Active Scene Understanding with Robot Interactions

Shuran Song

COLUMBIA | ENGINEERING
The Fu Foundation School of Engineering and Applied Science

Collaborators

Zhenjia Xu

Zhanpeng He

Andy Zeng

Jiajun Wu

Joshua B. Tenenbaum

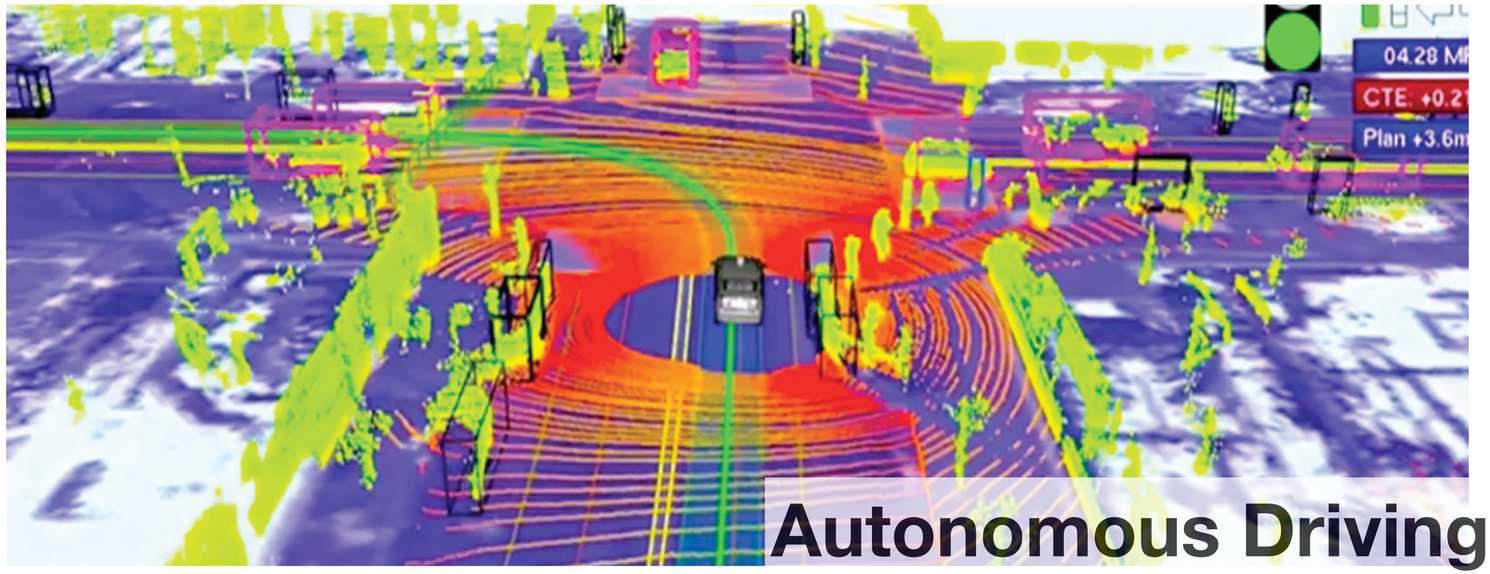
Johnny Lee

Thomas A. Funkhosuer

Alberto Rodríguez

See, Understand, Act

Service Robots

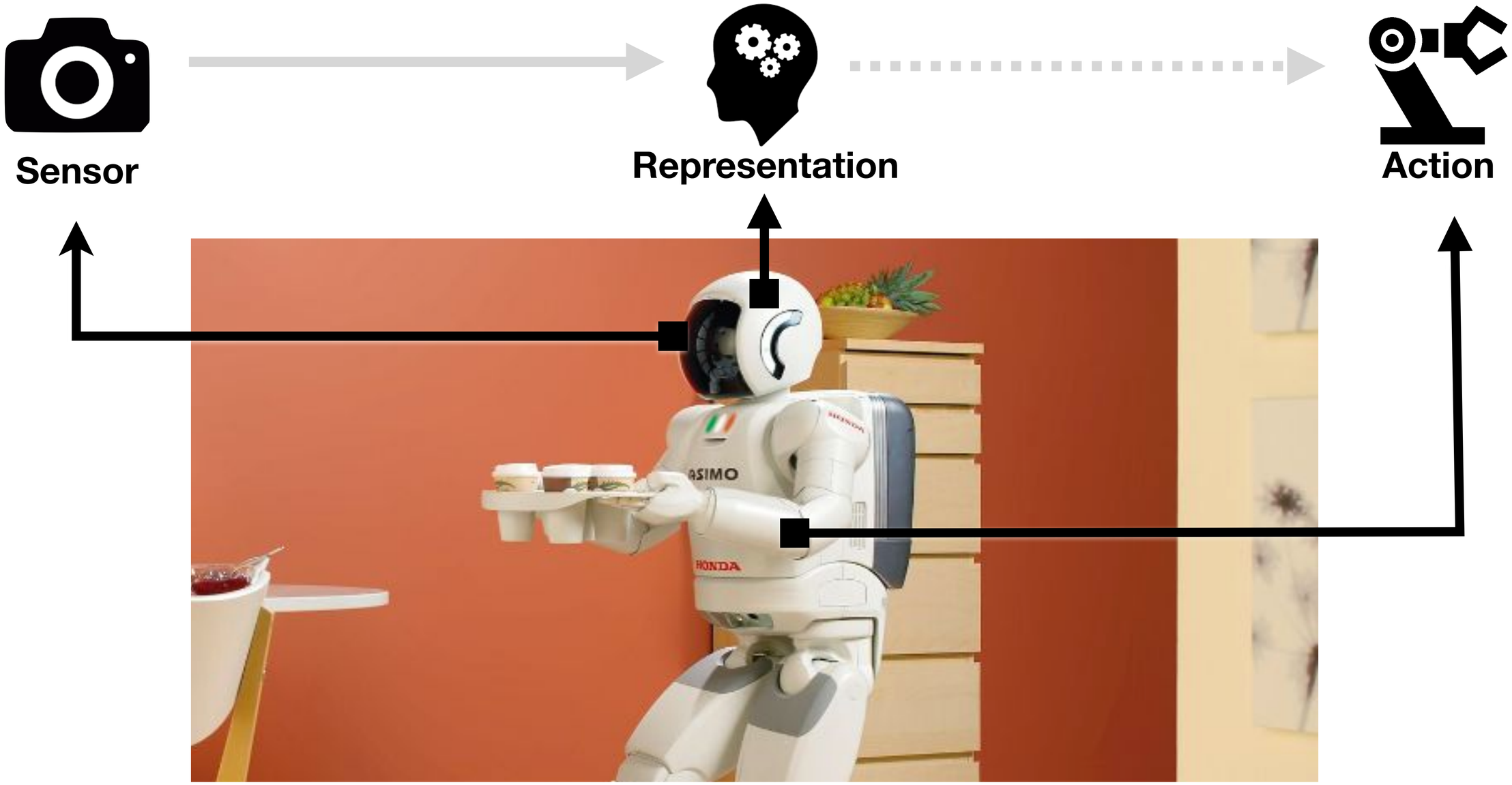


Autonomous Driving

Warehouse Logistics

Manufacturing

See, Understand, Act

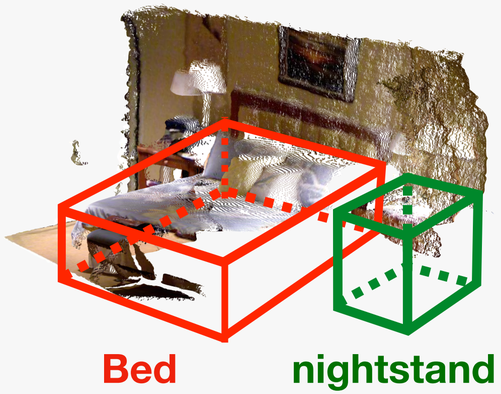


Scene Understanding

Scene Understanding

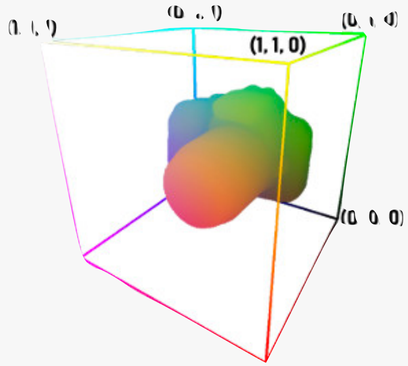
Scene Representations

Object Detection



SlidingShapes
ECCV 2014,CVPR 2016

Pose Estimation

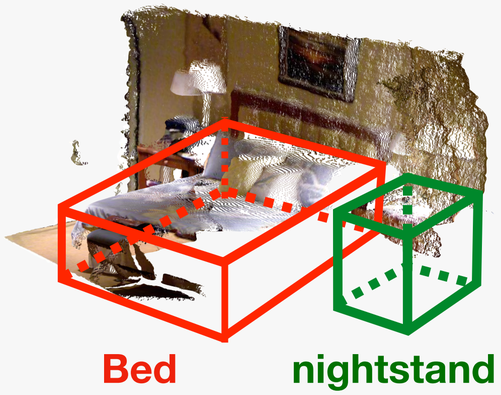


NOCS
CVPR2019

Scene Understanding

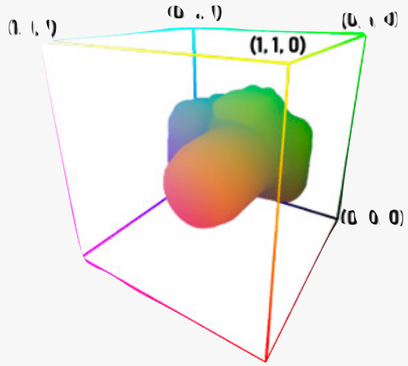
Scene Representations

Object Detection



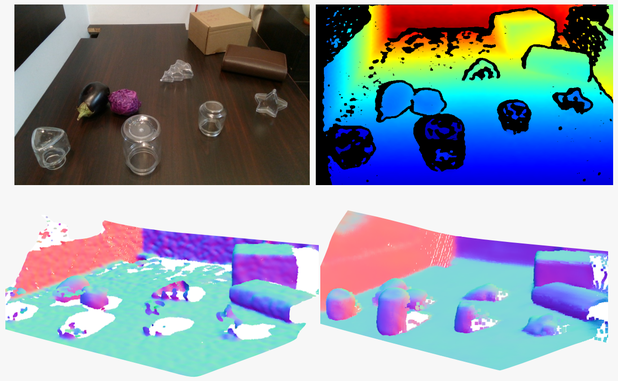
SlidingShapes
ECCV 2014,CVPR 2016

Pose Estimation



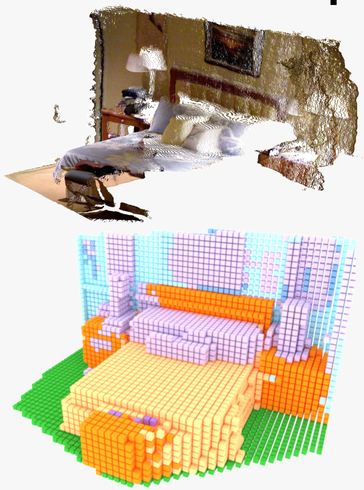
NOCS
CVPR2019

Geometry Estimation
(transparent object)



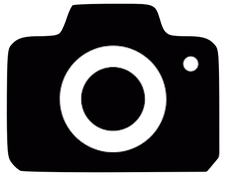
ClearGrasp
ICRA2020

Semantic Scene Completion



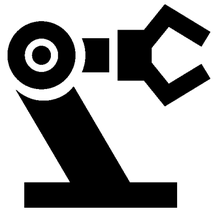
SSCNet
CVPR 2017

Scene Understanding



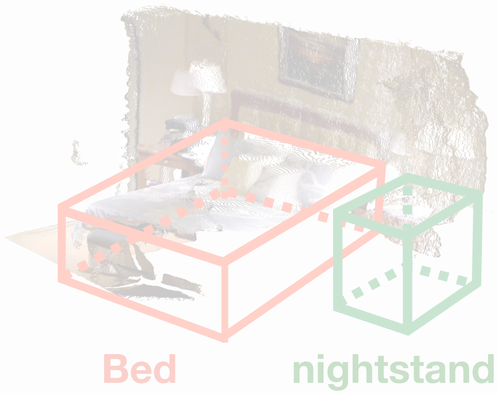
Sensor

Representation



Action

Object detection



SlidingShapes

ECCV 2014,CVPR 2016

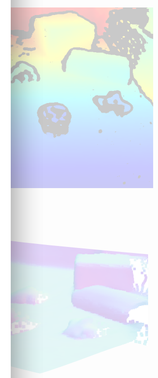
Pose Est

NO

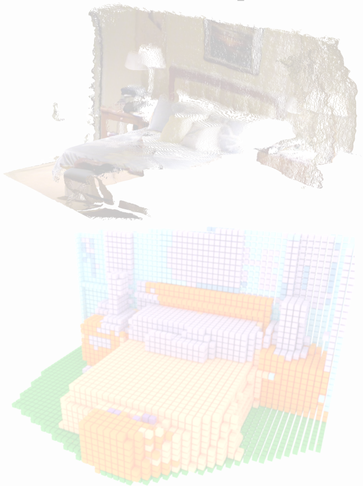
CVPR

Passive Observers

tion



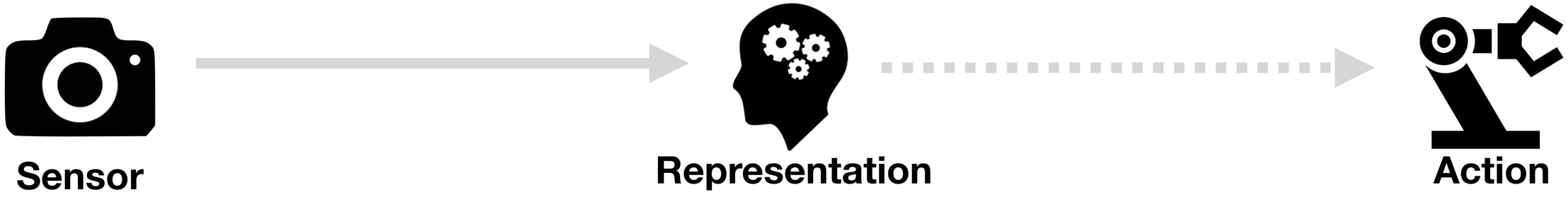
Scene Completion



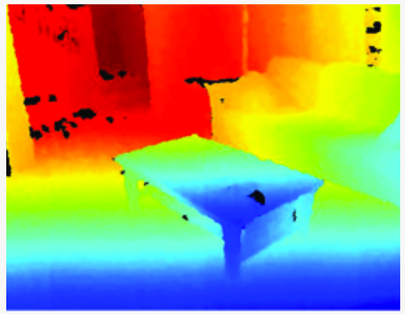
SSCNet

CVPR 2017,18

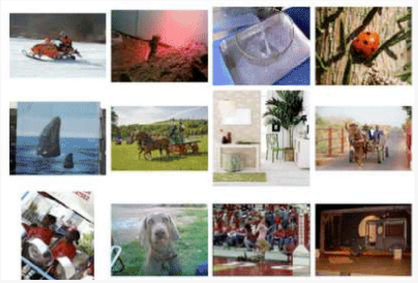
Scene Understanding



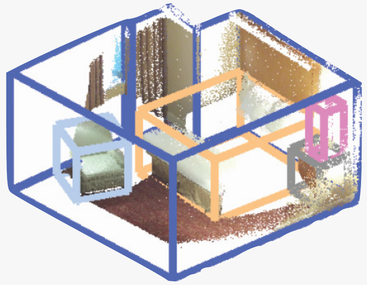
PASCAL VOC



NYU depth



ImageNet

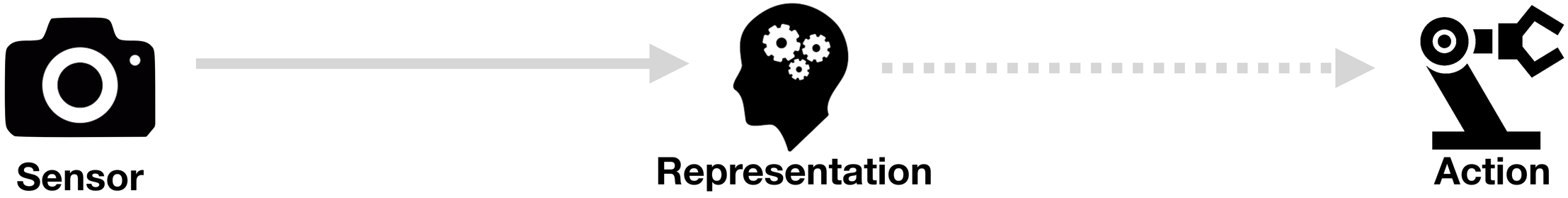


SUN RGB-D

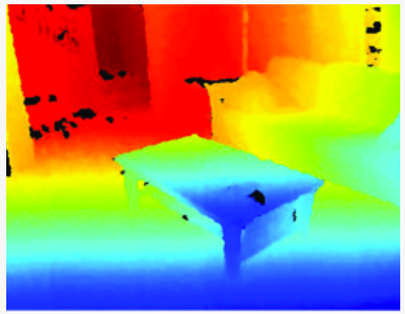
Computer Vision Benchmarks

- Static images

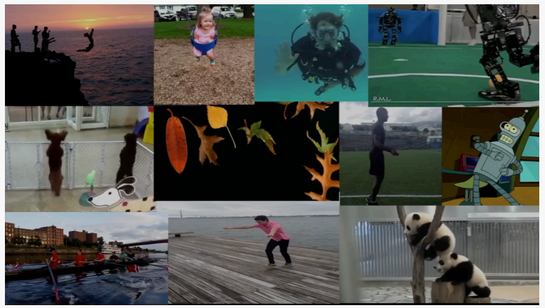
Scene Understanding



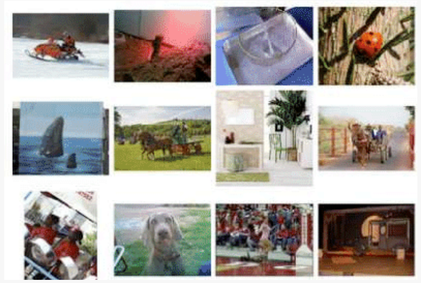
PASCAL VOC



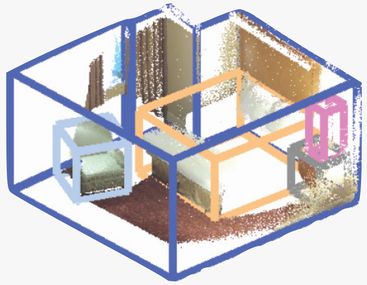
NYU depth



Moment in Time



ImageNet



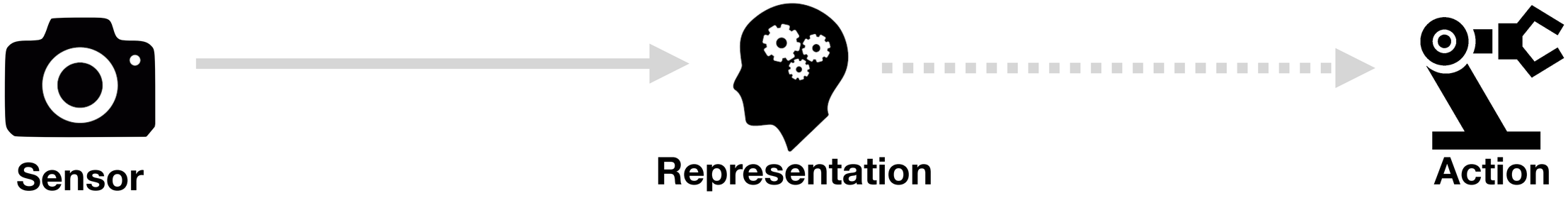
SUN RGB-D

CrowdPose

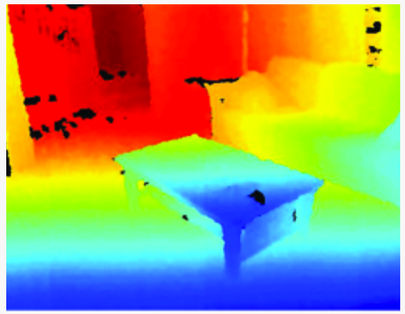
Computer Vision Benchmarks

- Static images
- Passive video

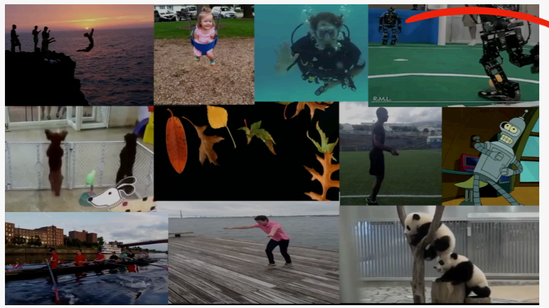
Scene Understanding



PASCAL VOC



NYU depth

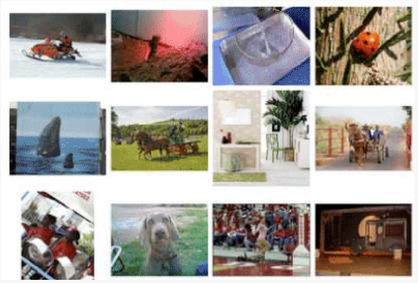


Moment in Time

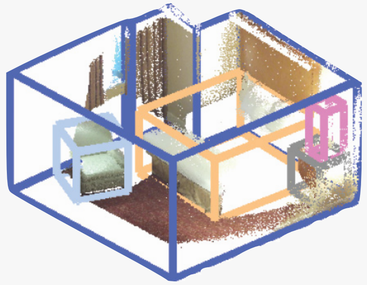
What causes all the motions?

✘ Casual relationship between action and motion

✘ Inform action planing



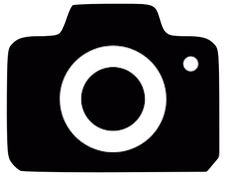
ImageNet



SUN RGB-D

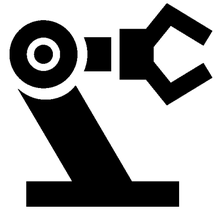
CrowdPose

Scene Understanding



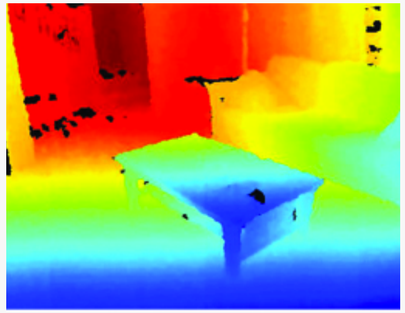
Sensor

Representation

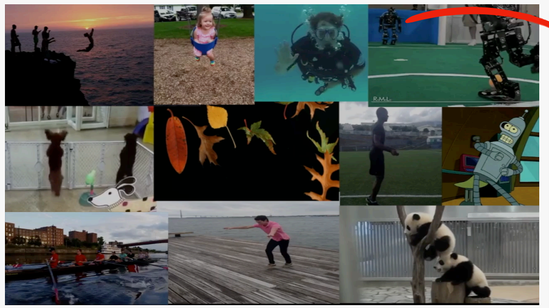


Action

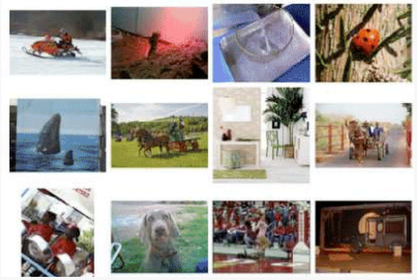
PASCAL VOC



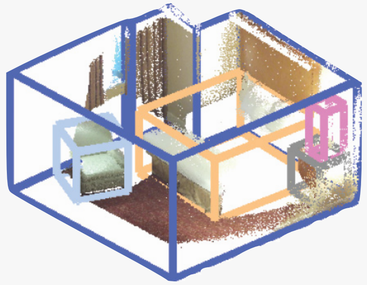
NYU depth



Moment in Time



ImageNet

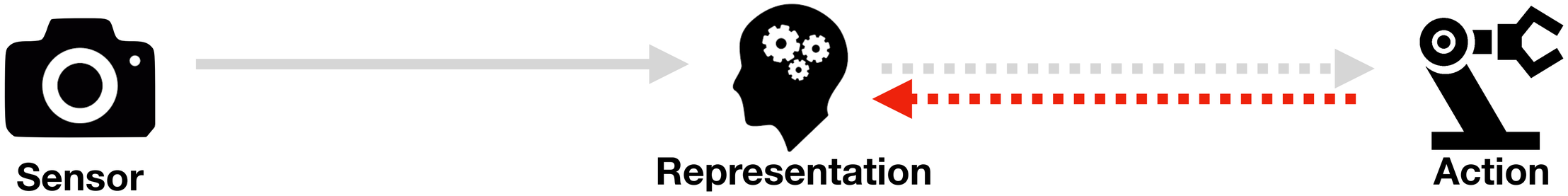


SUN RGB-D

CrowdPose

Passive Observers

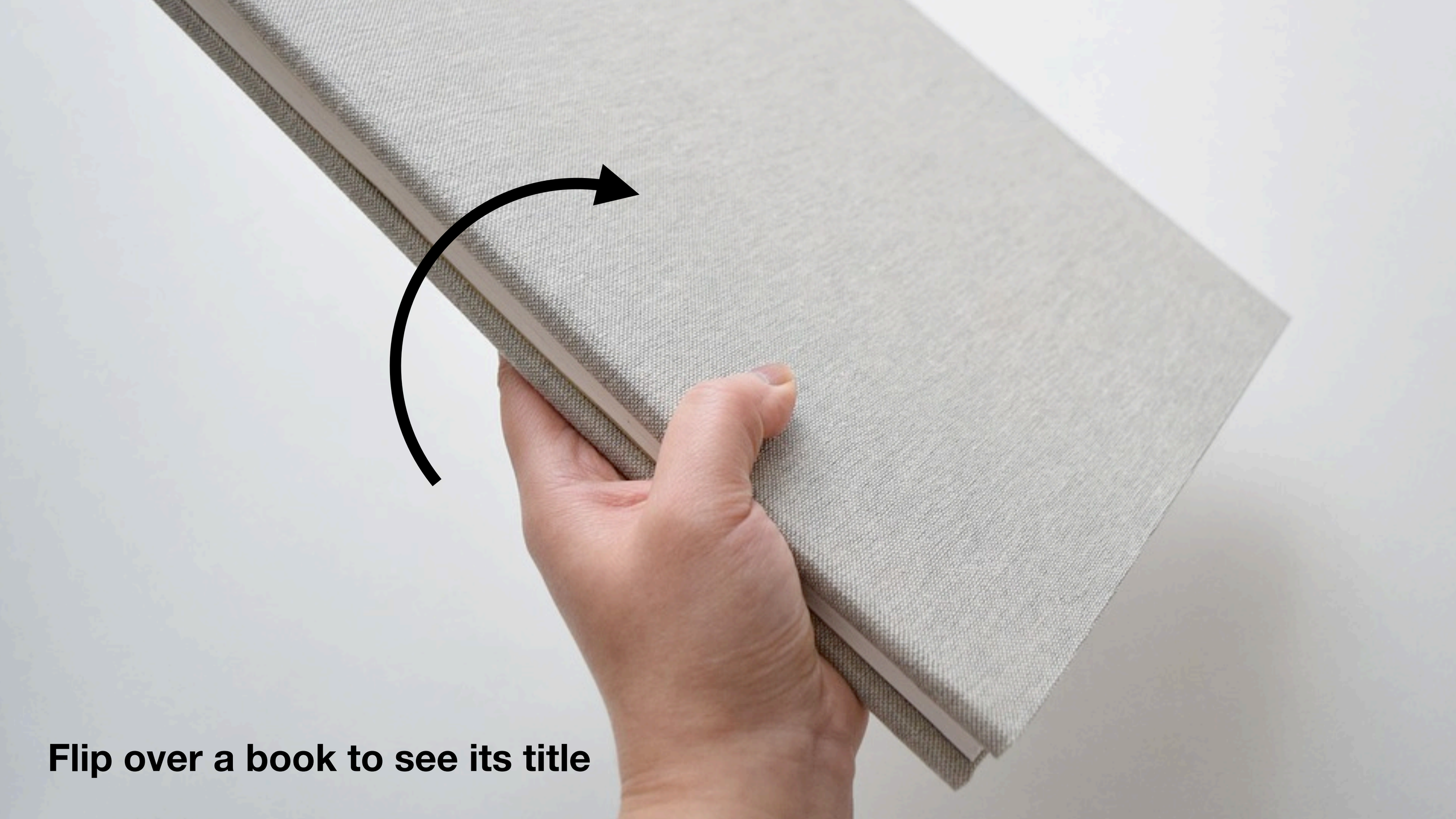
Scene Understanding



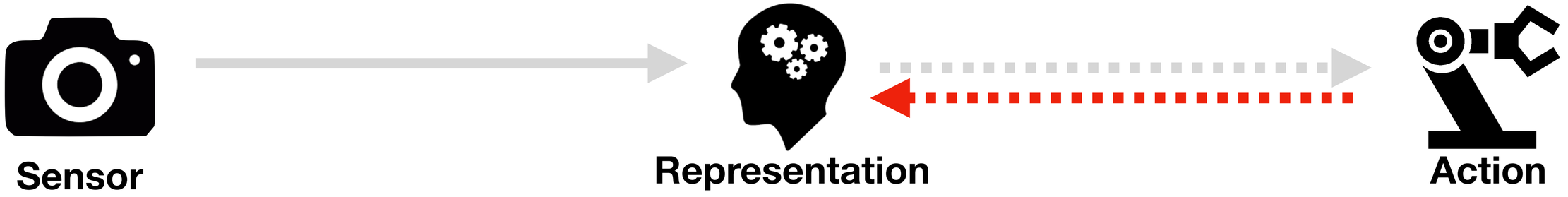
Using active exploration to retrieve useful information

