object might be positioned further along the path defined by the water stream, so that the stream might cause the red object to slide into the yellow object, or fail to move.

Mass-Collision. The scenario is identical to the Mass-Waterpush scenario, except that instead of a stream of water that pushes the object, a ball rolls into and collides with the object in both the inference and testing phases.

Elasticity-Wall. In the inference phase, an object is in ballistic motion towards a wall, bounces off, and lands on the floor. This provides information about the elasticity of the collision. The transition phase then occurs, and the object is again placed in ballistic motion towards the wall (which does not move) and marked as the 'red' object, with a 'yellow' mat being placed on the floor. Depending on the elasticity, after bouncing off of the wall, the red object may land on, or over- or under-shoot the yellow mat.

Elasticity-Platform. This scenario contains a raised platform that ends and drops onto a surface, with a wall at the end of that surface, followed by a mat (marked as 'yellow') on the floor. It starts

with an object (marked as 'red') bouncing onto the platform and continuing to bounce/slide to its edge. This provides information about the objects' elasticity, and thus how it will bounce when it falls onto the surface. The key question is whether the red object will touch the mat. In some cases, the wall is short, so the object must have high elasticity to bounce over and hit the mat. In other cases, the wall is tall but has a slot at the bottom; in these cases if the object bounces too much it will hit the wall and stop, but if it is less elastic it will slide through the slot and contact the mat. In this way we decorrelate the elasticity from the outcome.

Friction-Slide. There were two subtypes of scenarios for Friction-Slide. In both cases, an object (marked as 'red') is positioned near the top of a ramp and begins to slide down. In the 'gap' situation, there is a divot in the slope, and the 'yellow' mat is positioned in that gap; thus depending on the friction of the red object it might slide into the gap and contact the mat or fly over the gap and miss it. In the 'no-gap' situation, there is no divot, and the 'yellow' object is placed in the runout area of the slope; thus the red object might stop before hitting the yellow object or might continue to slide into it. We use both subtypes so that there is not a correlation between low- or high-friction objects and the outcome. No transition phase is needed in this case; the inference can be performed from the first part of the video when the object is sliding down the slope.

Friction-Collision. This scenario starts with the key 'red' object sliding along the floor, providing information about the friction of that object. The transition phase occurs and then the red object is reset to a different position with a new velocity, and a 'yellow' object is dropped from above (similar to the Mass-Collision and Mass-Waterpush scenarios). The crucial judgment is whether the friction of the red object will slow it down just enough so the yellow object will land on it, or whether the red object will under- or over-shoot the mark.

Friction-Clothslide. This scenario is identical to the Friction-Slide scenario, except instead of a rigid object sliding down the ramp, the 'red' object is a piece of cloth.

Deform-Roll. This scenario starts with a cloth hanging from posts either vertically or horizontally, and a key object is launched or dropped on the cloth respectively. This provides information about the deformability of the cloth. The transition phase occurs and the cloth is now hung from posts at an angle. The critical 'red' object is dropped from above the cloth, and the 'yellow' object is set in one of two places. It is either positioned on the ground towards the base of the cloth, so that it is important to determine whether the red object will sink into the cloth, or roll off of it and hit the yellow object. Or the yellow object is dropped from above the red object, so that if the cloth is deformable enough both will sink in and touch, but if it is not, the red object will roll off before the yellow and they will never contact.

The examples of all 9 scenarios are shown in our project page.

C Details of the Dataset

In some scenarios, the inference and prediction phases can be included in the same video (*e.g.*, Elasticity-Platform, Friction-Slide, and Friction-ClothSlide). However, in many cases, the physical event that provides information in the inference phase irrevocably changes the configuration of objects so that there is no way to use the inferred information for future predictions (*e.g.*, judging mass from seeing one domino topple into another leaves them both on the floor at the top row of Figure 2 in the main paper). In these cases, we include a "transition phase": a curtain slides in to block the scene, then while the scene is occluded the objects are rearranged for the prediction phase, and finally the curtain moves out of the way. And the cueing of the two target objects is done immediately after the transition phase, followed by a short observation of the rearranged objects in motion. Examples of all 9 scenarios are shown in our project page.

