Carnegie Mellon
School of Computer Science

Deep Reinforcement Learning and Control

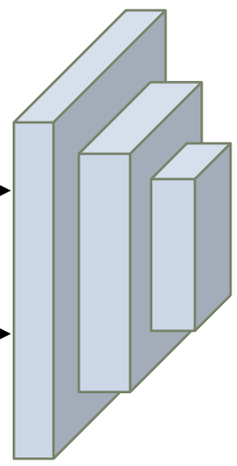
Visual Imitation Learning

CMU 10-703

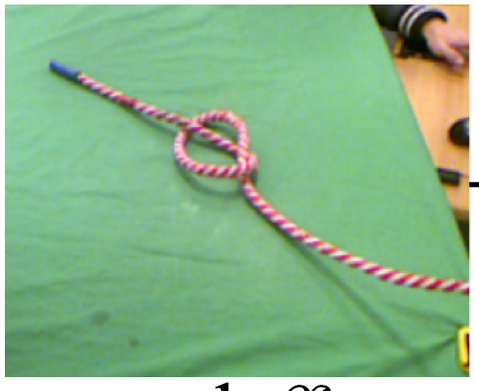
Katerina Fragkiadaki

How likely is to tie a knob only with trial-and-error?

current x_t



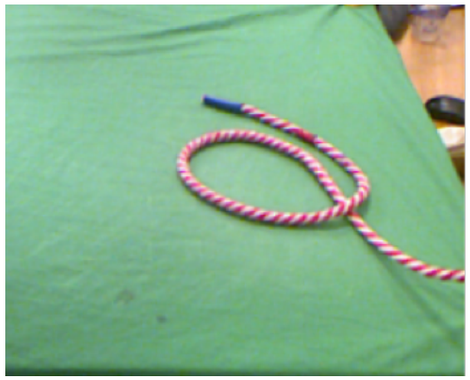
$\longrightarrow$ actions?



goal x_G

Consider a robot with a camera: no access to groundtruth low-dimensional state descriptions of the world.

How likely is to tie a knob only with trial-and-error?



current x_t

$\pi(x_t;\theta)$
policy

a_t

initially random

goal x_G

r_{t+1}

x_{t+1}

$$\max_{\theta}\left(\sum_{t=1}^{T} r_t\right)$$

Reward depends on how well the resulting state matches embedding-wise the goal state.

Embeddings can be trained with autoencoders, or under a combination of forward and inverse model learning.

Not very likely.

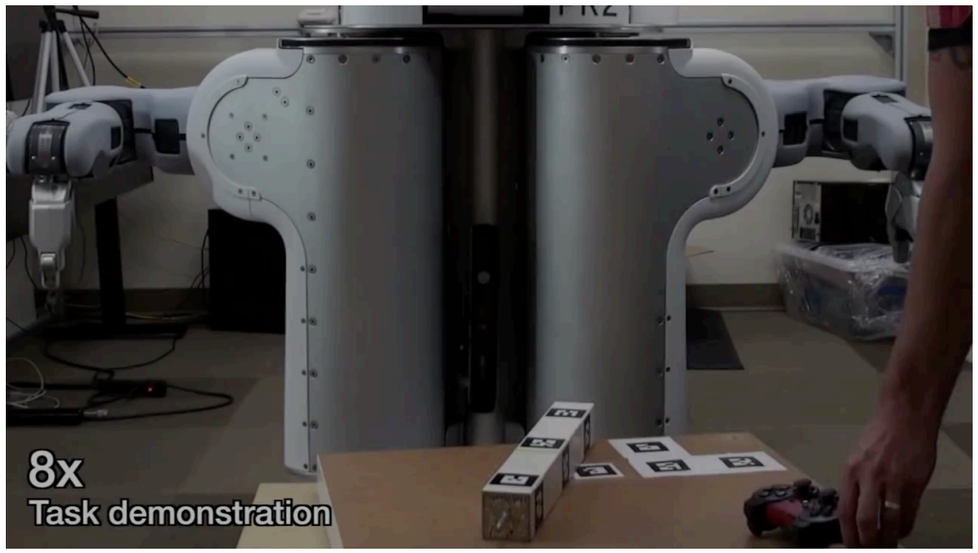
Indeed, we all learnt how to tie knobs with help from our parents

Imitation learning to the rescue

Learning from kinesthetic demonstations

a.k.a. kinesthetic teaching

Learning skills by having people or other agents performing the skill by taking over the end-effectors of the robot. The expert demos are given in the the action space of the robot agent.



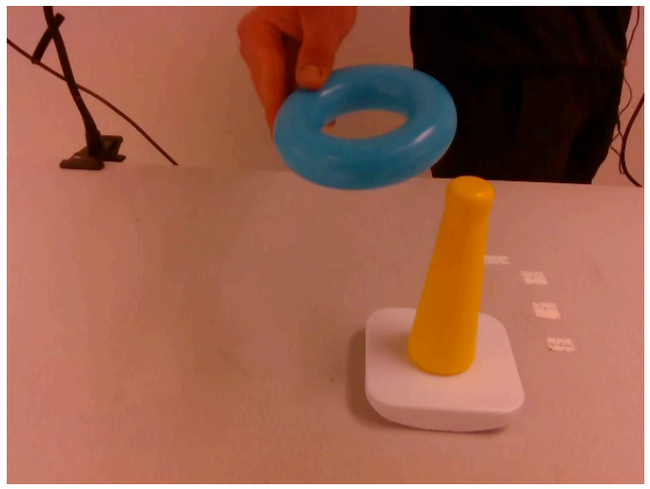
- Behaviour Cloning: simply regressing to the expert's actions
- Adversarial Imitation learning: reinforcement learning with denser reward depending on how well we match the expert's state-action densities
- Adding states from the expert demos in the experience buffer to get a chance to visit them (simple combination of RL and Imitation learning)

Learning from visual demonstations

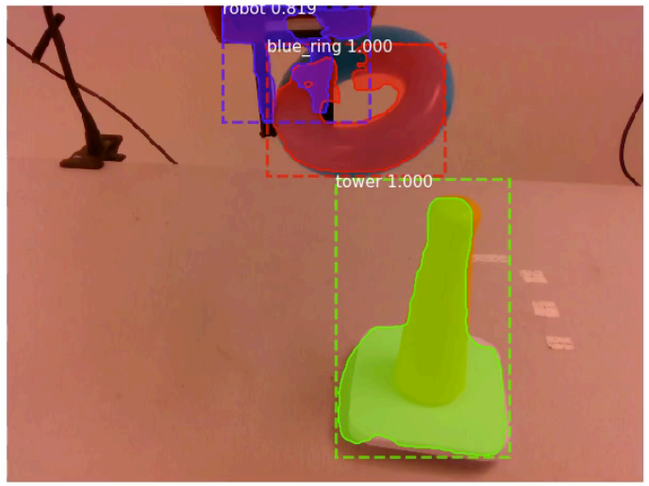
a.k.a. visual imitation or 3rd person imitation

Learning skills by watching people or other agents performing the skill

human demonstration



robot's imitation



Q: Can we use the imitation methods we have learnt to achieve this?

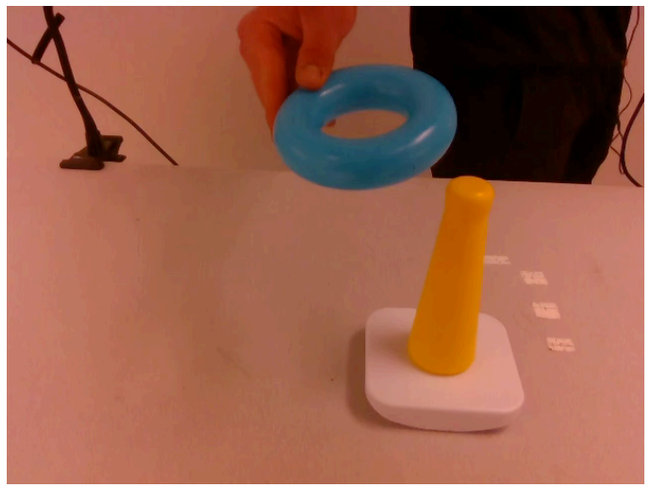
- Behaviour Cloning: not same action space, so no.
- Adversarial Imitation learning: reinforcement learning with denser reward depending on how well we match the expert's state only densities, so yes.
- Adding states from the expert demos in the experience buffer: that's possible only if the state does not involve active actuation: forces applied by the actor

Learning from visual demonstations

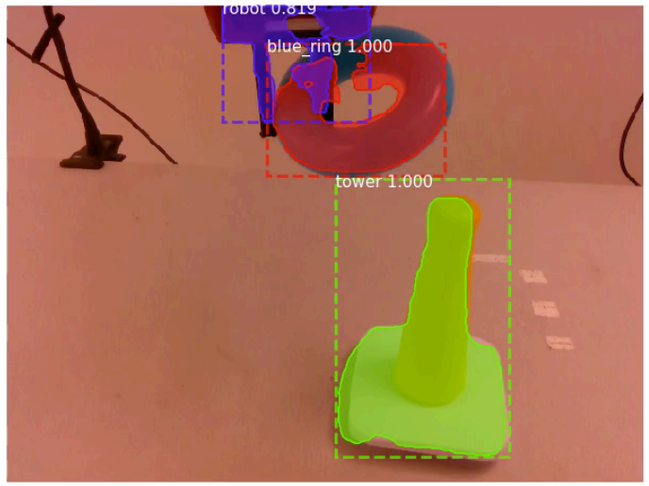
a.k.a. visual imitation or 3rd person imitation

Learning skills by watching people or other agents performing the skill

human demonstration



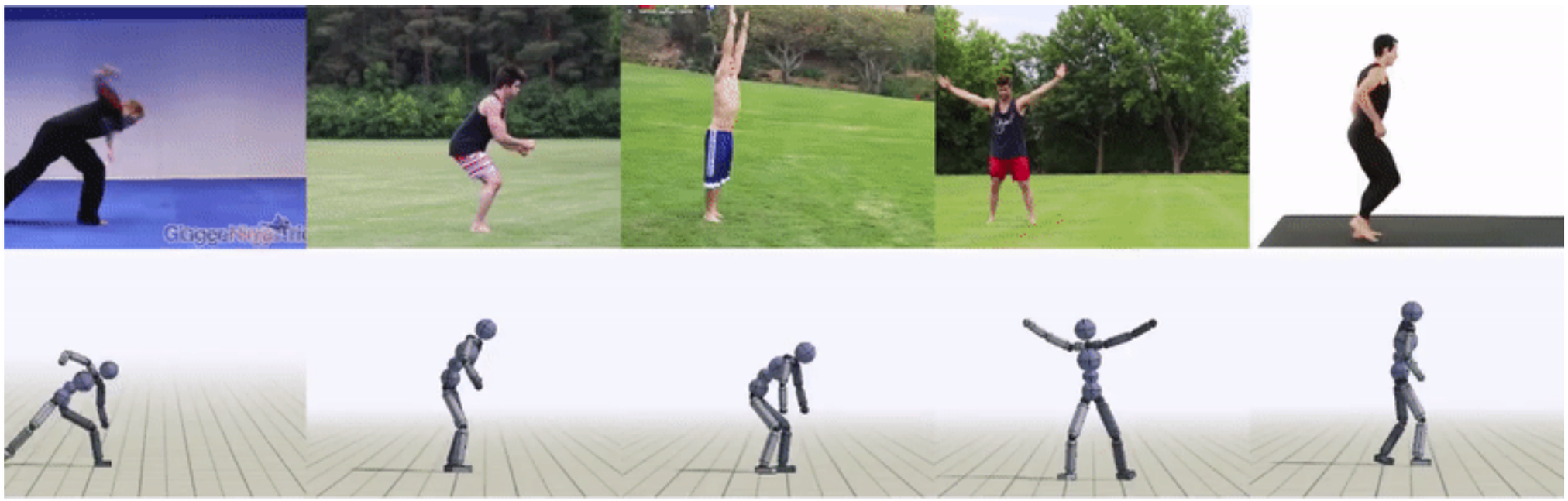
robot's imitation



- Central difficulty in visual imitation is perceiving the world state: where the objects are, in which pose, what velocities, etc.
- World state is available in simulation, e.g., if a human is demonstrating using a glove.
- For visual imitation though demonstation should be given in the real world else we beat the purpose.
- This means that Computer Vision really needs to work.

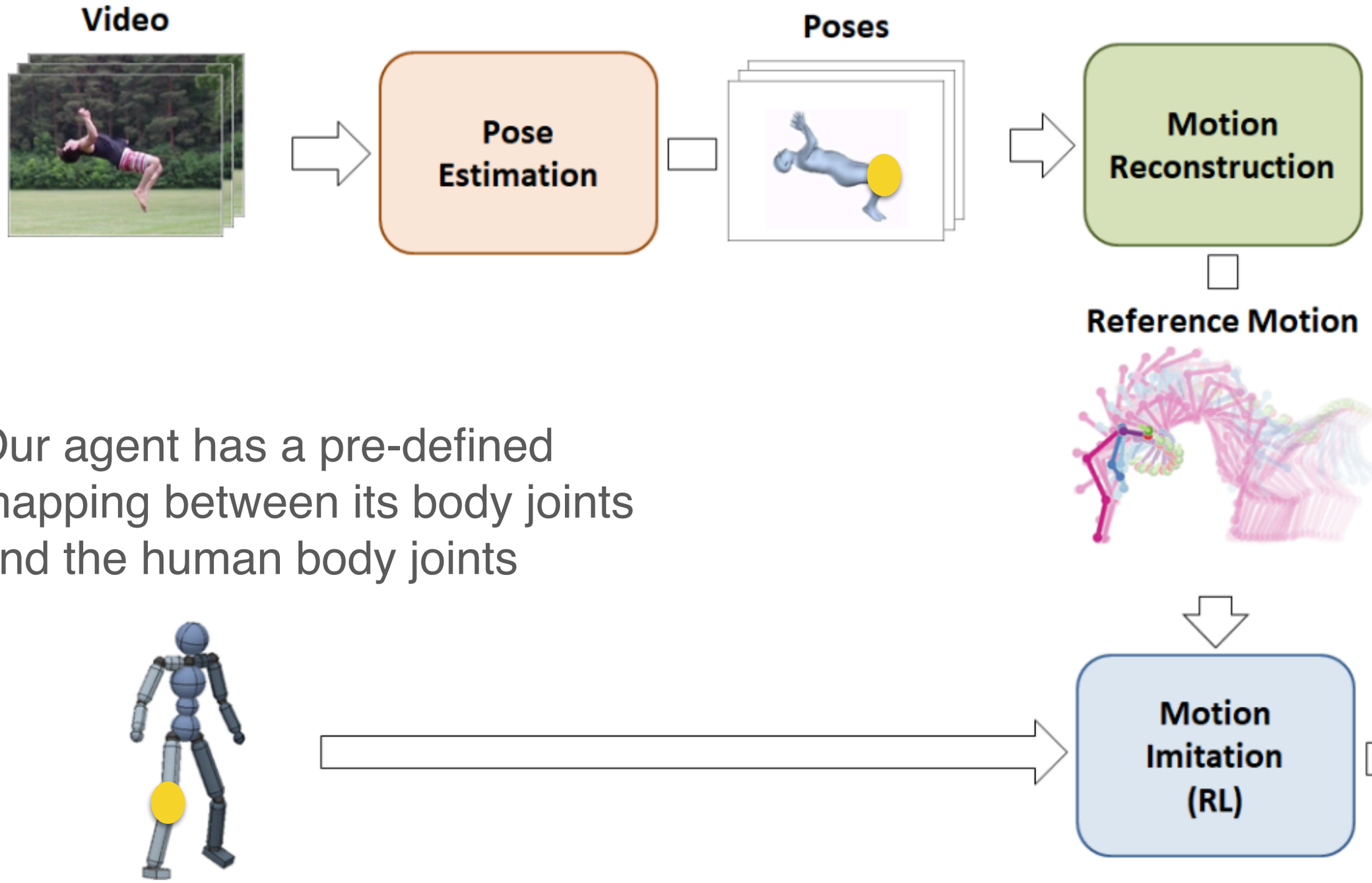
Learning acrobatics from watching Youtube

not by engineering the robot's actions



SFV: Reinforcement Learning of Physical Skills from Videos, Peng et al. 2018

Learning acrobatics from watching Youtube

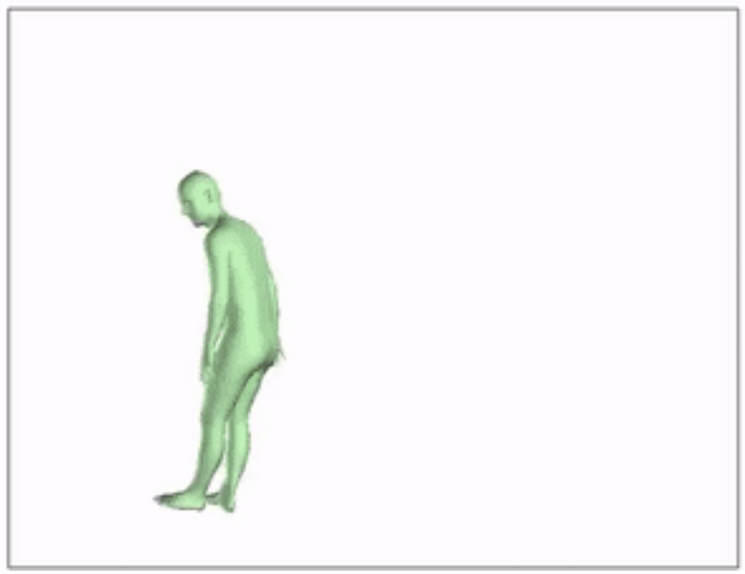


Our agent has a pre-defined mapping between its body joints and the human body joints

Q: Why we need RL? Why we do not just do behaviour cloning to imitate the reference motion sequence?

Imitating Humans by Inferring their 3D Poses

Handspring A



Backflip A

SMLP: a 3D human shape model

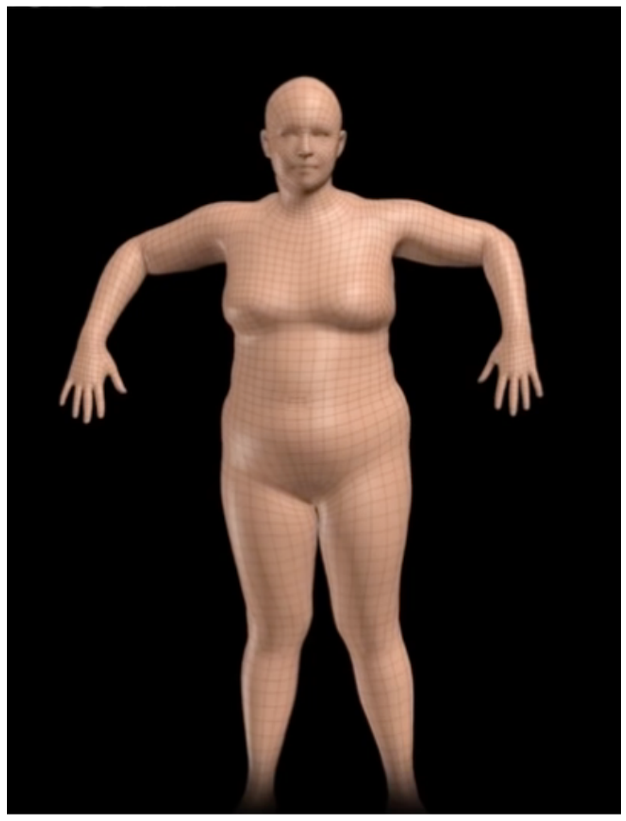
SMPL [M. Loper et al.]: a low-parametric model learned from aligning high-resolution 3D scans.