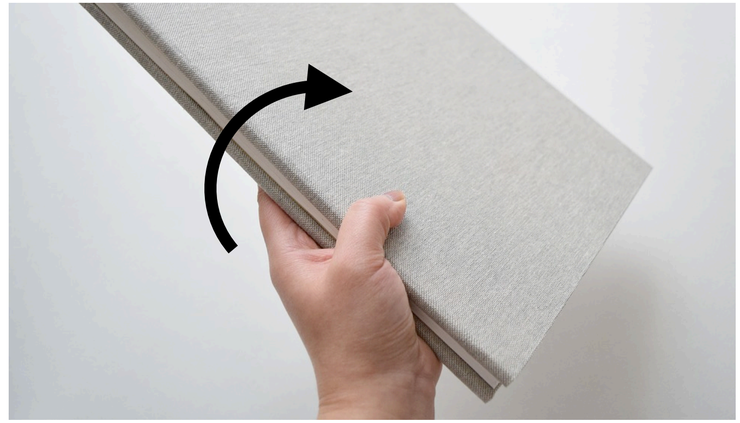
Dip our toes into the water to sense its temperature

Push a large box to sense its weight


Flip over a book to see its title

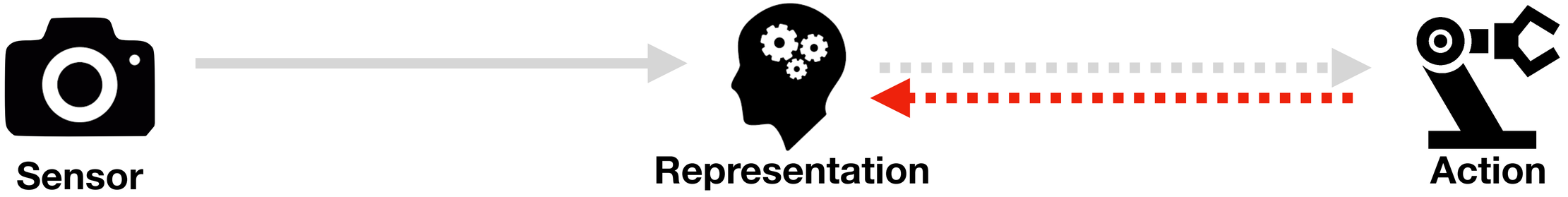
Scene Understanding

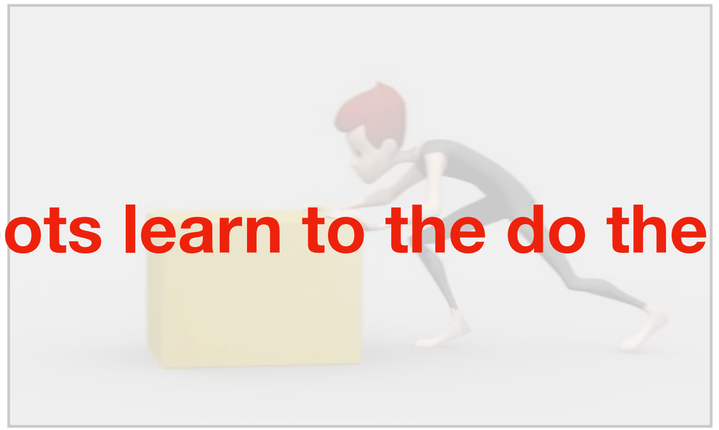




Action	Dipping	Pushing	Flipping
Information	Temperature	Weight	Title
Planing	Swim	Lift up the box	Read the book

Scene Understanding

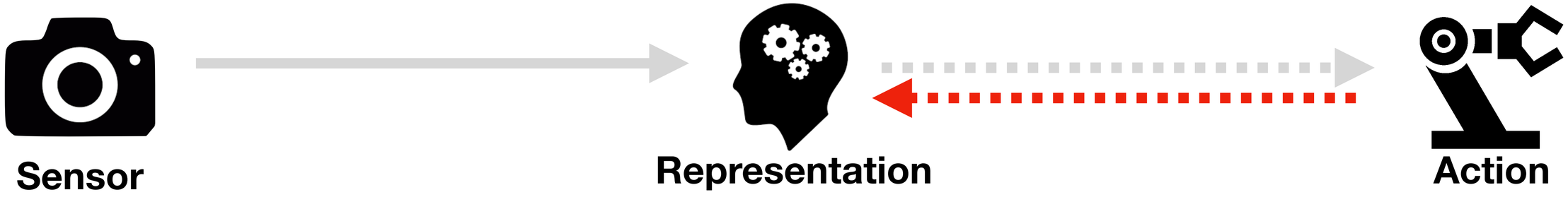




Can robots learn to the do the same?

Action	Dipping	Pushing	Flipping
Information	Temperature	Weight	Title
Planing	Swim	Lift up the box	Read the book

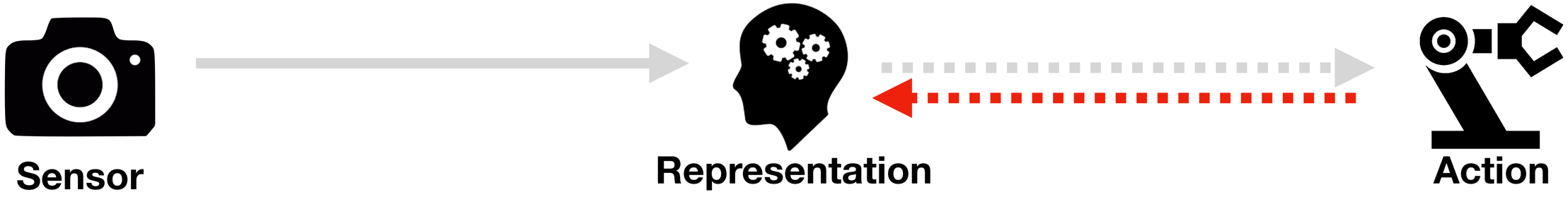
Scene Understanding



Active Explorers

Passive Observers

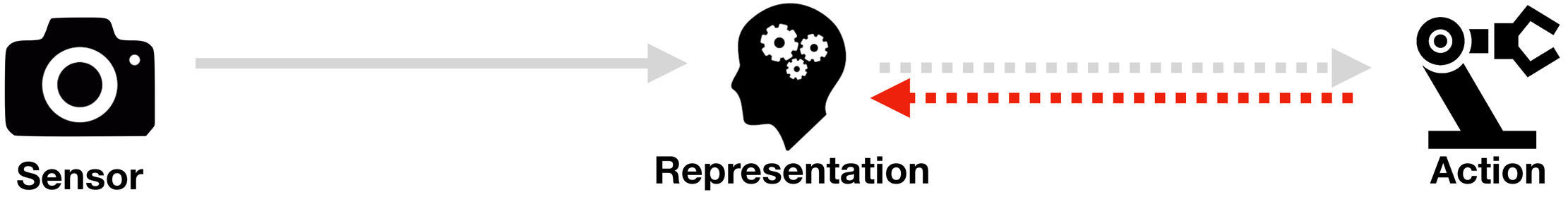
Active Scene Understanding

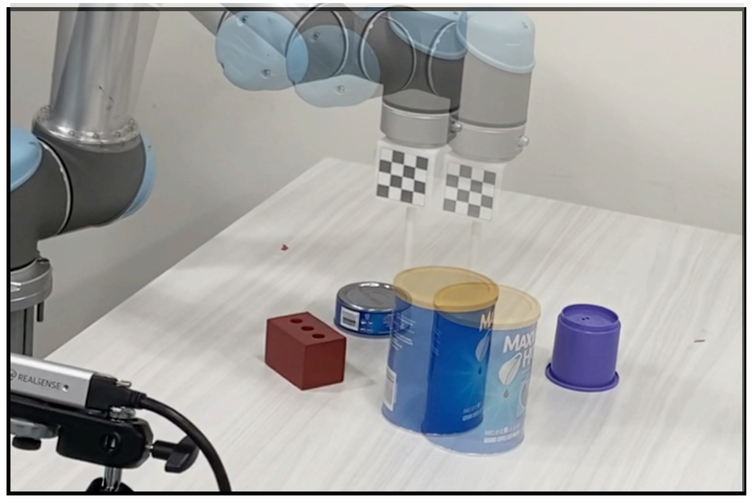


Active Explorers

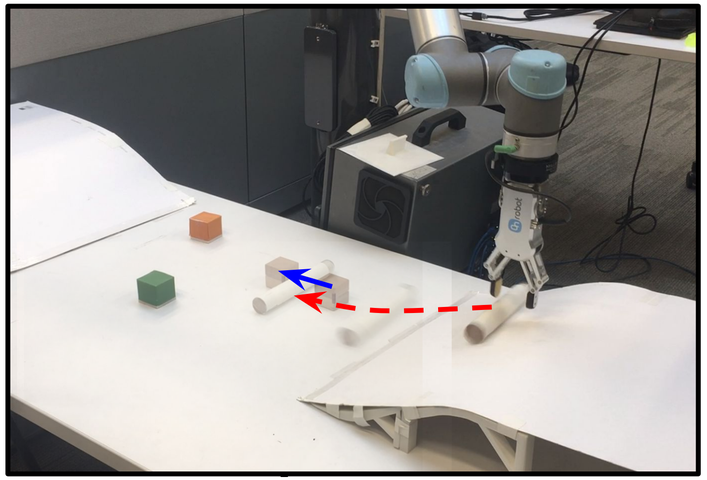
Passive Observers

Active Scene Understanding

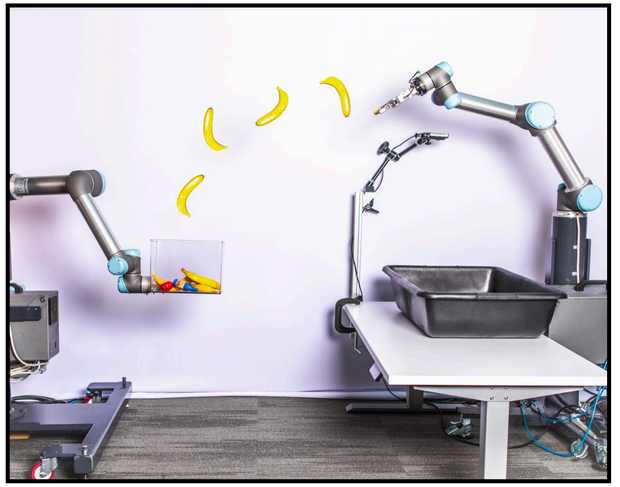




Dynamic Scene Representation
CoRL 2020

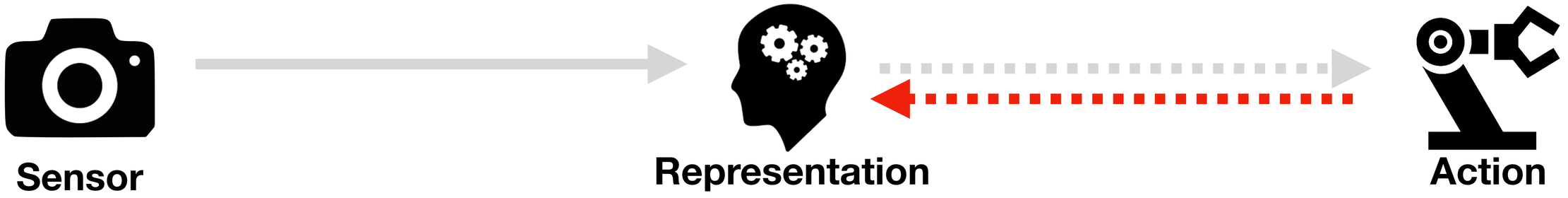


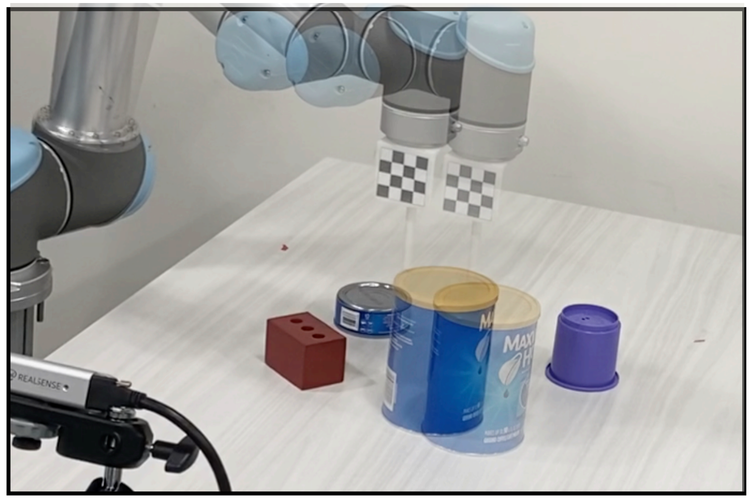
DensePhysNet
RSS2019



TossingBot
RSS2019

Active Scene Understanding





Dynamic Scene Representation
CoRL 2020

Learning 3D Dynamic Scene Representations for Robot Manipulation

Zhenjia Xu, Zhanpeng He, Jiajun Wu, Shuran Song
CoRL 2020

https://dsr-net.cs.columbia.edu/

DensePhysNet
RSS2019

TossingBot
RSS2019

Interaction for Perception

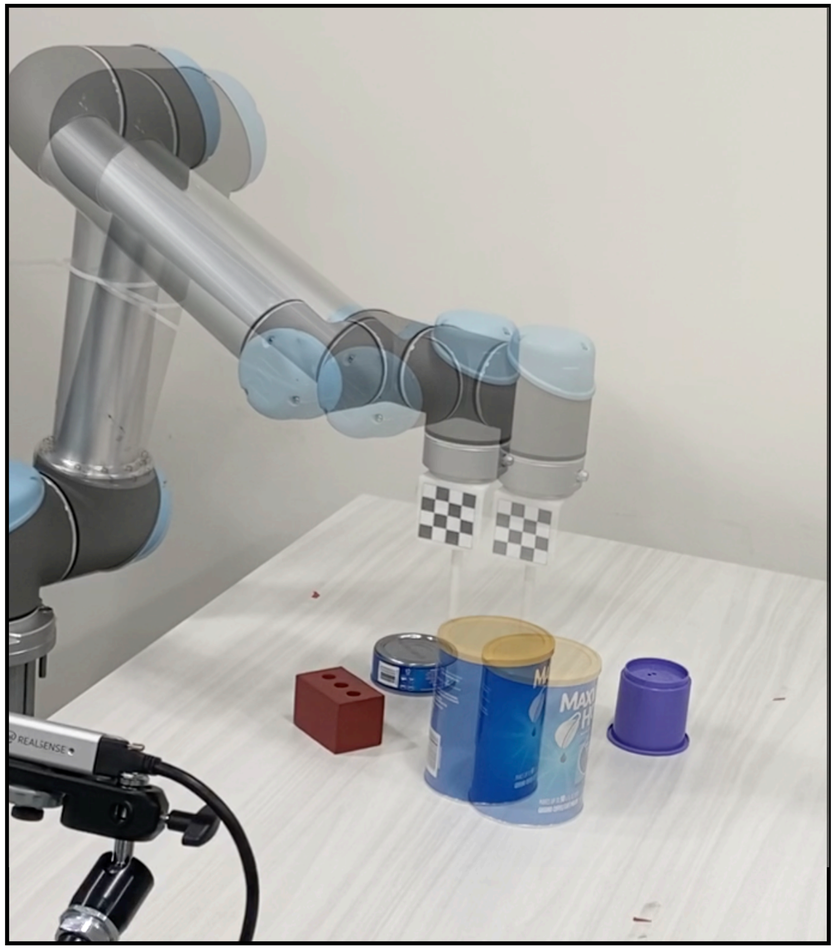
Goal:

Learning a Scene Representation with Interactions

Objects'
Instance Identity
3D geometry
Dynamics

How to do it?

Learning to predict object movement
under robot's interaction

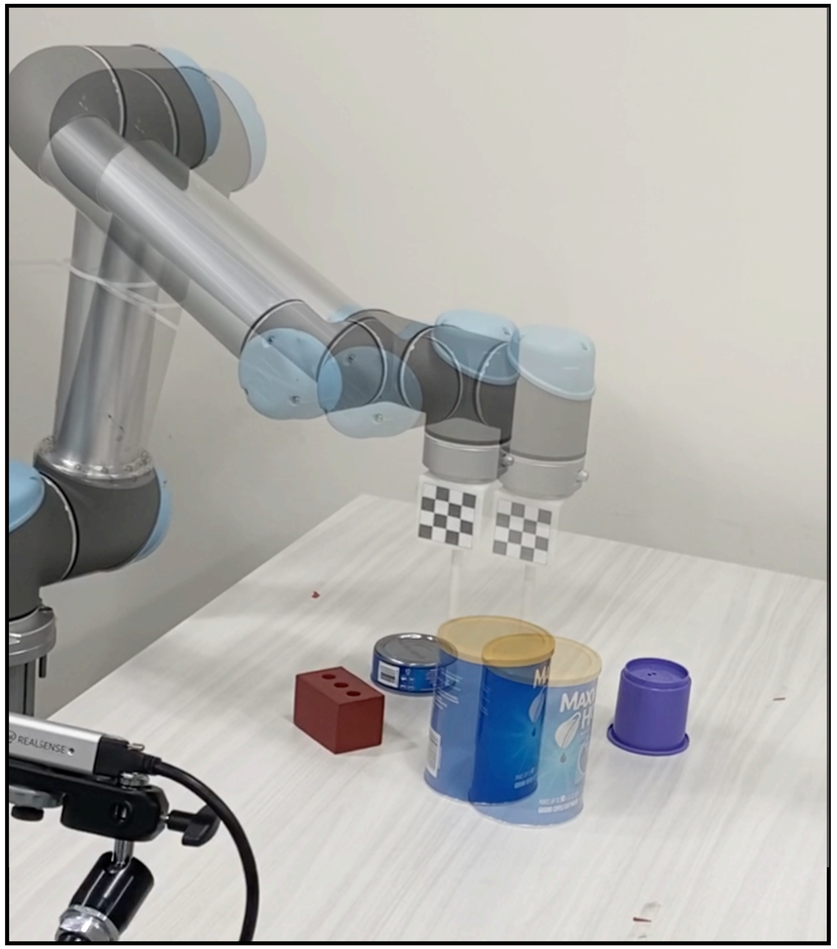


Interaction for Perception

Why does it work?

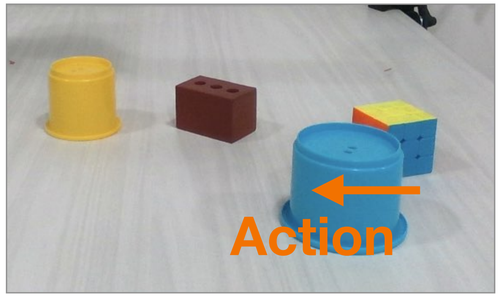
We know that the points on the same rigid object should move together. Formally, they should be described by the **same** SE(3) transformation.

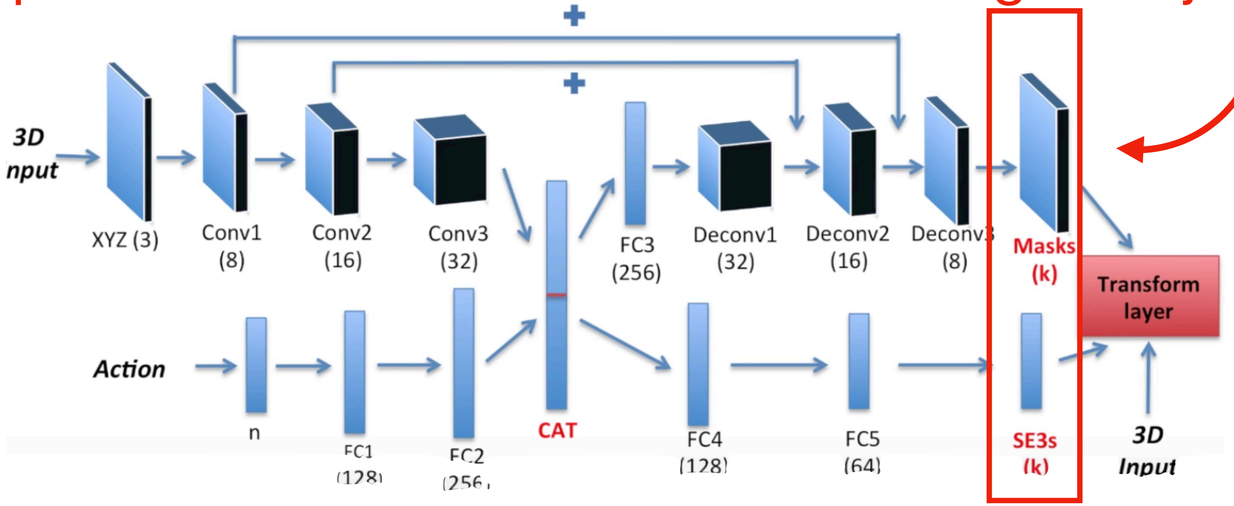
Therefore, by analyzing the motion of the whole scene, the system will be able to identify the each individual rigid object that would best explain the motion.

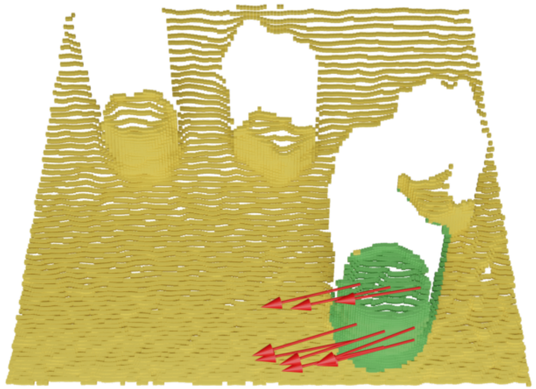


Prior work: SE3-Net

Masks represent the networks's understanding of object instances





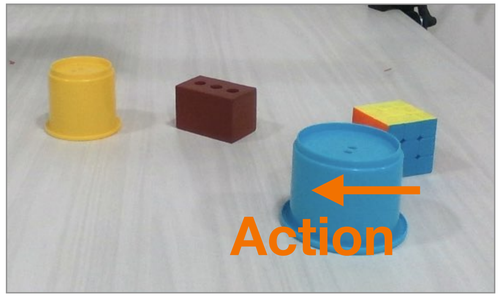


Predicting
K SE(3) transformation for K different masks

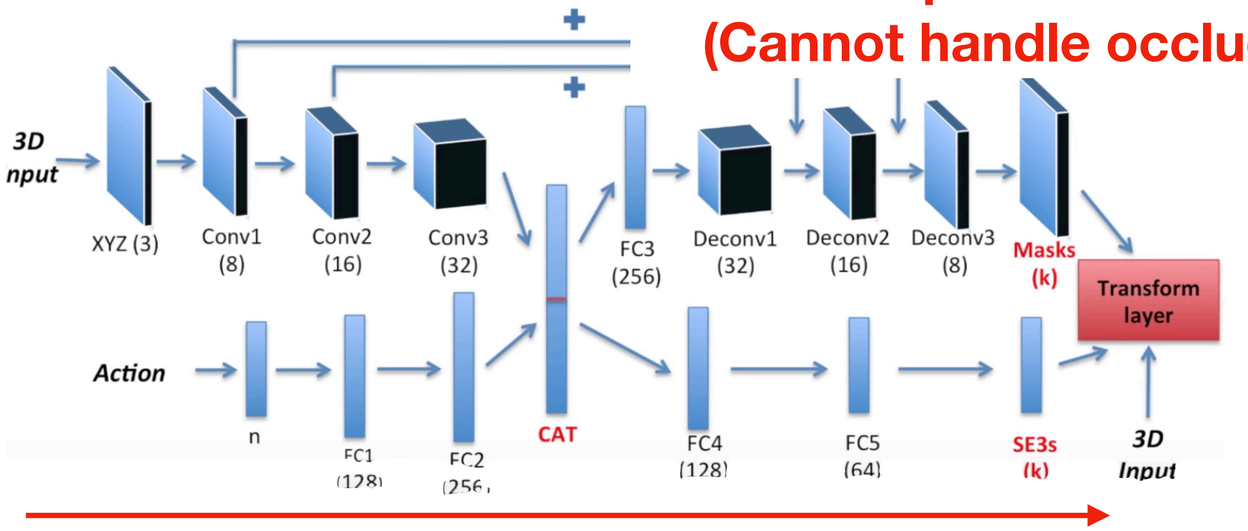
Output point-wise
scene flow
(supervision)

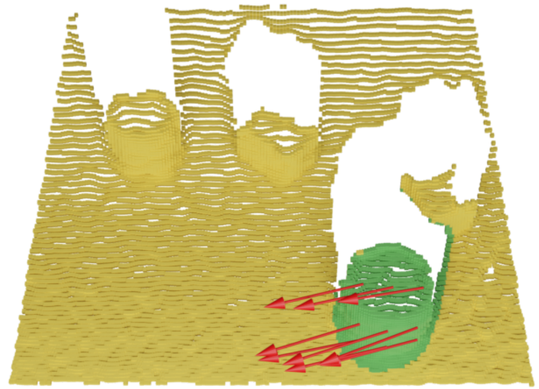
Se3-nets: Learning rigid body motion using deep neural networks
A Byravan, D Fox, ICRA, 2017

SE3-Net - Issues



Motion prediction for visible surface only
(Cannot handle occluded or new surfaces)

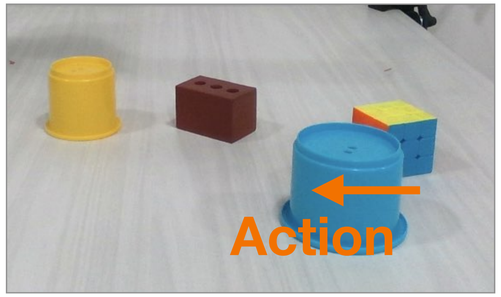




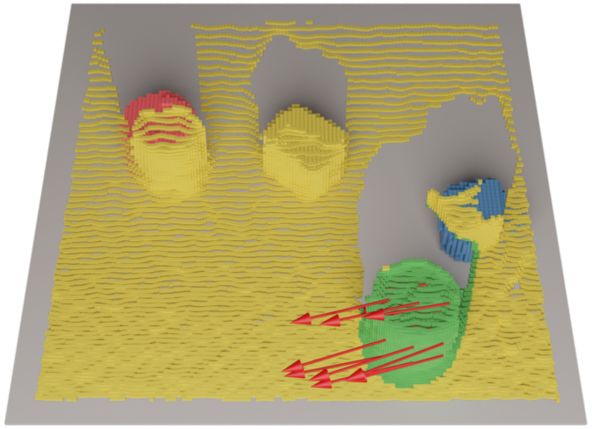
Single step prediction (no history)

1. The mask only describe object that moved in **this** step
2. The representation cannot encode object **permanence** (once the object is occluded it disappears from the representation)
3. The representation cannot **consistently** track object identity over time

Dynamic Scene Representation (DSR)

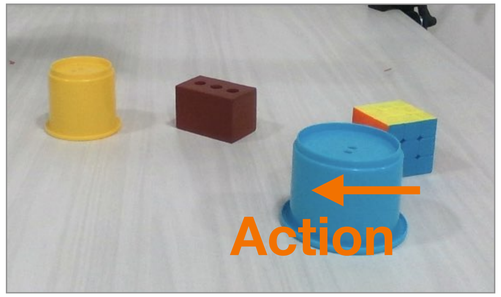


DSR-Net

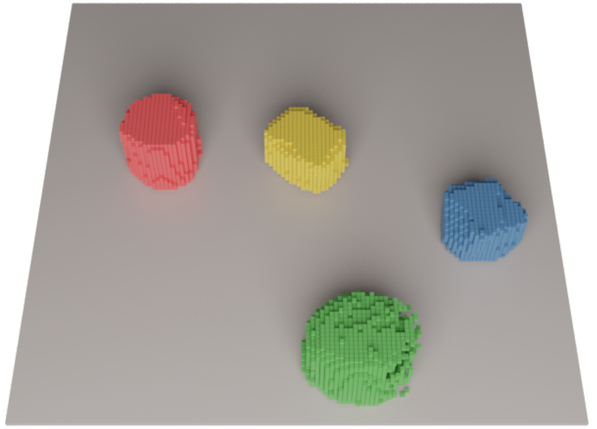


Amodal 3D representation:
Encode objects' complete 3D shape, regardless of occlusion

Dynamic Scene Representation (DSR)



DSR-Net

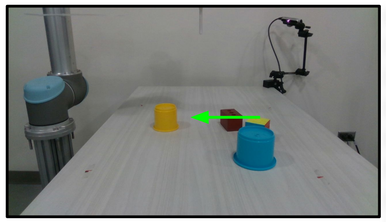


Warped Scene using Motion

history aggregation

✓ Multiple object ✓ Object Permanence ✓ Continuity

DSR-Net



Action

ActionEncoder

MotionPred.

SE(3)
Transformations

Scene Flow F_t

Observation (TSDF)

SceneEncoder

Scene Rep S_t

MaskPred.