Training set. The training dataset is used for the agents to learn dynamics prediction, and the learned representations can be discriminative enough to distinguish whether the red object hit the yellow object, and can generalize to the testing dataset. We generate 2000 trials for each mechanical property without YES/NO labels for dynamics pretraining. For each physical scenario, we make half of the trials where the output answers are YES and half of them are NO, so as to ensure the balance of learning.

Readout fitting set. The readout fitting set is a small dataset containing 192 trials used to map the dynamic representation learned in the training set to YES/NO of the video question-answering (*i.e.*, OCP) task.

Testing set. The final testing benchmark consists of 192 trials (96 pairs) for each mechanical property. We aim to avoid strong associations between superficial visual cues with the final YES/NO outcome by designing the readout and test dataset to be "paired" trials, where the paired video scenes are visually identical in the first frame during the prediction phase yet they unfold into different event outcomes due to different latent physical properties assigned to the objects in the videos. We achieve this by fixing object configuration during the prediction phase and regenerating the stimuli with uniformly sampled mechanical values on a target object until we get one stimulus with a positive outcome, and the other with a negative outcome. For each scene configuration, we sample a maximum of 5 different property values, and we drop scenes where we sample all true or all false outcomes. It has the same overall visual and physical statistics as the readout fitting set so that the learned mapping from the readout set can be directly evaluated on the test set.

D Pipelines and settings

For each video, we truncate (or pad) both the inference phase and the prediction phase to 160 frames, and sub-sample the videos by a factor of 5 for training the representation or dynamics models. All frames are resized to 128×128 to reduce the computational cost. For SlotFormer [8] and ALOE [3], we first pre-train the object-centric models STEVE [9] and MONet [10] on all sub-sampled frames for scene decomposition, and extract all slot representations for subsequent training. The dynamics models are then trained on the slot representations from the training set under future prediction loss. For MCVD, the frames are directly fed into the model for dynamics prediction under image reconstruction loss. For pRESNET-mlp, pVGG-mlp, and pDEIT-mlp, we leverage pretrained ResNet50 [4], VGG16 [6], and DeiT-small [5] on ImageNet as our feature extractors. For DPI-Net, we represent the scene with particle representation provided by the annotation. For the oracle model with property inference, we add the ground-truth property values into the attribute embedding input of DPI-Net in both training and testing. For the model without property inference, we simply mask all property values with padding zero vectors. For the full video observed, we feed the ground-truth particle-based representation to the model. We calculate the distance d_{min} between the closet particles in the two target objects. We consider the two objects will contact if d_{min} is smaller than a threshold η that is learned from the training set. η is set to 0.075. For other parameters, we follow the same setting as Physion [11].

For models to generate YES/NO responses from their learned representations, we use the readout fitting set for the models to learn to map from their latent representation to the target response. We perform rollout to generate future scene representations (e.g. feature maps for image-based methods, or object slots for object-centric methods) based on the inference phase in the readout set. We implement a multilayer perceptron (MLP) with intermediate dimensions of 256 and 64 as our readout model, which is trained on rollout scene representations from the readout set to classify whether the two cued objects contact. All experiments were run on 8 NVIDIA TITAN X GPUs using the Adam optimizer and a learning rate of 1e-4. The models learned on the training and readout sets are then evaluated on our final benchmark (testing set) by applying the learned visual representations and the readout model. The best testing results among all readout training epochs are reported.

E Explicit and implicit physical property inference

We selected four representative models (DeiT-mlp, ResNet-mlp, ALOE, and SlotFormer) for the experiment. A parallel layer of MLP is added at the end of the model (two-branch multitask: OCP and explicit property estimation) during the readout fitting process. With explicit property estimation as an auxiliary task, we report the performance comparison with our original setting as in Table 2. With explicit inference, performance barely improves, suggesting that the networks do not have access to the properties even when prompted, not that they understand properties but fail to use them for prediction.

Table 2: Ablation study on implicit physical property inference.

Method	DeiT-mlp	ResNet-mlp	ALOE	SlotFormer
implicit (OCP only)	55.4	55.1	53.4	56.7
explicit as auxiliary	54.2	55.7	54.7	56.7

F Datasheets for dataset

Here are our responses in reference to the Datasheets for Datasets [12] standards.

