# Model-predictive policy learning with uncertainty regularisation for driving in dense traffic

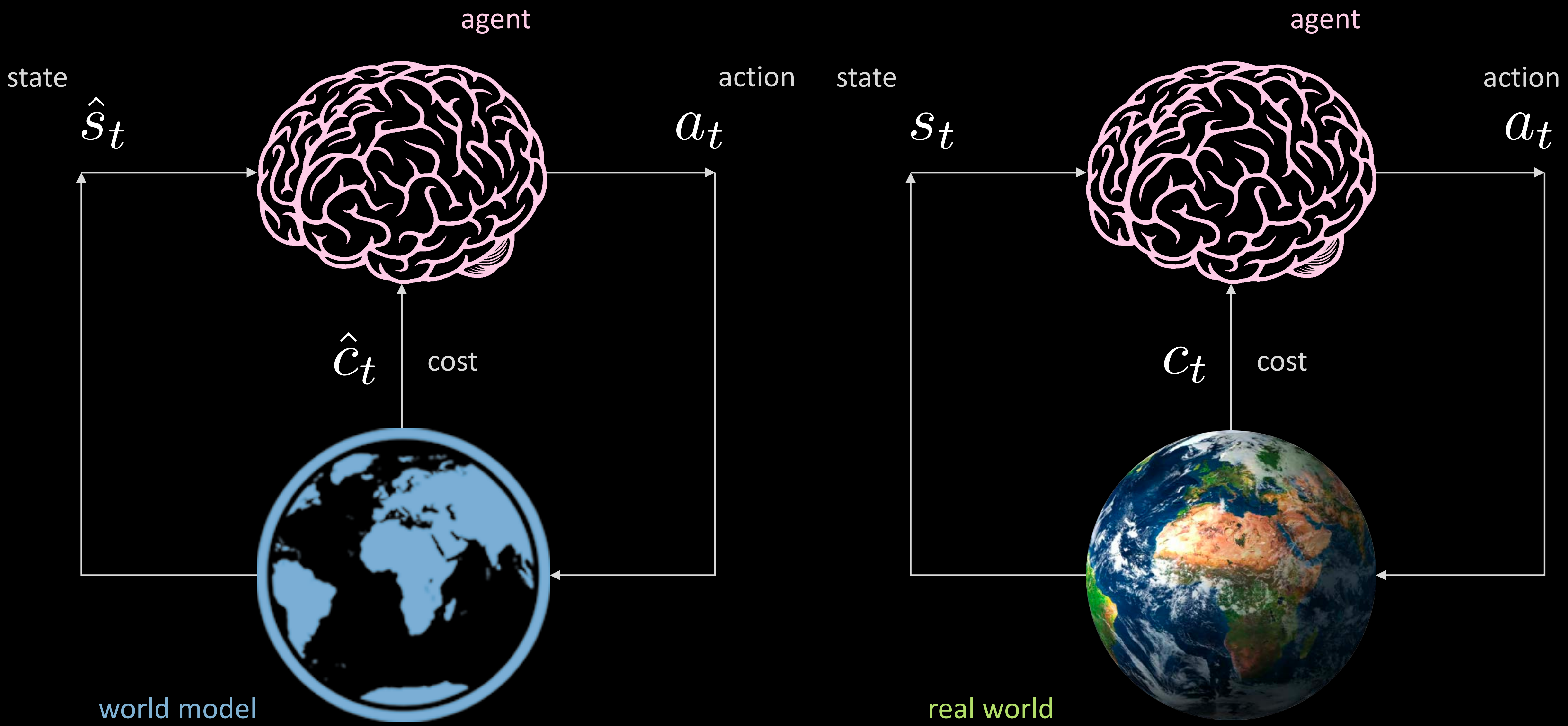
Mikael Henaff*, Alfredo Canziani*, Yann LeCun