Instance Mask M_t

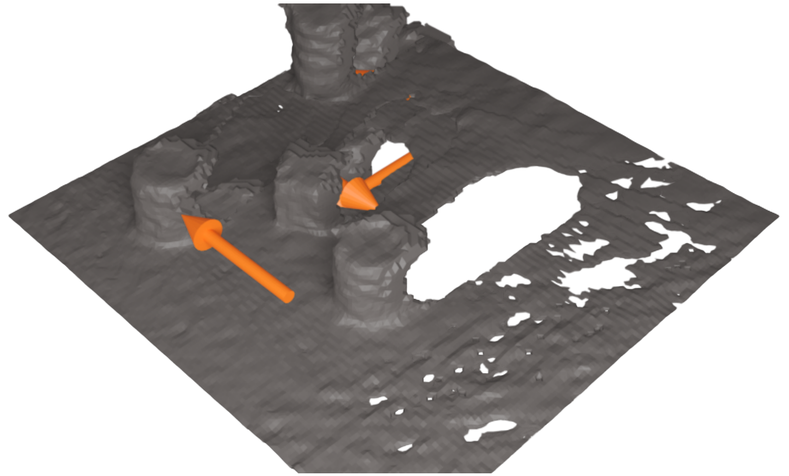
Forward
Warping

History Aggregation

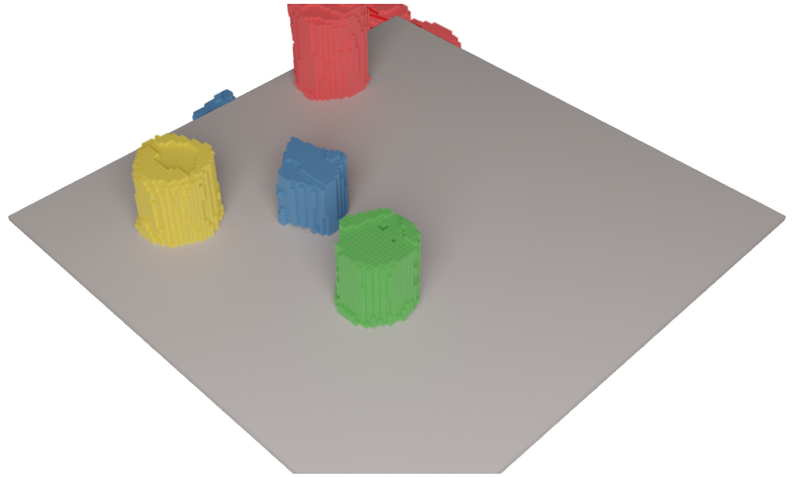
Warped Representation S'_t

DSR-Net in Action

Real-World **Novel** object



Depth Observation + Action



Mask and Motion Prediction

Evaluation

We want to see whether DSR-Net is able to

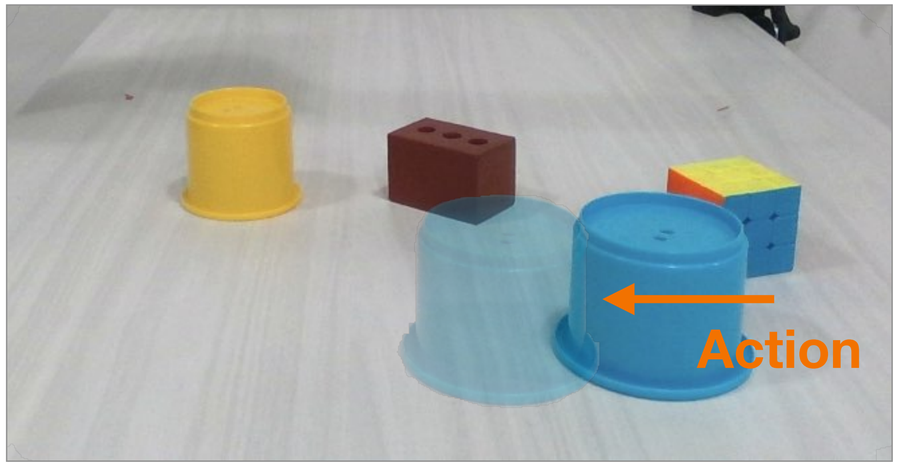
1. Accurately predict object motion under different robot interactions;
2. Aggregate the history and encodes object permanence and continuity;
3. Improve the performance of down-stream manipulation tasks.

Evaluation

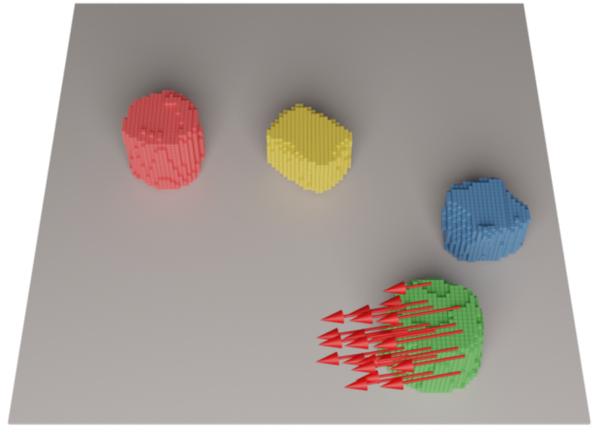
We want to see whether DSR-Net is able to

1. Accurately predict object motion under different robot interactions;
2. Aggregate the history and encodes object permanence and continuity;
3. Improve the performance of down-stream tasks.

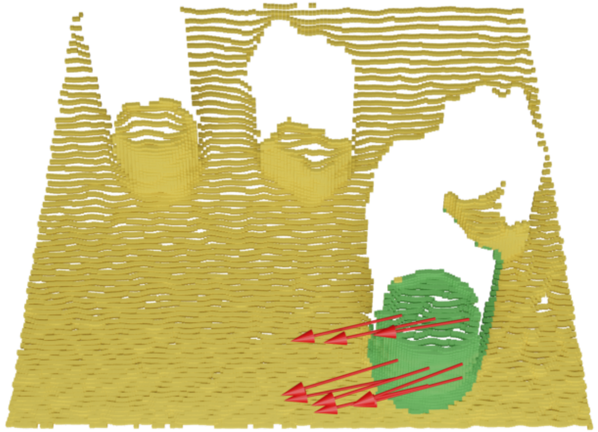
Result - Motion Prediction



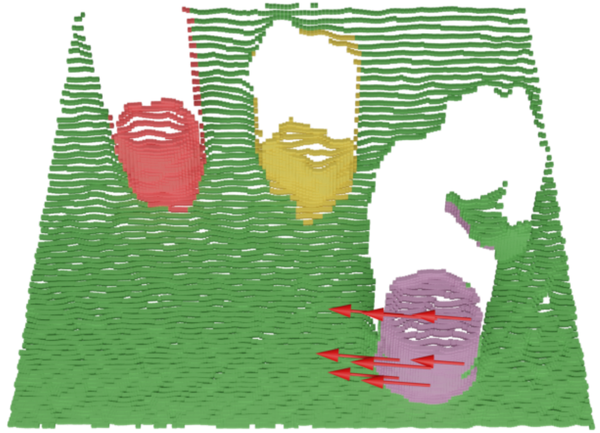
Color image



DSR-Net

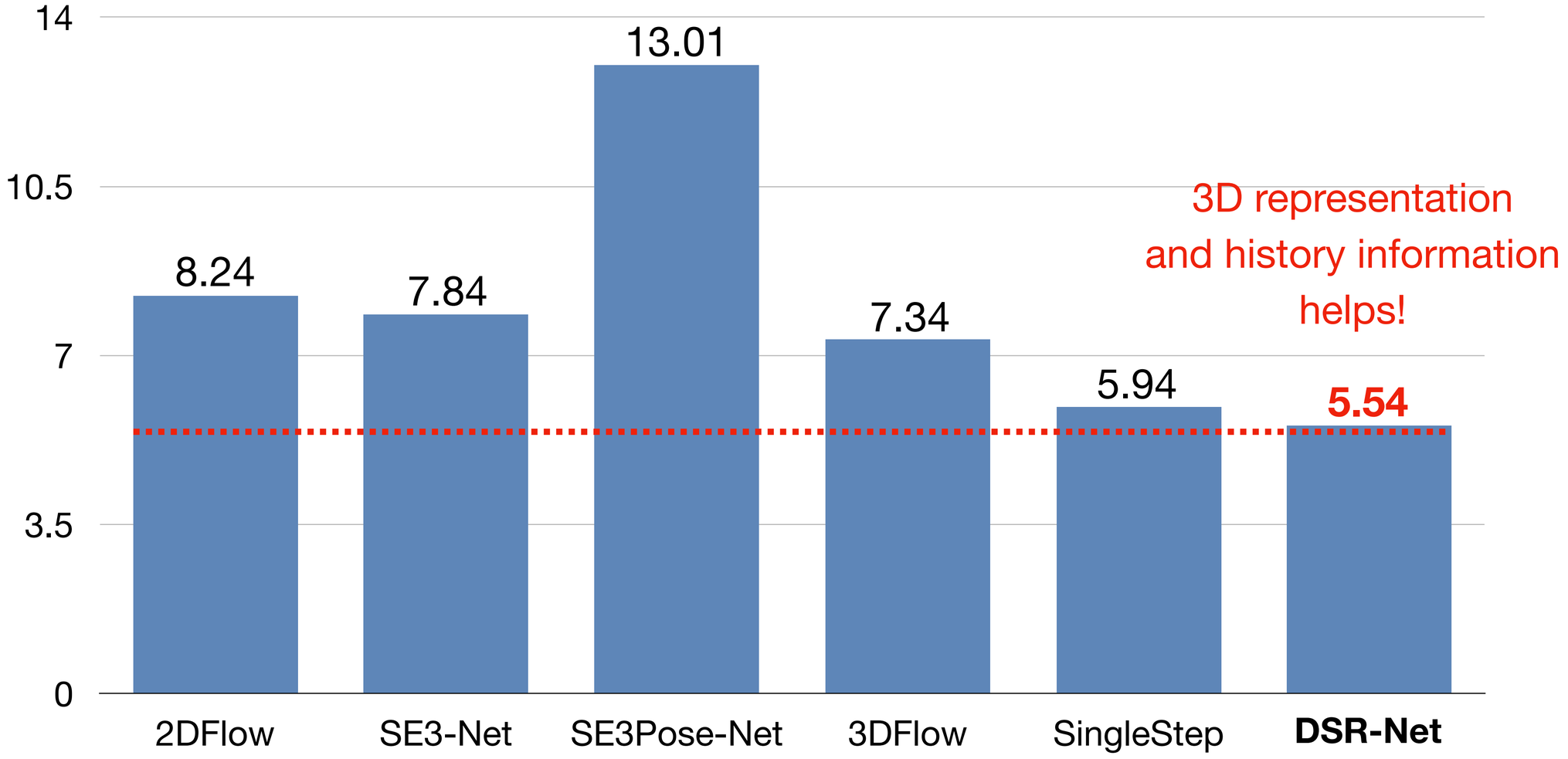


SE3-Net



SE3Pose-Net

Result - Motion Prediction



MSE of scene flow prediction on **visible** surface

Evaluation

We want to see whether DSR-Net is able to

1. Accurately predict object motion under different robot interactions;
2. Aggregate the history and encodes object permanence and continuity;
3. Improve the performance of down-stream tasks.

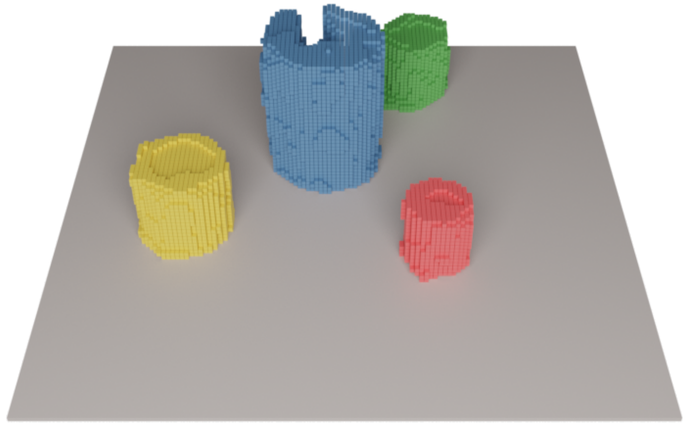
Object Permanence

Object Permanence: is the understanding that objects continue to exist even if they disappear from view due to occlusion.

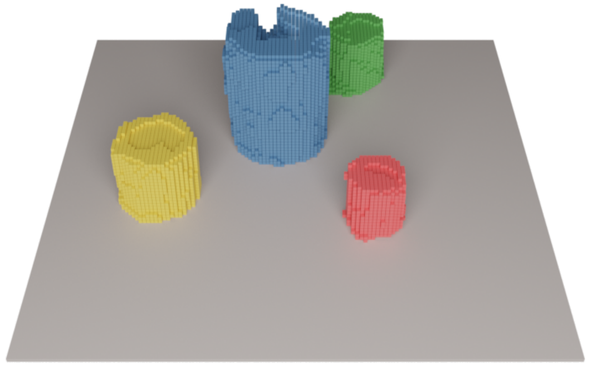
Object Permanence Result (Real)

Step 1

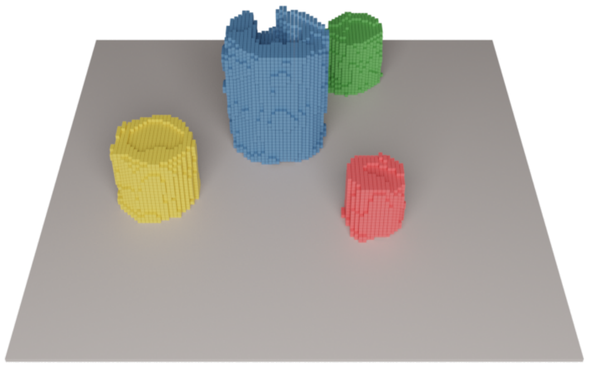
Camera View



DSR



NoWarp

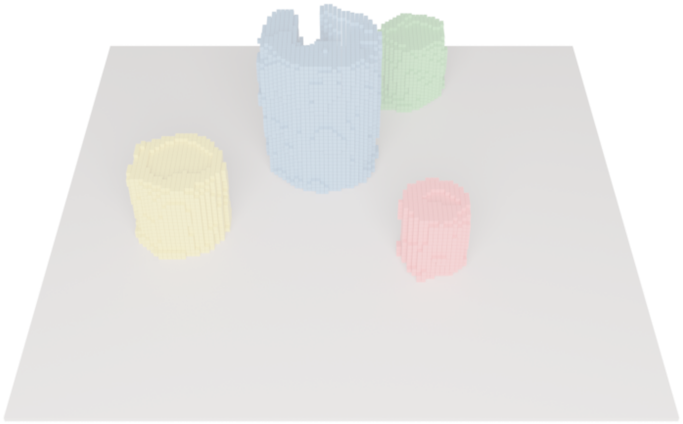


Single

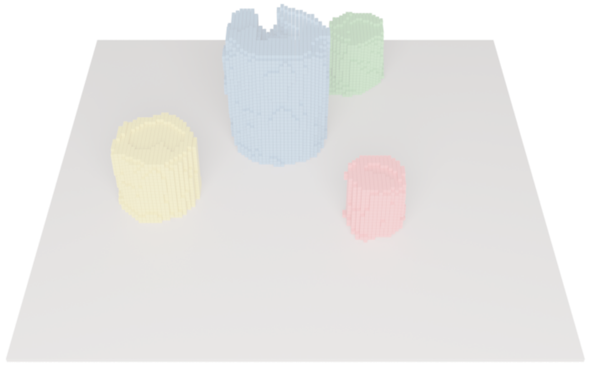
Object Permanence Result (Real)

Step 1

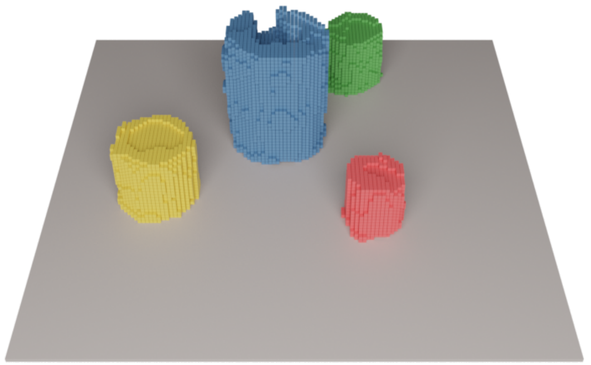
Camera View



DSR



NoWarp

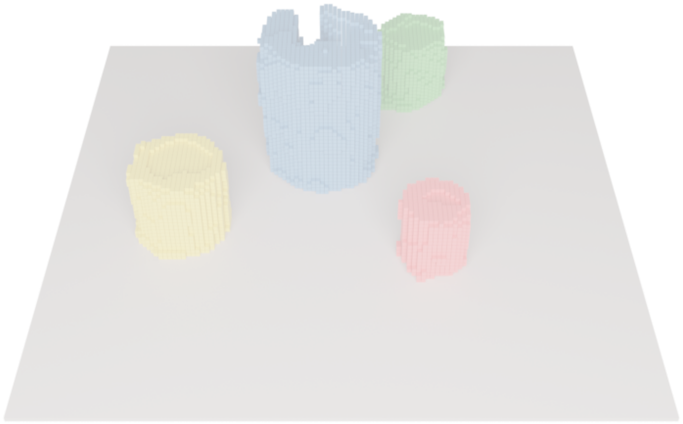


Single

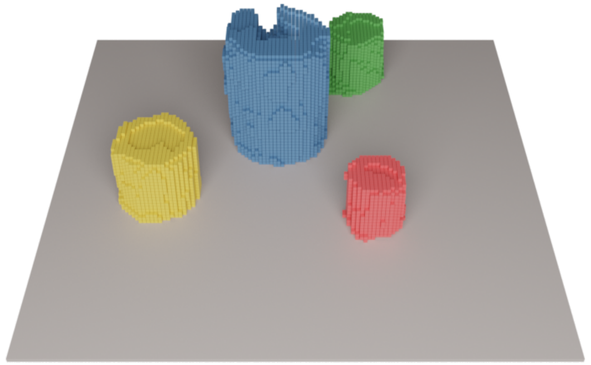
Object Permanence Result (Real)

Step 1

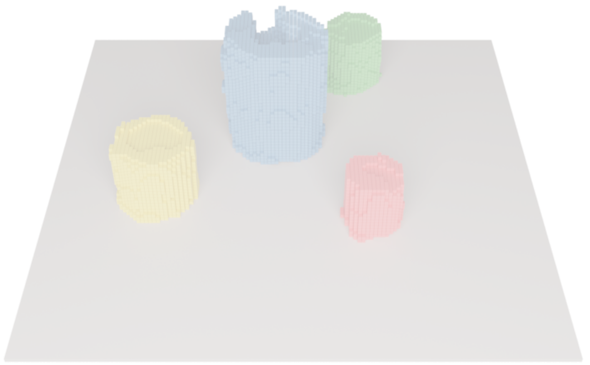
Camera View



DSR



NoWarp

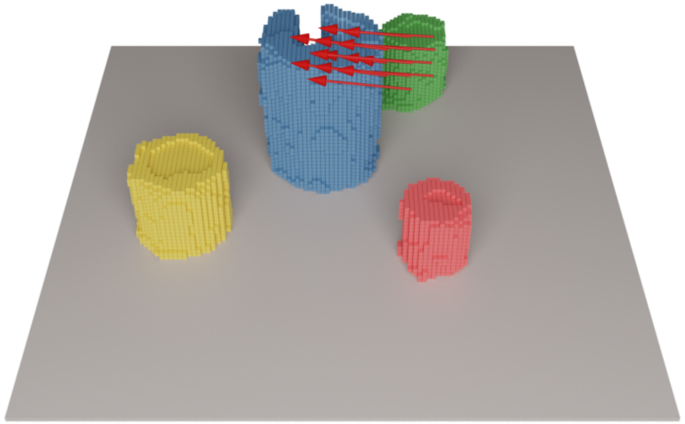


Single

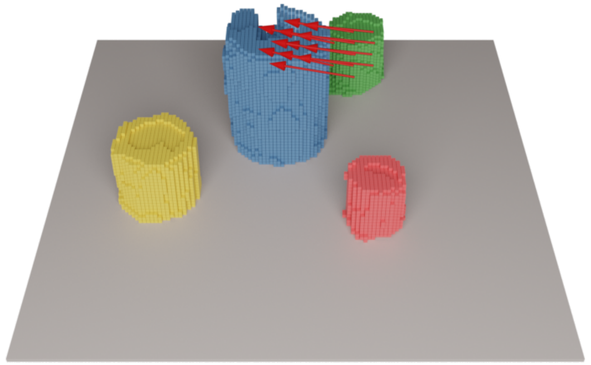
Object Permanence Result (Real)

Step 1

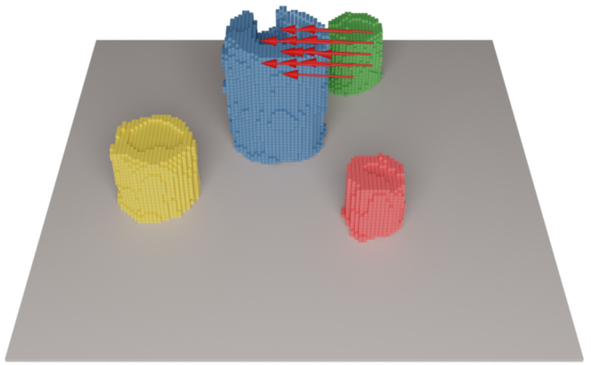
Camera View



DSR



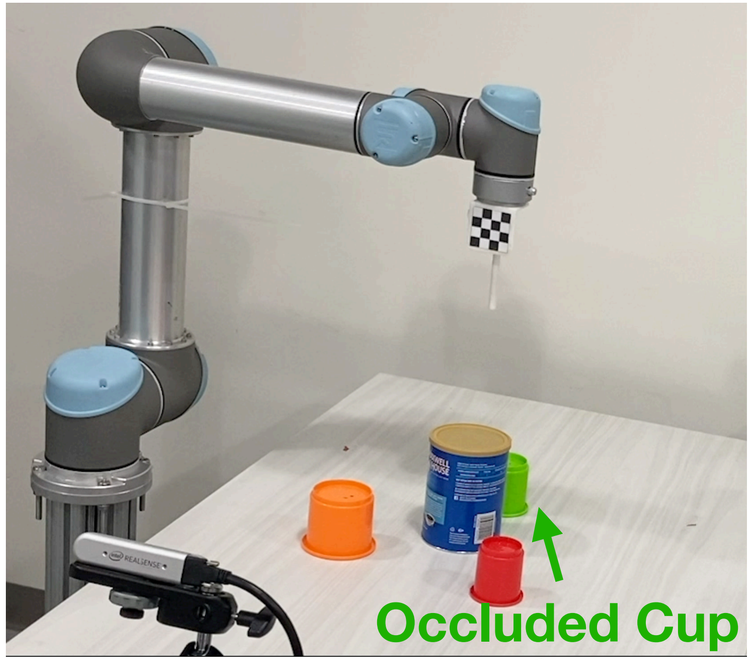
NoWarp



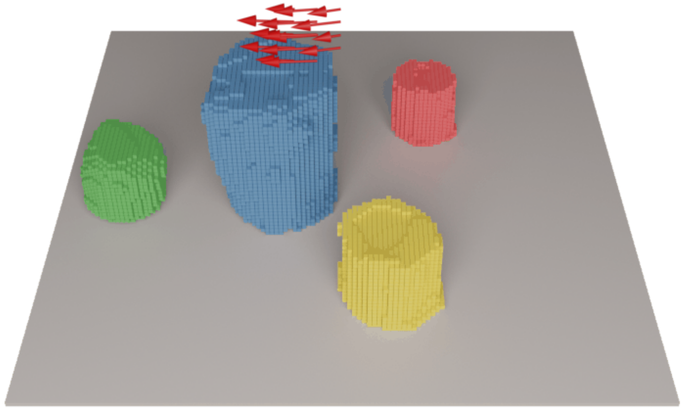
Single

Object Permanence Result (Real)

Step 2

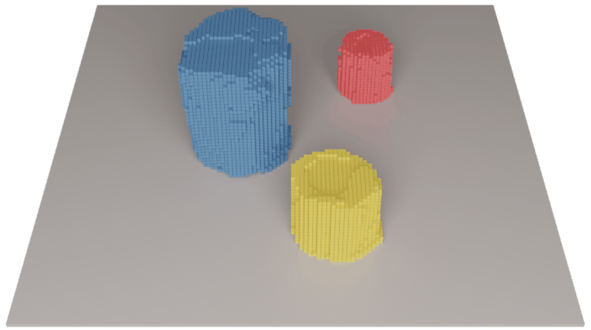


Camera View

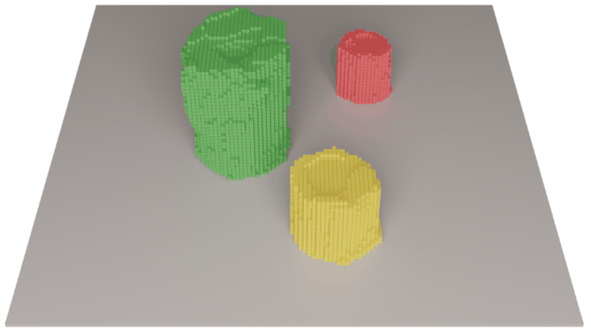


DSR

No Motion Prediction



NoWarp



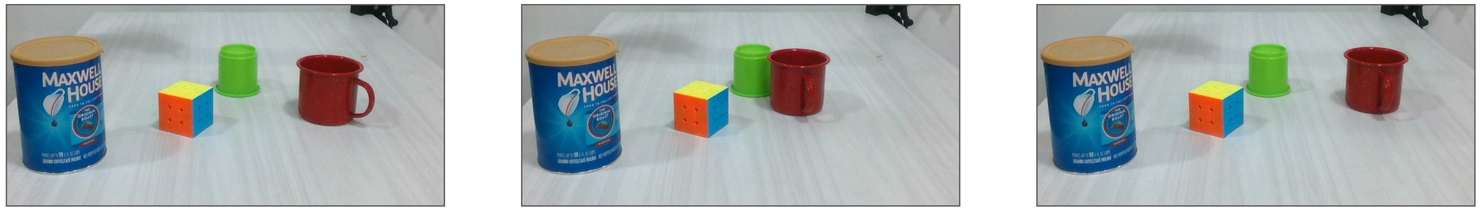
Single

Object Continuity

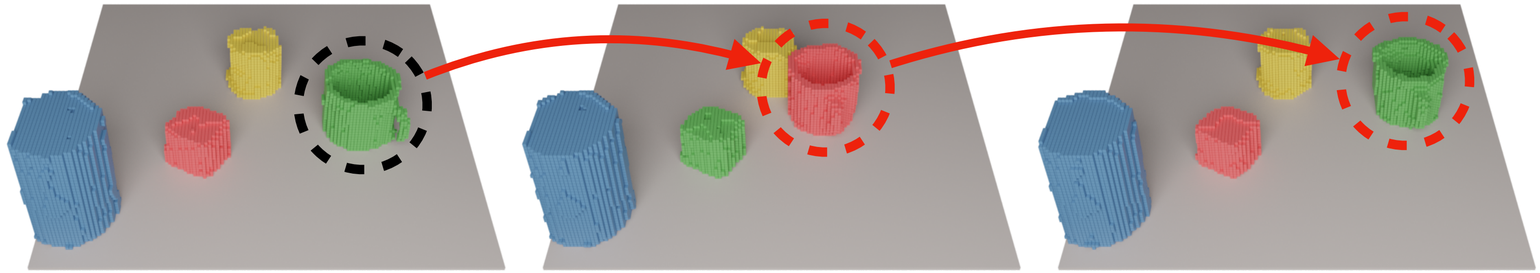
Object Continuity: Representation can recognize individual object instance and track their identity over time.