Motivation.

- **For what purpose was the dataset created?** To measure deep models' physical future prediction abilities and latent property inference capabilities, and compare these to predictions made by humans.
- **Who created the dataset and on behalf of which entity?** The authors listed on this paper, including researchers from MIT, Stanford, UC Berkeley, MIT-IBM Watson AI Lab, and UMass Amherst.
- **Who funded the creation of the dataset?** The various granting agencies supporting the above-named researchers, including both grants to the PIs as well as individual fellowships for graduate students and postdoctoral fellows involved with the project.

Composition.

- **What do the instances that comprise the dataset represent?** Each instance is a video of a simulated physical scene (e.g. a tower of blocks as it either collapses or remains steady), together with some metadata about that video, including map-structured metadata with segmentation maps and information about object-object collisions at each timepoint.
- **How many instances are there in total?** The dynamics prediction model training dataset consists of 2000 examples for each of the 4 physical properties. The OCP readout fitting dataset consists of 192 examples per each of the 4 physical properties. The test dataset (on which human responses were obtained) consists of 192 examples per physical property.
- **Does the dataset contain all possible instances or is it a sample of instances from a larger set?** Data is generated by a simulator; in a sense, the set of datapoints we created is an infinitesimally small subset of data that *could* have been generated. However, we are all here releasing all the examples we did actually generate.
- **What data does each instance consist of?** It consists of a video depicting a physical situation (e.g a tower of blocks falling over), together with simulator-generated metadata about the situation.
- **Is there a label or target associated with each instance?** For the training dataset, there are no labels. For both the OCP readout fitting dataset and the human testing dataset, there are binary labels describing whether the red object collided with the yellow zone during the duration of the trajectory.
- **Is any information missing from individual instances?** No.
- **Are relationships between individual instances made explicit?** Yes. All data is provided in a simple data structure that indicates which instances of data are connected with which instances of metadata.
- **Are there recommended data splits?** Yes, for each of the scenarios in the datasets, there are three splits: (a) a large training split for training physical prediction models from scratch; (b) a smaller readout-training set that is to be used for training the yes/no binary readout training as described in the paper, and (c) the test dataset on which human responses were obtained.
- **Are there any errors, sources of noise, or redundancies in the dataset?** We have not found any as of this publication. As these are discovered, they will be fixed and versioned.

- **Is the dataset self-contained, or does it link to or otherwise rely on external resources?** It is self-contained.
- **Does the dataset contain data that might be considered confidential?** No.
- **Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety?** No.
- **Does the dataset relate to people?** No.

Collection Process.

- **How was the data associated with each instance acquired? What mechanisms or procedures were used to collect the data? How was it verified?** Videos (for training, readout fitting, and human testing) were generated using the TDW simulation environment. Online crowdsourcing was used to obtain human judgements for each testing video. During the creation of the simulated videos, the researchers looked at the generated videos by eye to verify if the scenarios were correct (e.g. actually depicted the situations desired by our experimental design). Prior to running the actual data collection procedure for humans, we verified that the experimental websites were correct by having several of the researchers complete the experiment themselves.
- **Who was involved in the data collection process and how were they compensated?** PIs, students, and postdocs generated simulator-generated videos. For human responses, 200 participants (50 for each of the mechanical properties) were recruited from Prolific and paid $15.50 per hour for their participation.
- **Over what timeframe was the data collected?** All simulator-generated scenarios were created and human data was collected during the second half of 2022.
- **Were any ethical review processes conducted?** All human data collection was approved by UC San Diego IRB.

Preprocessing, clearning and labelling.

- **Was any preprocessing/cleaning/labeling of the data done?** We reviewed the test scenarios to make sure we videos with non-informative situations were not included (e.g., one of the key objects is fully occluded during the entirety of the video). No other preprocessing was done, and labeling was produced automatically by the system.

Uses.

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

Distribution.

- **Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created?** Yes it will be completely publicly available via our project page and the links listed thereupon.
- **How will the dataset will be distributed?** It will be available via links to the project page, and which will refer to permanent Amazon S3 resources.

- **When will the dataset be distributed?** Immediately.
- **Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)?** The dataset and associated code will be licensed under the MIT license.
- **Have any third parties imposed IP-based or other restrictions on the data associated with the instances?** No.
- **Do any export controls or other regulatory restrictions apply to the dataset or to individual instances?** No.