@HenaffMikael @alfcnz @ylecun

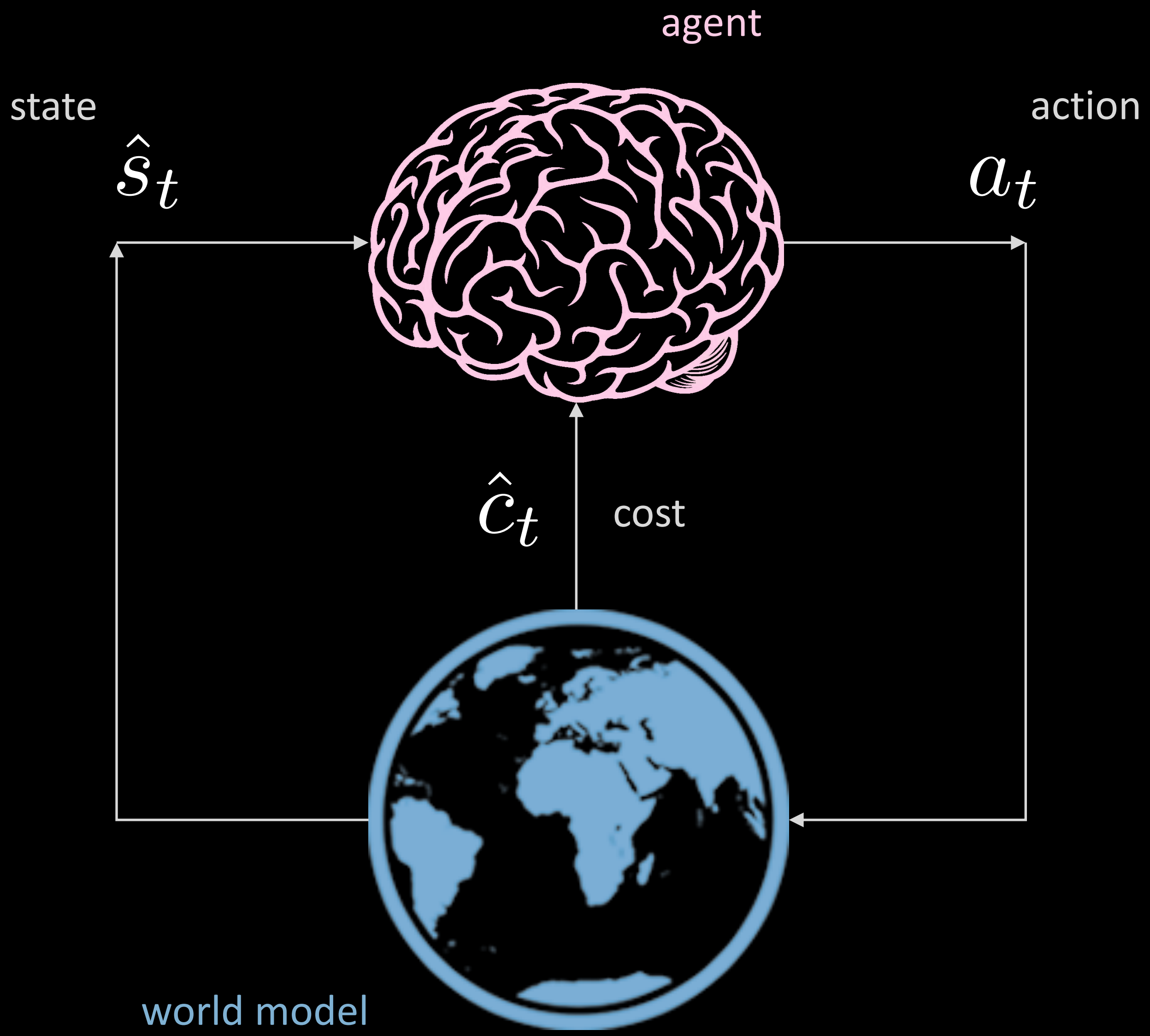
NYU Courant