Object Continuity Result

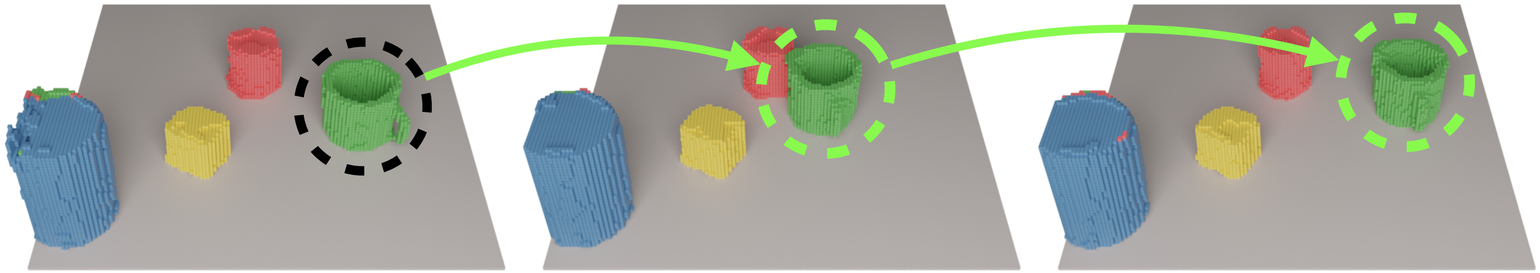
t = 3



Single



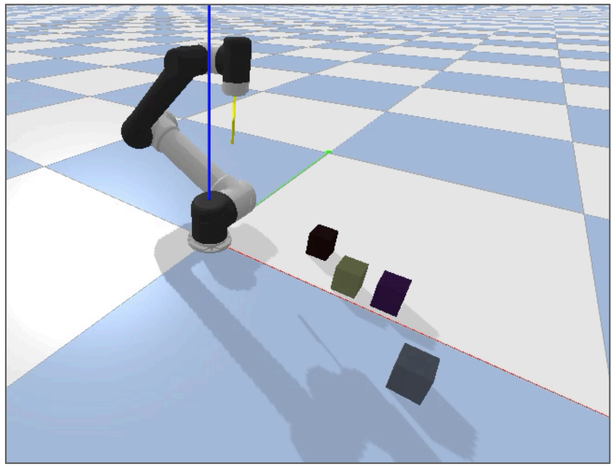
DSR



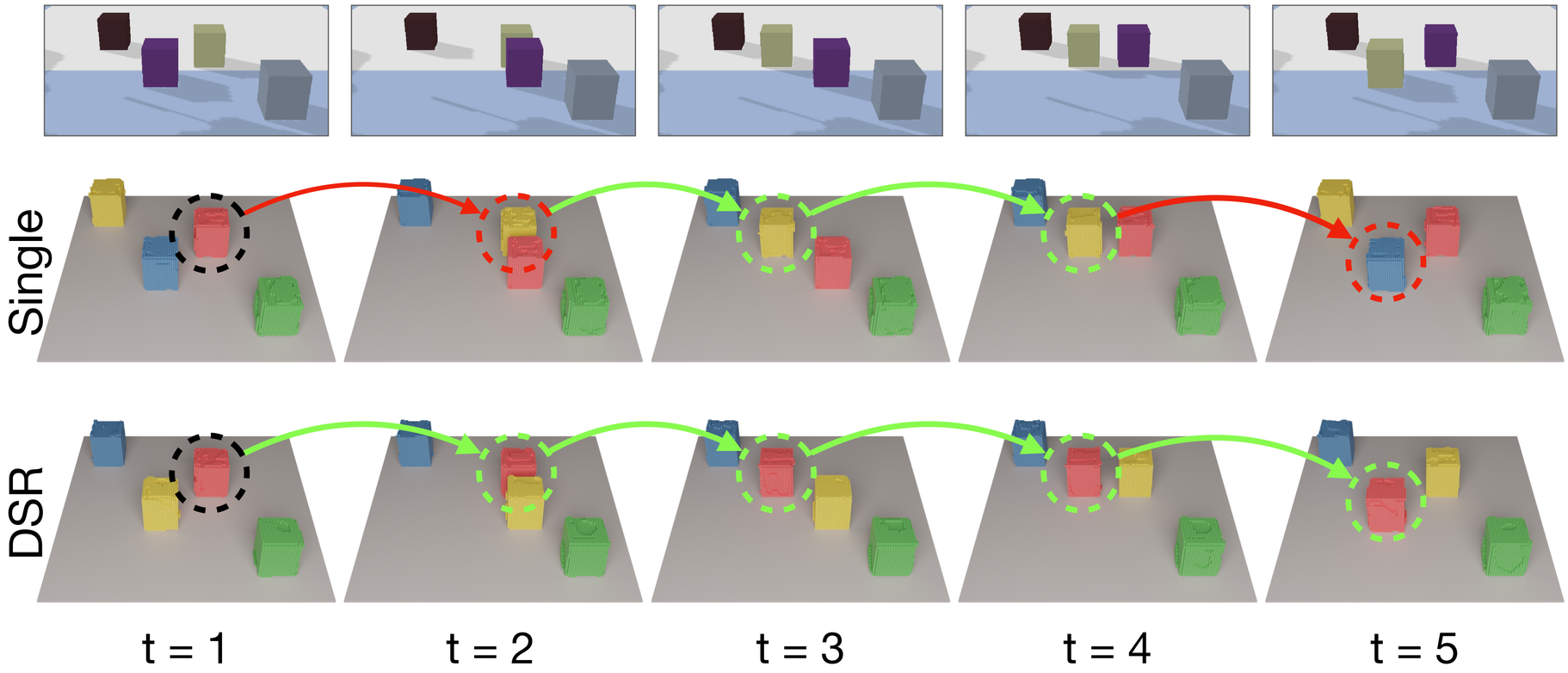
t = 1 t = 2 t = 3

Object Continuity Result

Objects are visually indistinguishable from depth input



t = 3



The continuity achieved by using history aggregation, instead of visual appearance.

Evaluation

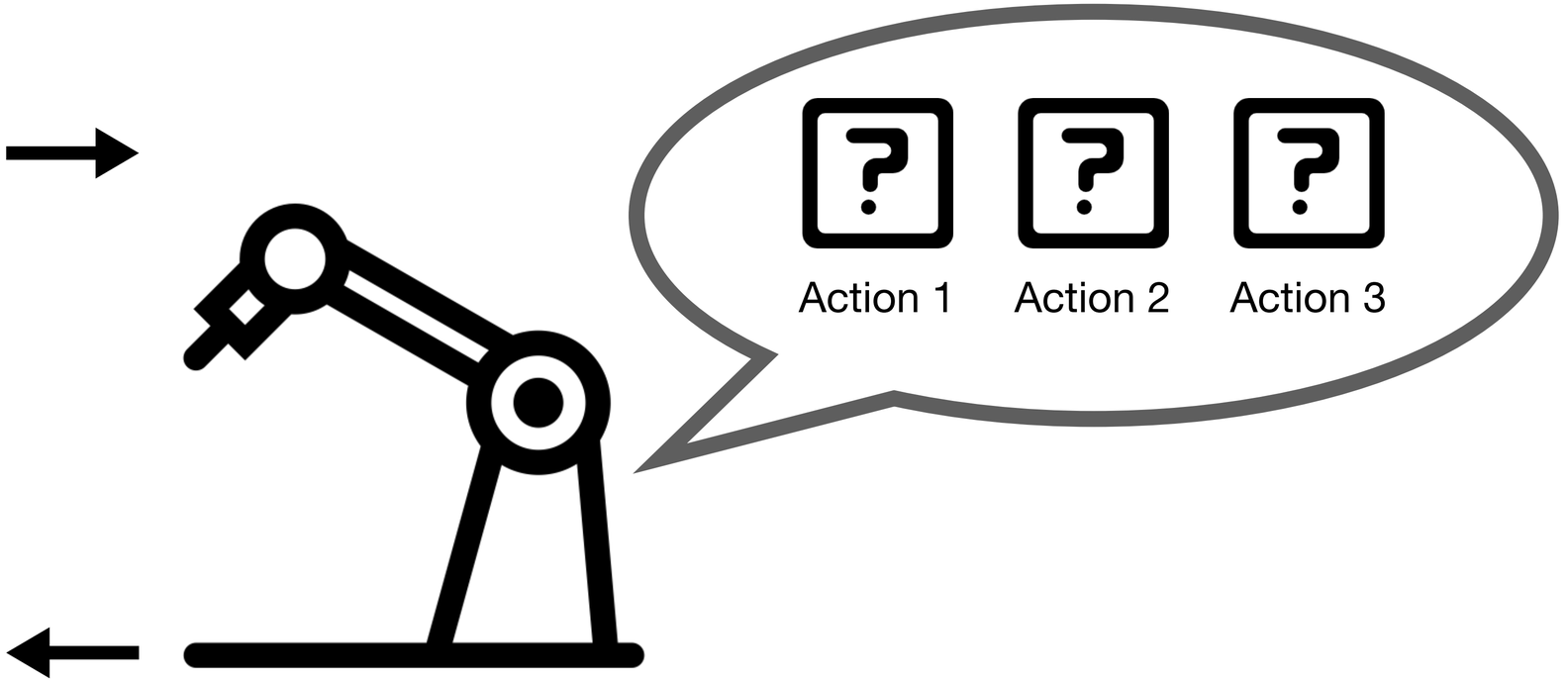
We want to see whether DSR-Net is able to

1. Accurately predict object motion under different robot interactions;
2. Aggregate the history and encodes object permanence and continuity;
3. Improve the performance of manipulation tasks.

Robot Manipulation: Planner Pushing

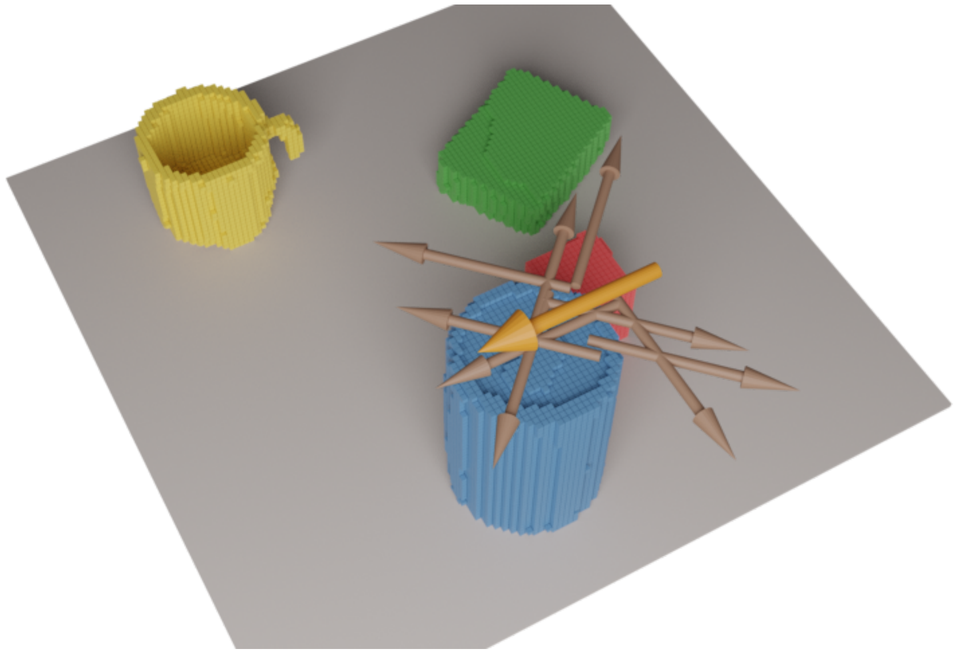
Initial state

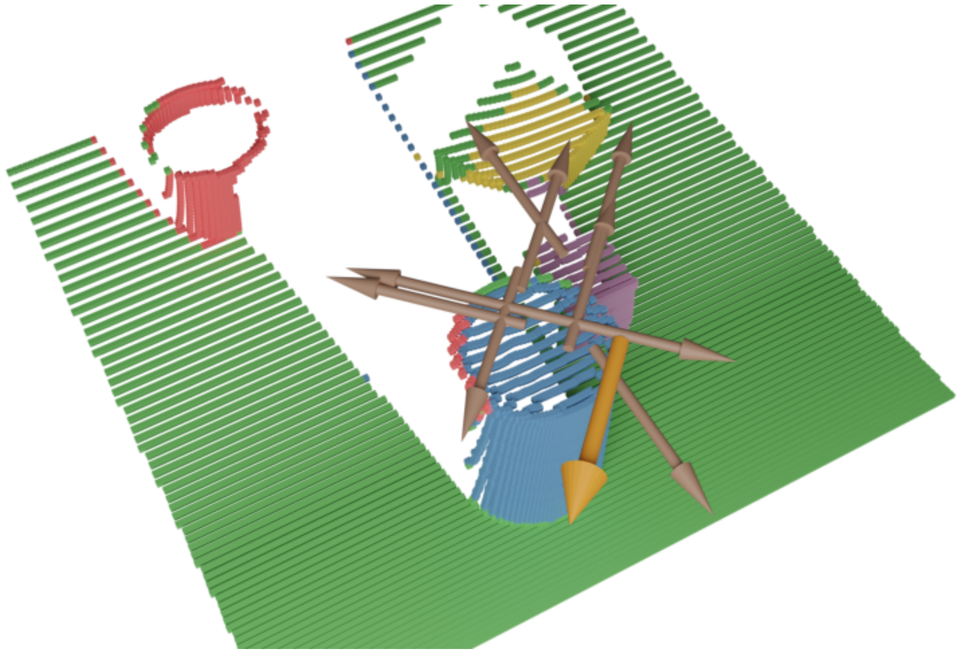
Target state



Robot Manipulation: Planner Pushing

t = 1



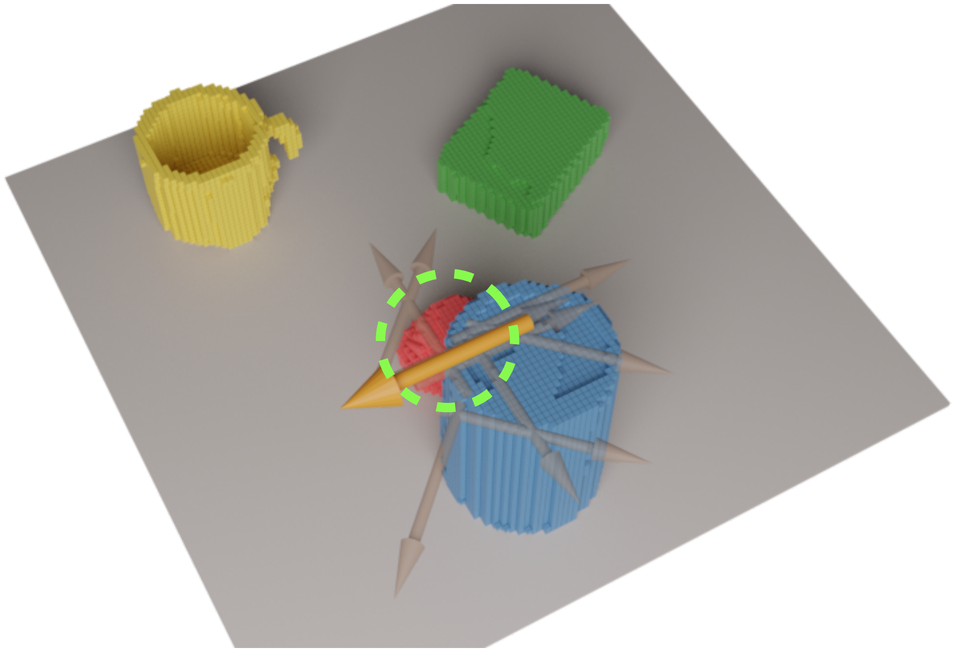


DSR-Net

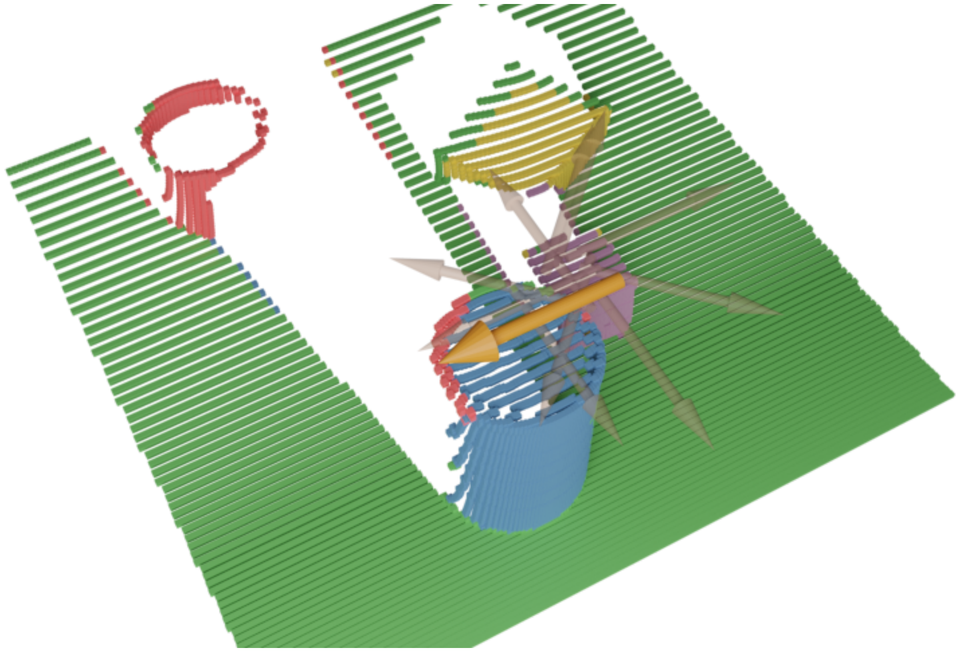
SE3Pose-Net

Robot Manipulation: Planner Pushing

t = 2



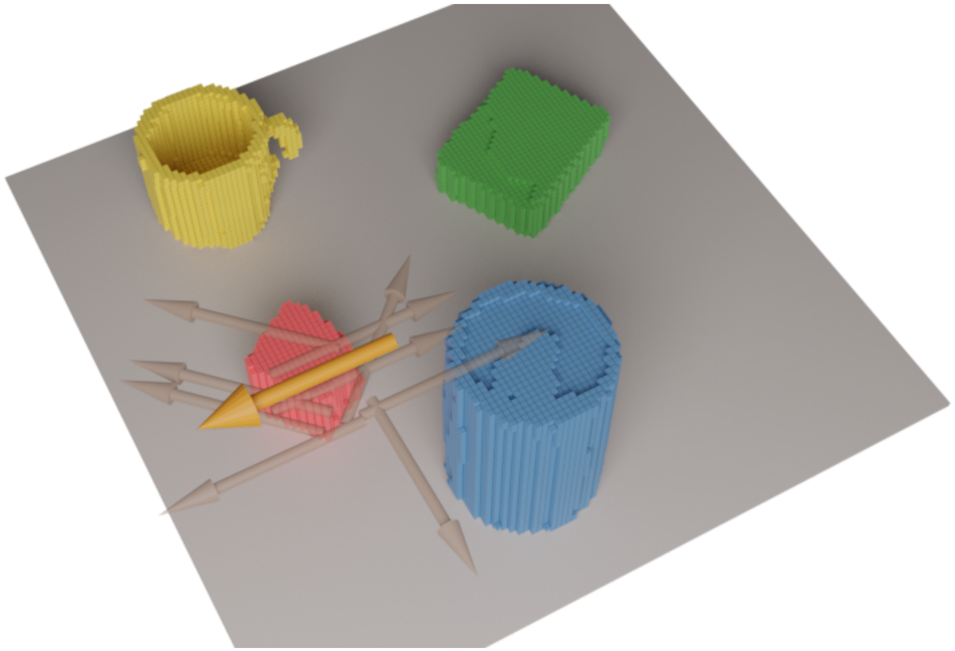
DSR-Net



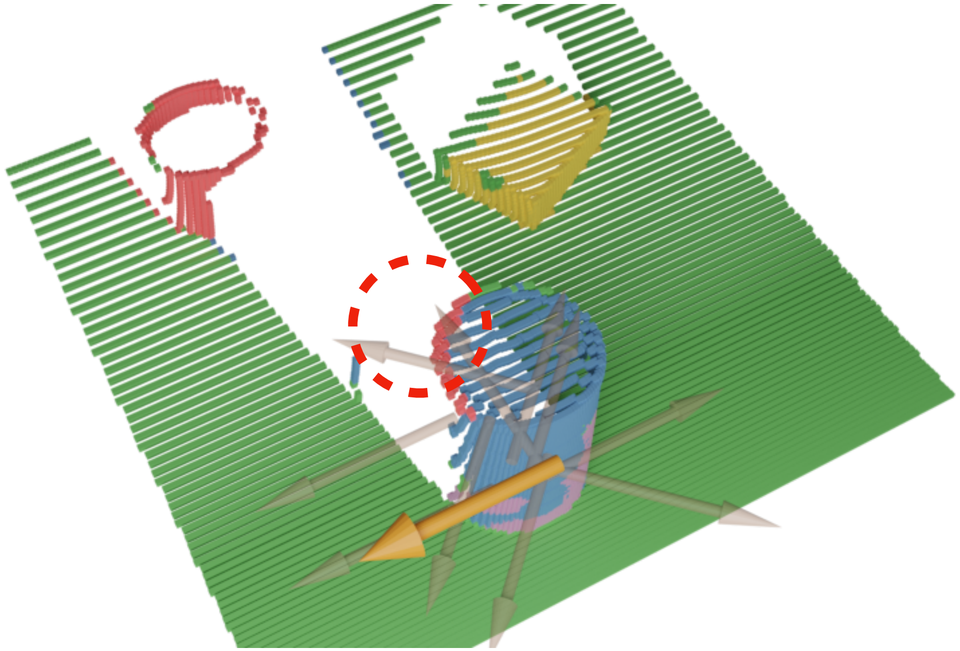
SE3Pose-Net

Robot Manipulation: Planner Pushing

t = 3



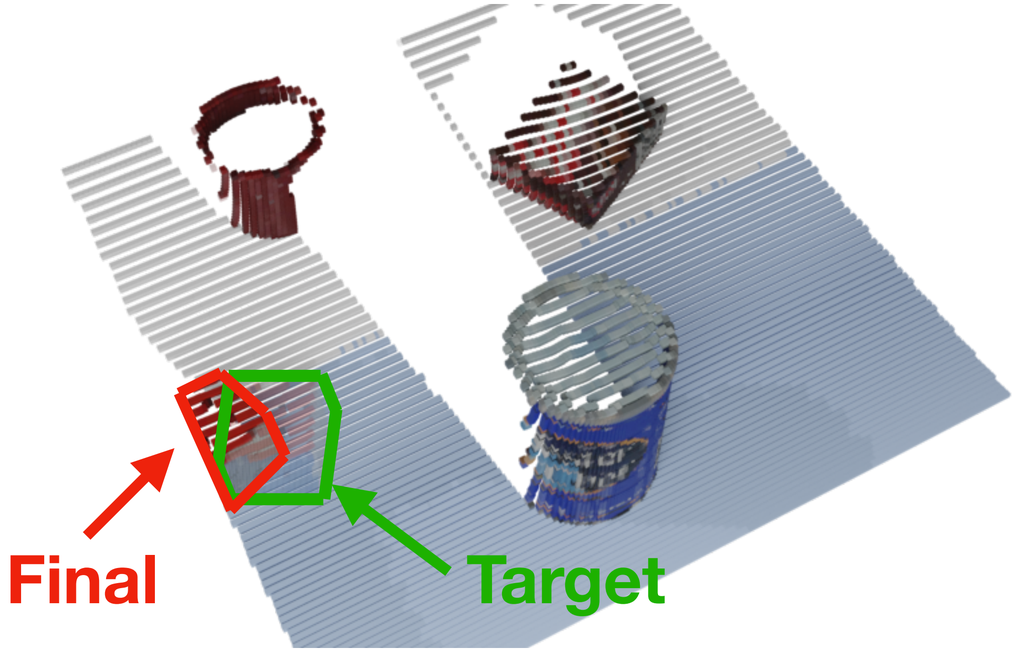
DSR-Net



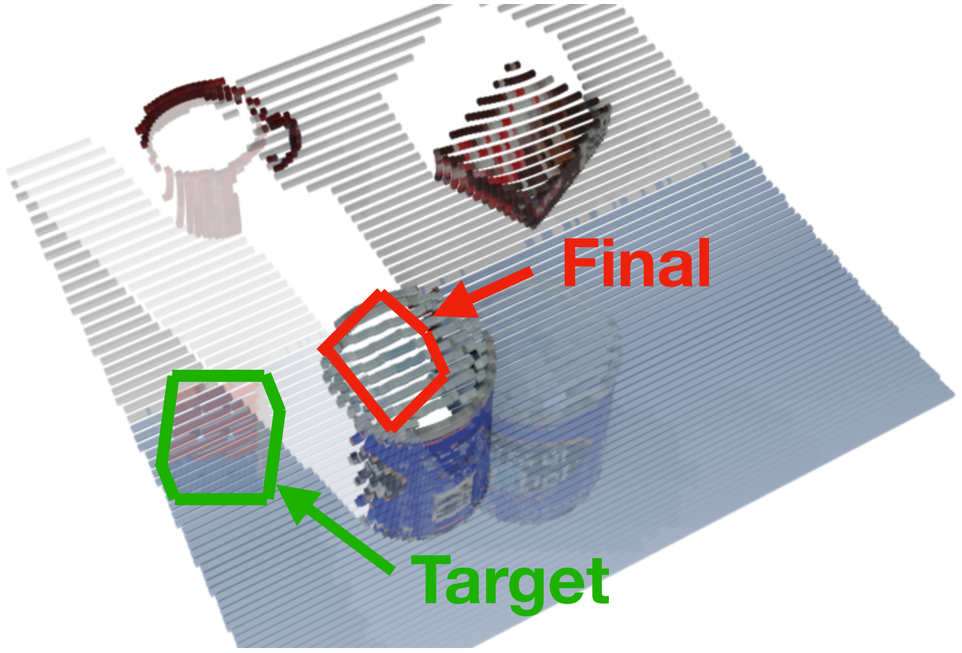
SE3Pose-Net

Robot Manipulation: Planner Pushing

Final State Comparison

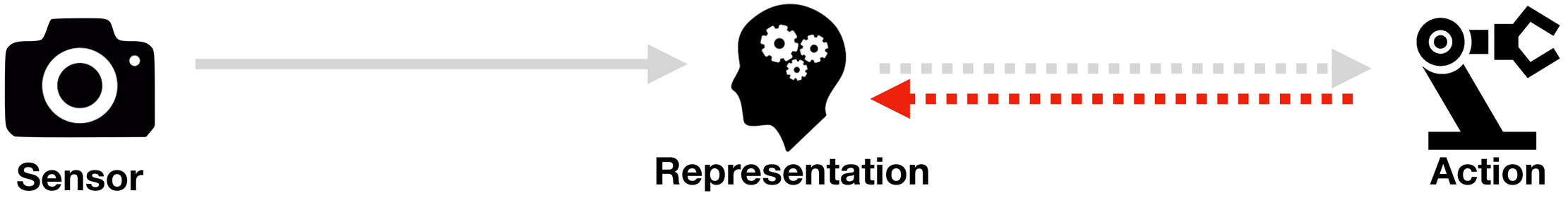


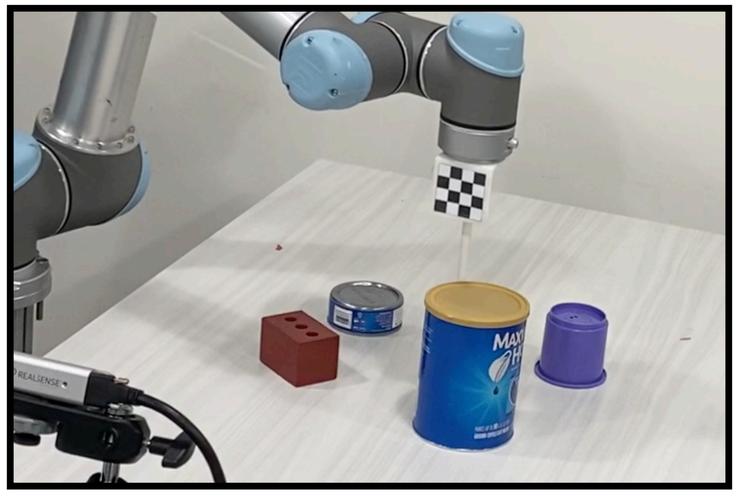
DSR-Net



SE3Pose-Net

Active Scene Understanding





Code + Data
https://dsr-net.cs.columbia.edu/

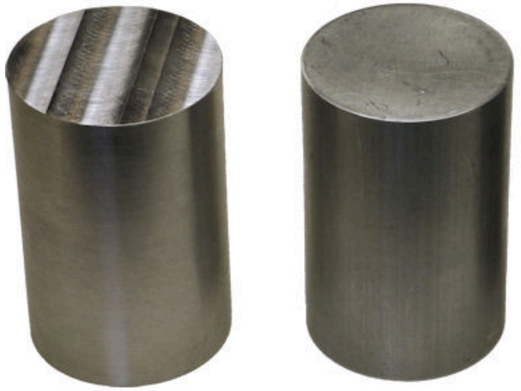
Dynamic Scene Representation:
Better 3D scene representation describes object instances, amodel 3D geometry, and their motion under interaction.

How about other object properties?

Mass? Friction? Other physical properties?

Why it is hard?

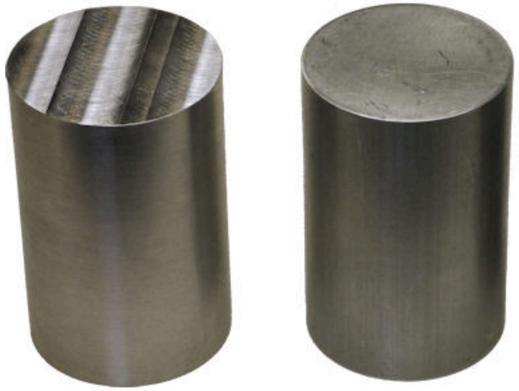
To learn physical properties though vision?



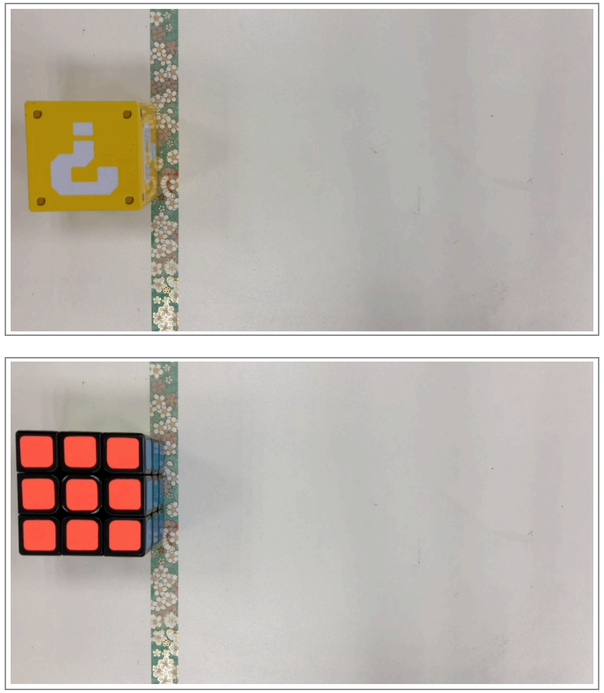
Cannot be inferred from appearance alone

Why it is hard?