Pose θ

Shape β

3D mesh

SMPL(θ , β)



Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, Michael J. Black, SMPL: A Skinned Multi-Person Linear Model (SIGGRAPH Asia 2015)

SMLP: a 3D human shape model

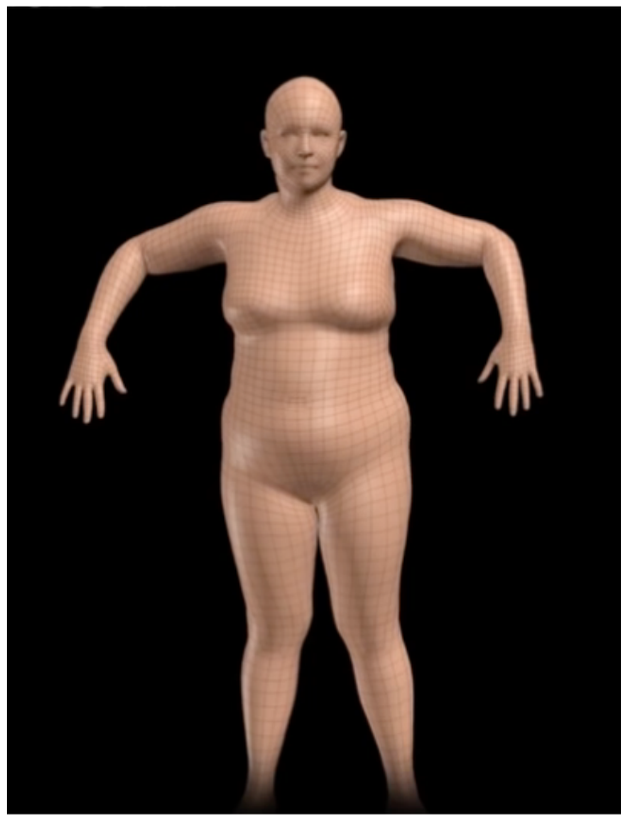
SMPL [M. Loper et al.]: a low-parametric model learned from aligning high-resolution 3D scans.

3D mesh

SMPL(θ , β)

Pose Shape

θ β



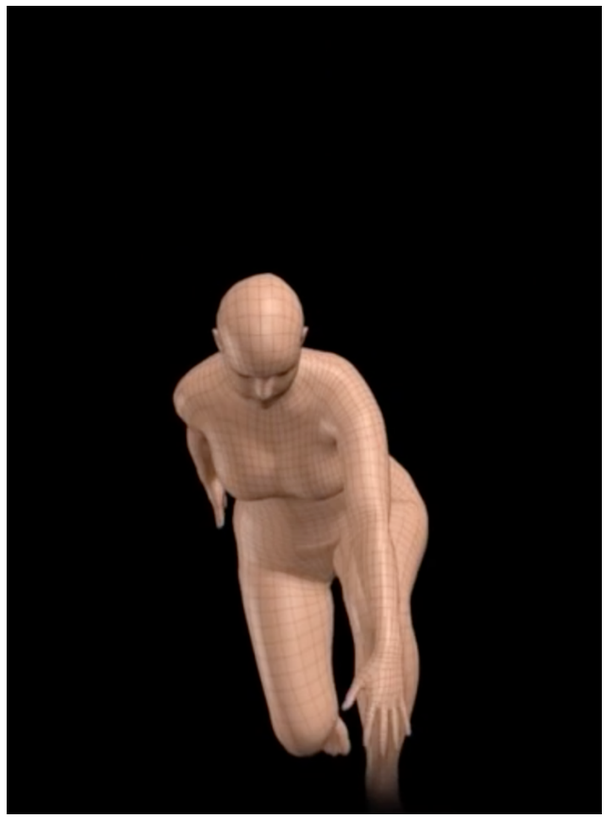
Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, Michael J. Black, SMPL: A Skinned Multi-Person Linear Model (SIGGRAPH Asia 2015)

SMLP: a 3D human shape model

SMPL [M. Loper et al.]: a low-parametric model learned from aligning high-resolution 3D scans.

Pose θ

Shape β



3D mesh

SMPL(θ , β)

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, Michael J. Black,
SMPL: A Skinned Multi-Person Linear Model (SIGGRAPH Asia 2015)

SMLP: a 3D human shape model

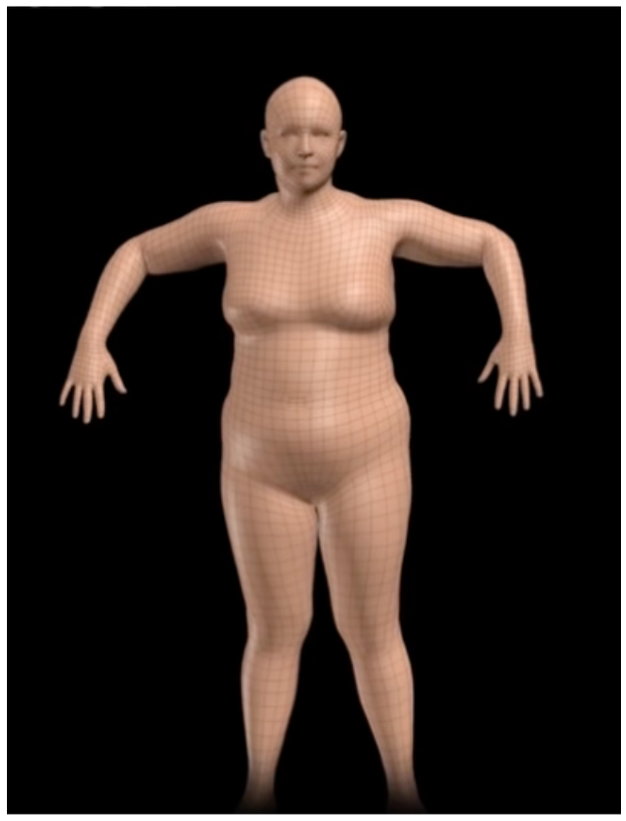
SMPL [M. Loper et al.]: a low-parametric model learned from aligning high-resolution 3D scans.

3D mesh

SMPL(θ , β)

Pose Shape

θ β



Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, Michael J. Black, SMPL: A Skinned Multi-Person Linear Model (SIGGRAPH Asia 2015)

SMLP: a 3D human shape model

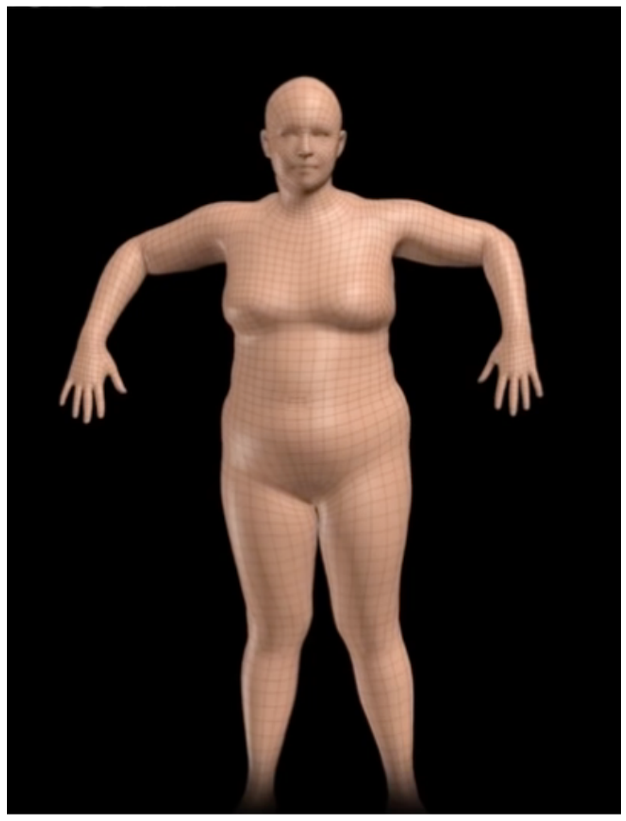
SMPL [M. Loper et al.]: a low-parametric model learned from aligning high-resolution 3D scans.

3D mesh

SMPL(θ , β)

Pose Shape

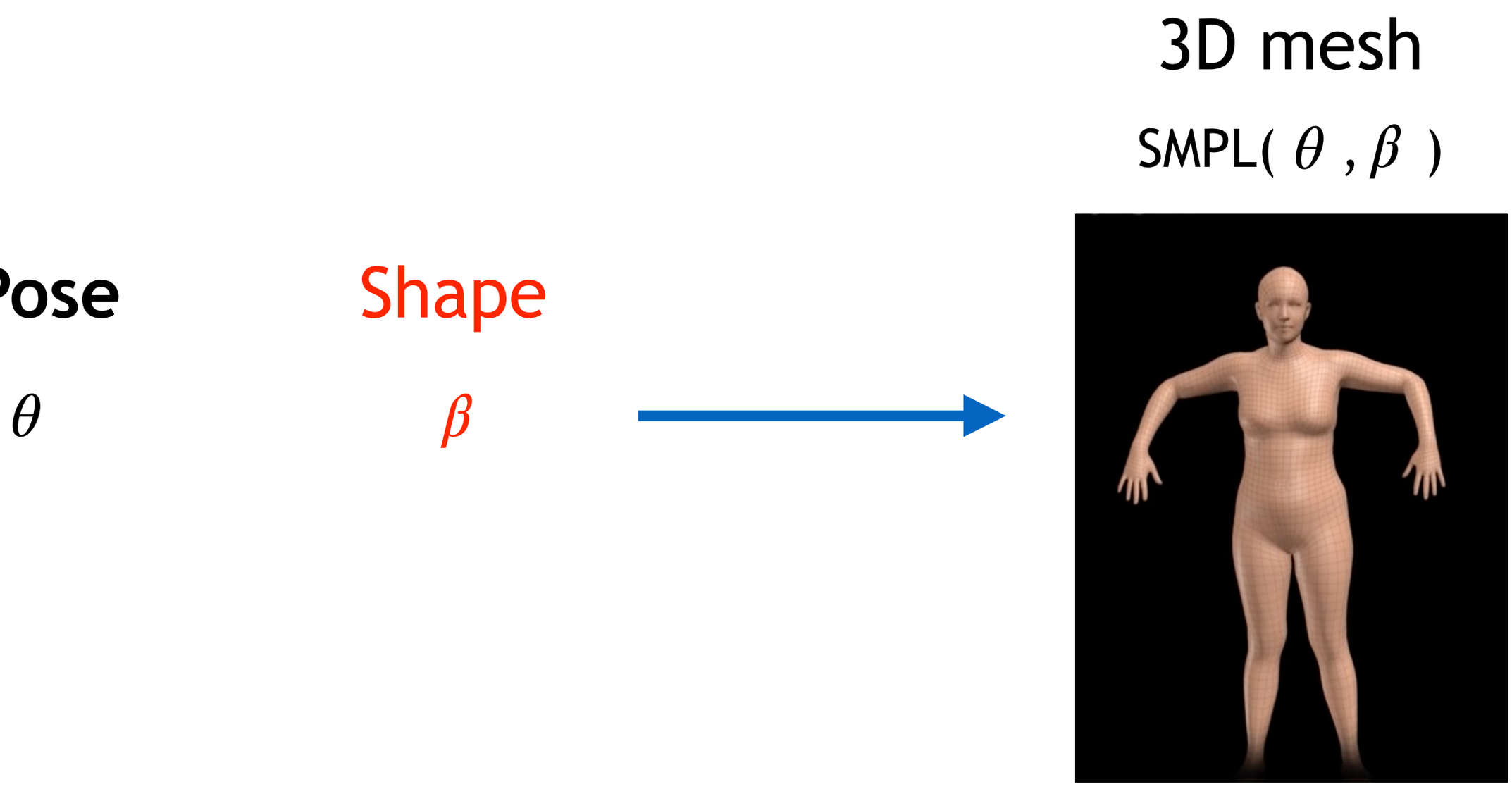
θ β



Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, Michael J. Black, SMPL: A Skinned Multi-Person Linear Model (SIGGRAPH Asia 2015)

SMLP: a 3D human shape model

SMPL [M. Loper et al.]: a low-parametric model learned from aligning high-resolution 3D scans.



Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, Michael J. Black, SMPL: A Skinned Multi-Person Linear Model (SIGGRAPH Asia 2015)

SMLP: a 3D human shape model

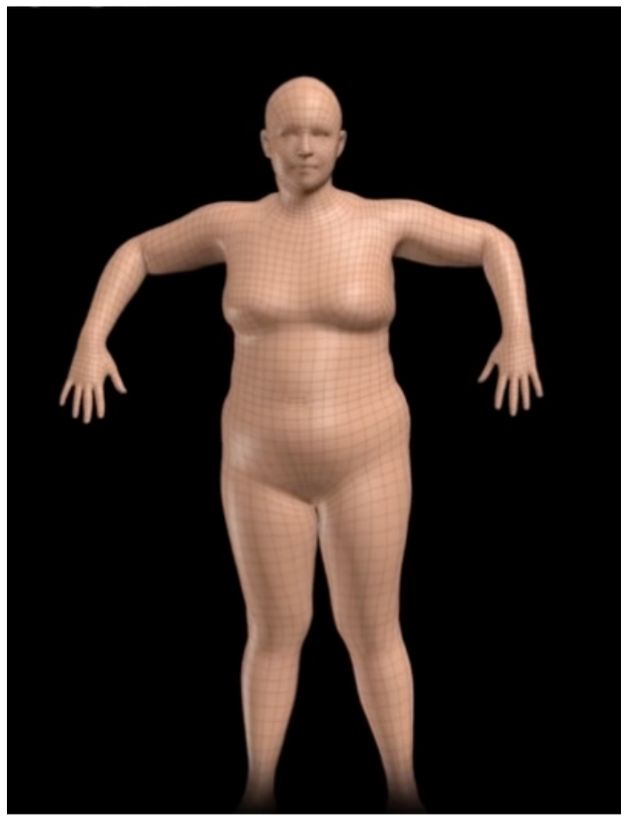
SMPL [M. Loper et al.]: a low-parametric model learned from aligning high-resolution 3D scans.

3D mesh

SMPL(θ , β)

Pose Shape

θ β



Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, Michael J. Black, SMPL: A Skinned Multi-Person Linear Model (SIGGRAPH Asia 2015)

SMLP: a 3D human shape model

SMPL [M. Loper et al.]: a low-parametric model learned from aligning high-resolution 3D scans.

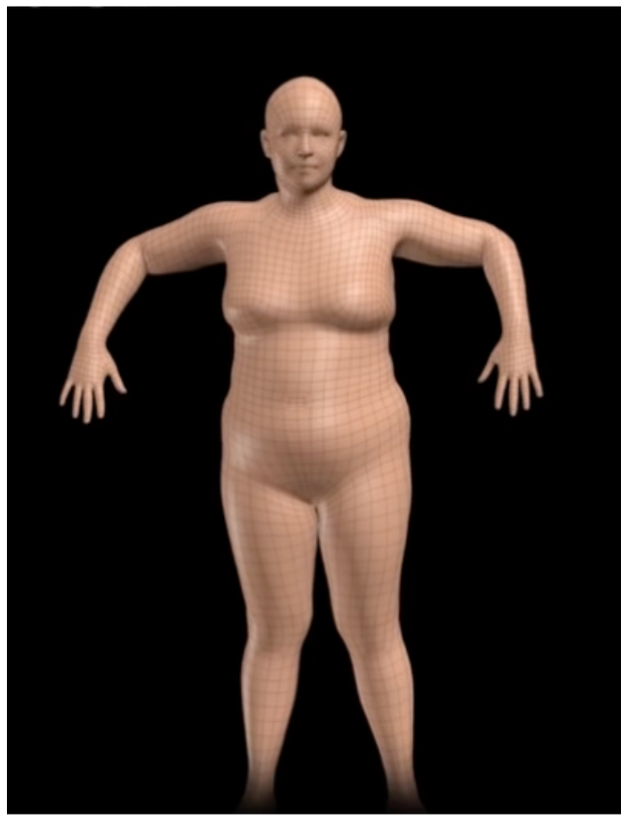
3D mesh

SMPL(θ , β)

Pose θ

Shape β

Differentiable mapping



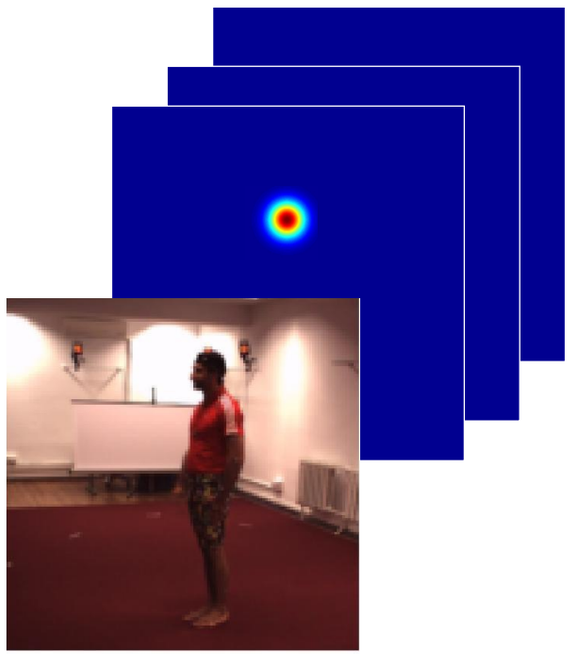
Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, Michael J. Black, SMPL: A Skinned Multi-Person Linear Model (SIGGRAPH Asia 2015)

3D body pose inference

Inputs:

RGB frame

2D keypoint heatmaps



3D body pose inference

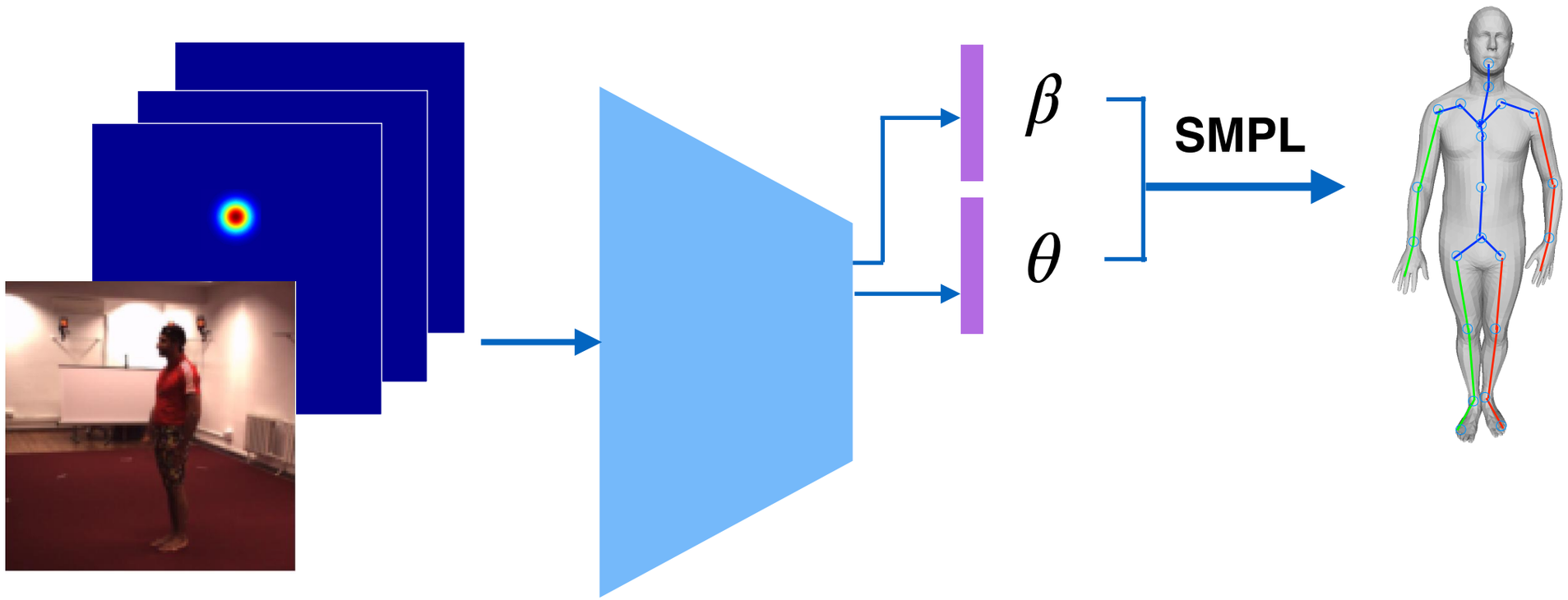
Inputs:

RGB frame

2D keypoint heatmaps

Outputs:

SMPL parameters (β, θ)



3D body pose inference

Inputs:

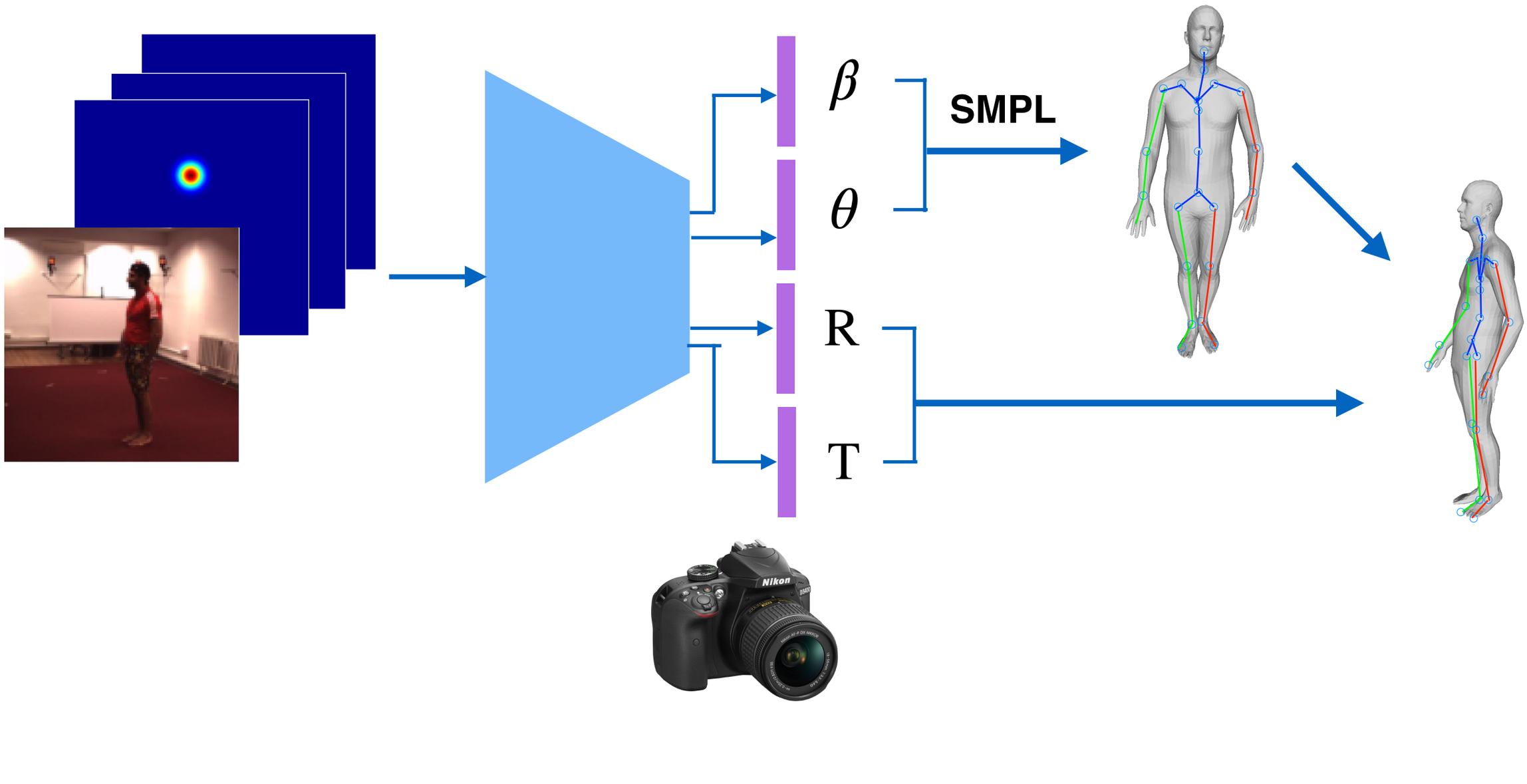
RGB frame

2D keypoint heatmaps

Outputs:

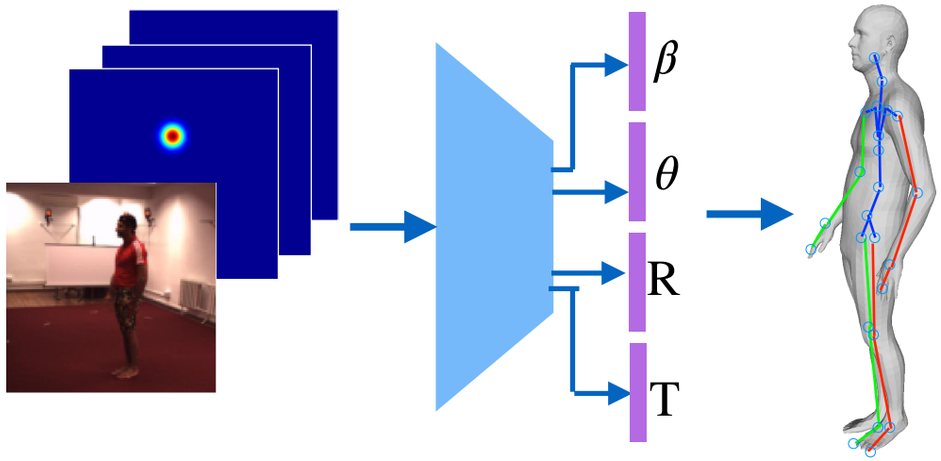
SMPL parameters (β, θ)

camera parameters(R, T)

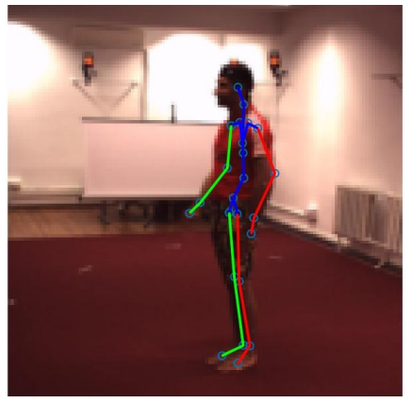


Self-supervised reprojection losses

Frame t

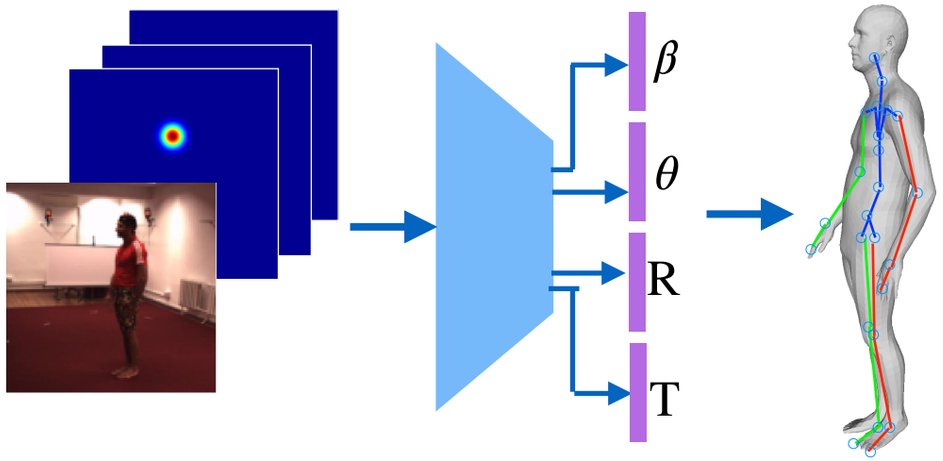


Keypoint re-projection

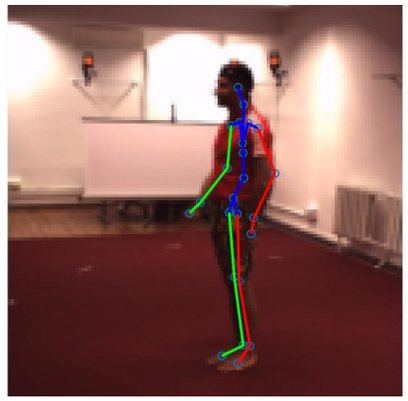


Self-supervised reprojection losses

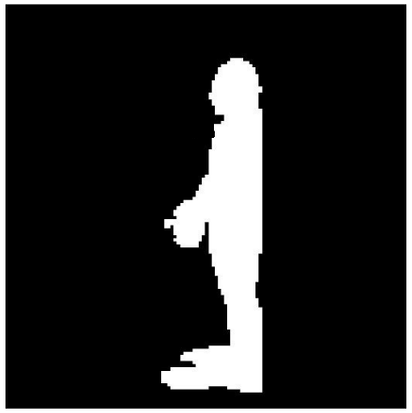
Frame t



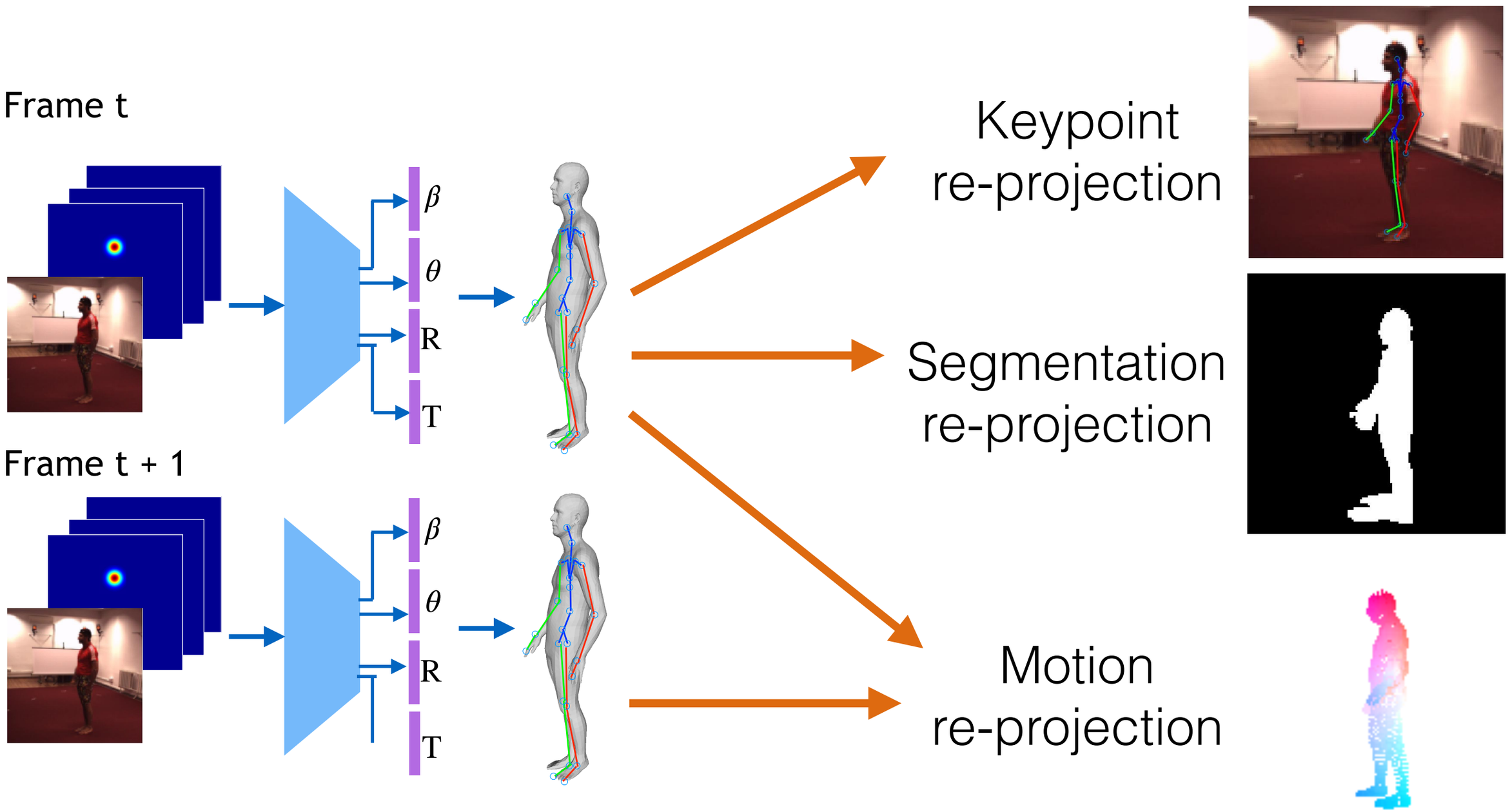
Keypoint re-projection



Segmentation re-projection



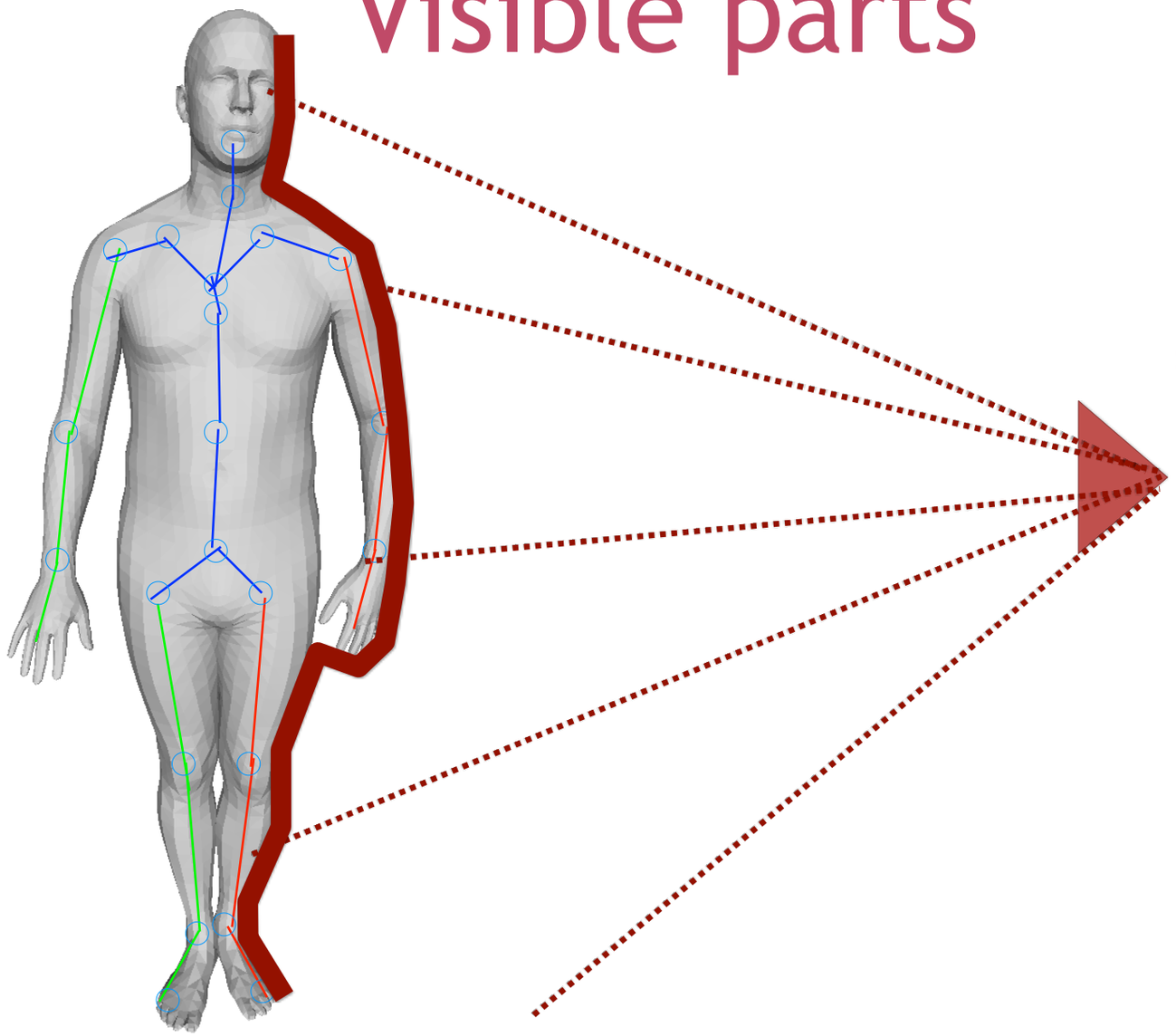
Self-supervised reprojection losses



Visibility-aware reprojection

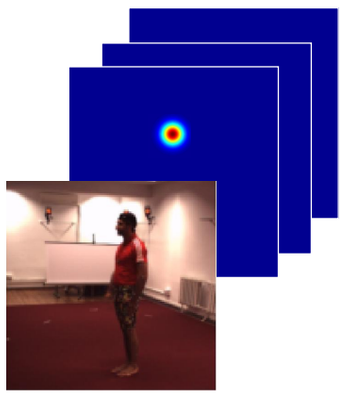
Visible parts

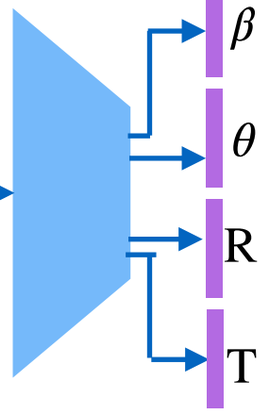
Occluded parts

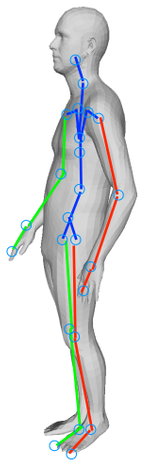


Camera

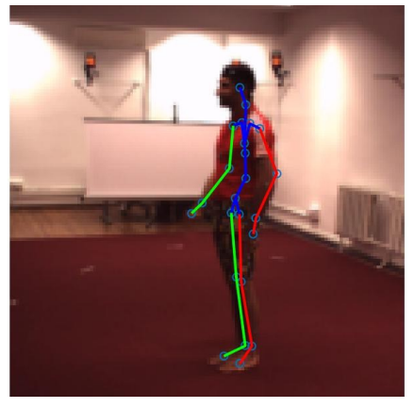
Frame t



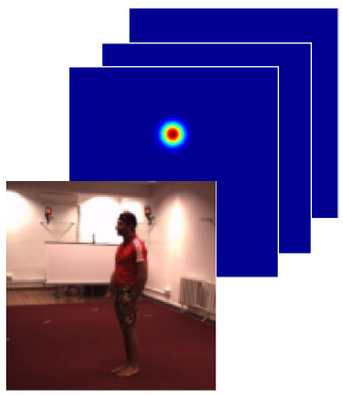


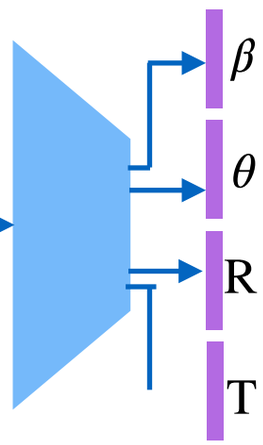


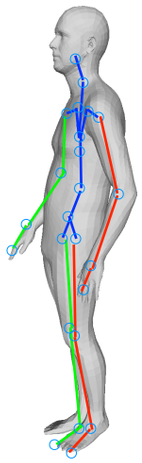
Keypoint re-projection



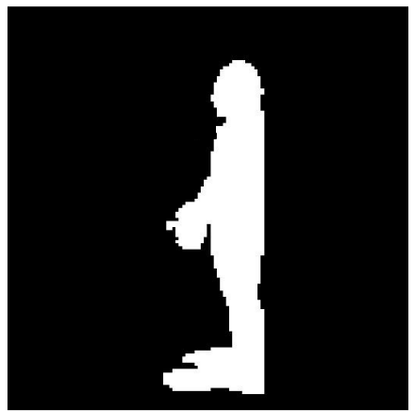
Frame t + 1







Segmentation re-projection

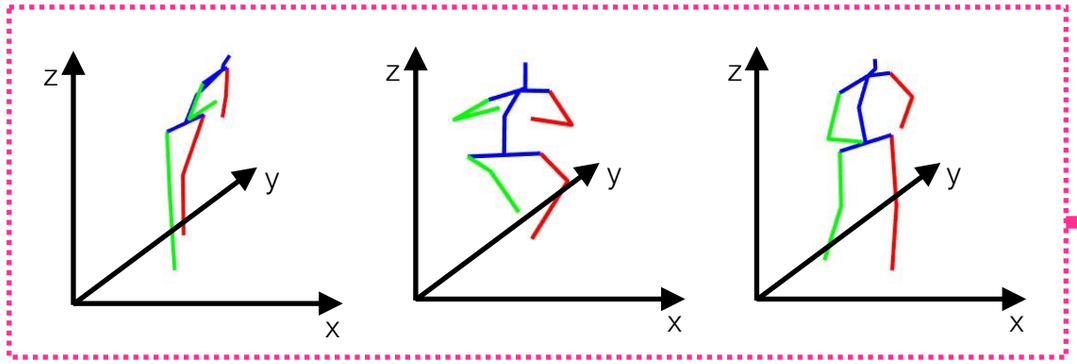


Motion re-projection

Q: Can such re-projection losses result in a non-anthropomorphic looking 3D human body pose?

