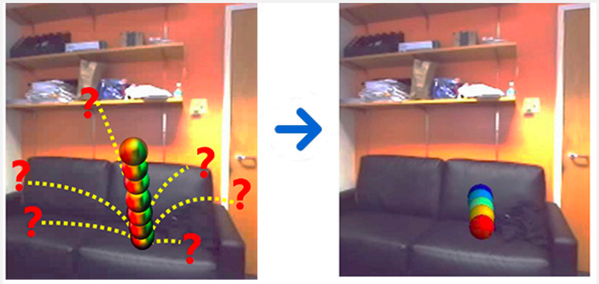
Bounce and Learn: Modeling Scene Dynamics with Real-World Bounces

Senthil Purushwalkam [1] Abhinav Gupta [1] Danny M. Kaufman [2] Bryan Russell [2]
[1]Carnegie Mellon University [2]Adobe Research
http://www.cs.cmu.edu/~spurushw/projects/bouncelearn.html

How do you predict the outcome of this bounce?



Humans:

→ Identify object/material (sofa)
→ Prior knowledge of sofa properties (soft, deformable)
→ Use properties and prior knowledge of interactions to predict outcome

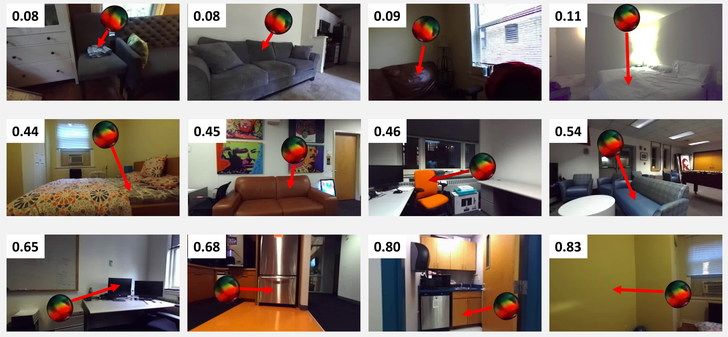
Transfer to machines:

→ Map appearance of objects to physical properties (coefficient of restitution (COR), normal, friction coefficients, etc)
→ Use physical properties + knowledge of physics to predict outcome

This Work

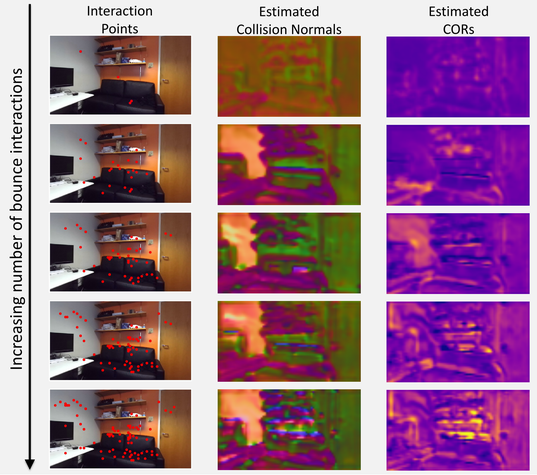
- Dataset of 5k real-world bounces
- Mapping visual appearance to coefficient of restitution (COR) and collision normal
- Learning a physics model to predict post-bounce trajectories using estimated properties

Bounce Dataset



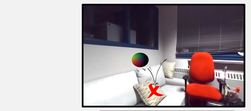
Foam Ball | Homes and Offices | 5K+ Bounces | 100 FPS RGB-D Videos

Online Physical Parameter Estimation



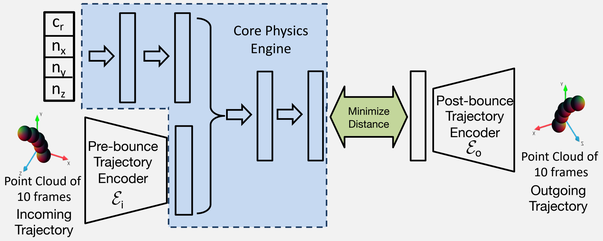
Bounce and Learn Model

Overview



Visual Inference Module (VIM)

Physics Inference Module (PIM)



Pre-bounce and post-bounce encoders are inspired from PointNet [3]

Pretraining the PIM

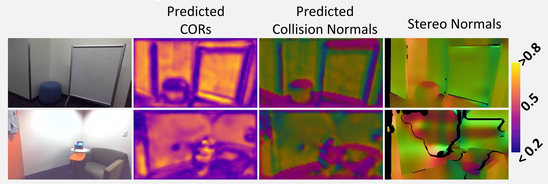
- PyBullet [1] simulated bounce point clouds
- Triplet Margin Loss on trajectory encodings*

Training VIM + PIM

- Use trajectories from Bounce Dataset
- VIM estimates physical params at bounce location $\rho_{x,y}$
- PIM predicts the output trajectory using $\rho_{x,y}$ as t_{pred}
- VIM + PIM can be trained jointly using L2 distance

Qualitative Results

Physical Parameter Estimation



Trajectory Forward Prediction

Quantitative Results

Model	Center Dist Err	%Normals within 30°	COR Median Abs. Err
Parabola encoding	26.3	17.52	0.179
Center encoding	23.1	23.41	0.178
Pretrain VIM + IN [2]	40.1	-	-
Ours	**21.3**	**24.08**	**0.168**
Ours + Stereo Normals	**22.7**	**50.14**	**0.159**

References:
1. Erwin Coumans and Yunfei Bai. pybullet, a Python module for physics simulation for games, robotics and machine learning. http://pybullet.org/, 2016–2017
2. Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, and Koray Kavukcuoglu. Interaction networks for learning about objects, relations and physics. In Advances in Neural Information Processing Systems (NIPS), 2016
3. Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. arXiv preprint arXiv:1612.00593, 2016