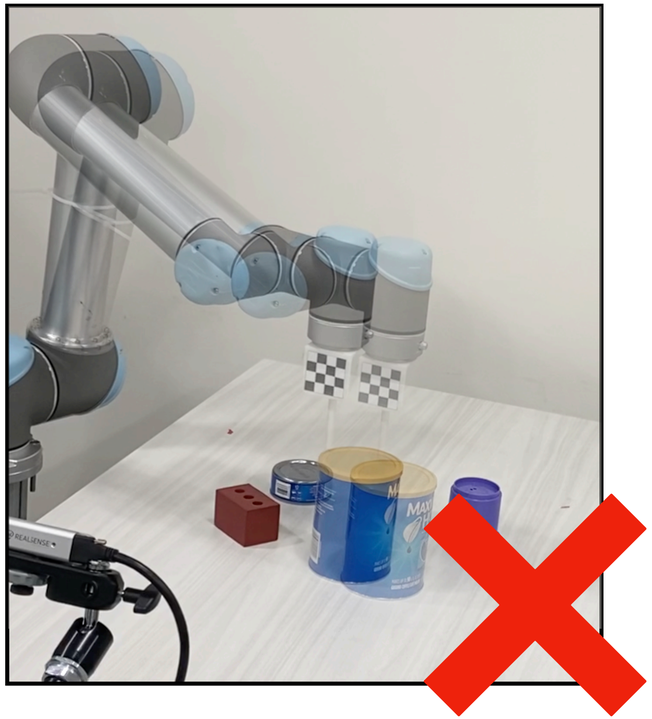
To learn physical properties though vision?



Cannot be inferred from appearance alone

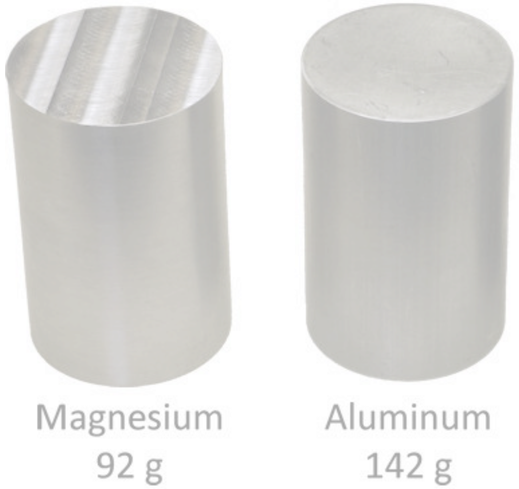


Not salient under quasi-static interactions

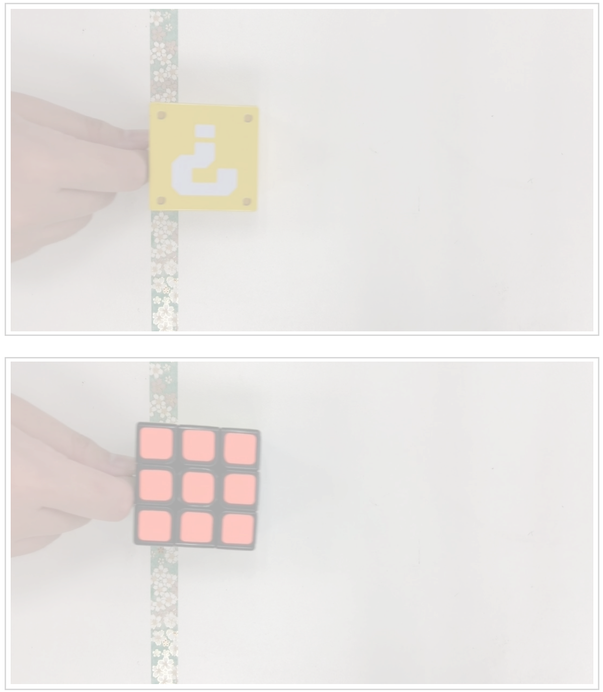


Interactions used in DSR (quasi-static pushing) is not enough

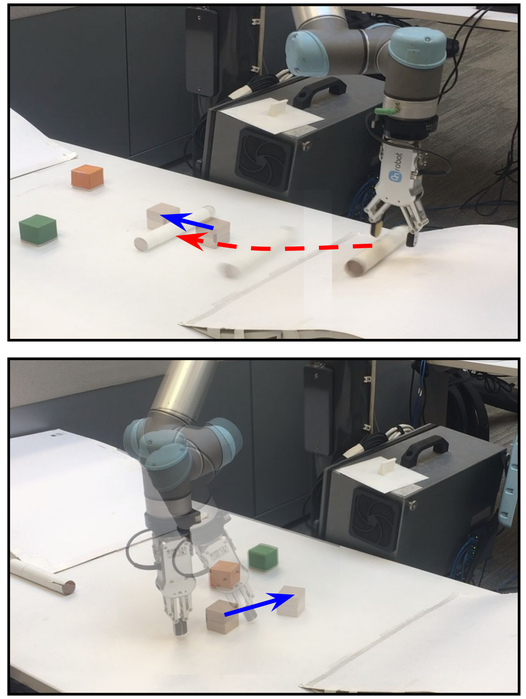
Why it is hard?



Cannot be inferred from appearance alone

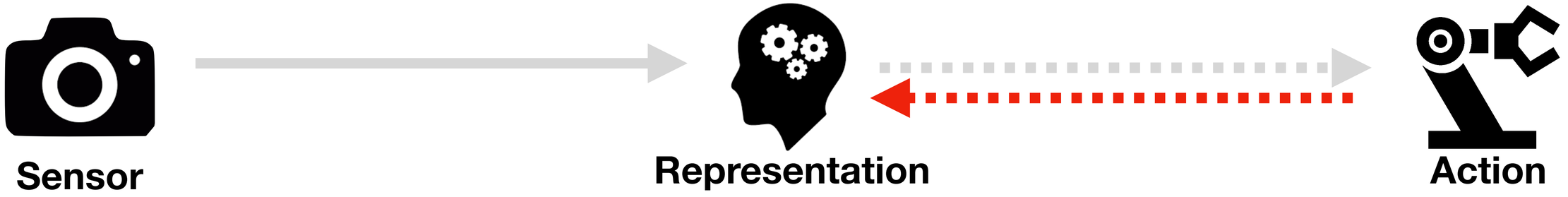


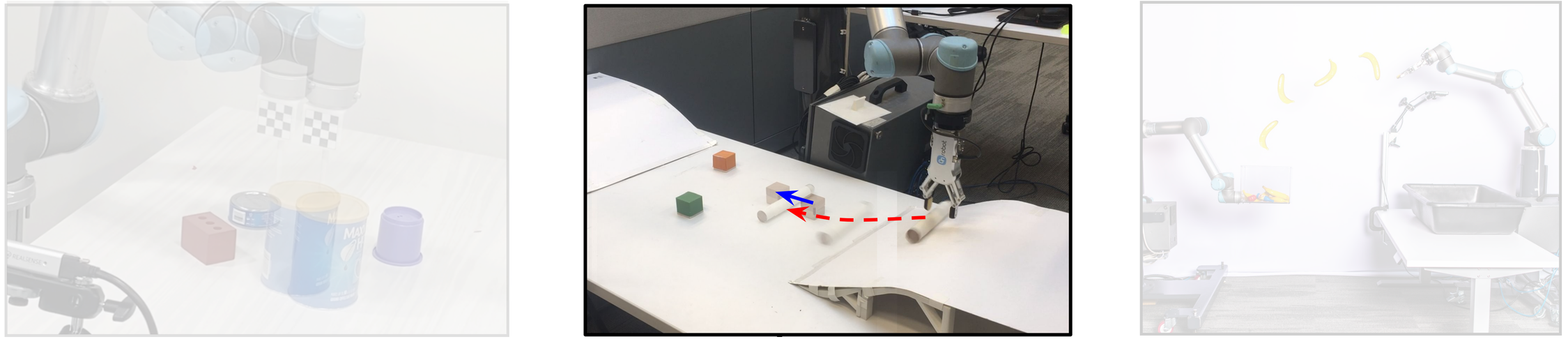
Not salient under quasi-static interactions



Need multiple interactions to decouple the properties

Active Scene Understanding

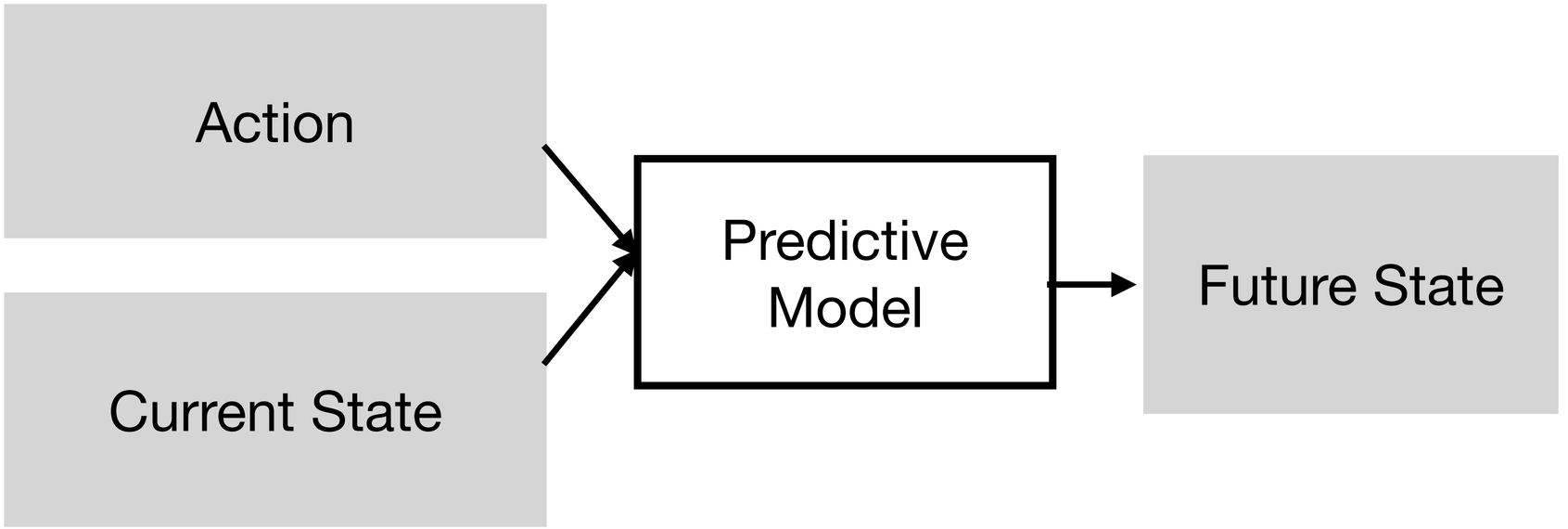




DensePhysNet: Learning Dense Physical Object Representations via Multi-step Dynamic Interactions (RSS2019)

Zhenjia Xu, Jiajun Wu, Andy Zeng, Joshua B. Tenenbaum, Shuran Song

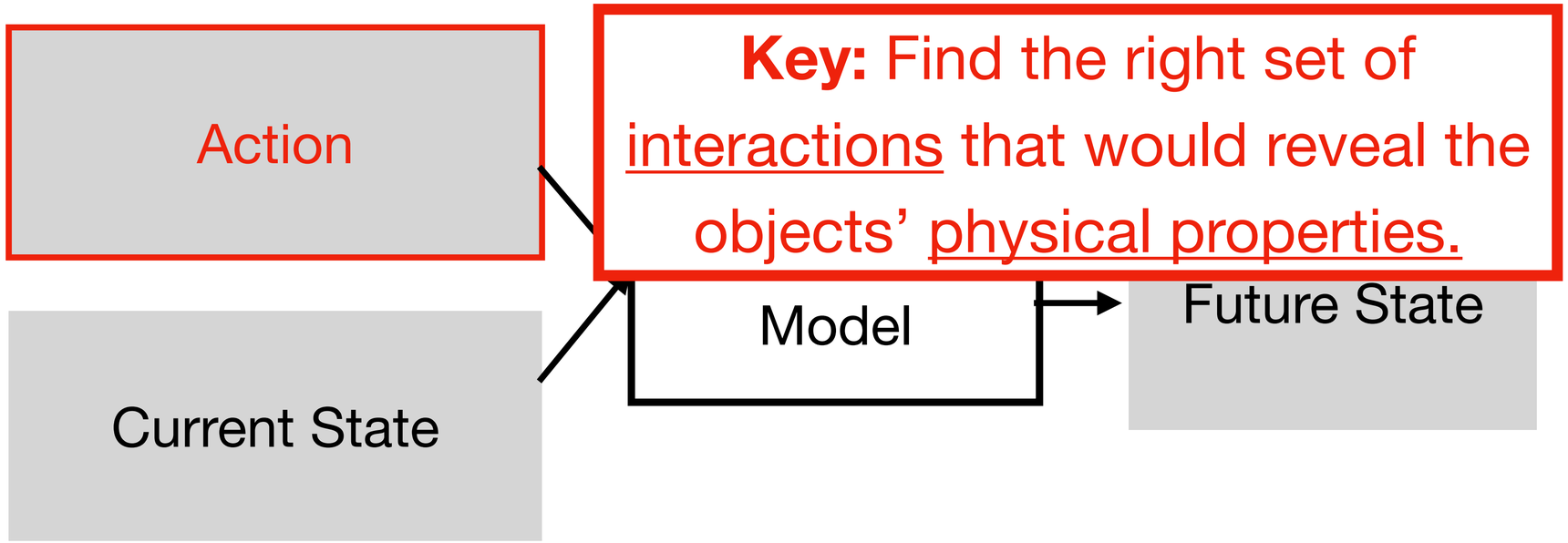
DensePhysNet



Hypothesis:

To accurately predict the future states, the system will need to acquire an implicit understanding of objects' physical properties and how they influence objects' motion.

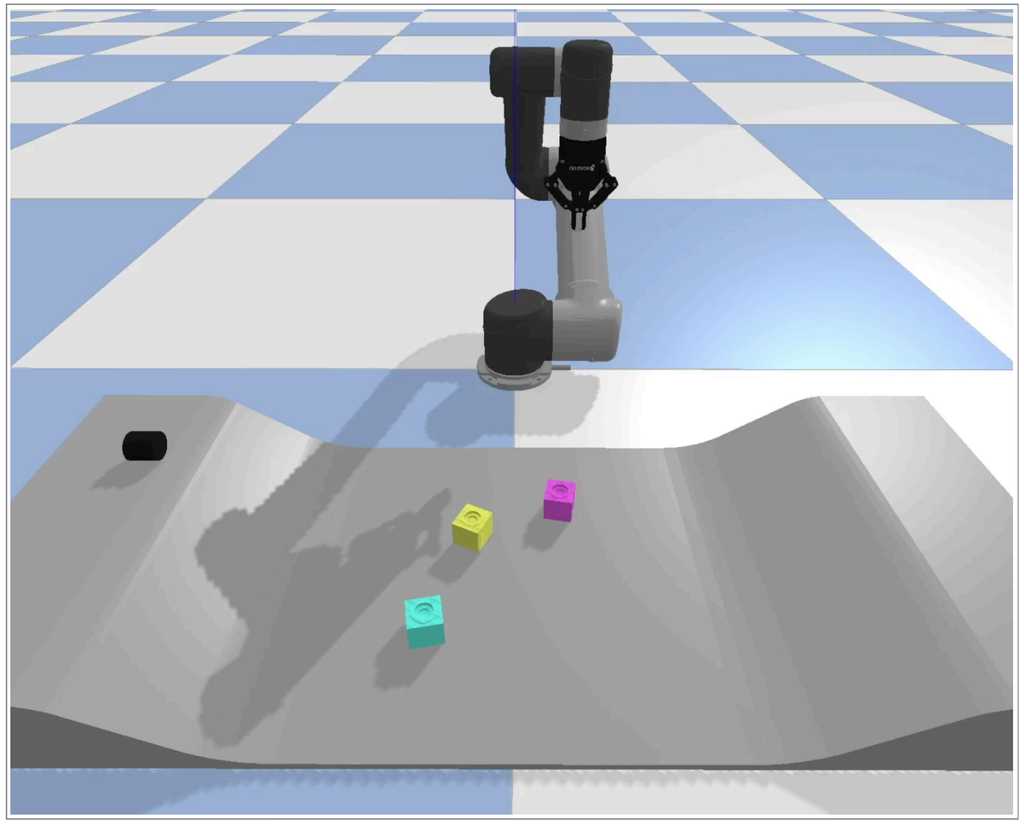
DensePhysNet



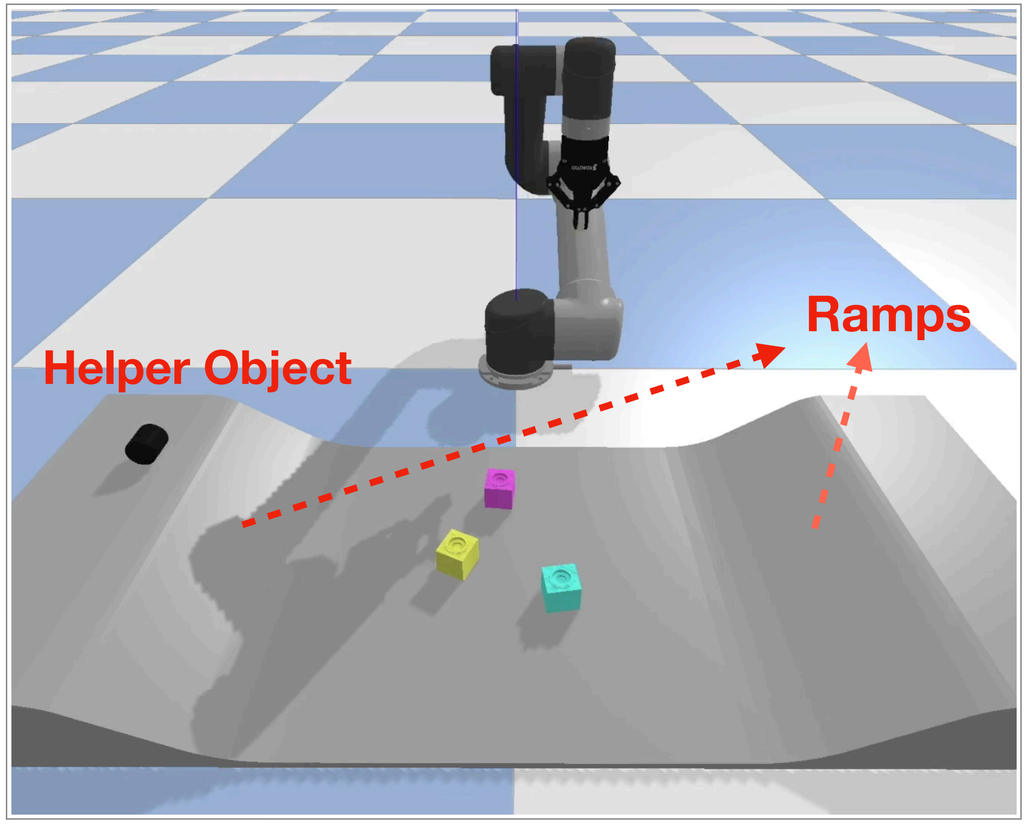
Hypothesis:

To accurately predict the future states, the system will need to acquire an implicit understanding of objects' physical properties and how they influence objects' motion.

Dynamic Interactions

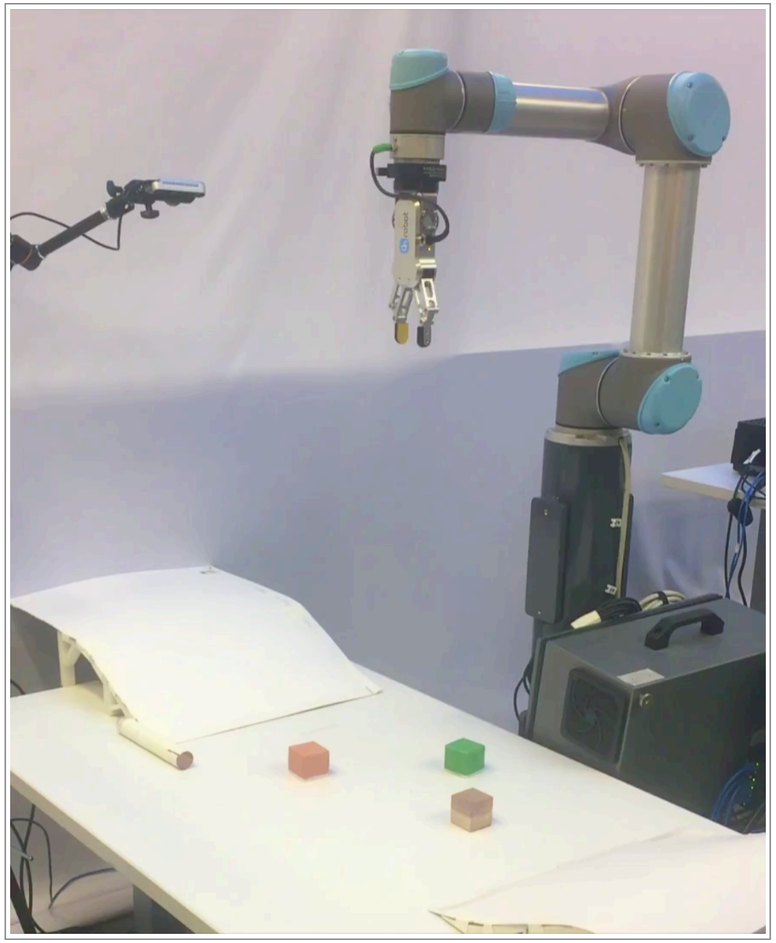


Sliding

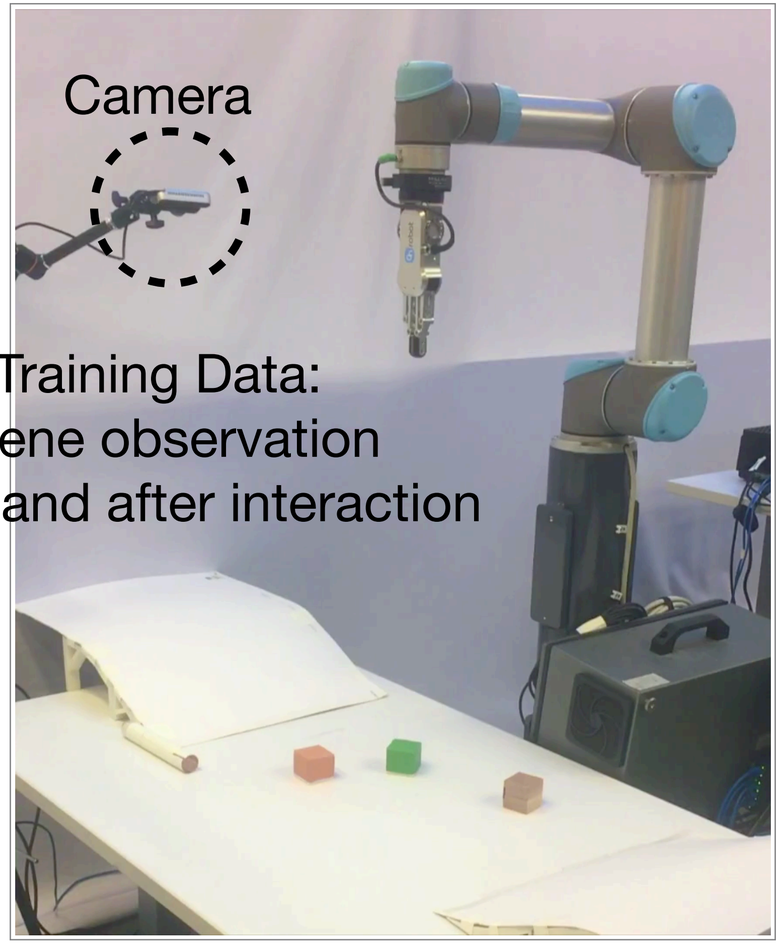


Collision

Dynamic Interactions

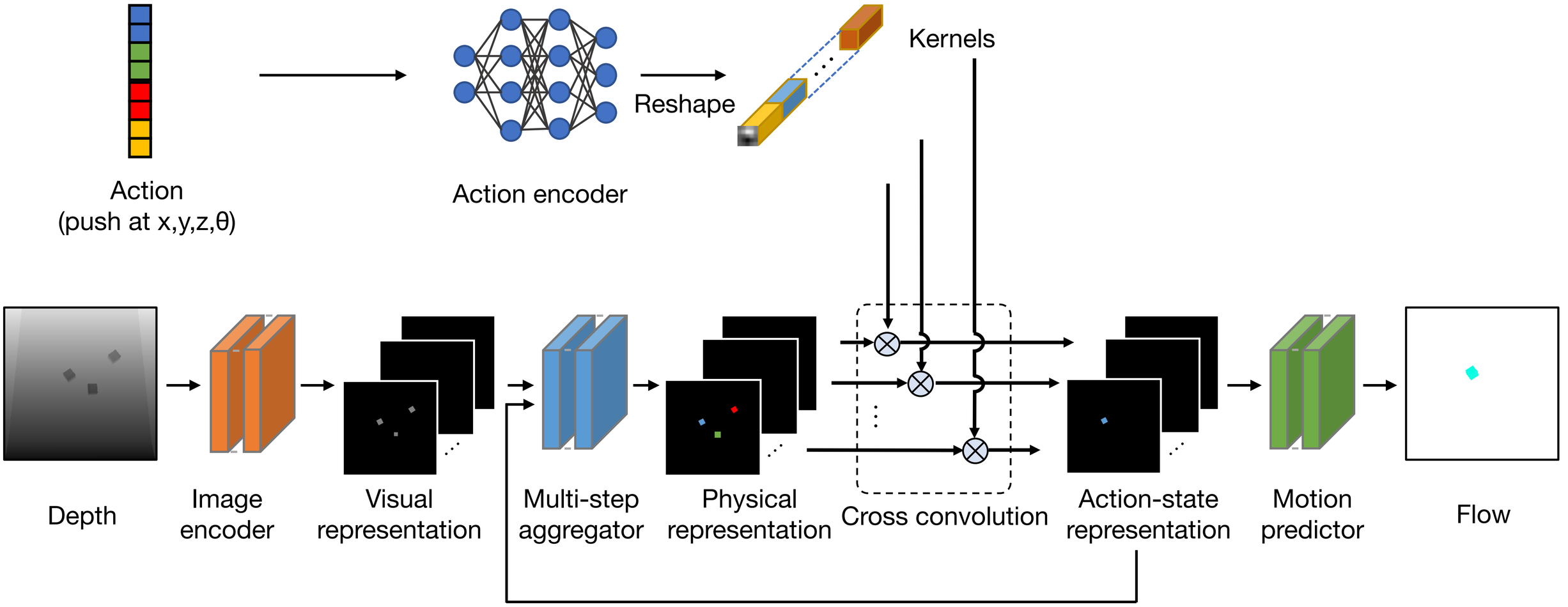


Sliding

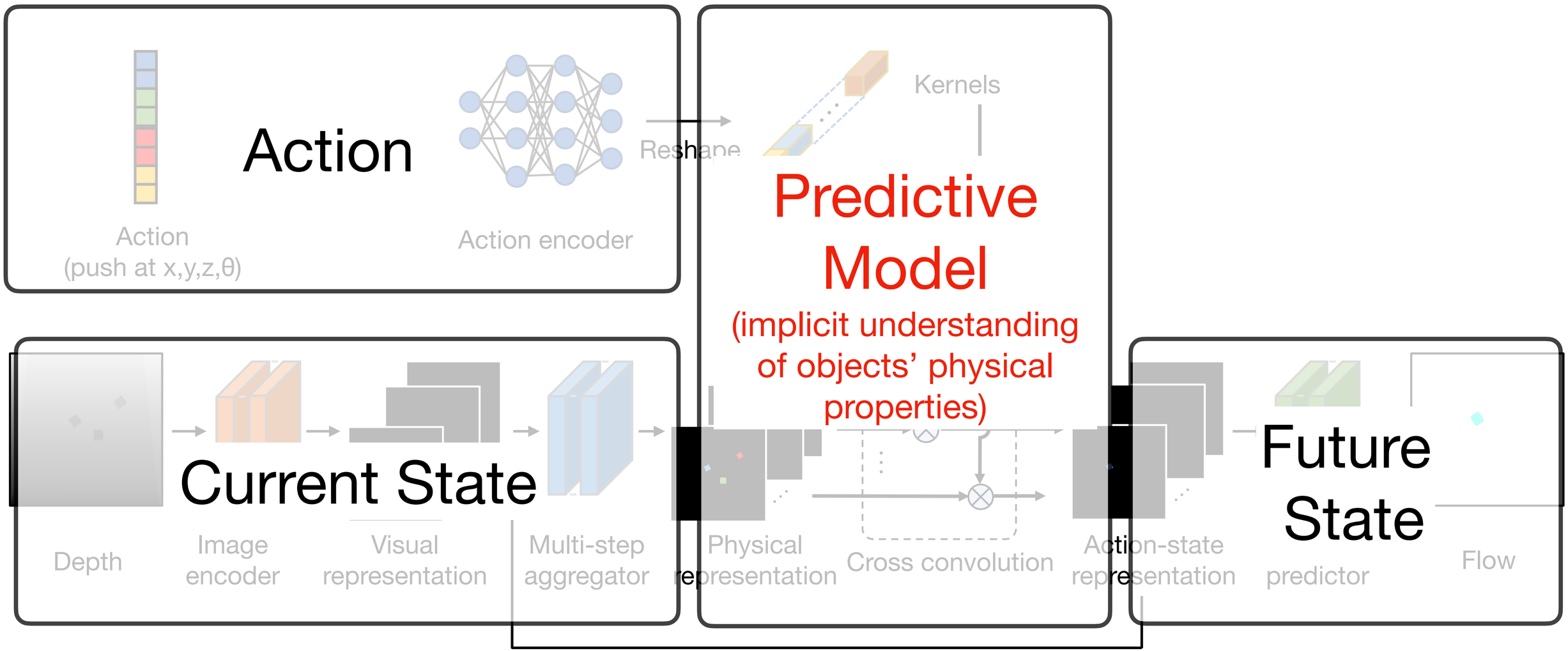


Collision

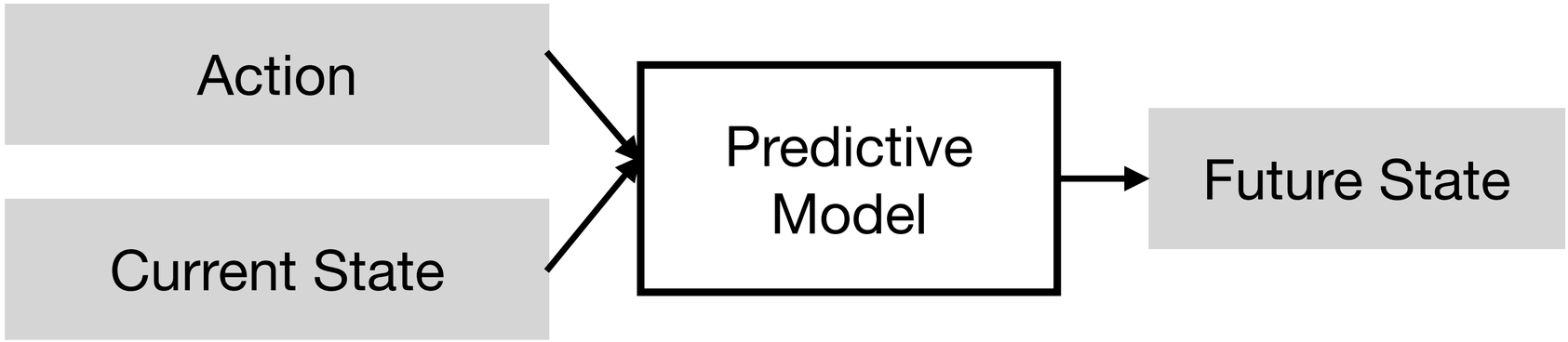
DensePhysNet



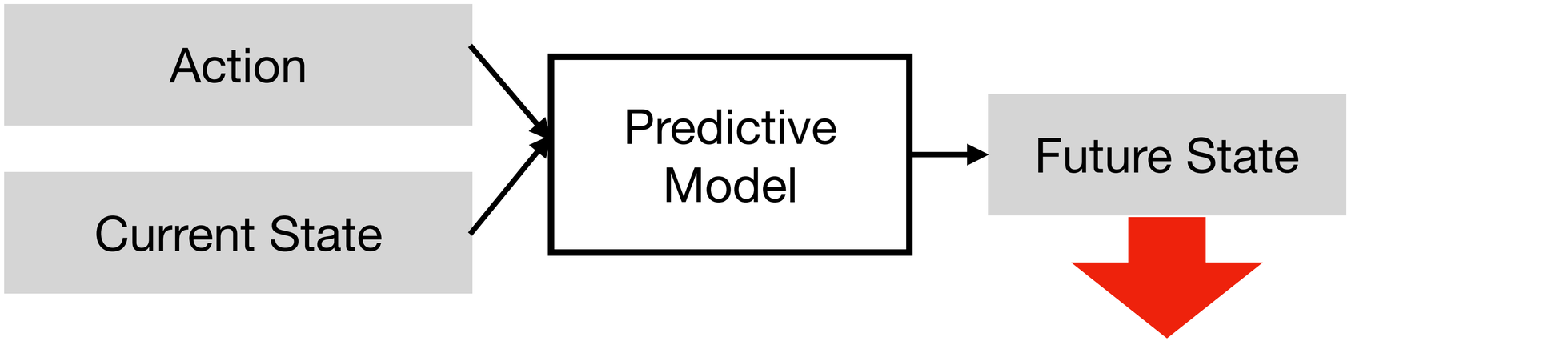
DensePhysNet



How to Evaluation DensePhysNet?