Maintenance.

- **Who is supporting/hosting/maintaining the dataset?** The dataset is hosted on Amazon S3 resource. The associated Amazon S3 account is the institutional account for the CogTools lab (at Stanford).
- **How can the owner/curator/manager of the dataset be contacted?** The corresponding author of the paper can be contacted via email as described in the front page of the paper.
- **Is there an erratum?** No. If needed, any future errata will be posted on the project page.
- **Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances)?** As this dataset becomes used by a larger audience, we will review the instances for errors that users uncover. These errors will be corrected as they are discovered on an ongoing basis.
- **Will older versions of the dataset continue to be supported/hosted/maintained?** If newer versions of the dataset are created, these will only be in additional to the existing data. Old versions will be maintained indefinitely.
- **If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so?** No. Making contributions to this dataset requires a very detailed understanding of a variety of components and how they interconnect – physics simulators, scenario generation modules, online psychophysical experimentation platforms, etc. – and we do not contemplate allowing third parties to (e.g.) add new examples of physical scenarios.

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

Dataset identifier. At the moment, we provide access to the dataset via Amazon S3 links that are visible via our project page. We have not yet pushed out data into a standard data repository or created a DOI for it. This is because we expect the specifics of how the data is made available to develop during the paper review process. Once this is complete, we will push the data into a standardized data repository and generate a DOI for it.

G Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]

(b) Did you include complete proofs of all theoretical results? [N/A]

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [Yes]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [Yes]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [Yes]

References

[1] Chuang Gan, Jeremy Schwartz, Seth Alter, Martin Schrimpf, James Traer, Julian De Freitas, Jonas Kubilius, Abhishek Bhandwaldar, Nick Haber, Megumi Sano, Kuno Kim, Elias Wang, Damian Mrowca, Michael Lingelbach, Aidan Curtis, Kevin T. Feigelis, Daniel M. Bear, Dan Gutfreund, David D. Cox, James J. DiCarlo, Josh H. McDermott, Joshua B. Tenenbaum, and Daniel L. K. Yamins. Threedworld: A platform for interactive multi-modal physical simulation. *CoRR*, abs/2007.04954, 2020.

[2] Vikram Voleti, Alexia Jolicoeur-Martineau, and Christopher Pal. Masked conditional video diffusion for prediction, generation, and interpolation. *arXiv preprint arXiv:2205.09853*, 2022.

[3] David Ding, Felix Hill, Adam Santoro, and Matt Botvinick. Object-based attention for spatio-temporal reasoning: Outperforming neuro-symbolic models with flexible distributed architectures. *arXiv preprint arXiv:2012.08508*, 1, 2020.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[5] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021.

[6] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[7] Yunzhu Li, Jiajun Wu, Russ Tedrake, Joshua B Tenenbaum, and Antonio Torralba. Learning particle dynamics for manipulating rigid bodies, deformable objects, and fluids. *arXiv preprint arXiv:1810.01566*, 2018.

[8] Ziyi Wu, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg. Slotformer: Unsupervised visual dynamics simulation with object-centric models. *arXiv preprint arXiv:2210.05861*, 2022.

[9] Gautam Singh, Yi-Fu Wu, and Sungjin Ahn. Simple unsupervised object-centric learning for complex and naturalistic videos. *arXiv preprint arXiv:2205.14065*, 2022.

[10] Christopher P Burgess, Loic Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matt Botvinick, and Alexander Lerchner. Monet: Unsupervised scene decomposition and representation. *arXiv preprint arXiv:1901.11390*, 2019.

[11] Daniel M. Bear, Elias Wang, Damian Mrowca, Felix J. Binder, Hsiau-Yu Fish Tung, R. T. Pramod, Cameron Holdaway, Sirui Tao, Kevin A. Smith, Fan-Yun Sun, Li Fei-Fei, Nancy Kanwisher, Joshua B. Tenenbaum, Daniel L. K. Yamins, and Judith E. Fan. Physion: Evaluating physical prediction from vision in humans and machines. *CoRR*, abs/2106.08261, 2021.

[12] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92, 2021.