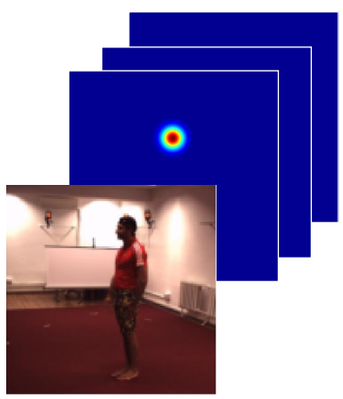
Adversarial matching

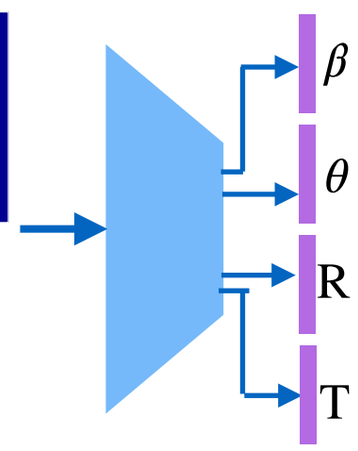
3D human poses

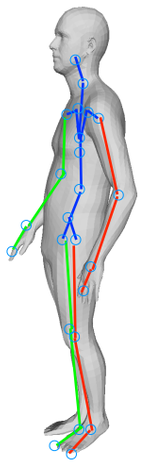


Discriminator

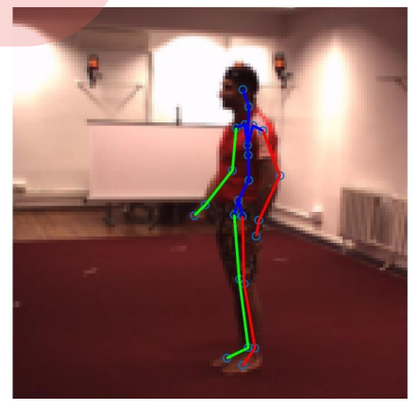
Frame t



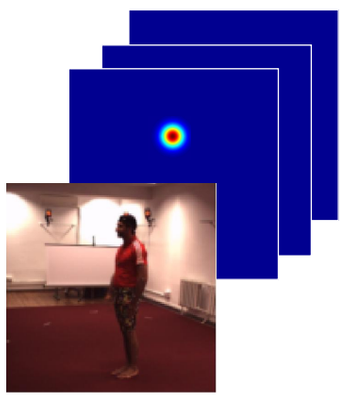


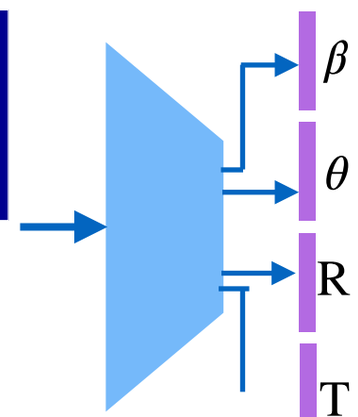


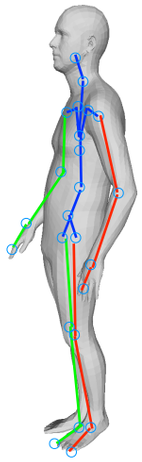
Keypoint re-projection



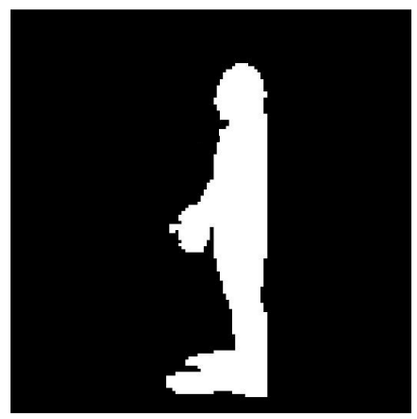
Frame t + 1







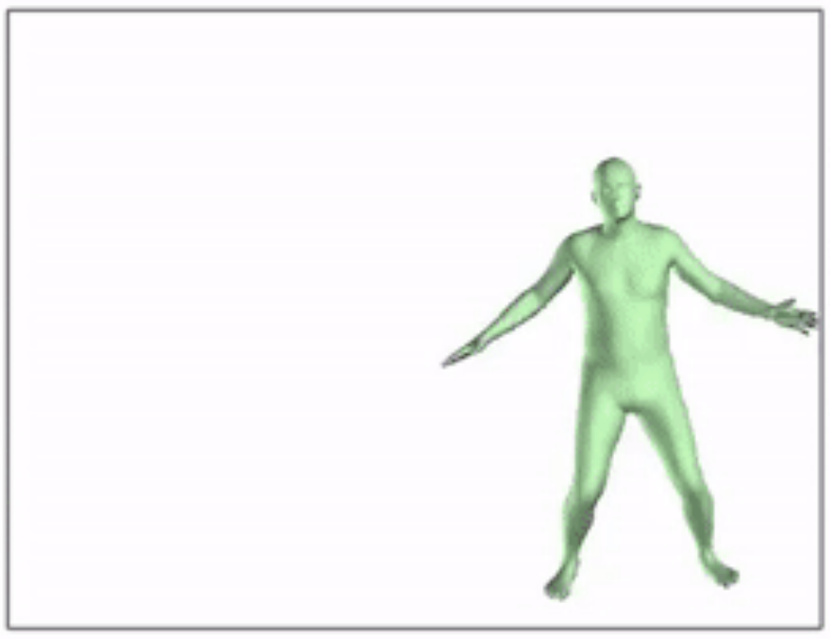
Segmentation re-projection



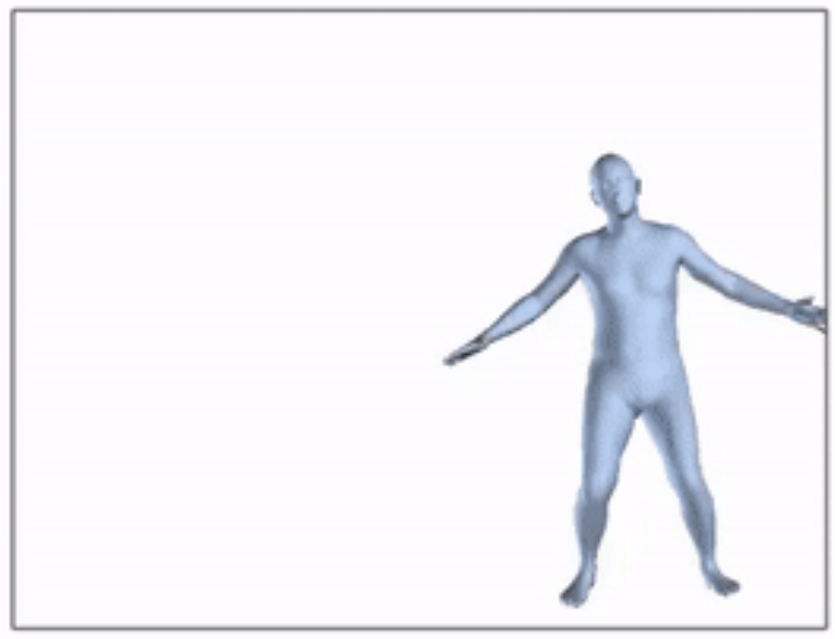
Motion re-projection

+temporal smoothing

Video: Cartwheel A

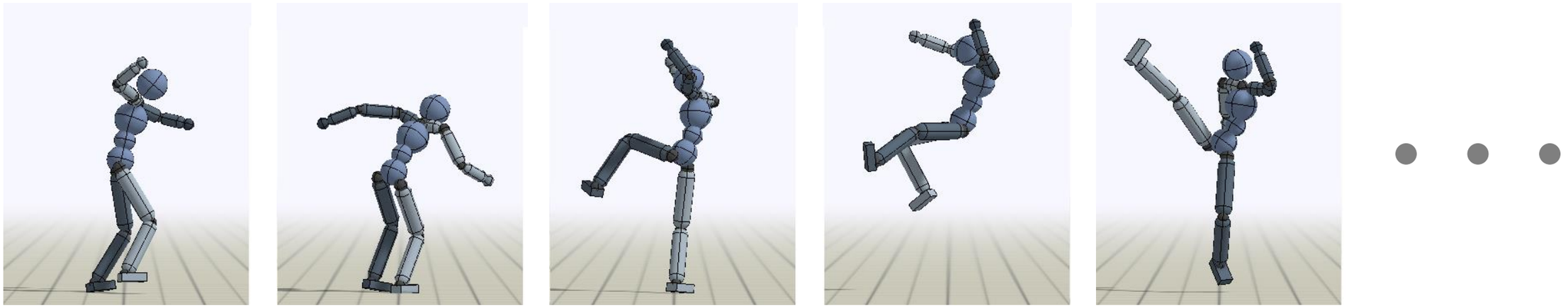


Before Reconstruction



After Reconstruction

Reference Motion

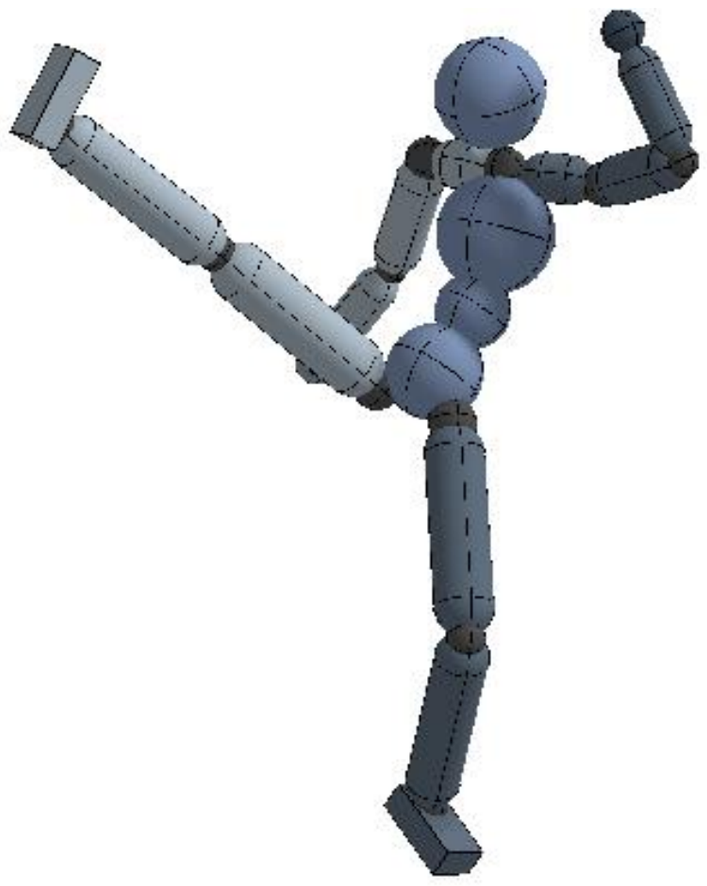


a_0 a_1 a_2 a_3 a_4 • • •

State + Action

State:

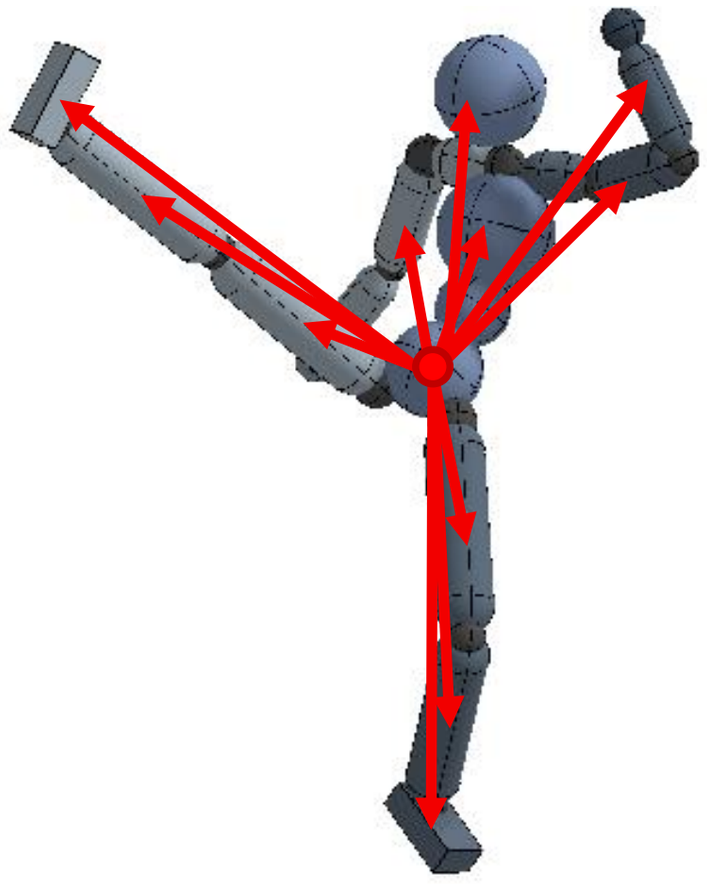
- link positions
- link velocities



State + Action

State:

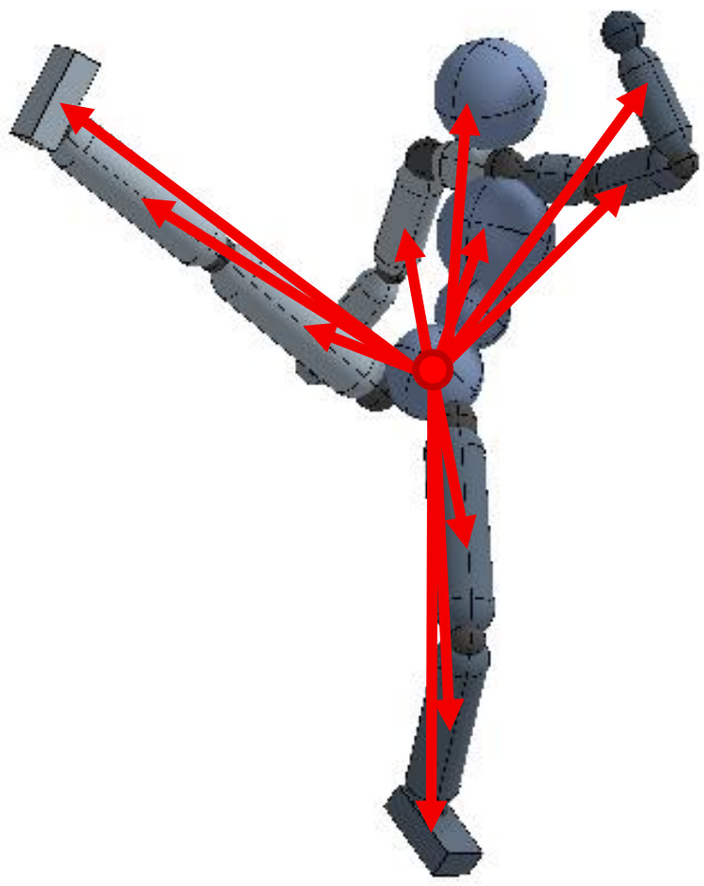
- link positions
- link velocities



State + Action

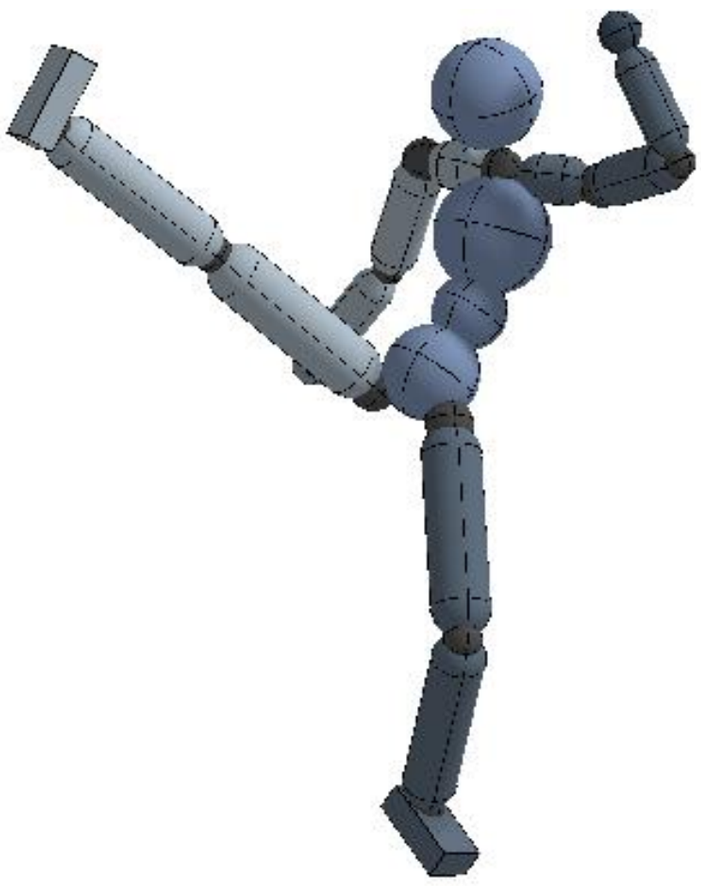
State:

- link positions
- link velocities



Action:

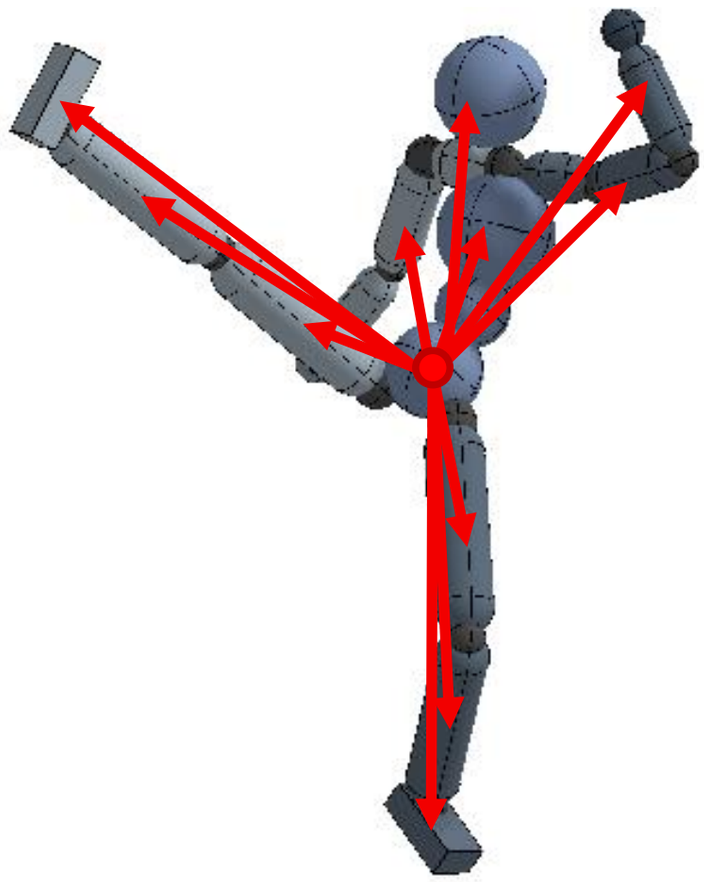
- PD targets



State + Action

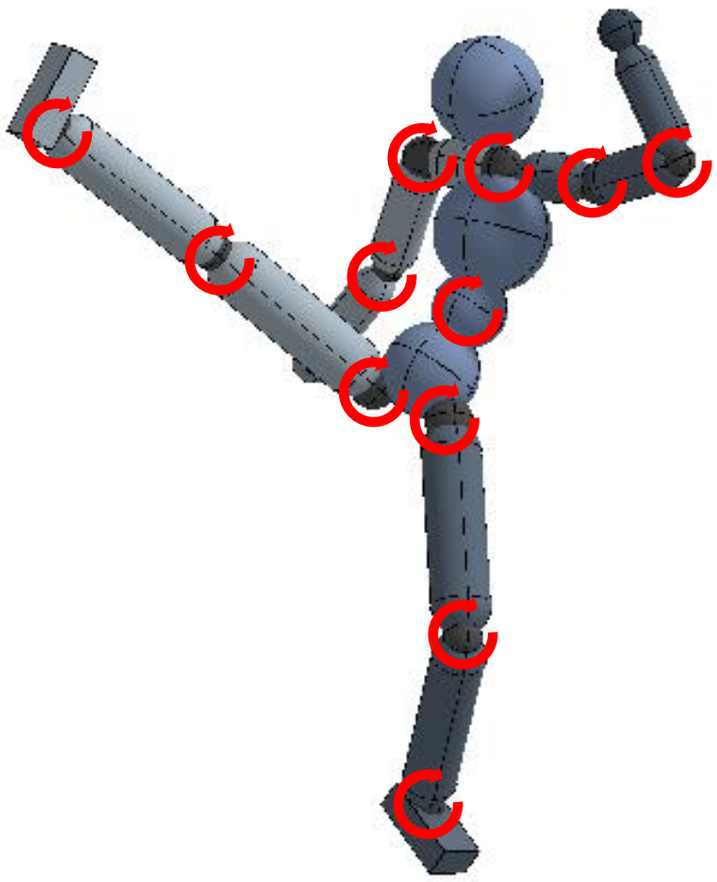
State:

- link positions
- link velocities



Action:

- PD targets

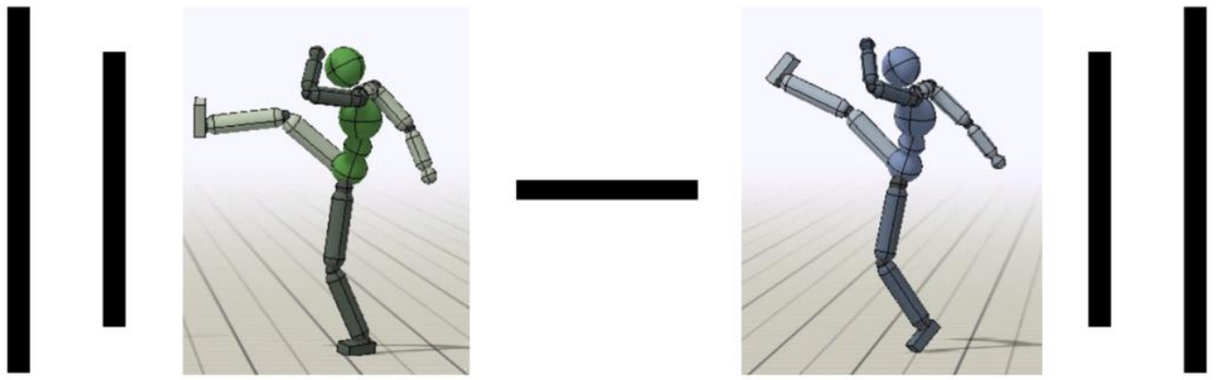


Imitation Objective

the reference trajectory

$$r_t = \exp\left(-2\|\hat{q}_t - q_t\|^2 \right)$$

Imitation Objective



Proximal Policy Optimization (PPO)

$$\max_{\theta} \quad J(\theta)$$

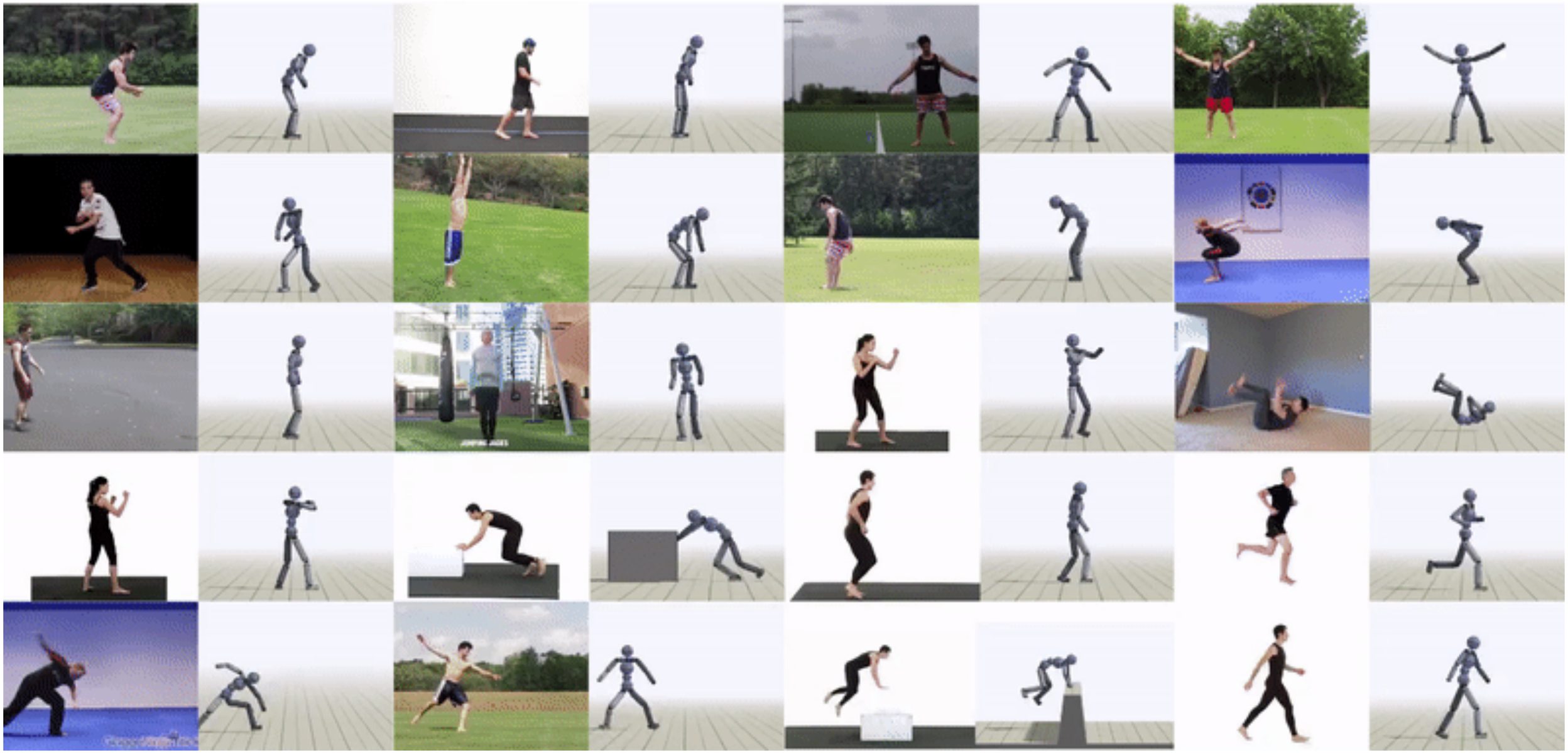
[Schulman et al. 2017]

Proximal Policy Optimization (PPO)

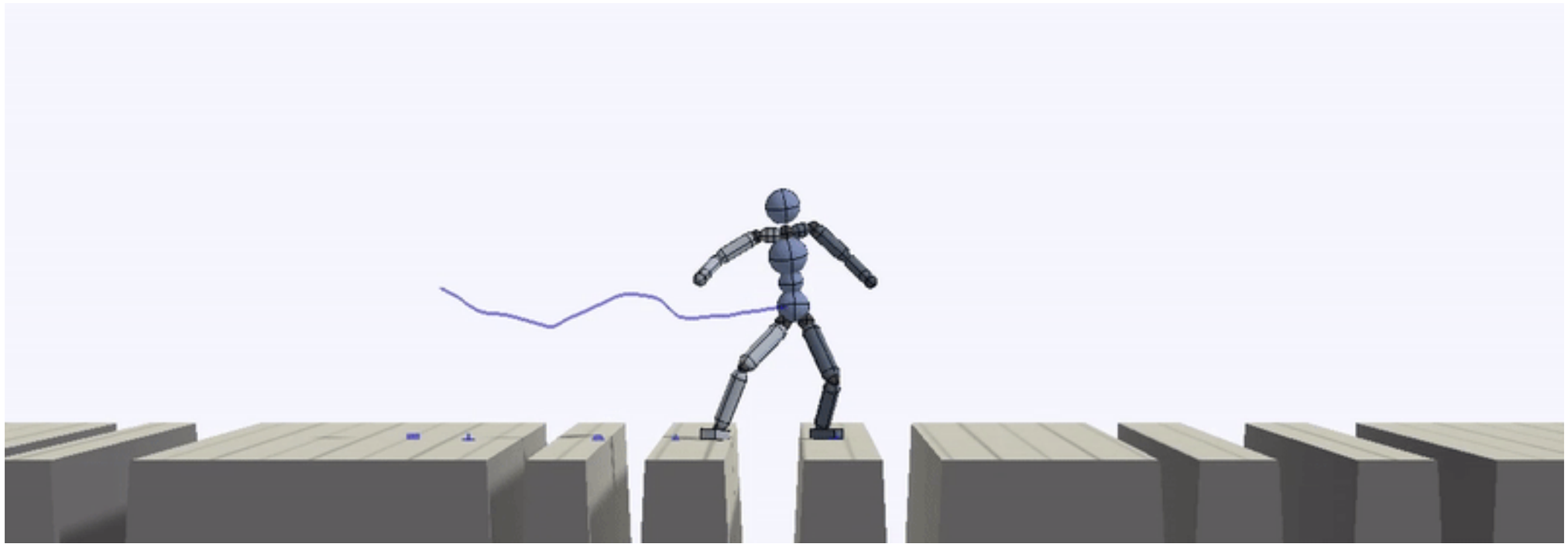
$$\max_{\theta} \quad J(\theta)$$

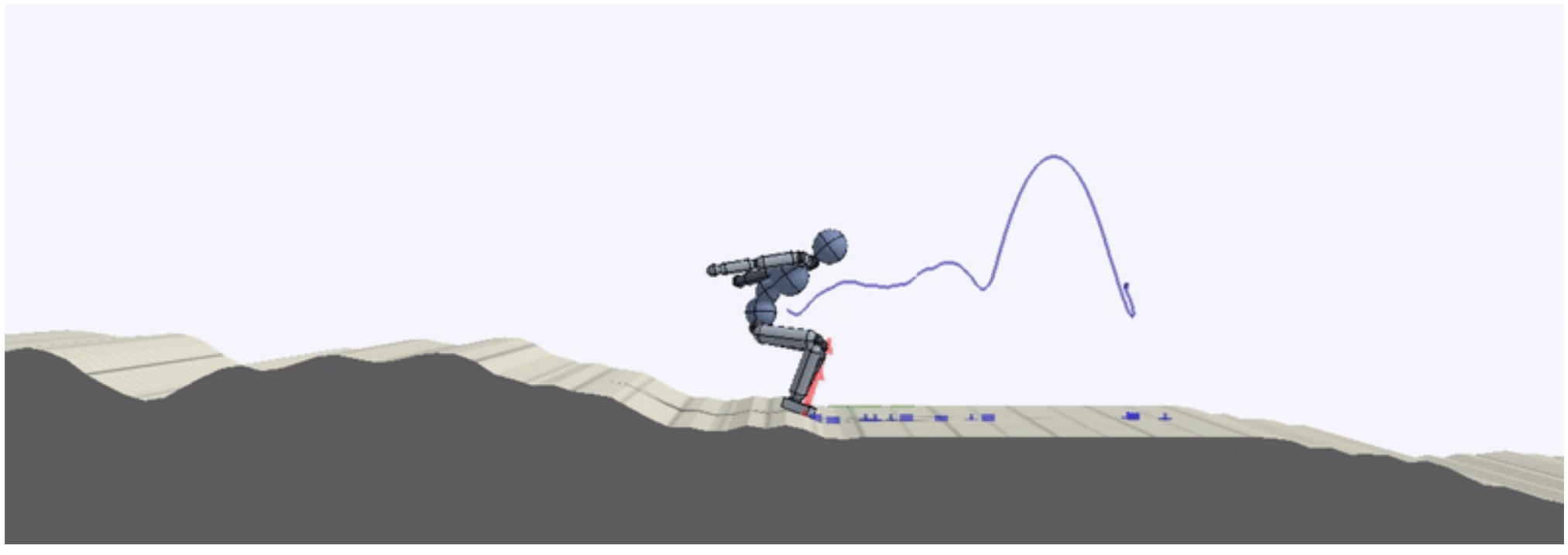
$$\text{s.t.} \quad \mathbb{E}_{s_t \sim d_\theta(s_t)} \left[KL \left(\pi_{\theta_{old}}(\cdot|s_t) \middle| \pi_\theta(\cdot|s_t) \right) \right] \leq \delta_{KL}$$

[Schulman et al. 2017]



Adapting a skill through RL to novel environments





Failure modes

Video: Gangnam Style

Reference Motion

Simulation

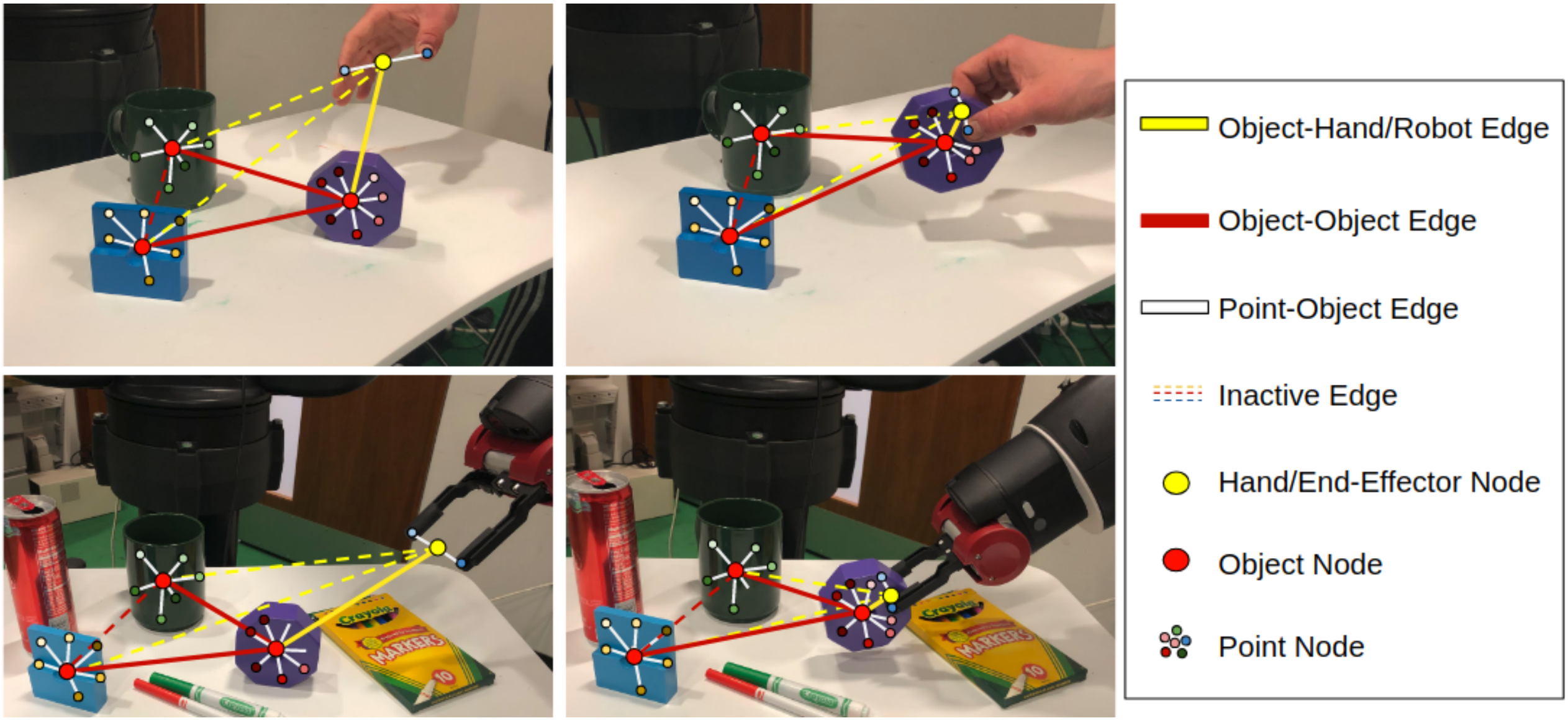
Imitating beyond human body pose

- Imitating human body was possible thanks to progress in Computer Vision that can detect 2D human body keypoints and reconstruct them in 3D very well.
- What about the rest of the objects in the world, that we cannot yet easily 3D reconstruct?