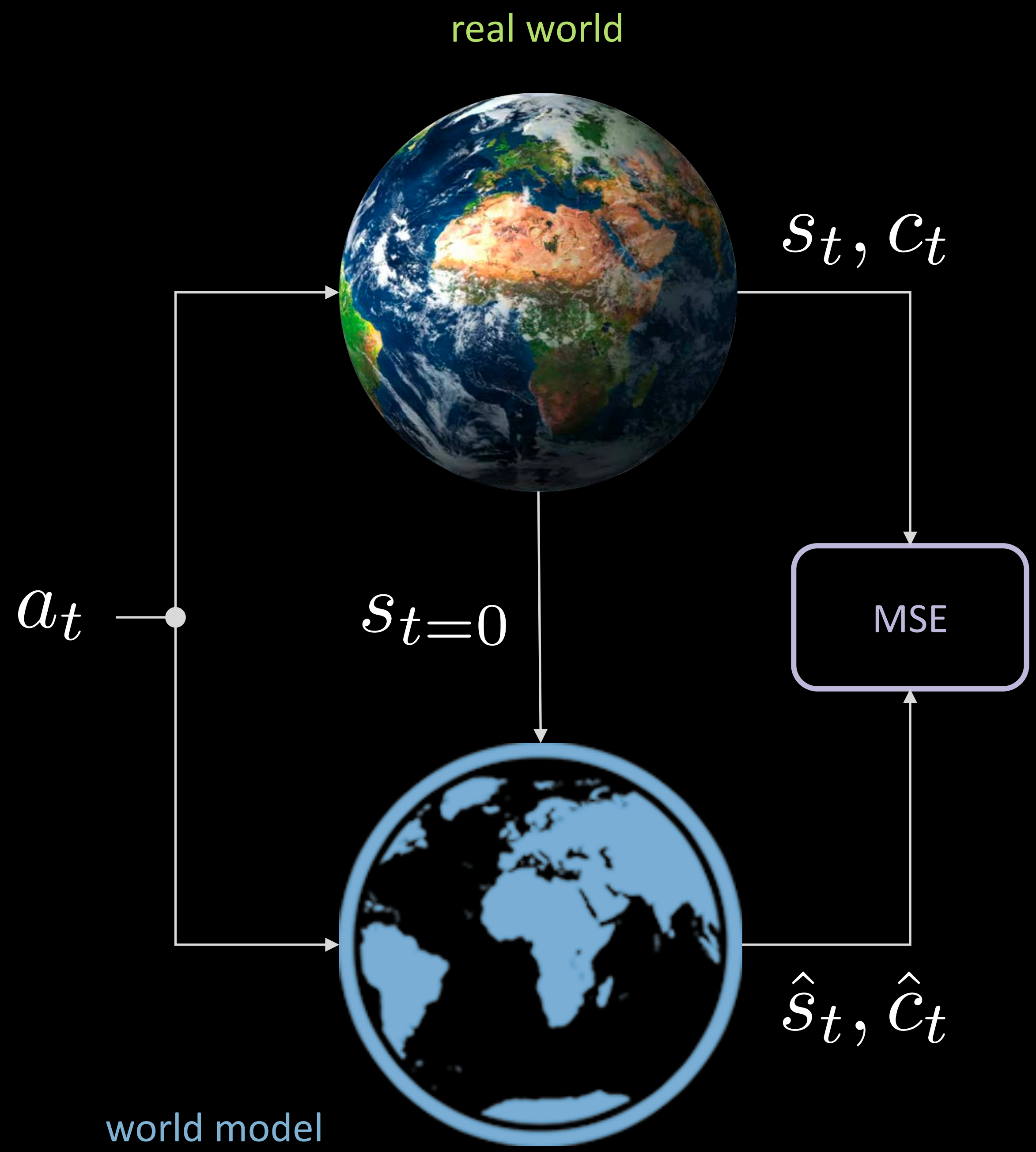
* equal contribution

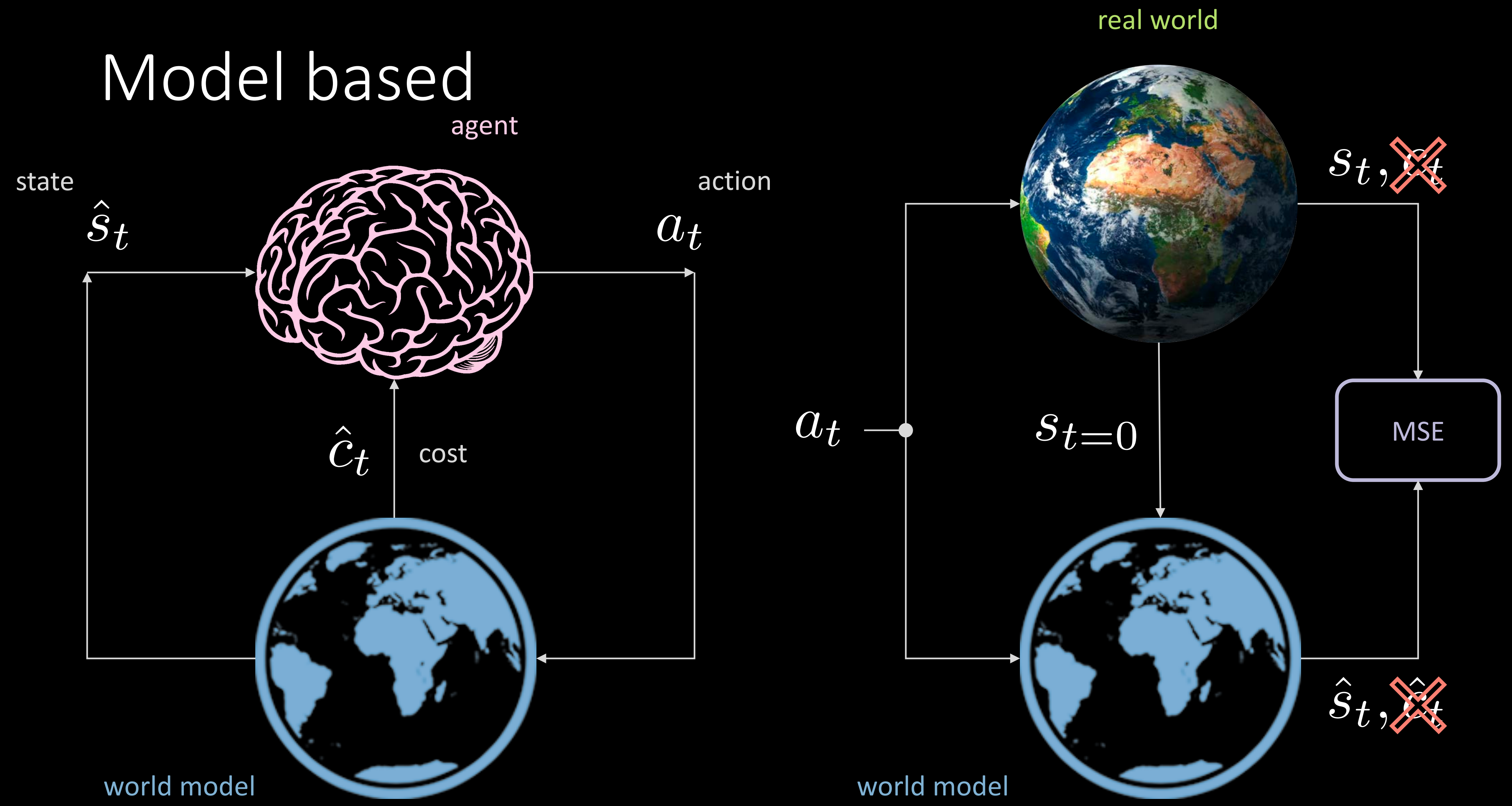