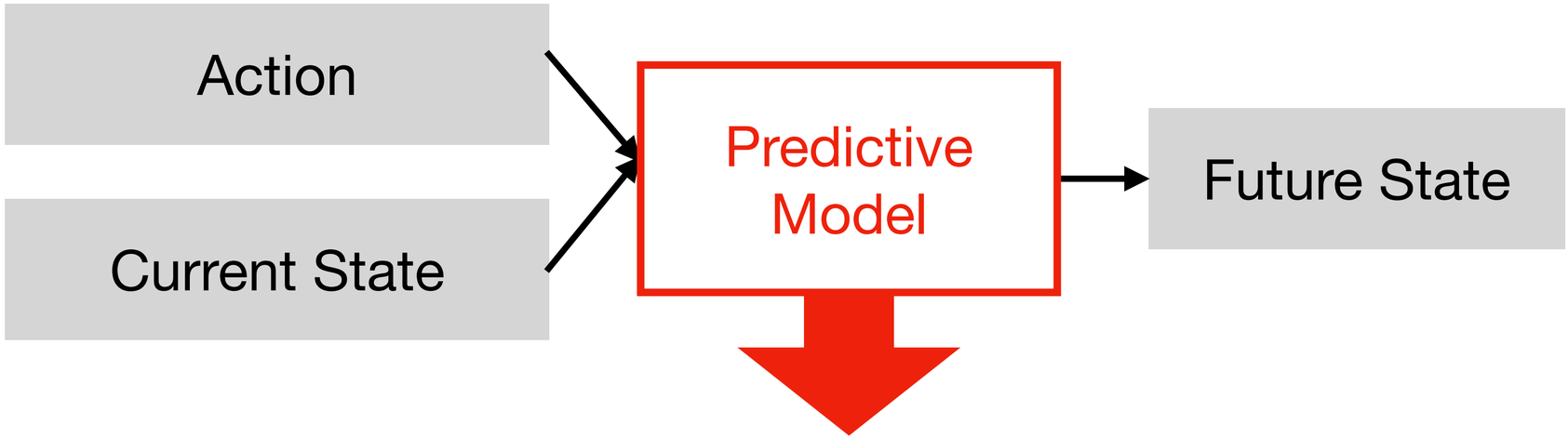


How to Evaluation DensePhysNet?



Although DensePhysNet is is trained as a predictive model, its predictive power is not the only thing we care about.

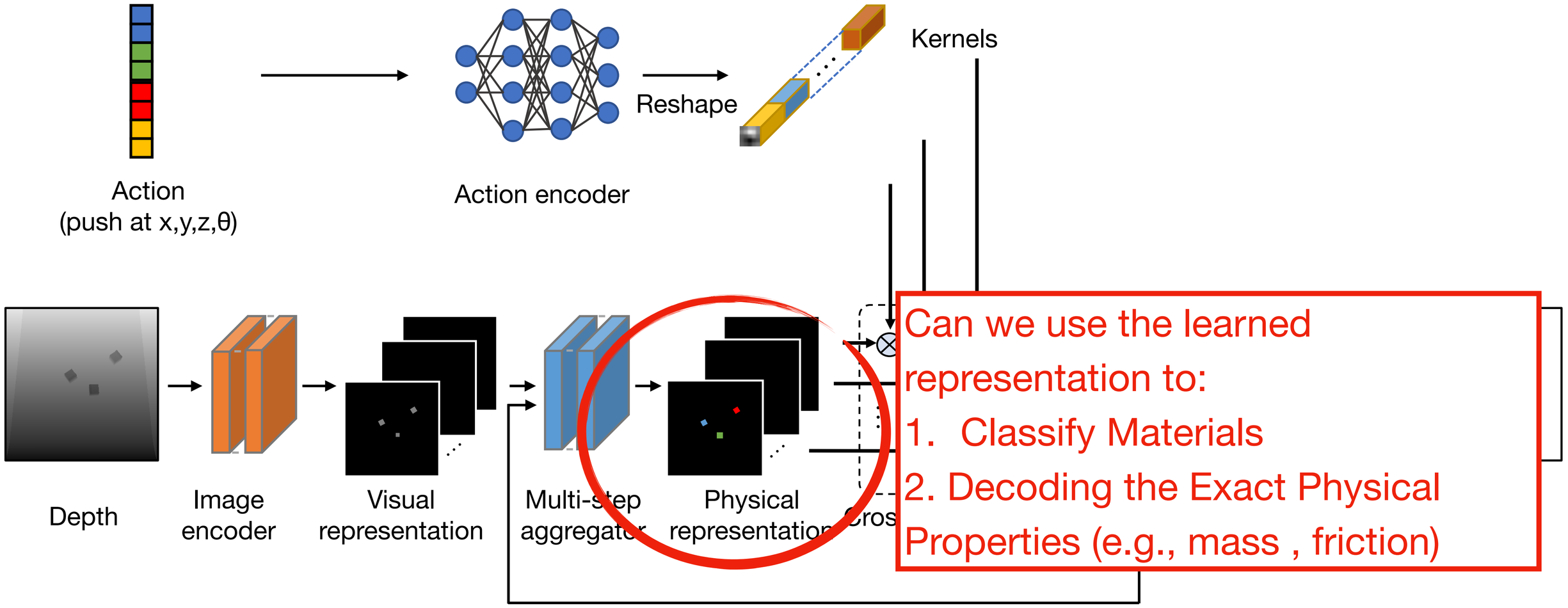
How to Evaluation DensePhysNet?



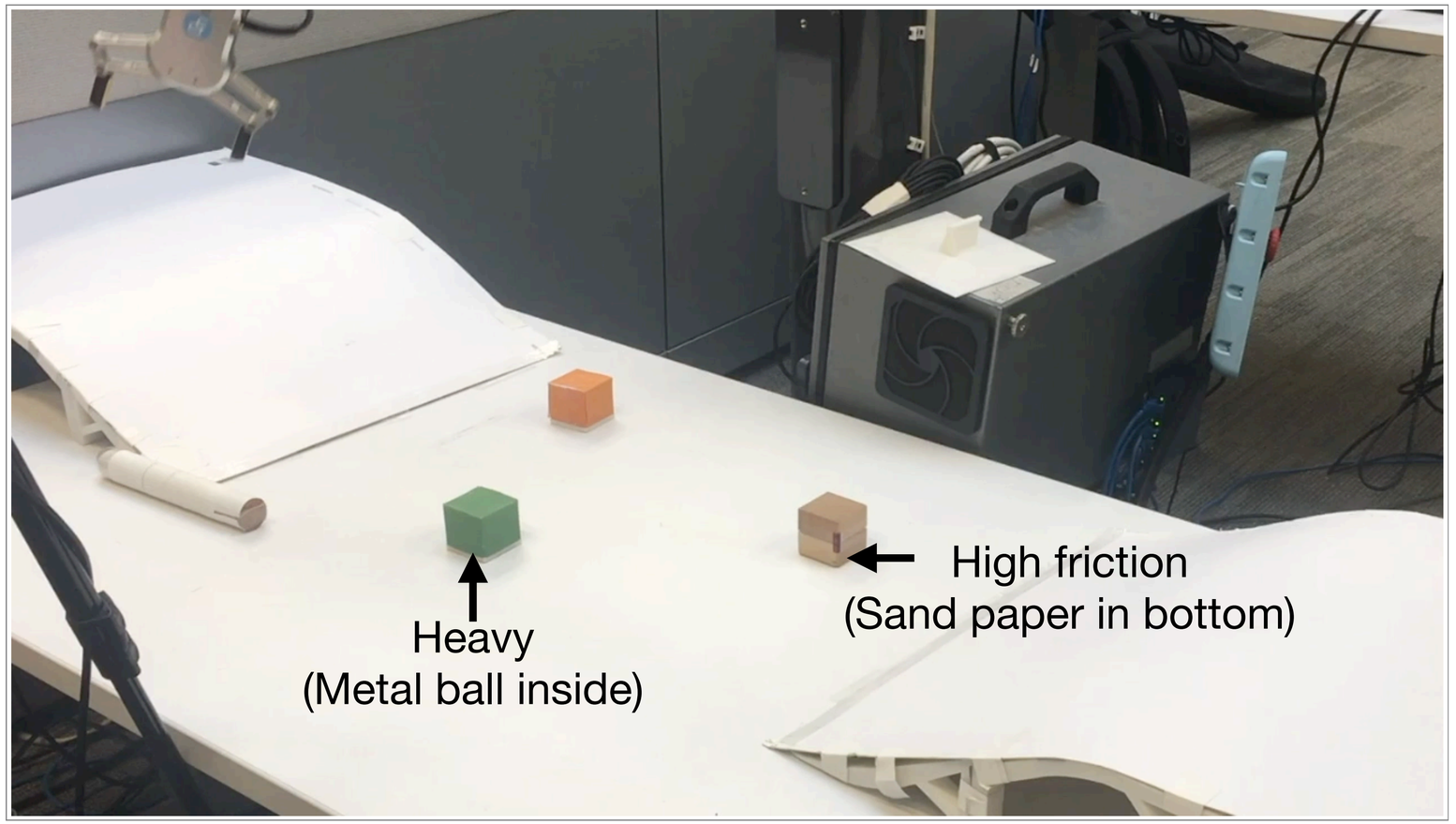
Although DensePhysNet is is trained as a predictive model, its predictive power is not the only thing we care about.

What we really care is whether the representation learns objects' physical properties.

Predictive Model



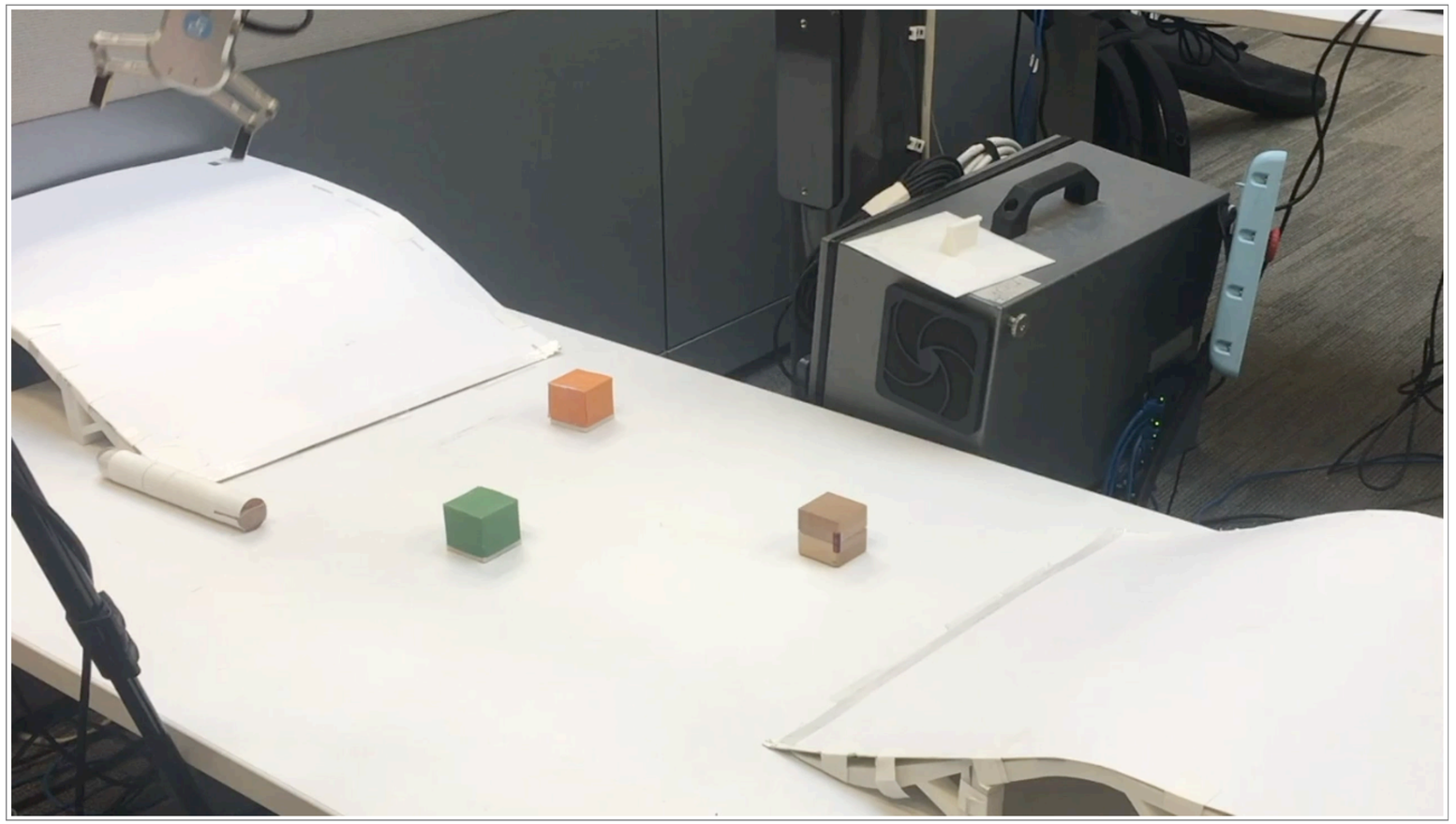
Material classification



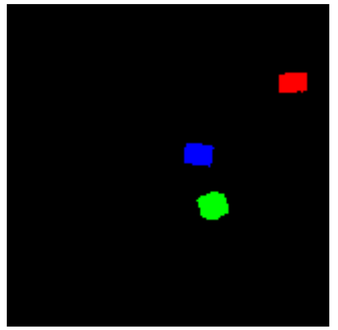
Real-time video (system use depth only)

Since the system only use depth these three block are visually indistinguishable.

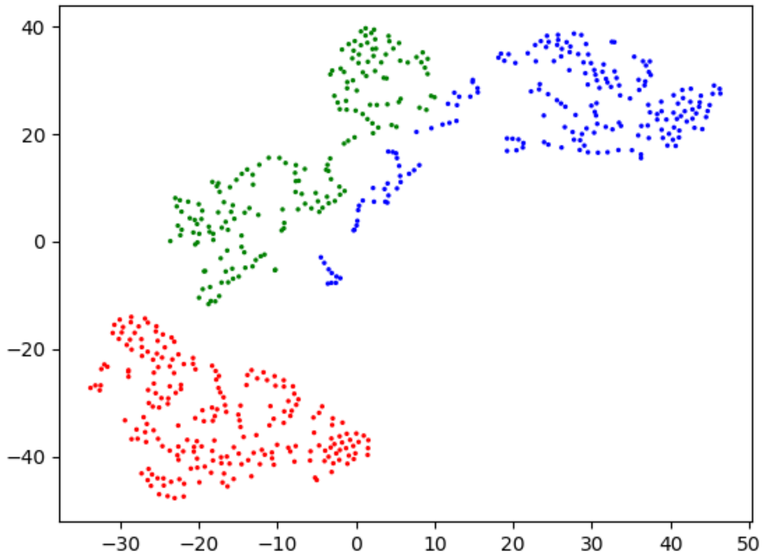
Material classification



Real-time video (system use depth only)

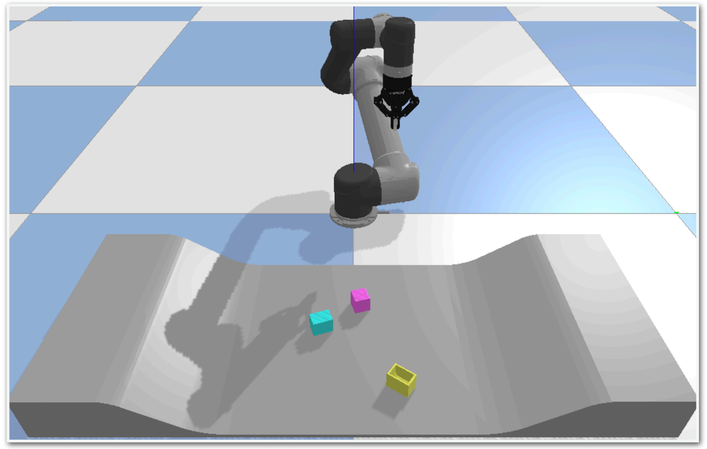


Mask

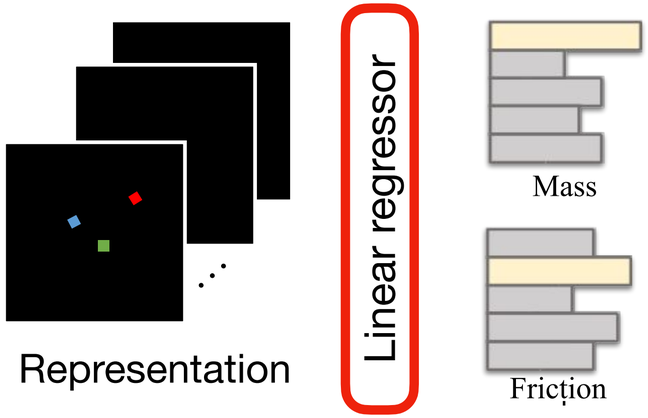


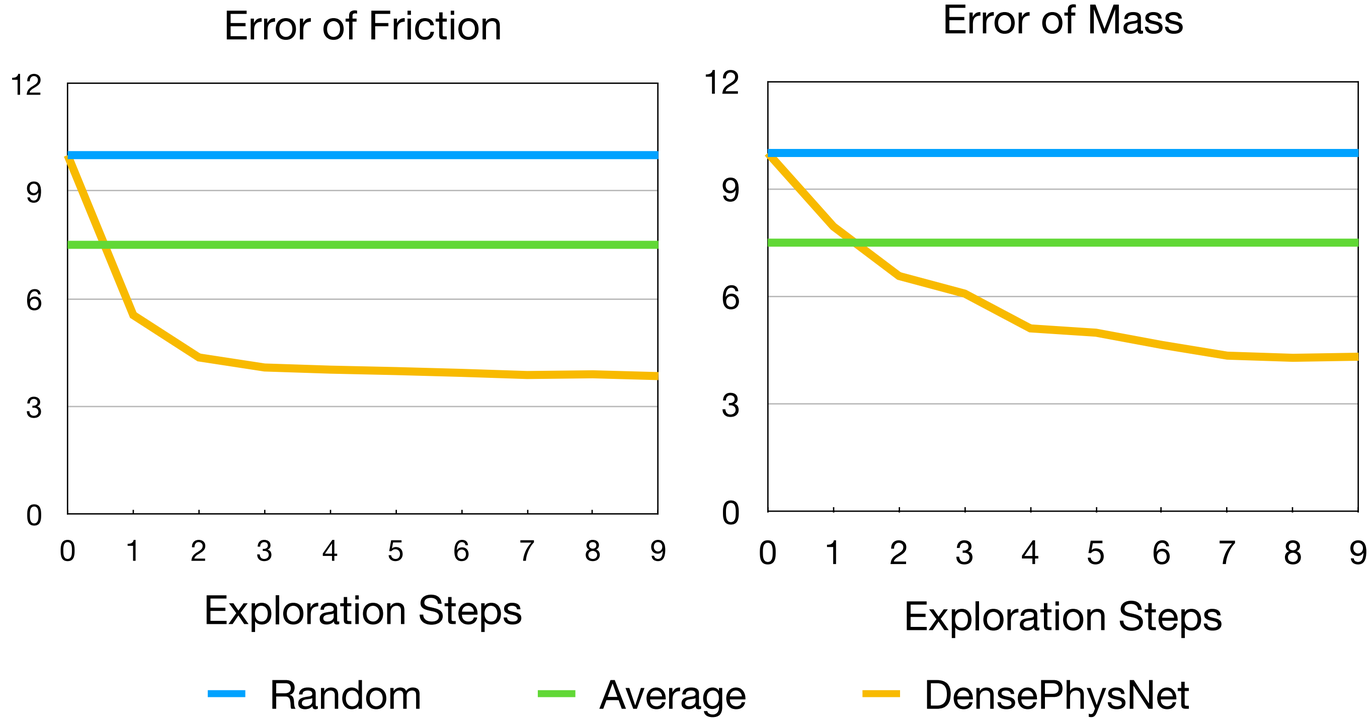
t-SNE visualization
(Color indicates object)

Decoding Physical Properties



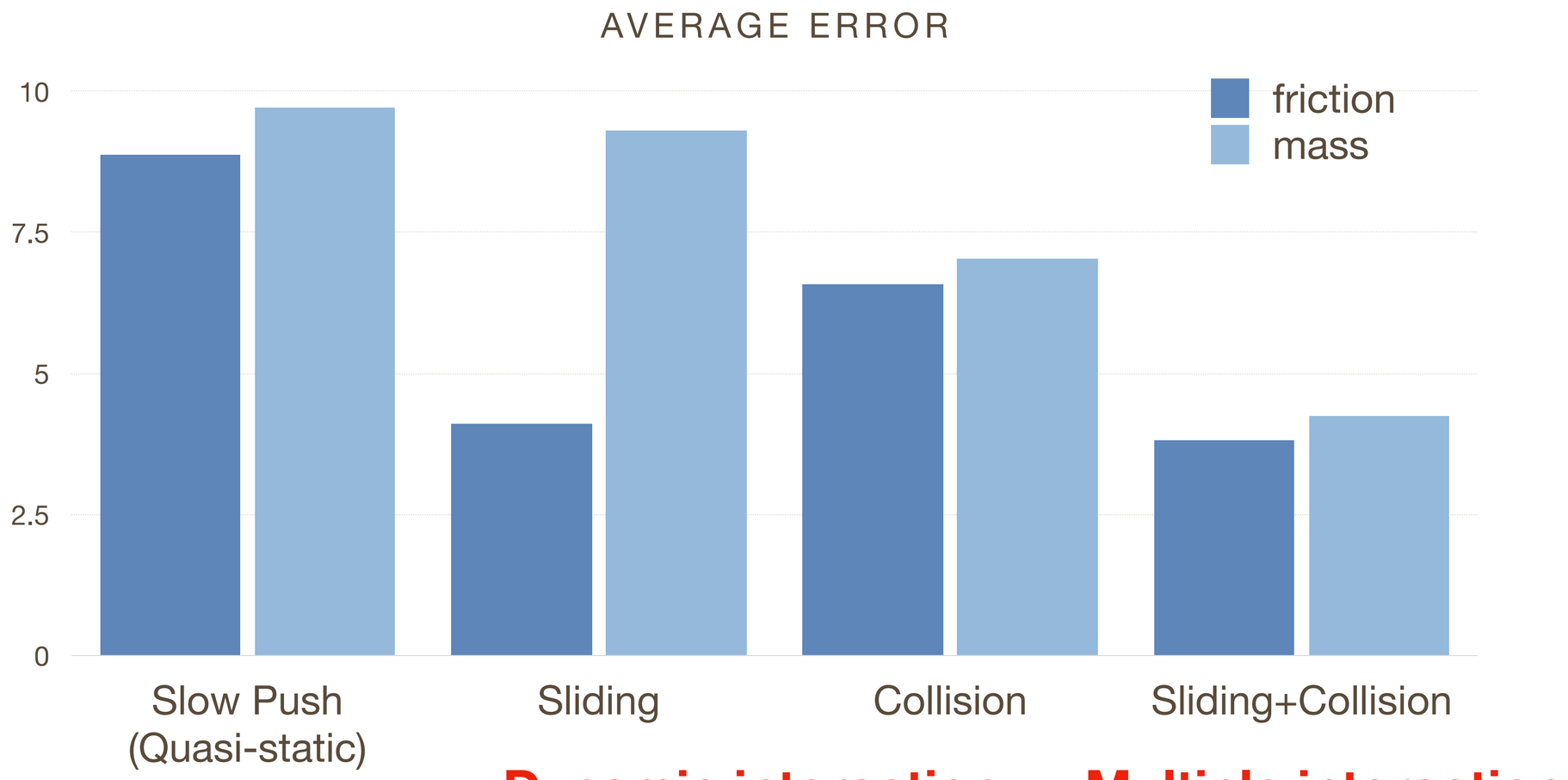
Test in Simulation





More interaction Helps!