Imitating beyond human body pose

- We will use object keypoints (similar to human keypoints) and learn detectors on-the-fly to track them and imitate them-> visual entity graphs for visual imitation
- We will imitate in the real world directly, using iLQR

Visual Entity Graphs for Visual Imitation

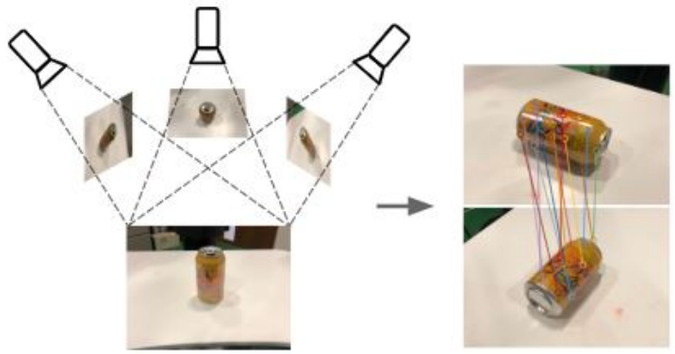


Detecting Visual Entities

Hand Keypoint Detection

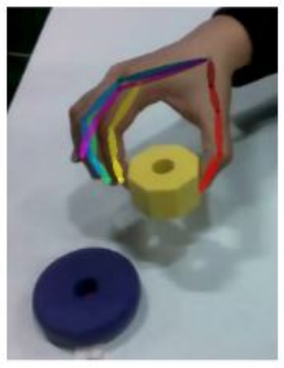
Detecting Visual Entities

Hand Keypoint Detection

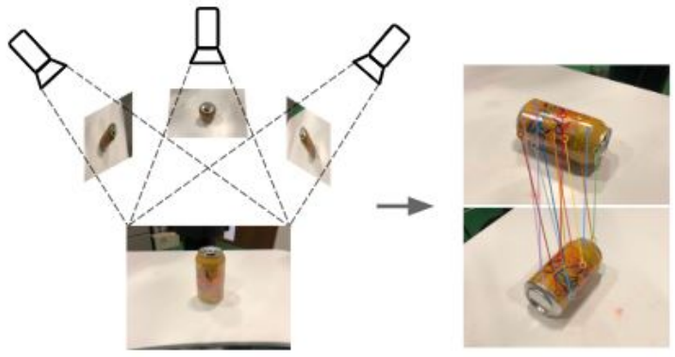


Multi-View Self-Supervised Point-Feature Learning

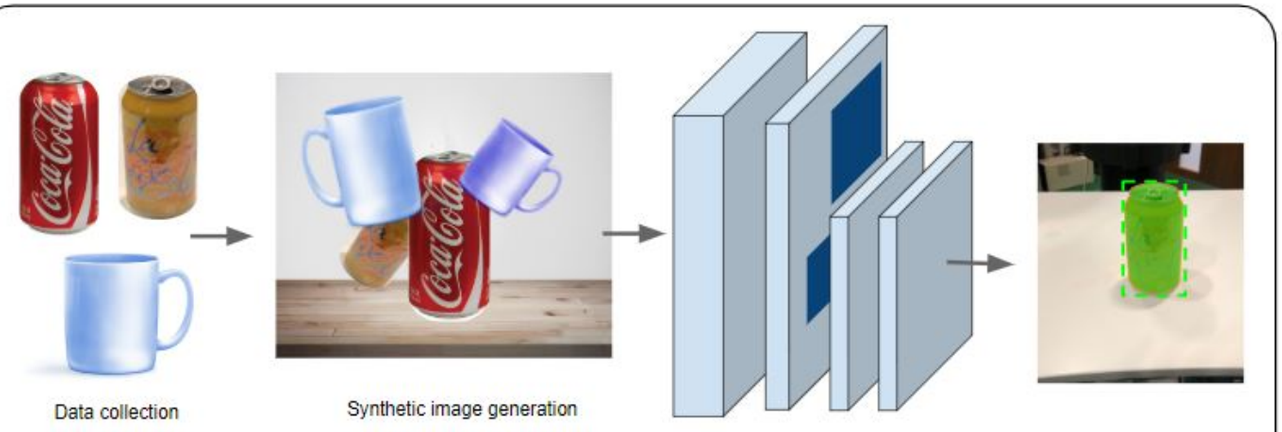
Detecting Visual Entities



Hand Keypoint Detection

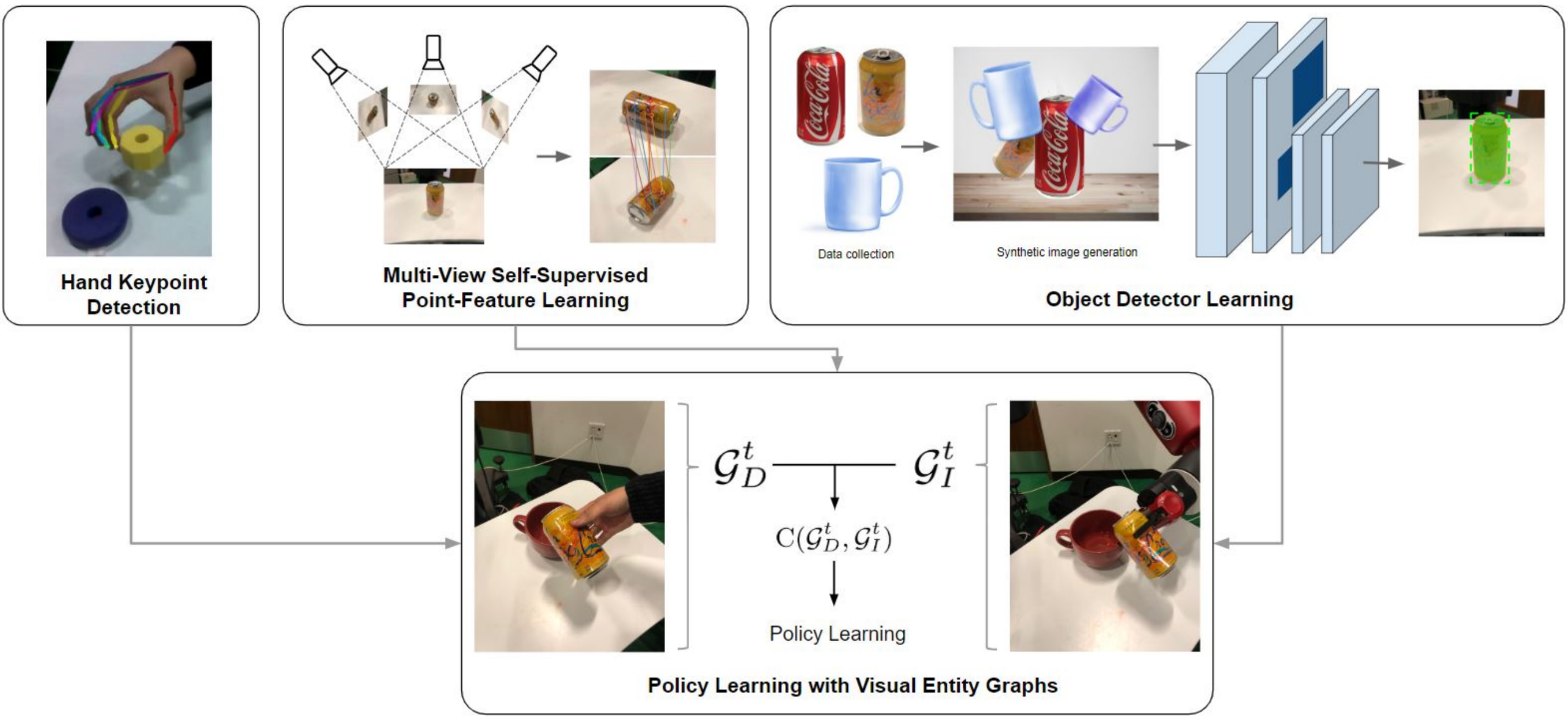


Multi-View Self-Supervised Point-Feature Learning



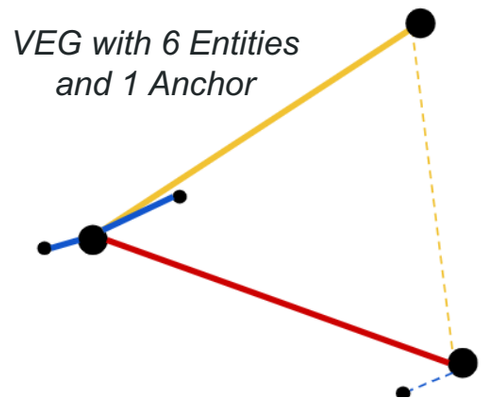
Object Detector Learning

Detecting Visual Entities



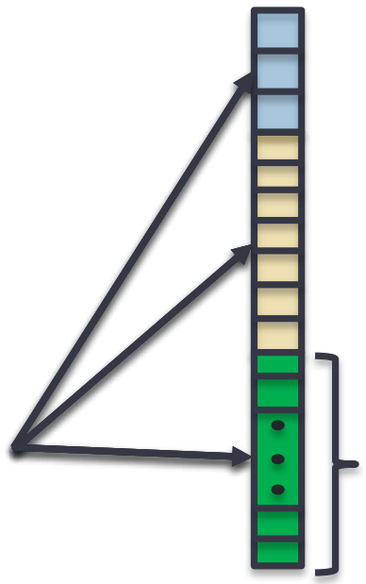
iLQR with VEGs

State Space



$$\mathbf{z} = \begin{bmatrix} \text{end-effector } (X, Y, Z) \text{ pose} \\ \text{joint positions} \\ \text{relative spatial arrangements} \end{bmatrix}$$

State



- end-effector (X, Y, Z) pose $\in \mathbb{R}^3$
- joint positions $\in \mathbb{R}^7$
- relative spatial arrangements $\in \mathbb{R}^{d_s}$, $d_s = \sum_{i=1}^{N_{anchors}} [(N_{total} - 1) * 3] + N_{anchors} * 3$

Object & hand/robot entities

Sum over all object-hand/robot edges if connected anchor

(x, y, z)-average over all point-object edges for each anchor

Cost

Sum over all entity pairs with **active** edges

$$\mathcal{C}(\mathcal{G}_D^t, \mathcal{G}_I^t) = \sum_{i,j,i<j} w(\mathcal{E}^{t,(i,j)}) \cdot \text{att}(\mathcal{E}^{t,(i,j)}) \cdot \left\| \left(\mathbf{x}_D^{t,i} - \mathbf{x}_D^{t,j}\right) - \left(\mathbf{x}_I^{t,i} - \mathbf{x}_I^{t,j}\right) \right\|$$

Scalar edge weight

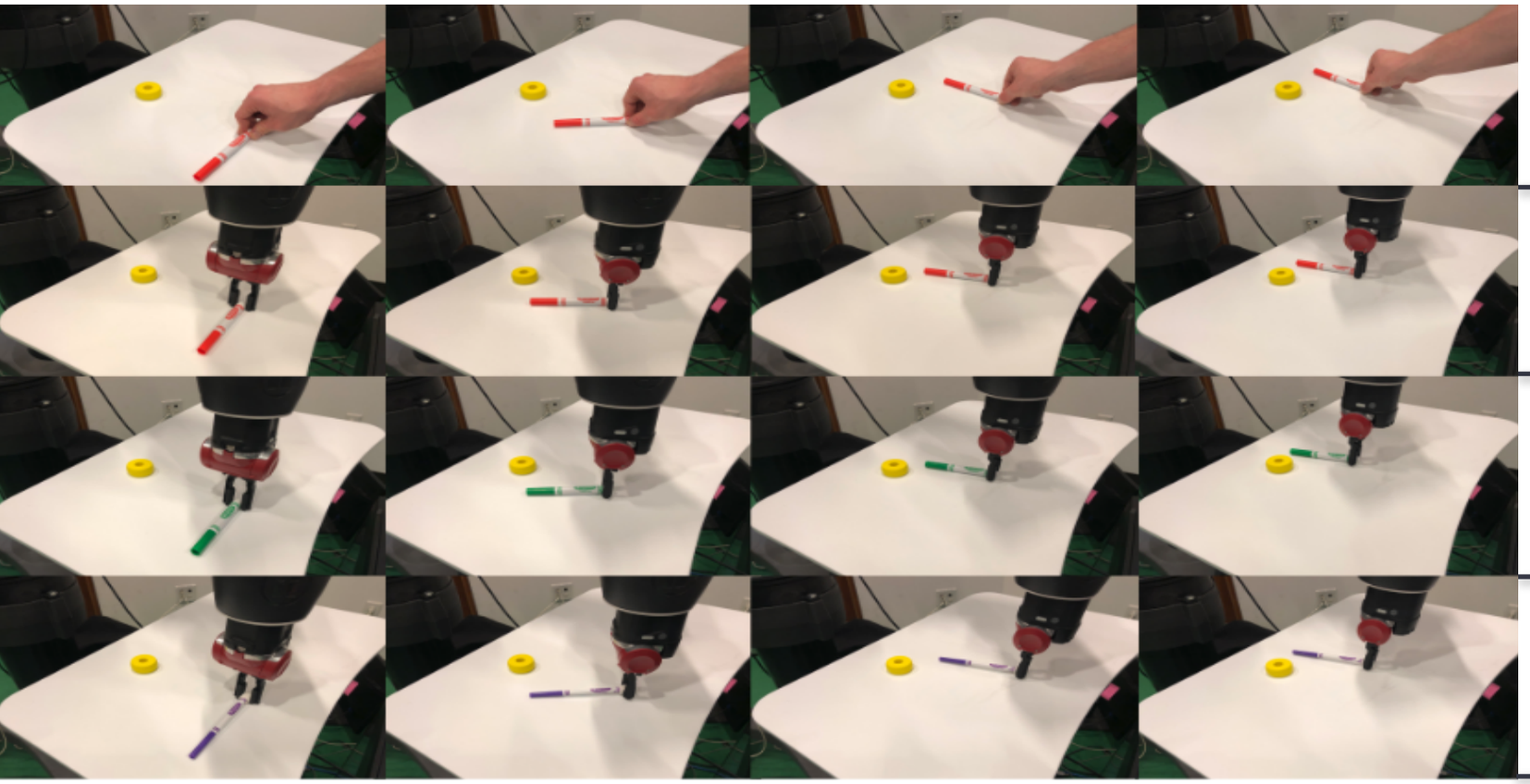
Binary **motion-guided** attention

Spatial arrangement for entity *i* and *j* in **demonstration**

Spatial arrangement for entity *i* and *j* in **imitation**

Difference in spatial arrangement for entity *i* and *j*

Generalization Capability of Visual Entity Graphs

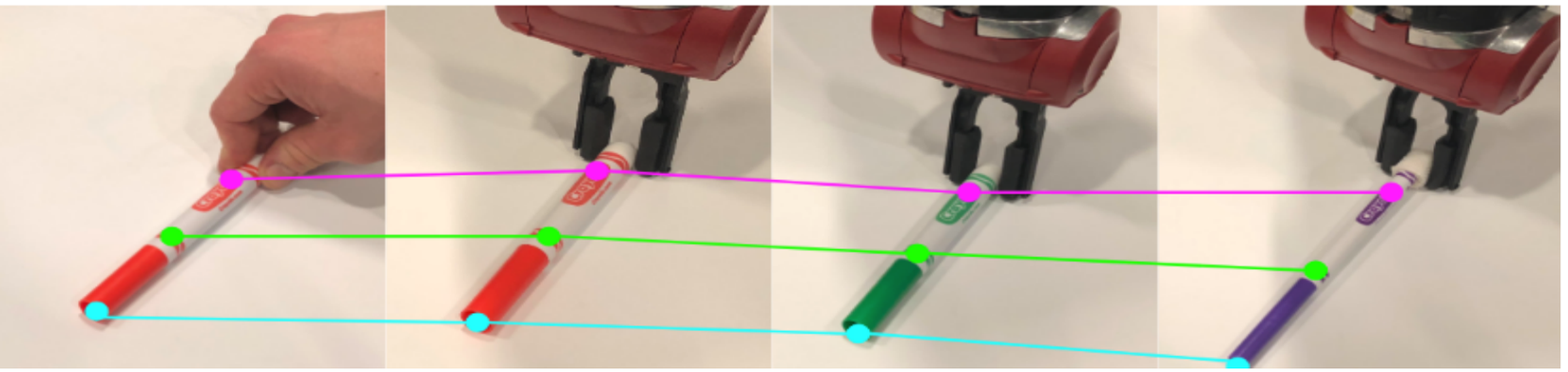


Human demonstration

Imitation **same** object

Imitation **different** object (in train set)

Imitation **different** object (**not** in train set)



Feature correspondence generalizes across different object instances

Is this how babies imitate?

- Doing nothing till someone shows them a visual demonstation, and then they get to work?
- No. They are quite busy even on their own exploring the world and building models for it.
- Then, they make use of those models to accelerate imitation.
- Q: Did the imitation methods we showed used any model knowledge?

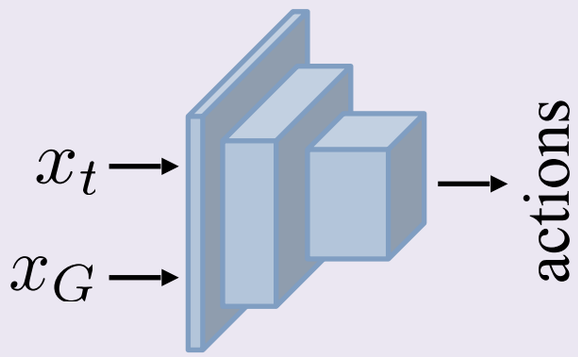
Hypothesis of how babies imitate

TRAINING TIME

TEST TIME

Explore the Environment

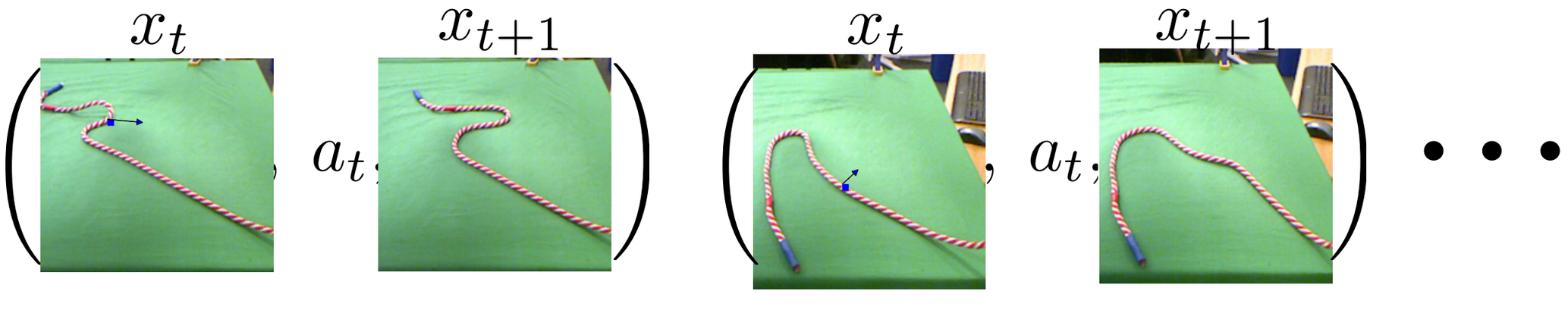
Distill Exploration into skills



Perform End-Tasks

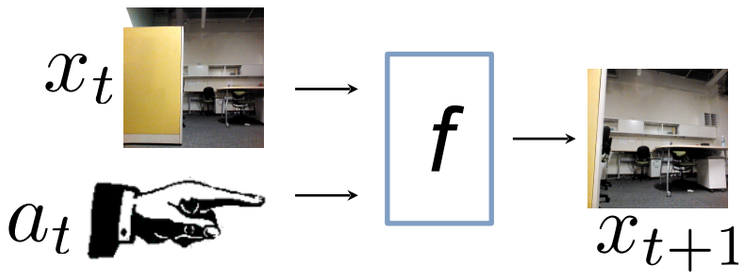
Model learning by random exploration

Rope Manipulation

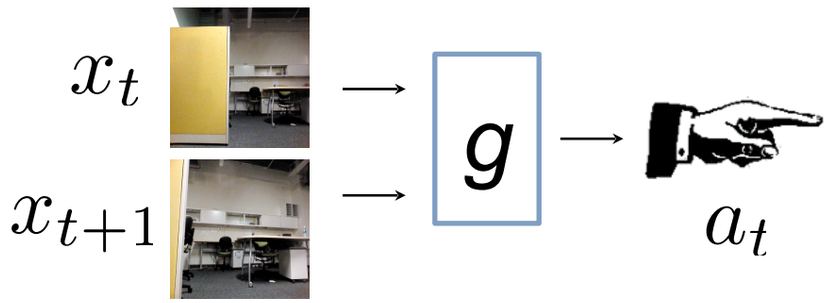


Visual Navigation

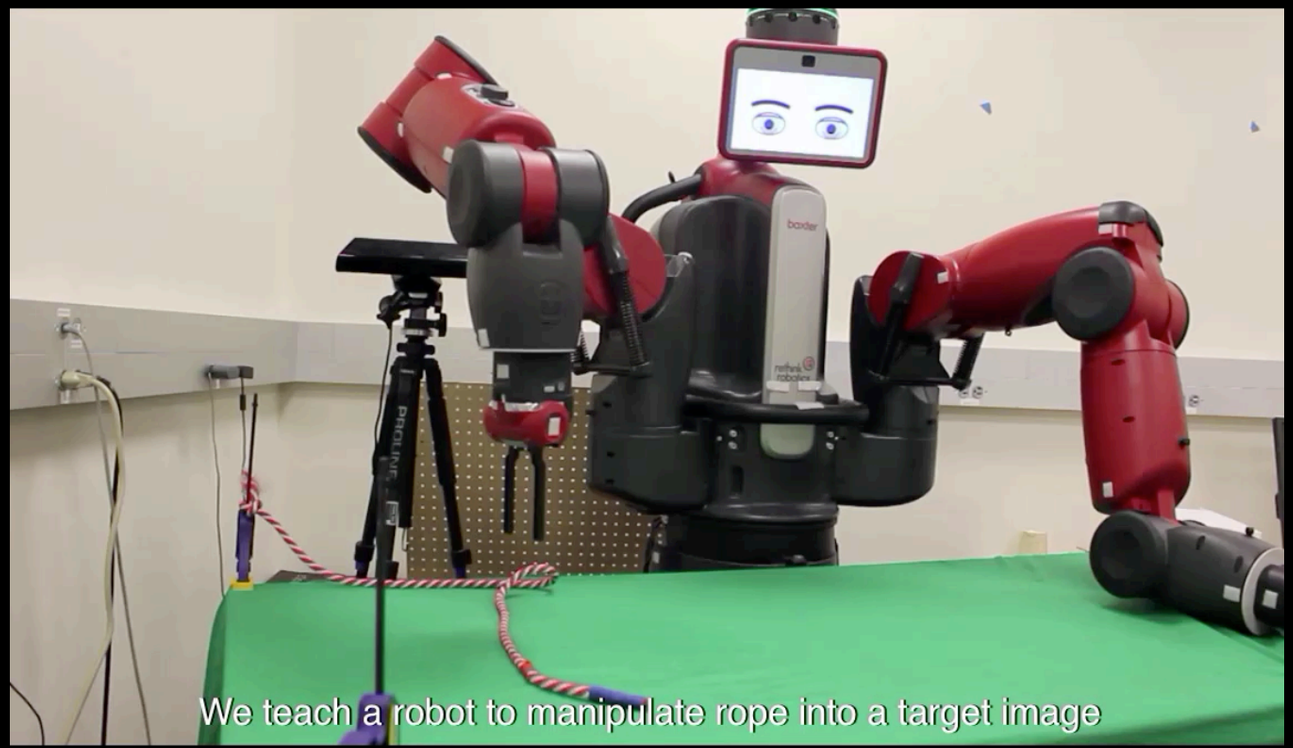
Forward Dynamics



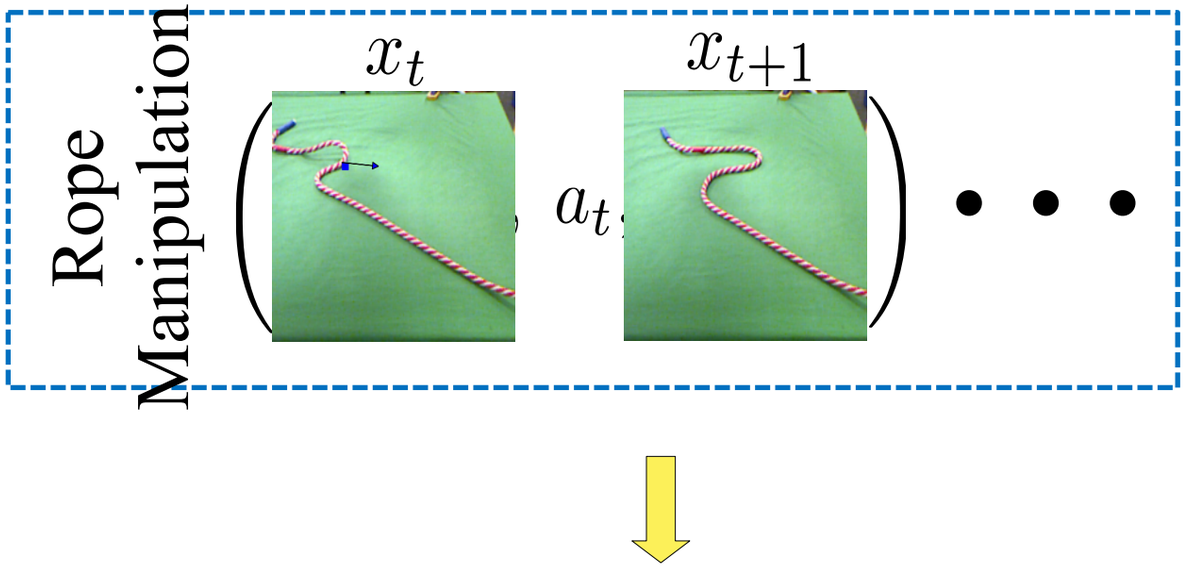
Inverse Dynamics

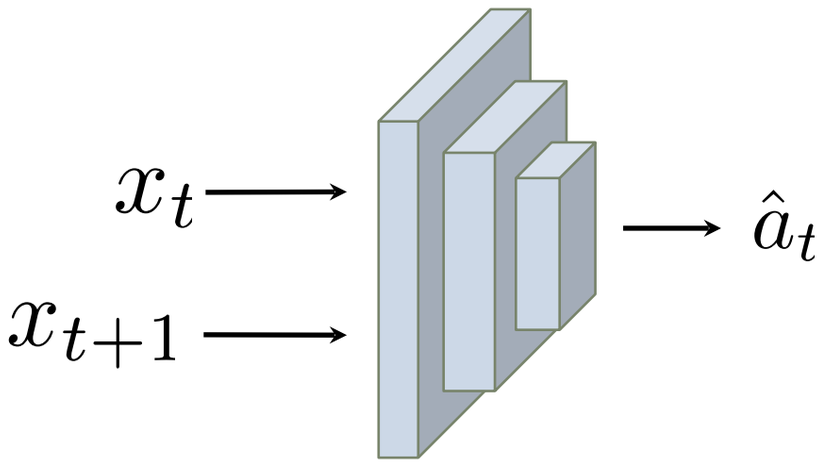


Exploring the Environment



Model learning by random exploration

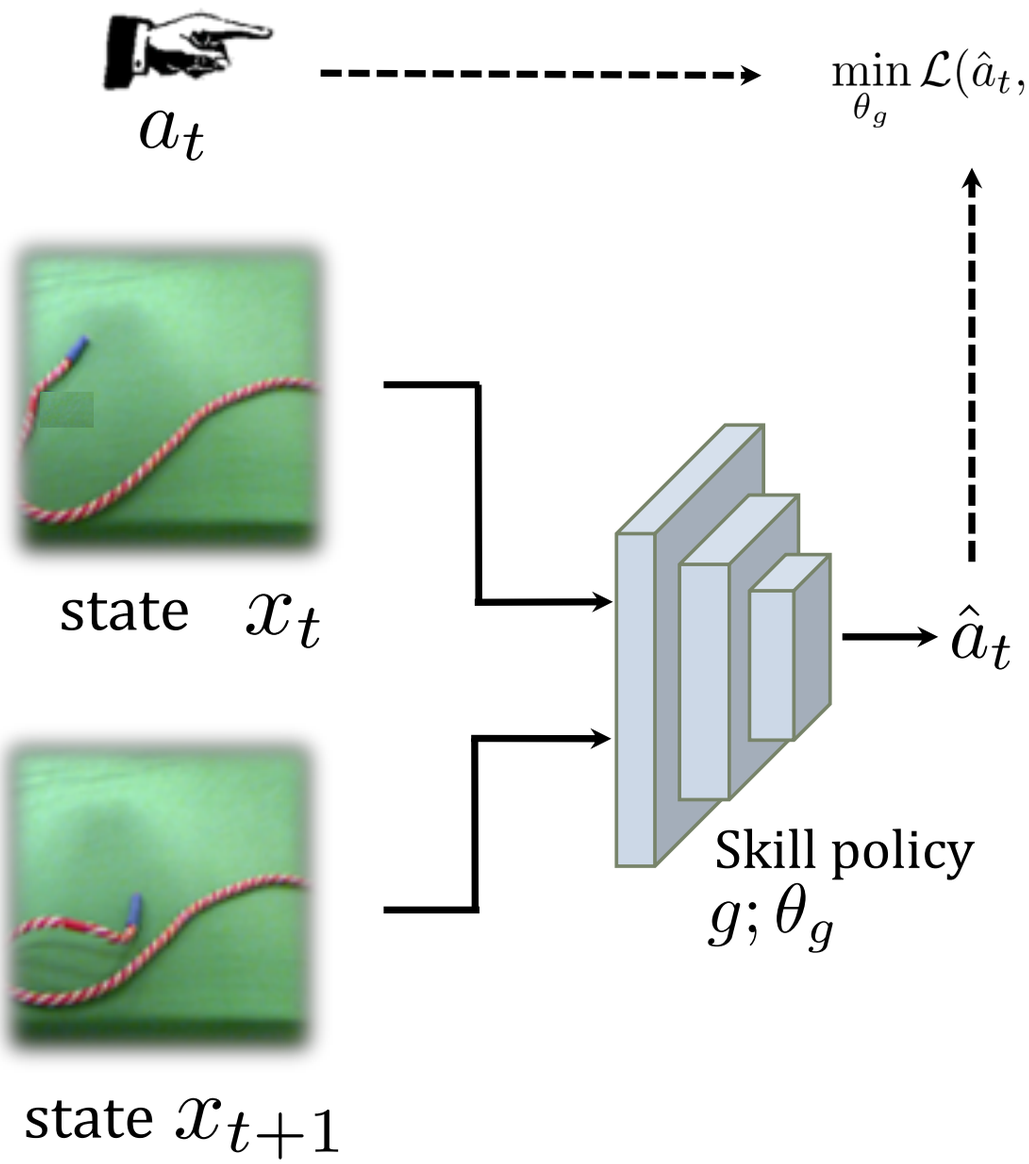


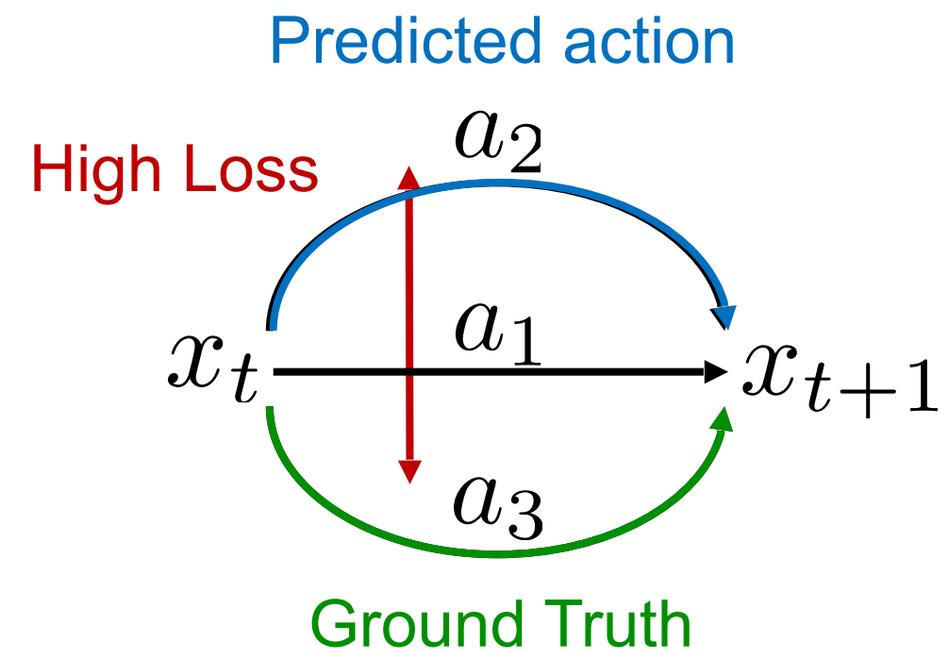


"skill" policy

Learning Inverse Models by regressing to actions

handle via VAEs or
auto-regressive models?

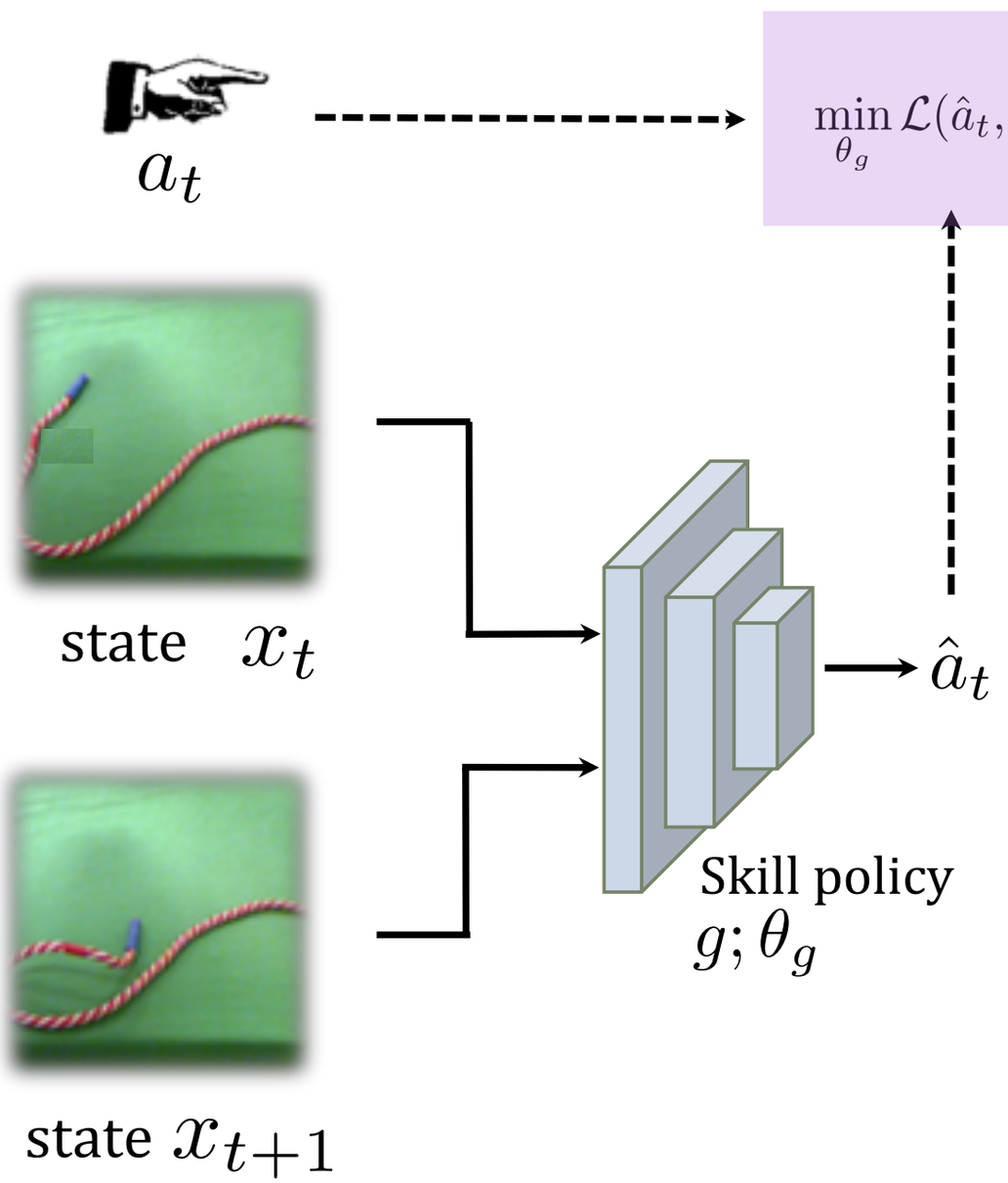


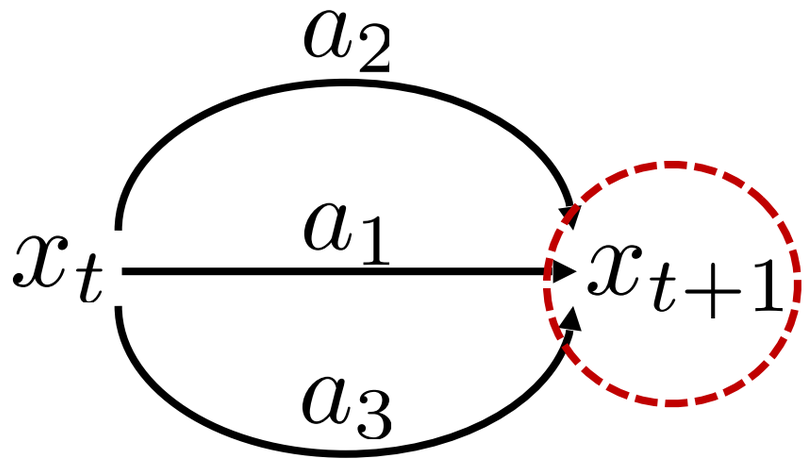


multi-modality in action space

Learning Inverse Models by penalizing resulting state

We do not care what actions to choose as long as it takes us to the right goal state-> let's penalize the resulting state as opposed to the action.