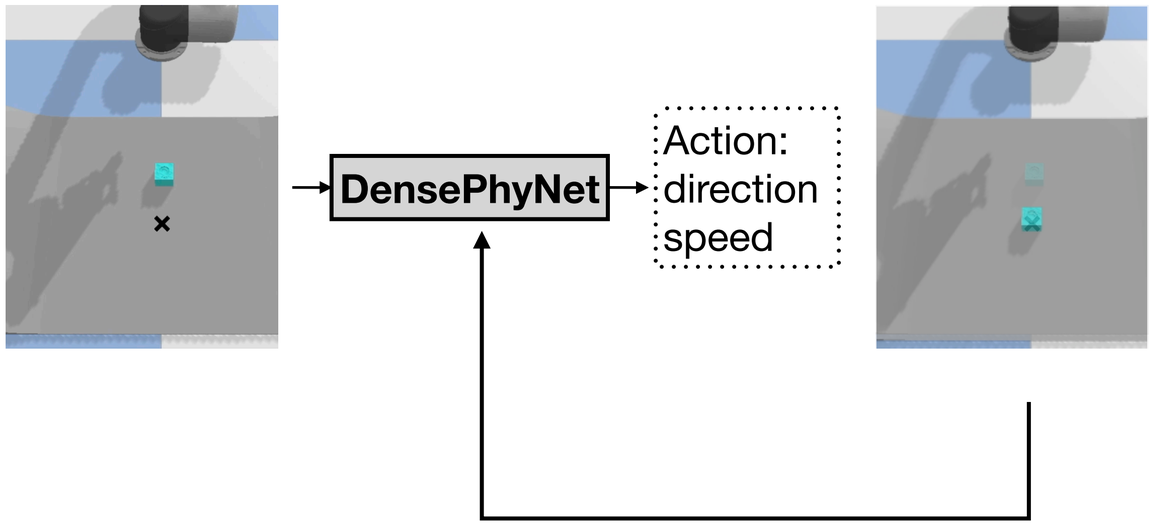
How about Interaction Types



Application in Manipulation

Application in Manipulation

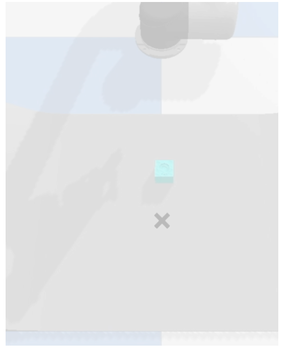
Learned predictive model
for **known** manipulation tasks



Updating the physical representation

Application in Manipulation

Learned predictive model
for **known** manipulation tasks

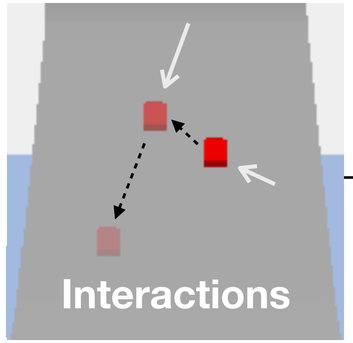


DensePhyNet

Action:
direction
speed

DSR-Net cannot do this, since it cannot explicitly decode object physical property

Decoded properties + physics engine
for **novel** manipulation tasks



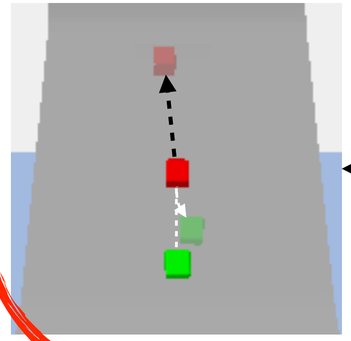
Interactions

DensePhyNet

Decode Properties

mass: m_a
friction: f_a

(+)



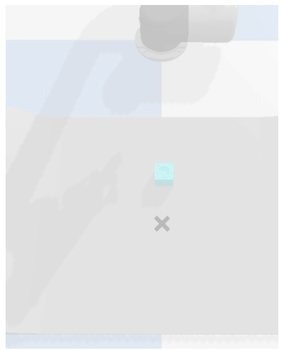
Novel Task

Physics Engine

Application in Manipulation

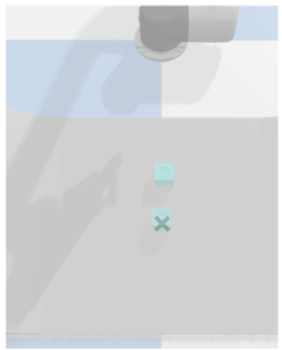
Learned predictive model
for **known** manipulation tasks

Decoded properties + physics engine
for **novel** manipulation tasks

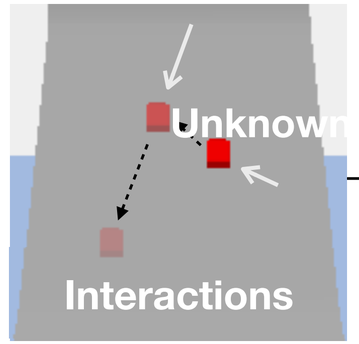


DensePhyNet

Action:
direction
speed



Hepler object selection:
Mass?
Initial speed?



DensePhyNet

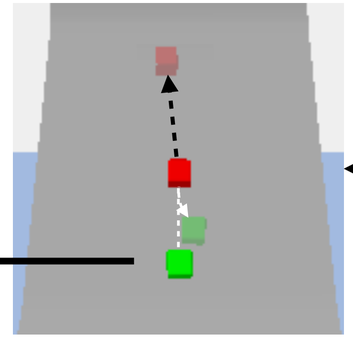
Decode Properties

mass: m_a
friction: f_a

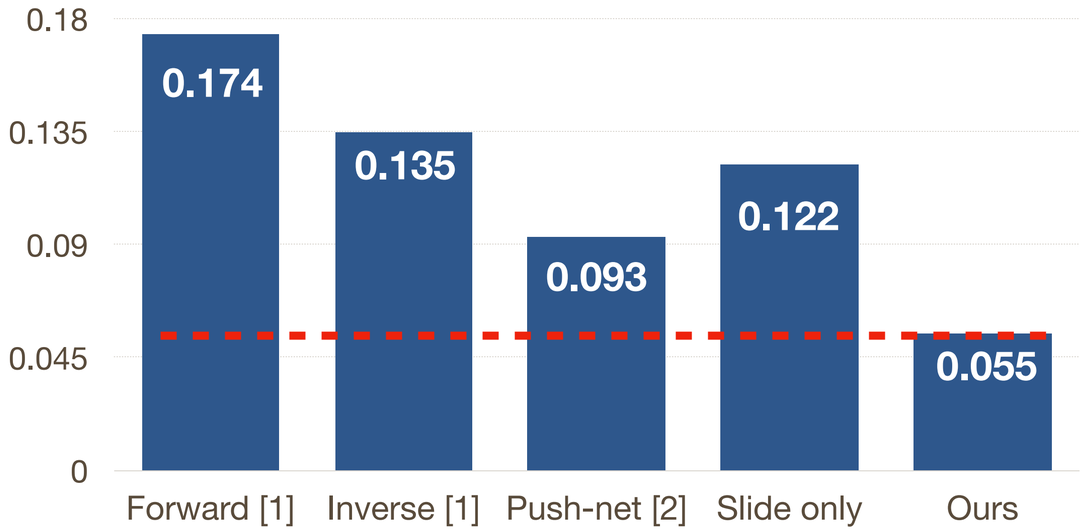
(+)

Physics Engine

Novel Task



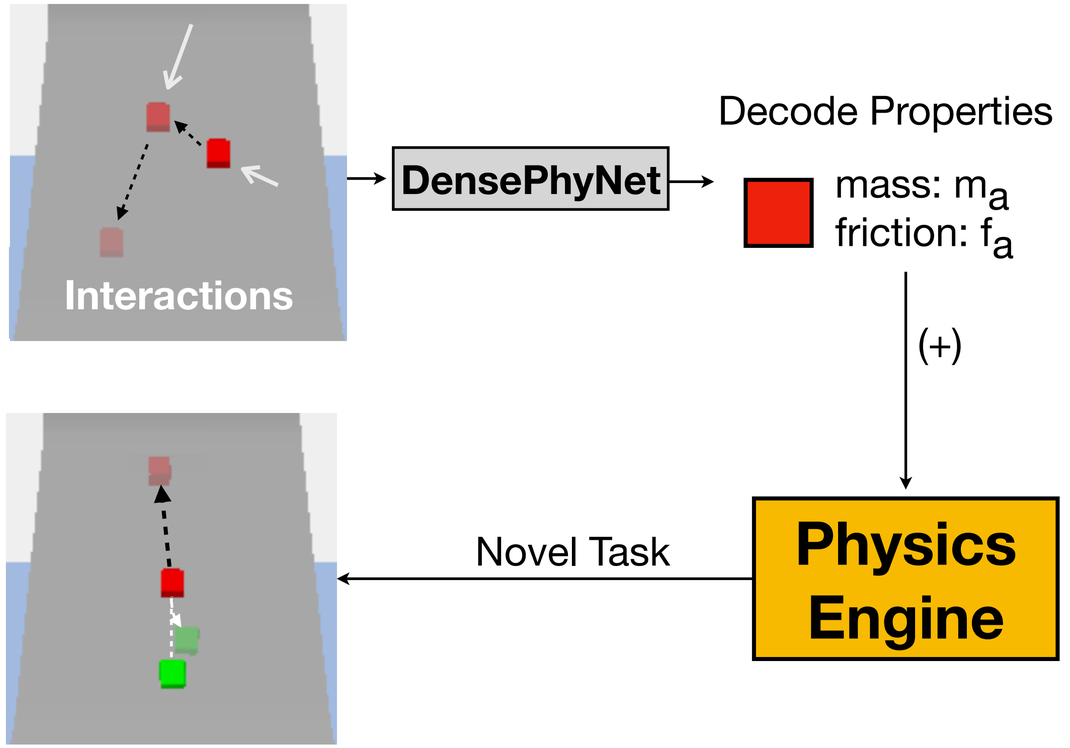
Application in Manipulation



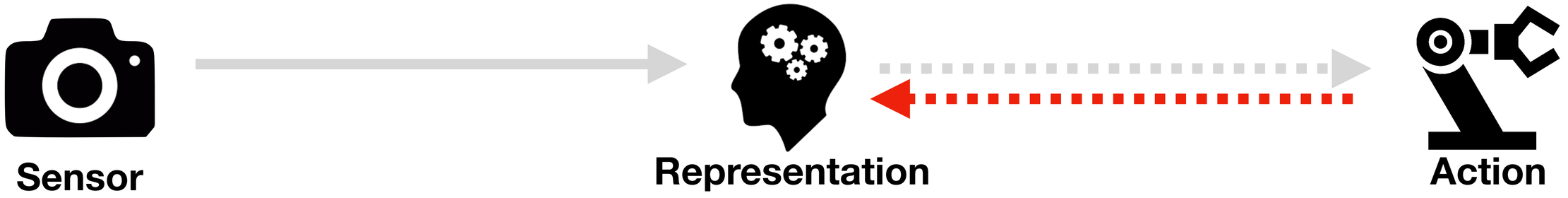
[1] Agrawal et al. NeurIPS, 2016
[2] Li et al. RSS 2018

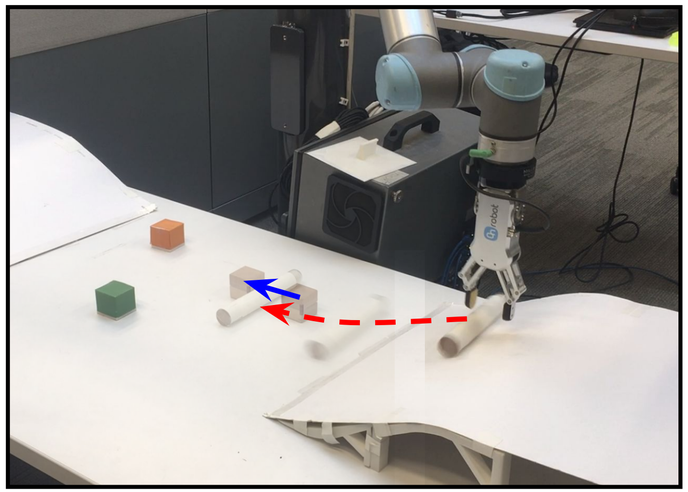
Error comparison

Decoded properties + physics engine for **novel** manipulation tasks



Active Scene Understanding





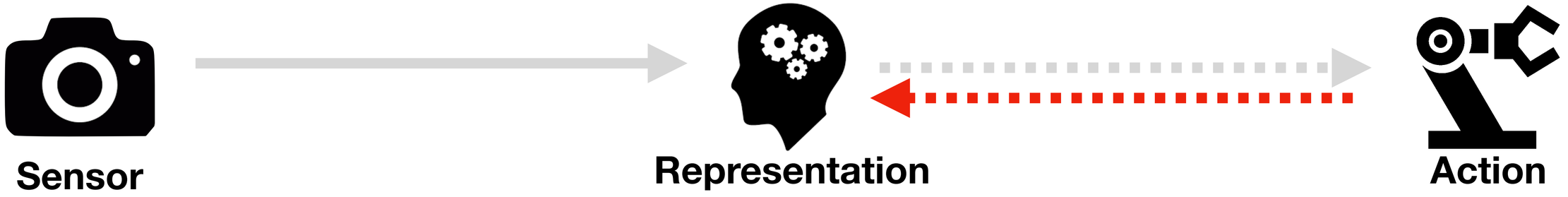
A model that learns physical object representations from self-supervised interactions.

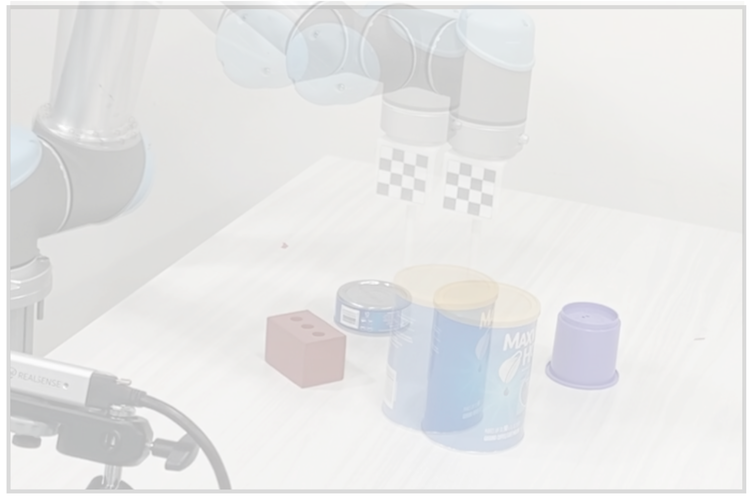
- Diverse set of interactions
- Dynamic interactions

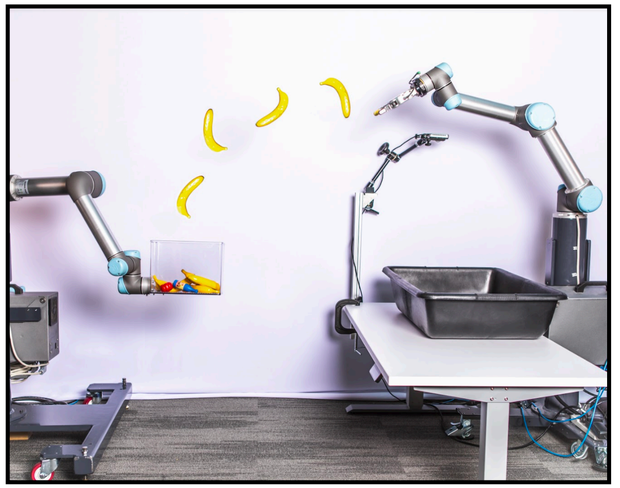
to reveal different physical properties

More Interactions for richer representations?

Active Scene Understanding







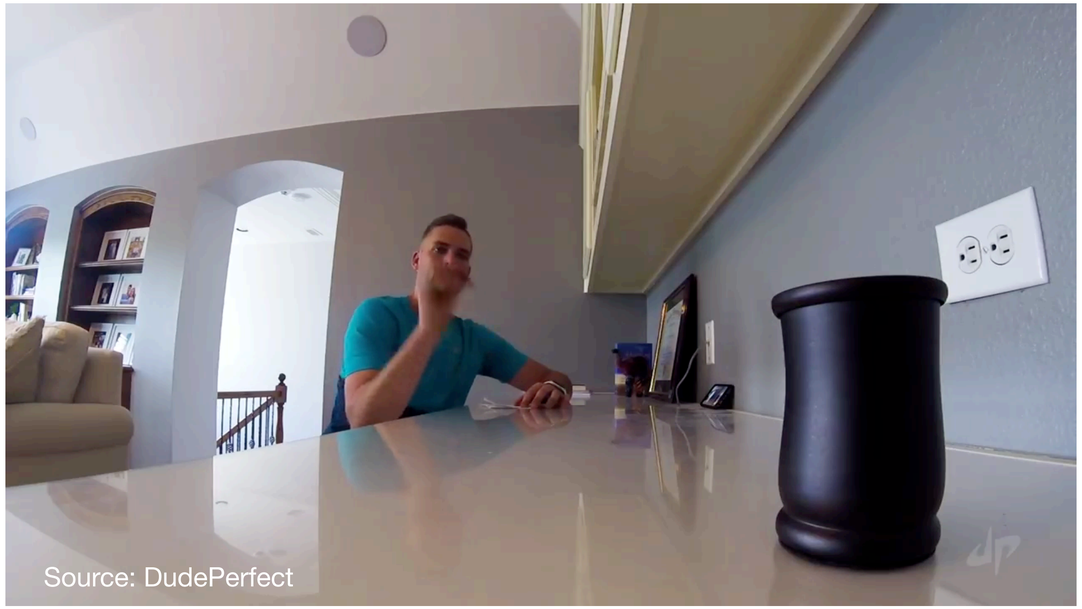
TossingBot: Learning to Throw Arbitrary Objects with Residual Physics

Andy Zeng, Shuran Song, Johnny Lee, Alberto Rodriguez, Thomas Funkhouser

Throwing is Useful

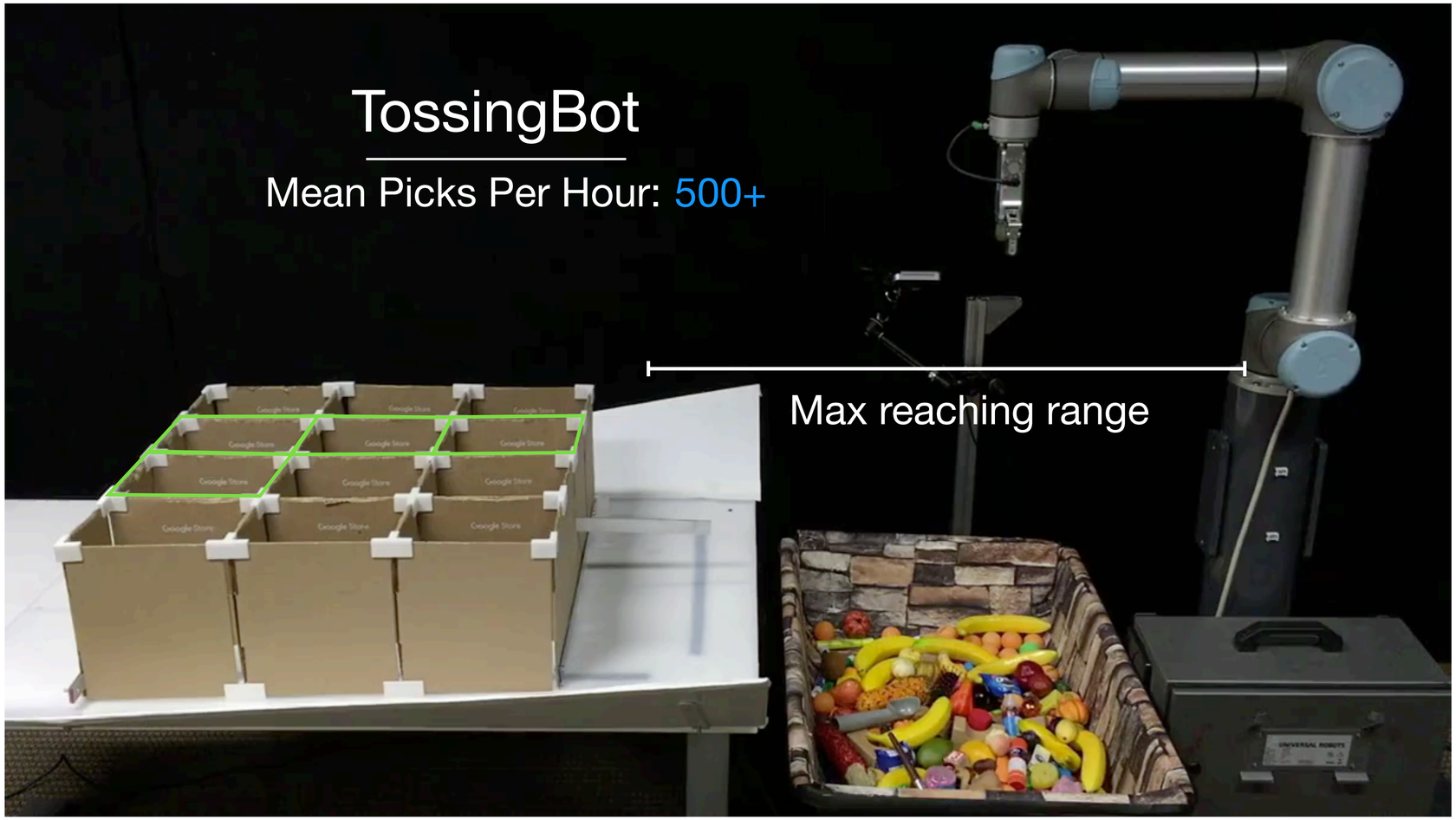
Throwing is Useful

People frequently use throwing to improve our efficiency.



Throwing is Useful

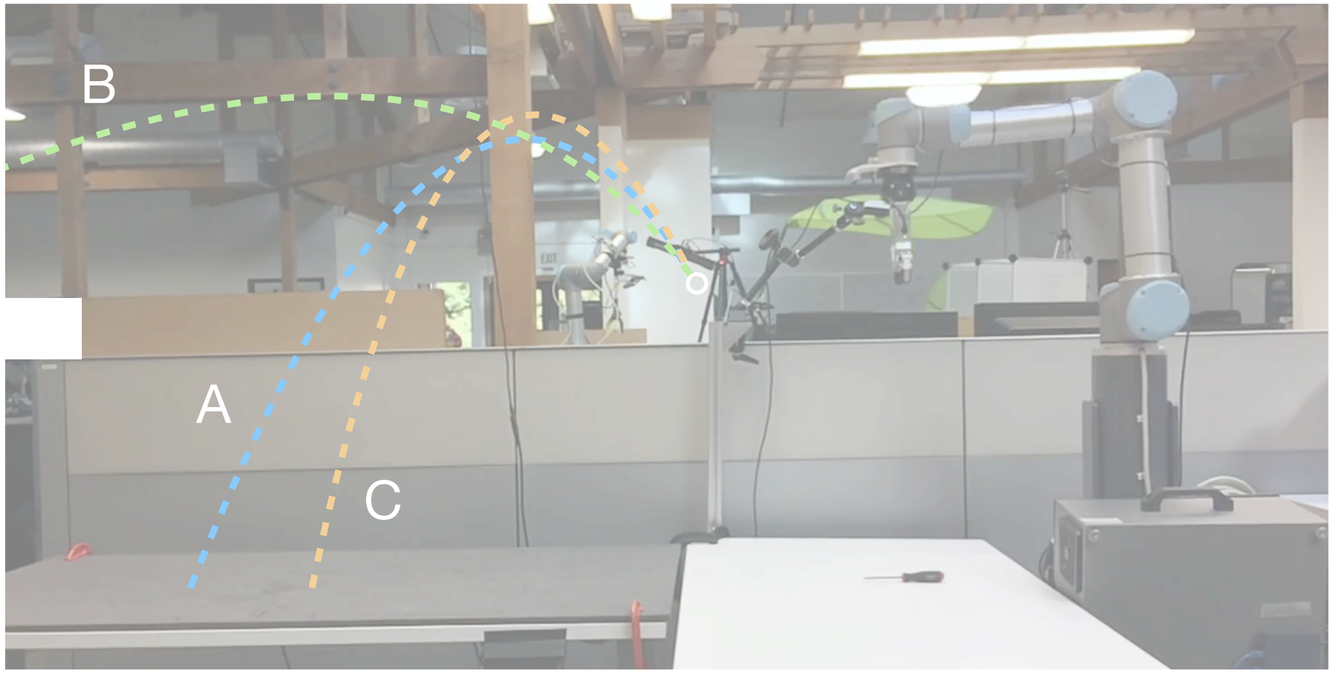
To improve speed and reachability



Side View

What the system need to learn?

What the system need to learn?



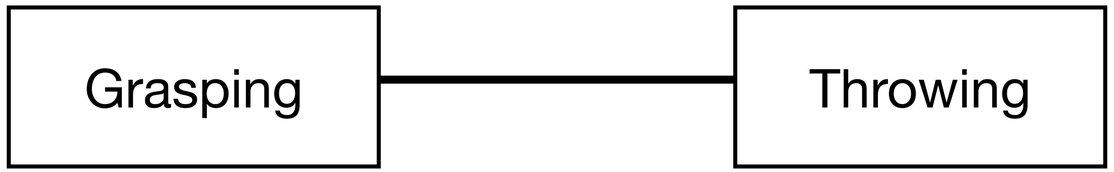
Grasp wrt Center of Mass

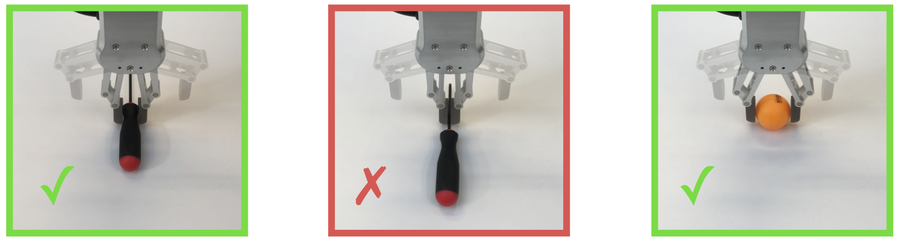
Varying Dynamics

Key Ideas

Acquire Pre-throw Conditions

Learned jointly





stable unpredictable stable

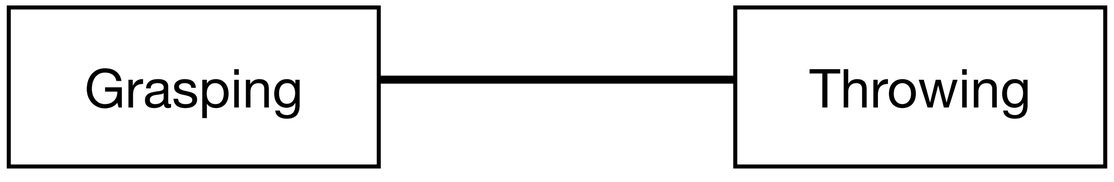
learned from experience

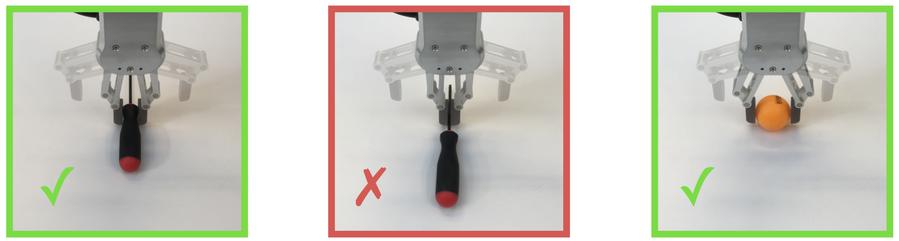
Handle Object Dynamics

Key Ideas

Acquire Pre-throw Conditions

Learned jointly



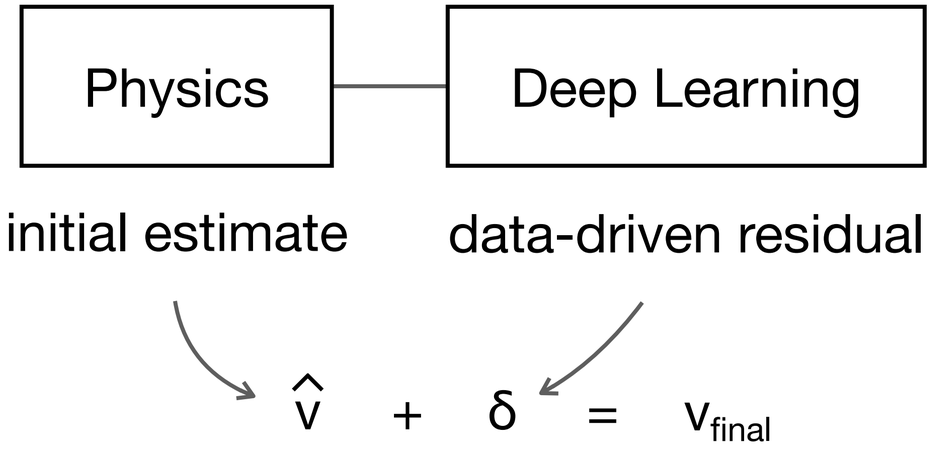


stable unpredictable stable

learned from experience

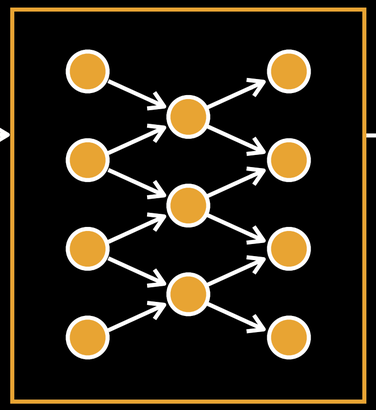
Handle Object Dynamics

Residual Physics



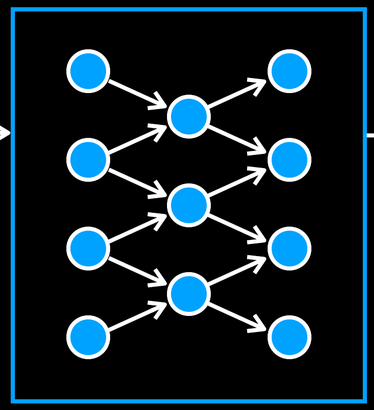
- v: generalizes to new target locations
- δ: learns to compensate obj. dynamics

RGB-D Heightmap



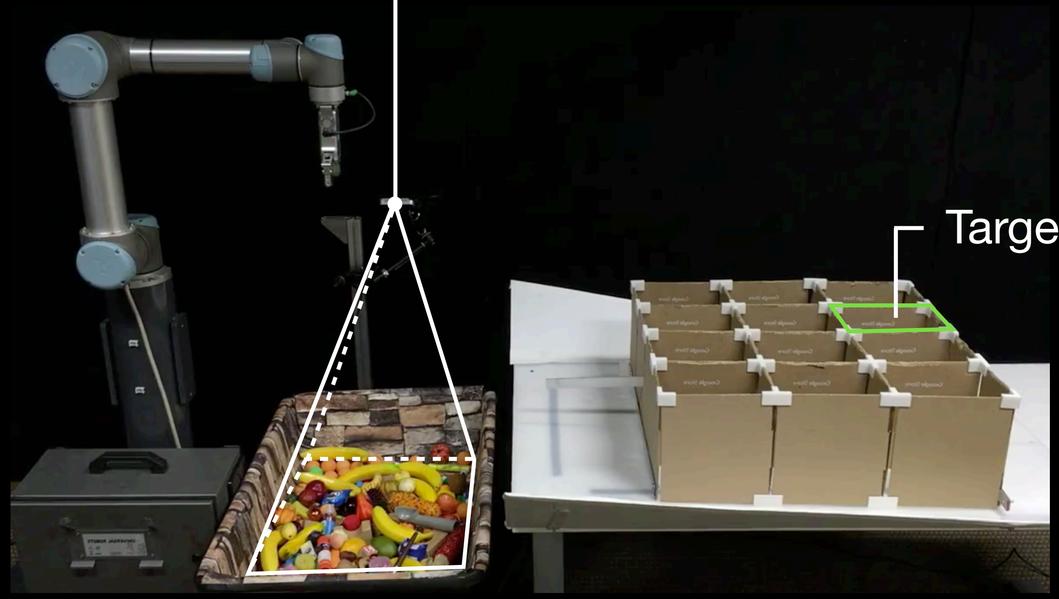
Perception Network

features

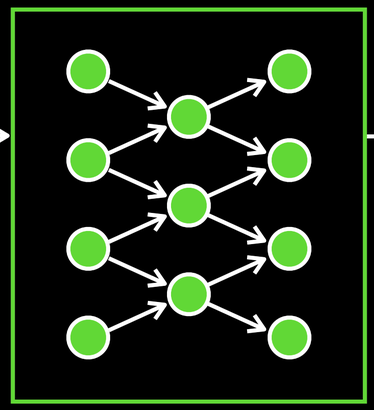


Grasping Network

Grasp Confidence
(dense pixel-wise)

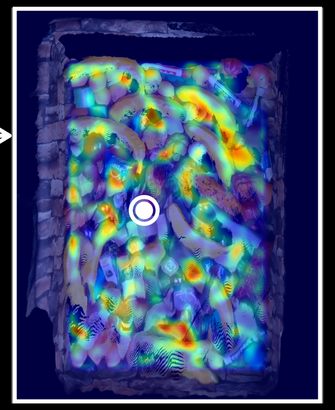


Target Location

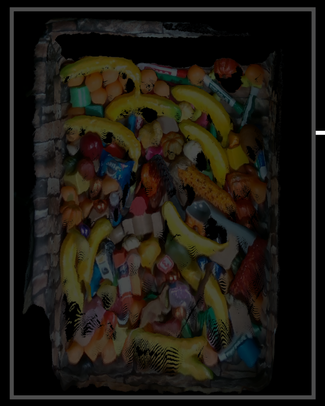


Throwing Network

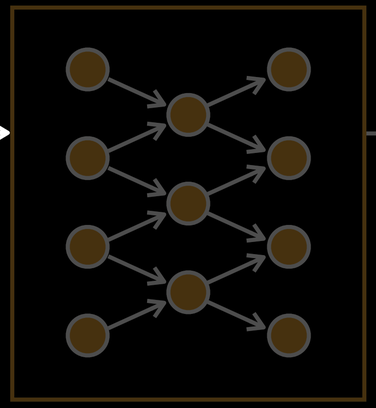
δ



Throw Velocities
(dense pixel-wise)

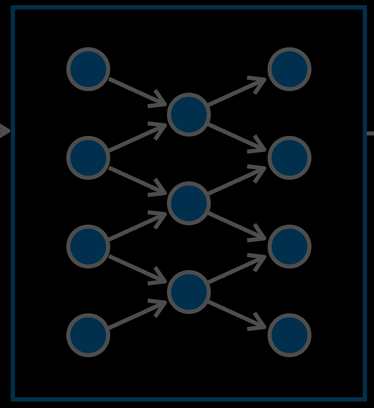


RGB-D Heightmap

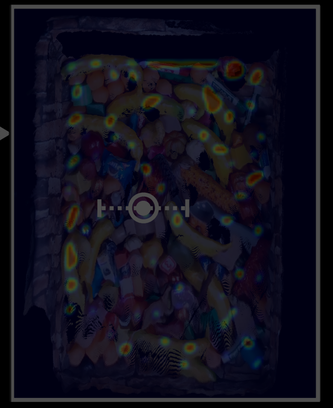


Perception Network

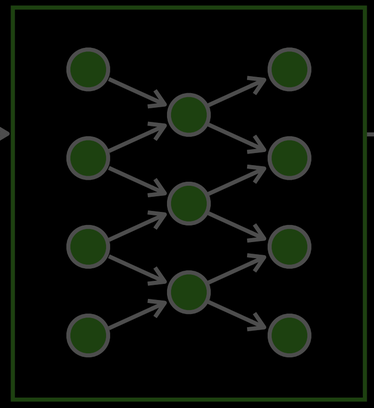
features



Grasping Network

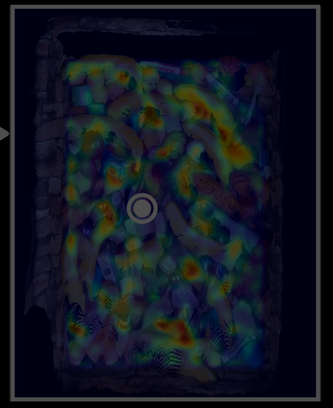


Grasp Confidence
(dense pixel-wise)



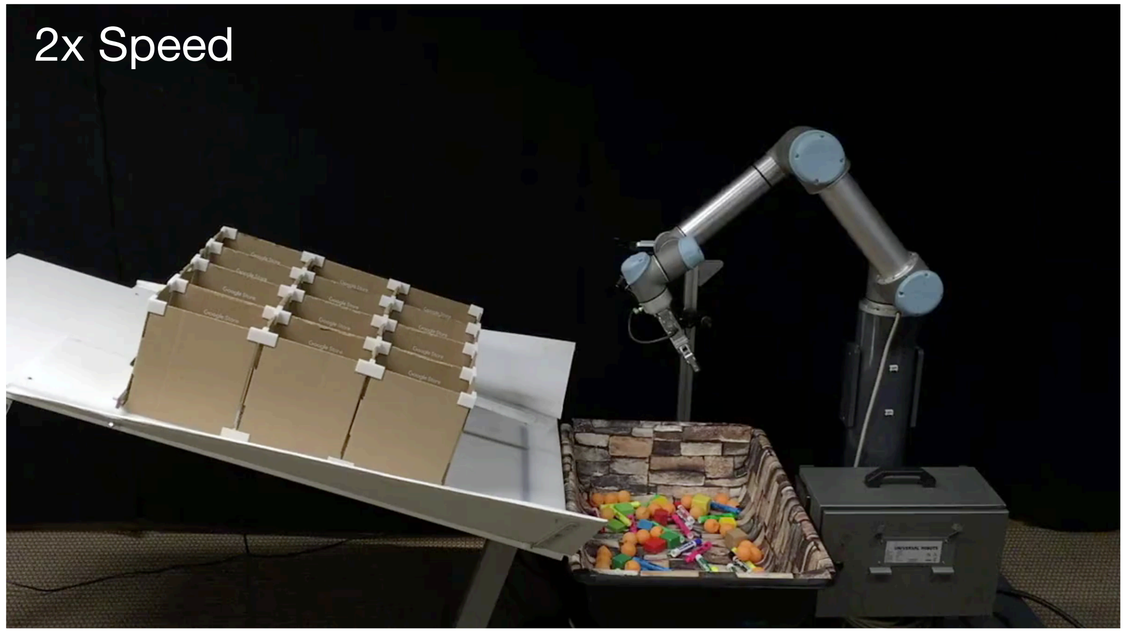
Throwing Network

δ



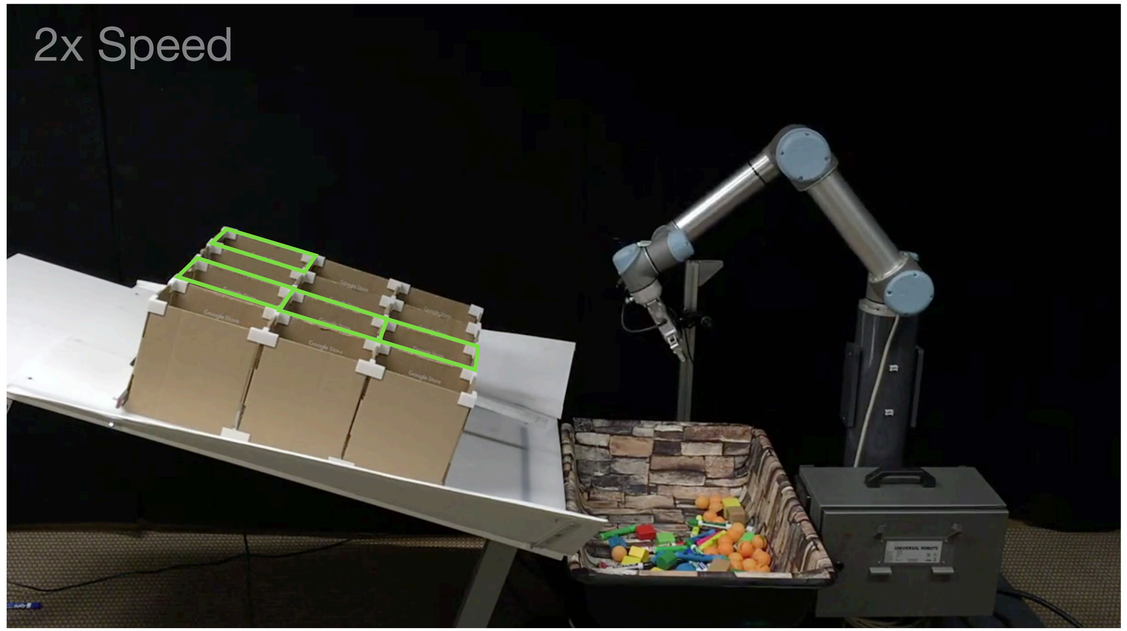
Throw Velocities
(dense pixel-wise)

Training Process



Random Initialization

Grasping: 5%
Throwing: 0%



14 Hrs Real-world Training

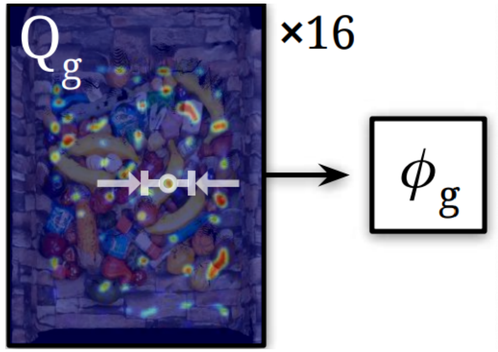
Grasping: 87%
Throwing: 85%

What does TossingBot learn?

RGB-D
Heightmap

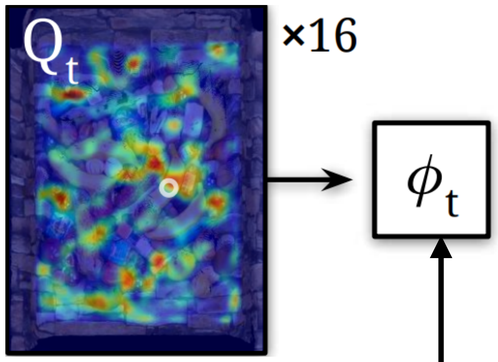
Learning Module

Physics-based
Controller



ϕ_g

Grasping
Score

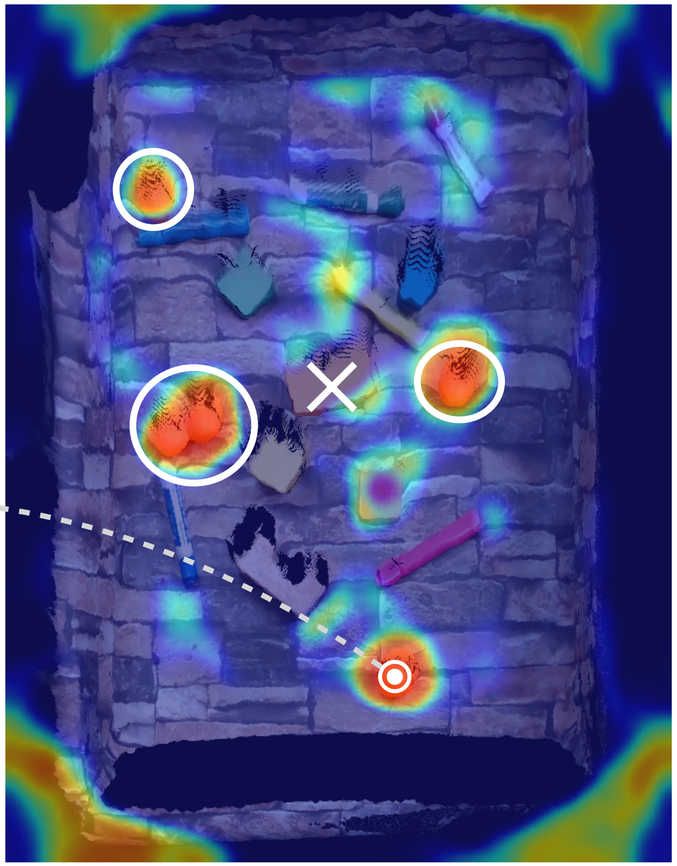


ϕ_t

Throwing
Velocity

What does TossingBot learn?

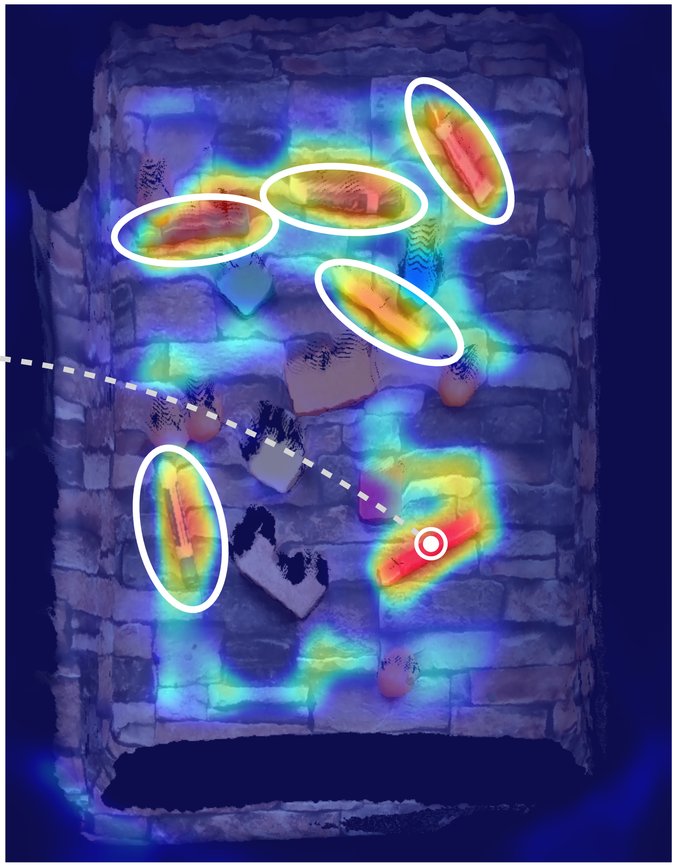
Camera View



Nearest Neighbor in Feature Space

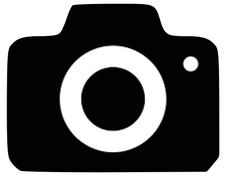
What does TossingBot learn?

Camera View



Nearest Neighbor in Feature Space

Active Scene Understanding



Sensor

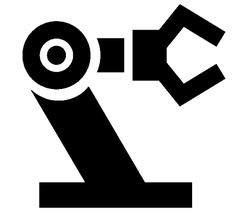
Representation

Object Instance
Amodal 3D Shape

Mass
Friction

Mass distribution
Aerodynamics

Accurate Throwing

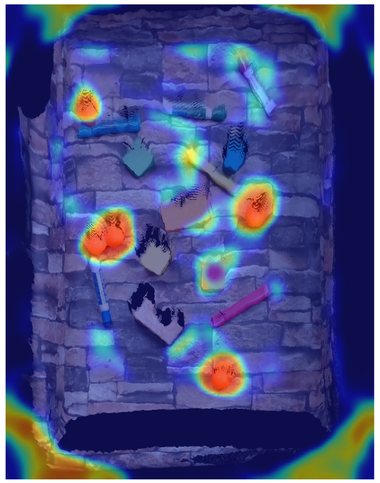


Action

Push

Sliding
Colliding

Throwing

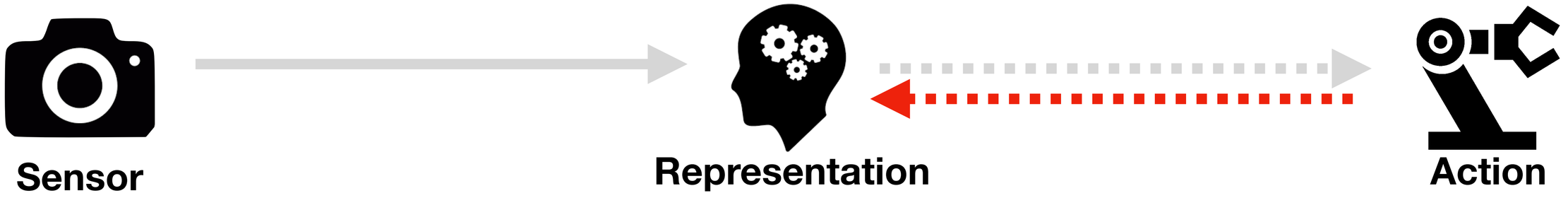


Active Scene Understanding

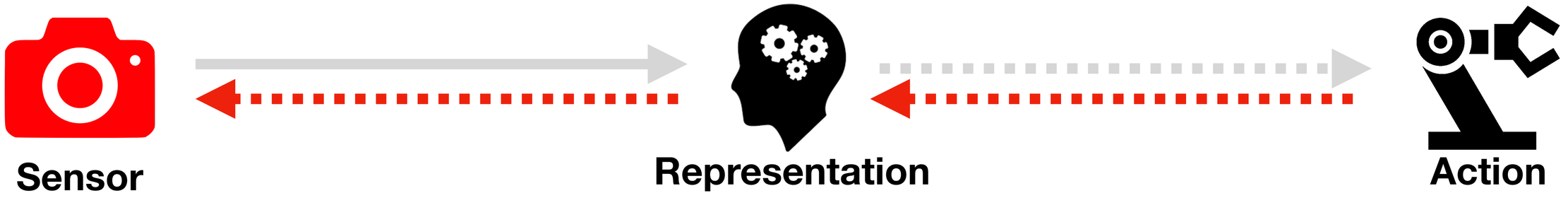
Summary:

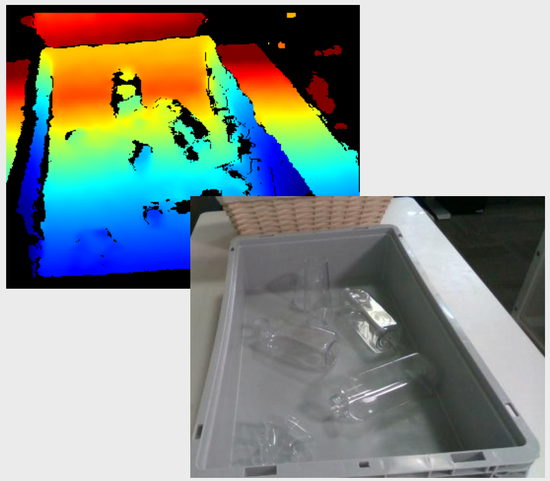
- Discover objects properties beyond visual appearance
    - e.g., object-instance, shape, mass, friction, aero-dynamics …
- Automatically acquire training data using action+future states
    - e.g., predictive model (motion), landing location
- Better representation to inform action planning
    - e.g., pushing, sliding, tossing

What's Next?



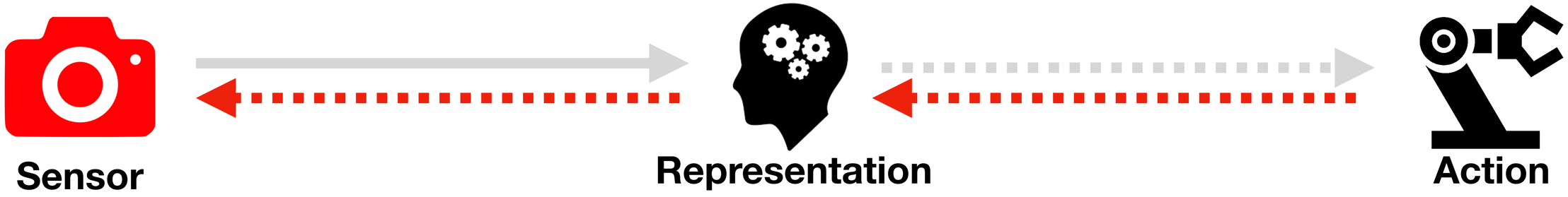
What's Next?



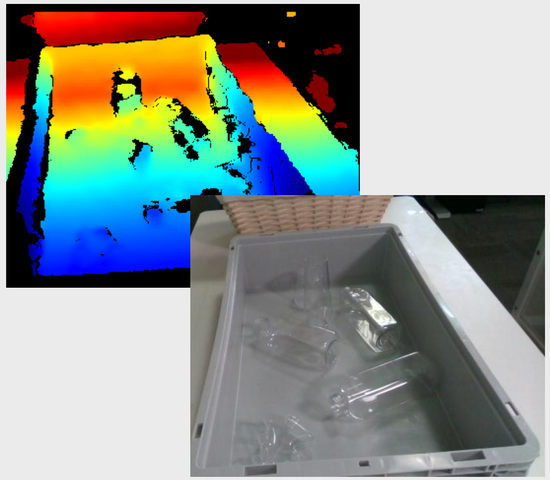


Visual

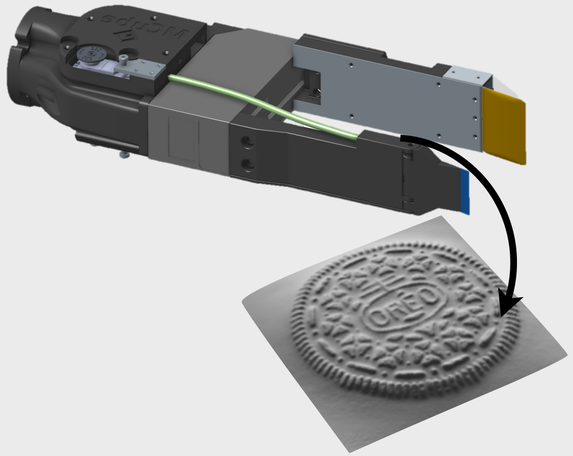
What's Next?



Sensor Fusion

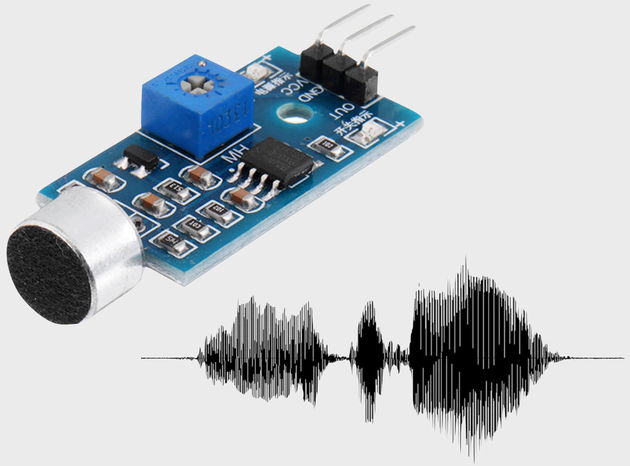


Visual



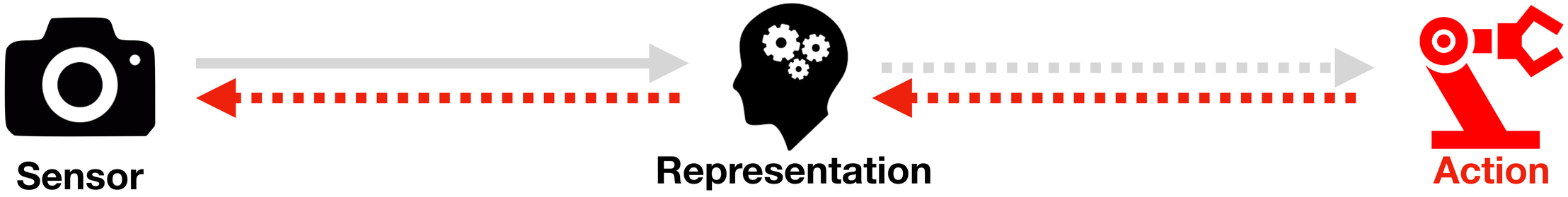
Tactile

Force/Torque



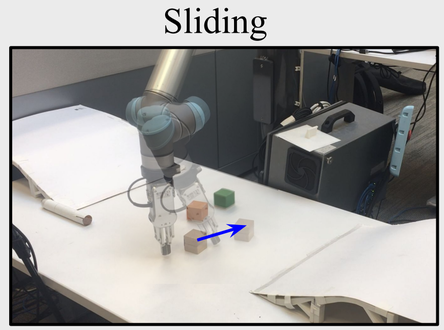
Sound

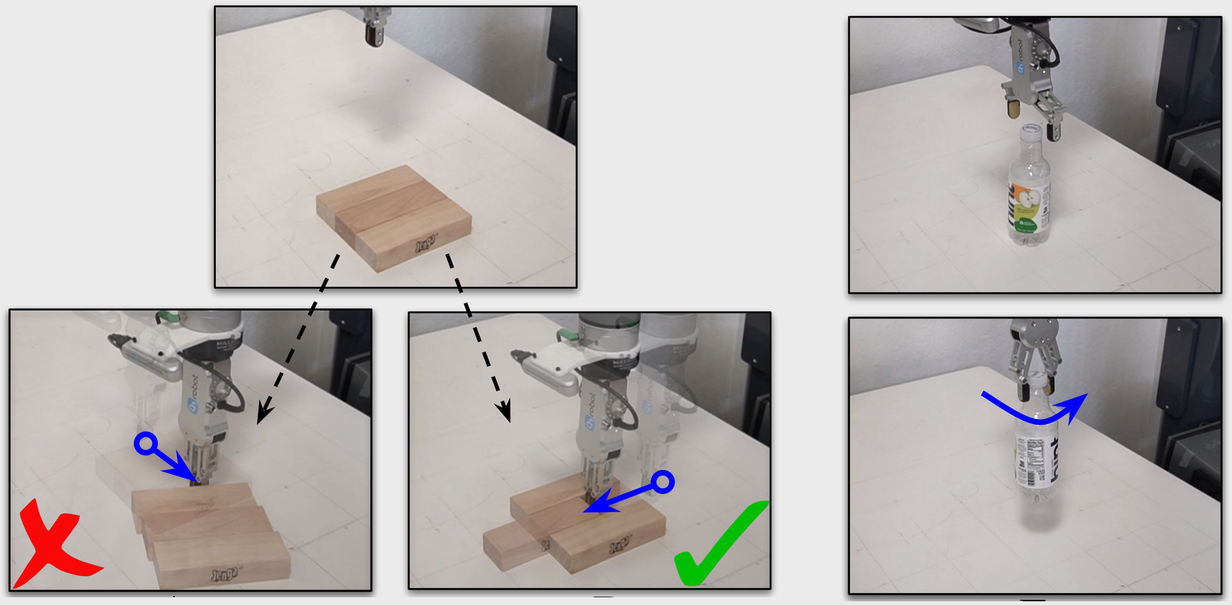
What's Next



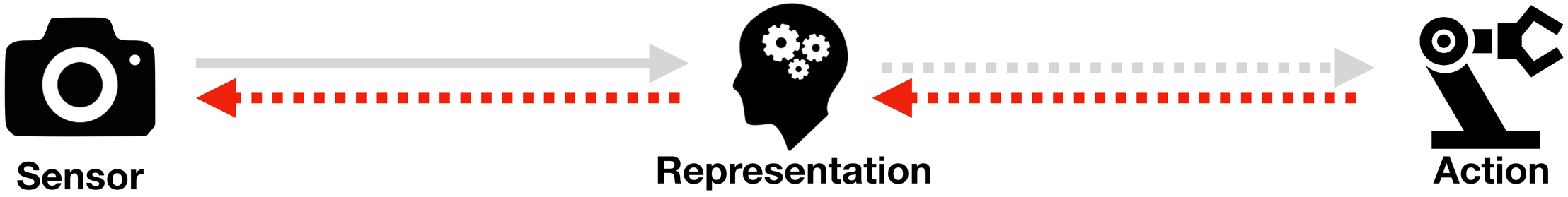
Action Selection

Which action will provide most useful information?





Active Scene Understanding

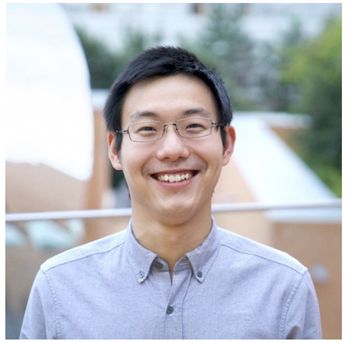


Active Explorers

Passive Observers

Acknowledgements

Zhenjia Xu



Jiajun Wu

Andy Zeng

Zhanpeng He

Google

amazon.com

Joshua B. Tenenbaum

Johnny Lee

Thomas Funkhosuer

Alberto Rodríguez

NSF

Thank You!