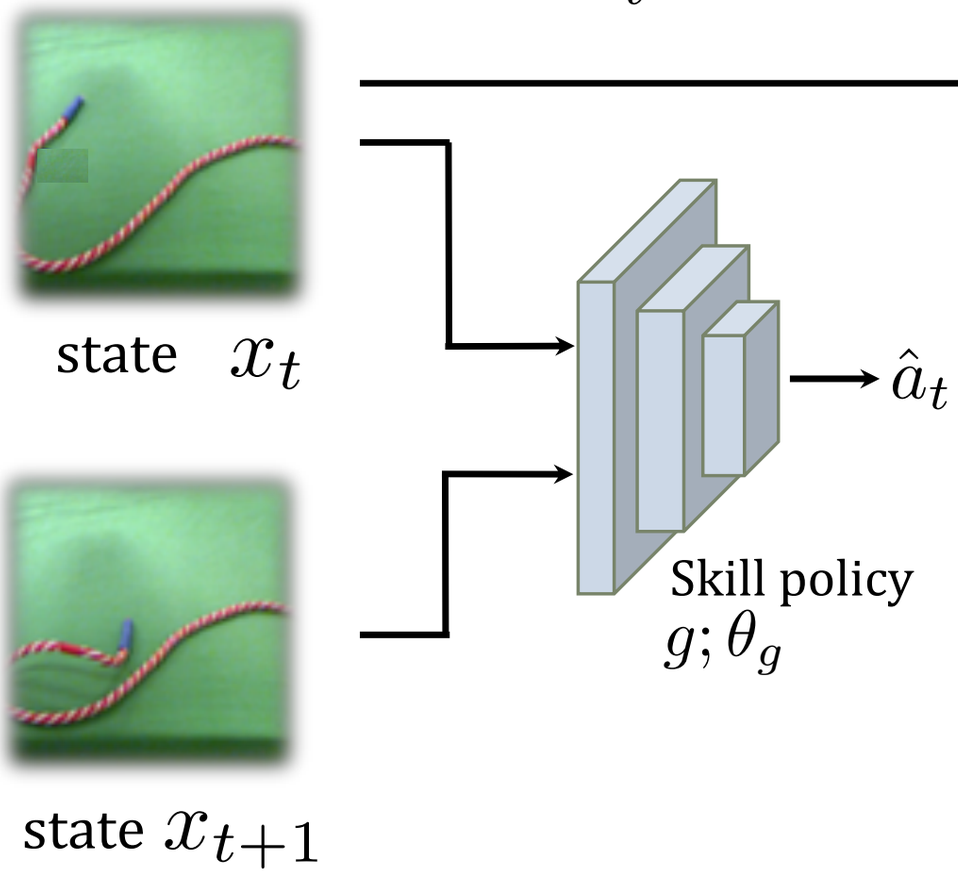
Penalize the "effect" of actions

How to operationalize it?

Learning Jointly Inverse and Forward Models

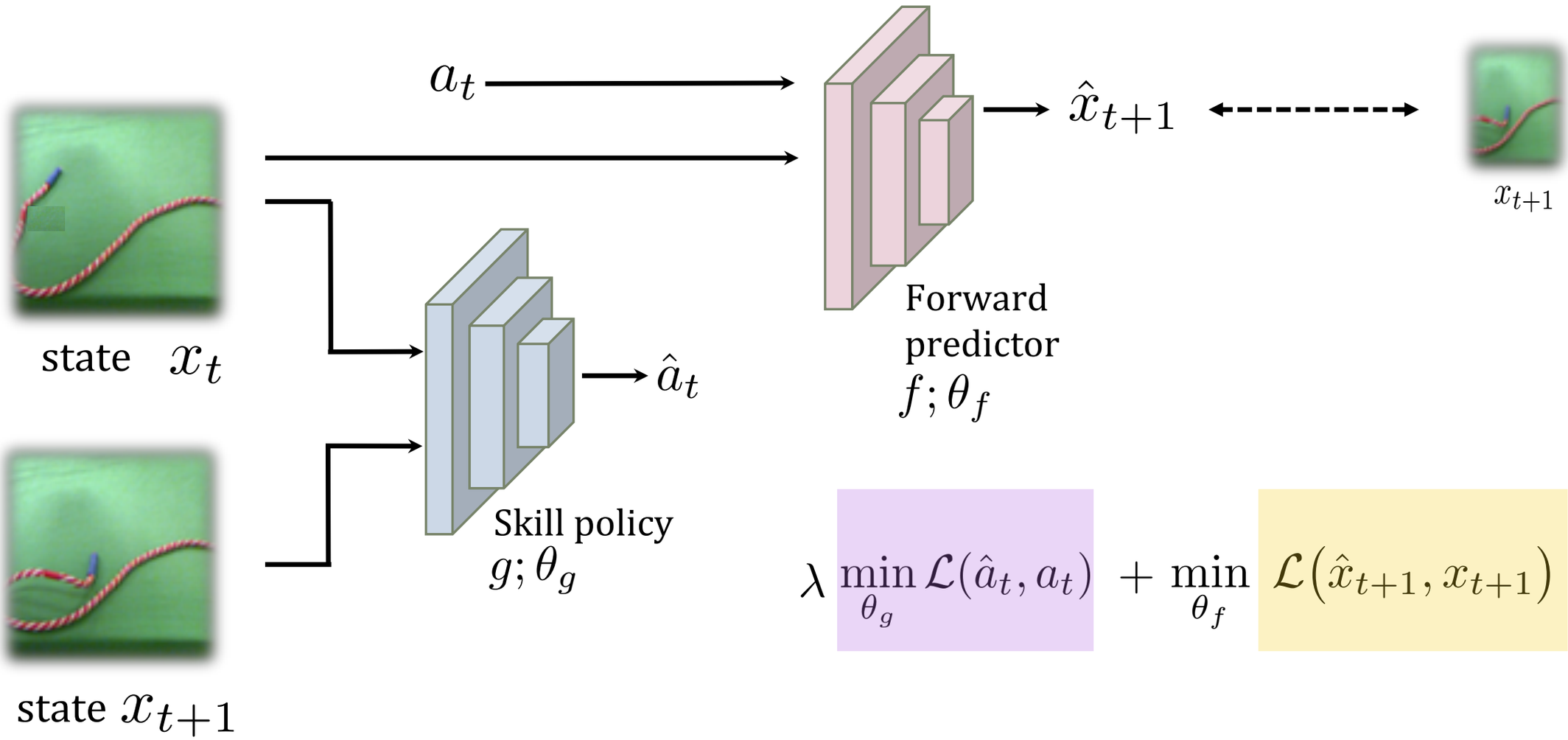
Train jointly a forward and an inverse model by regressing to the next state: next state will also be multimodal, but less multimodal than the action!

$$x_{t+1} \approx \hat{x}_{t+1} = f(x_t, a_t; \theta_f)$$



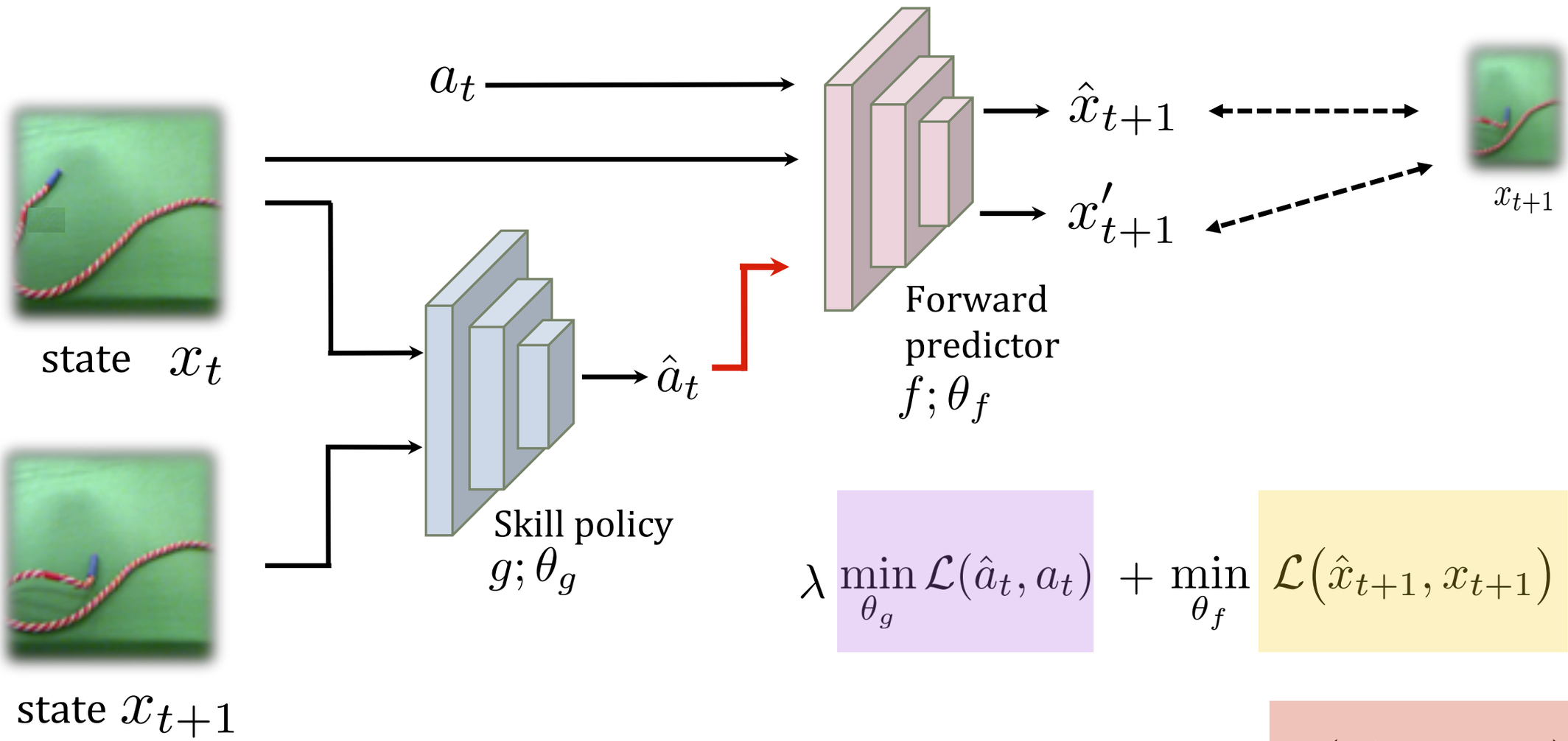
$$\lambda \min_{\theta_g} \mathcal{L}(\hat{a}_t, a_t) + \min_{\theta_f} \mathcal{L}\left(\hat{x}_{t+1}, x_{t+1}\right)$$

Handling Multimodality with Forward Consistency



$$\lambda \min_{\theta_g} \mathcal{L}(\hat{a}_t, a_t) + \min_{\theta_f} \mathcal{L}\left(\hat{x}_{t+1}, x_{t+1}\right)$$

Handling Multimodality with Forward Consistency

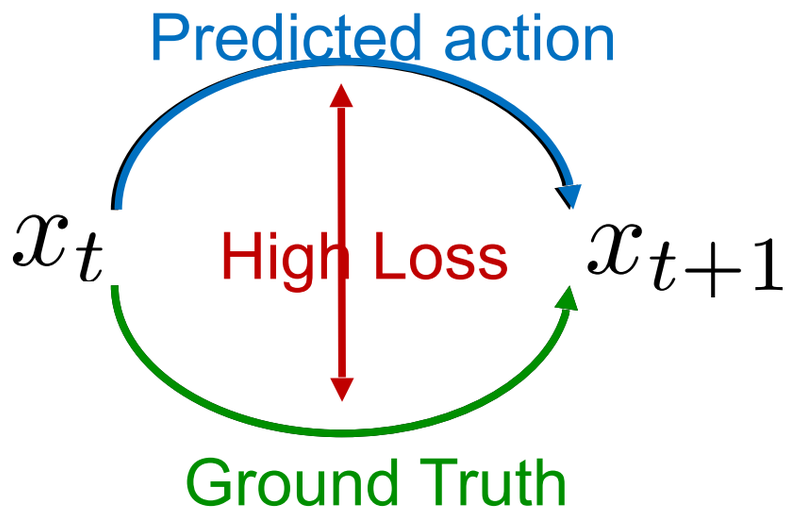


$$\lambda \min_{\theta_g} \mathcal{L}(\hat{a}_t, a_t) + \min_{\theta_f} \mathcal{L}\left(\hat{x}_{t+1}, x_{t+1}\right)$$

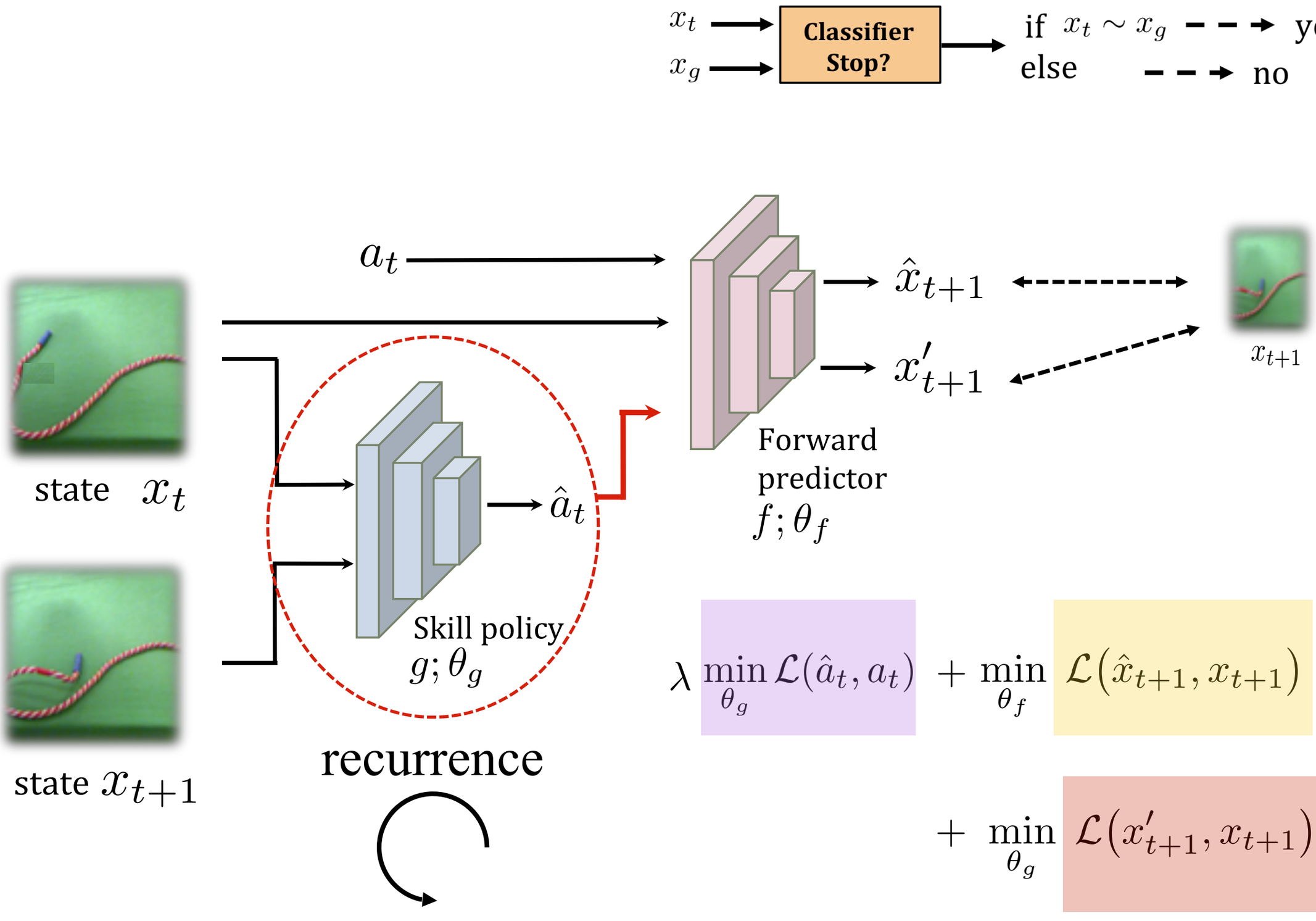
$$+ \min_{\theta_g} \mathcal{L}\left(x'_{t+1}, x_{t+1}\right)$$

In practice: Multi-step planning

multi-modality gets even worse with multi-step policy



Forward Consistency in Multi-step Planning

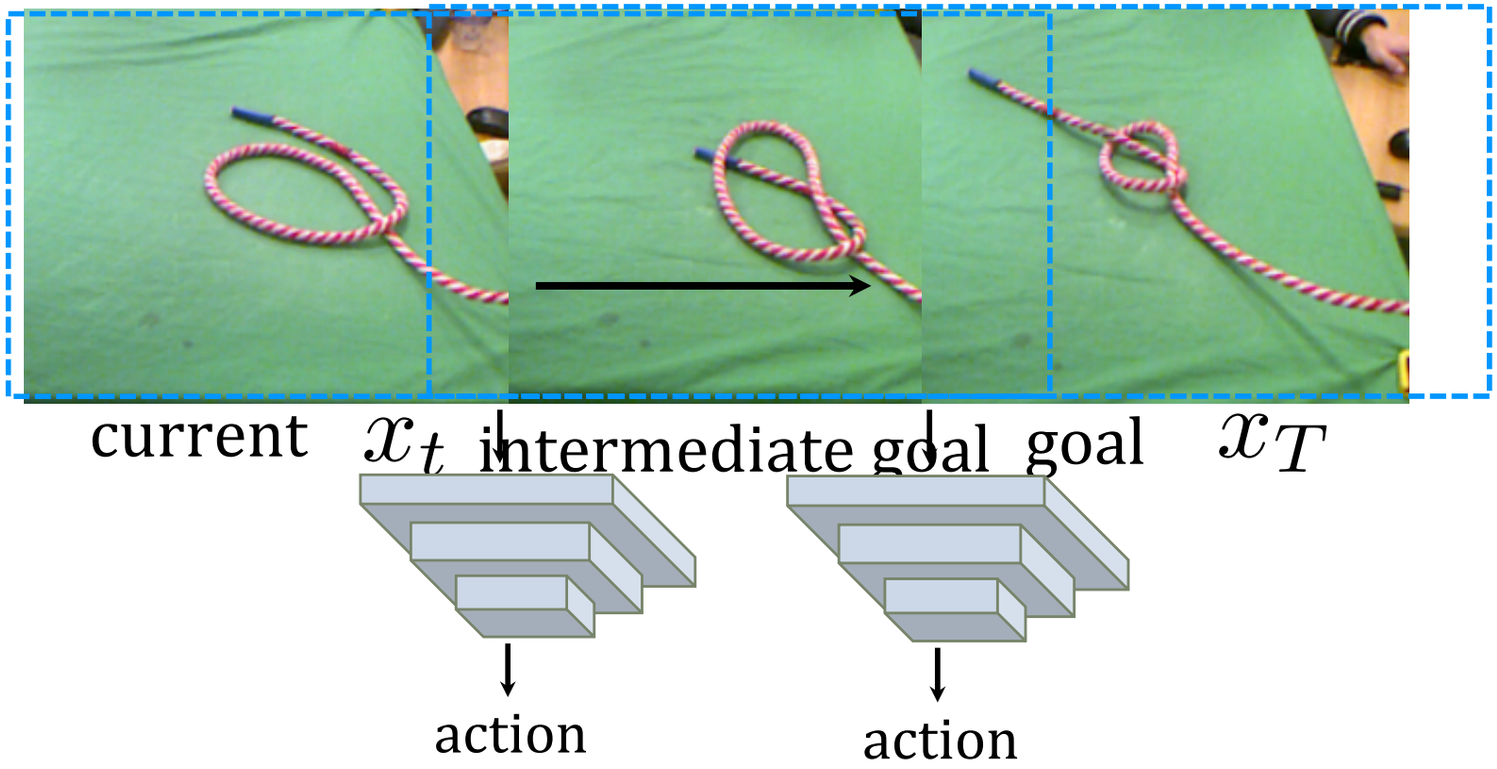


$$\lambda \min_{\theta_g} \mathcal{L}(\hat{a}_t, a_t) + \min_{\theta_f} \mathcal{L}\left(\hat{x}_{t+1}, x_{t+1}\right)$$

$$+ \min_{\theta_g} \mathcal{L}\left(x'_{t+1}, x_{t+1}\right)$$

Model-guided Visual Imitation

1. Use the visual demonstation to obtain subgoals: intermediate states you need to reach
2. Use the skills you have learns to reach those subgoals



Knot-tying Rope Manipulation