# Model based *vs.* model free



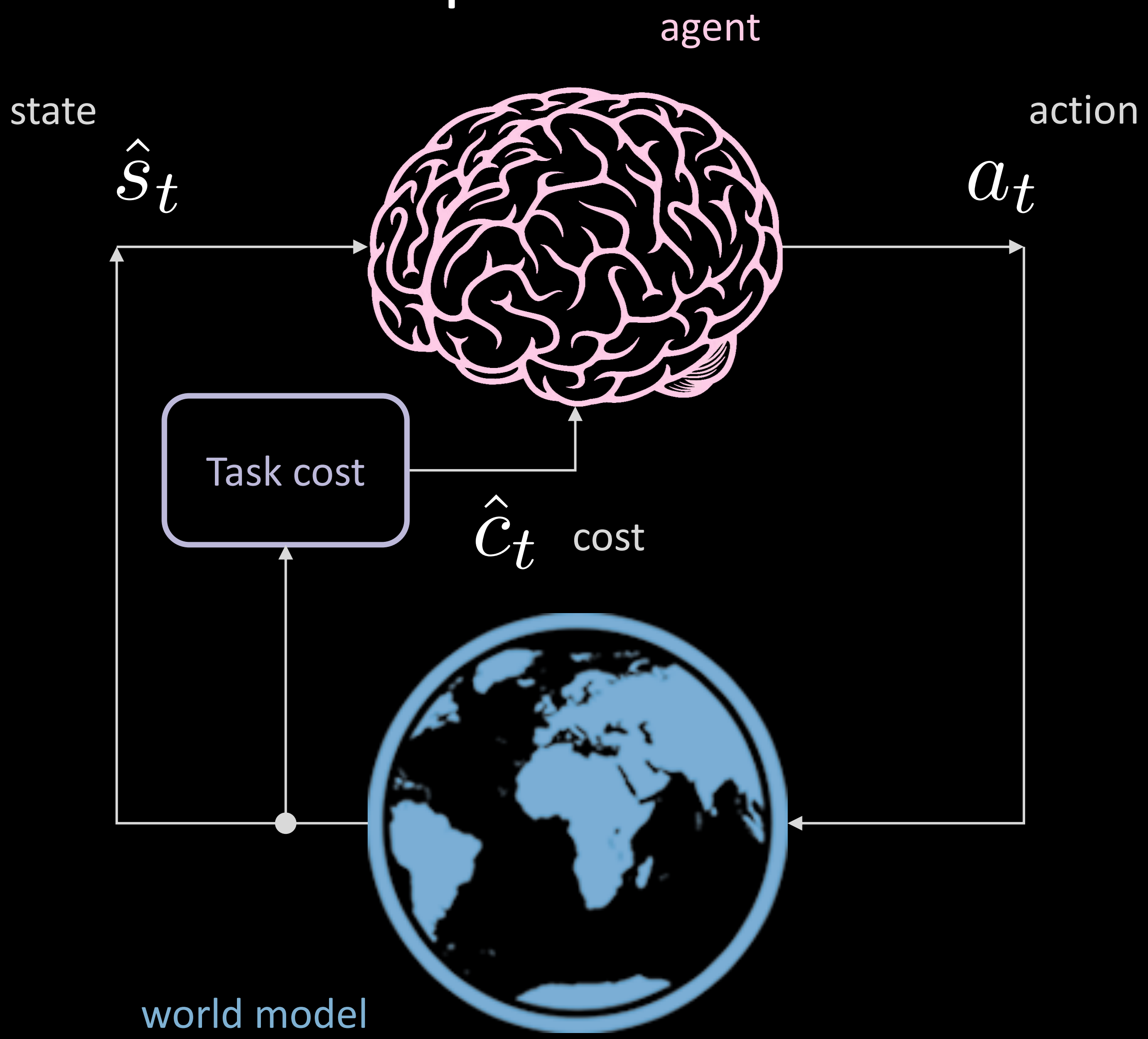
# Model based

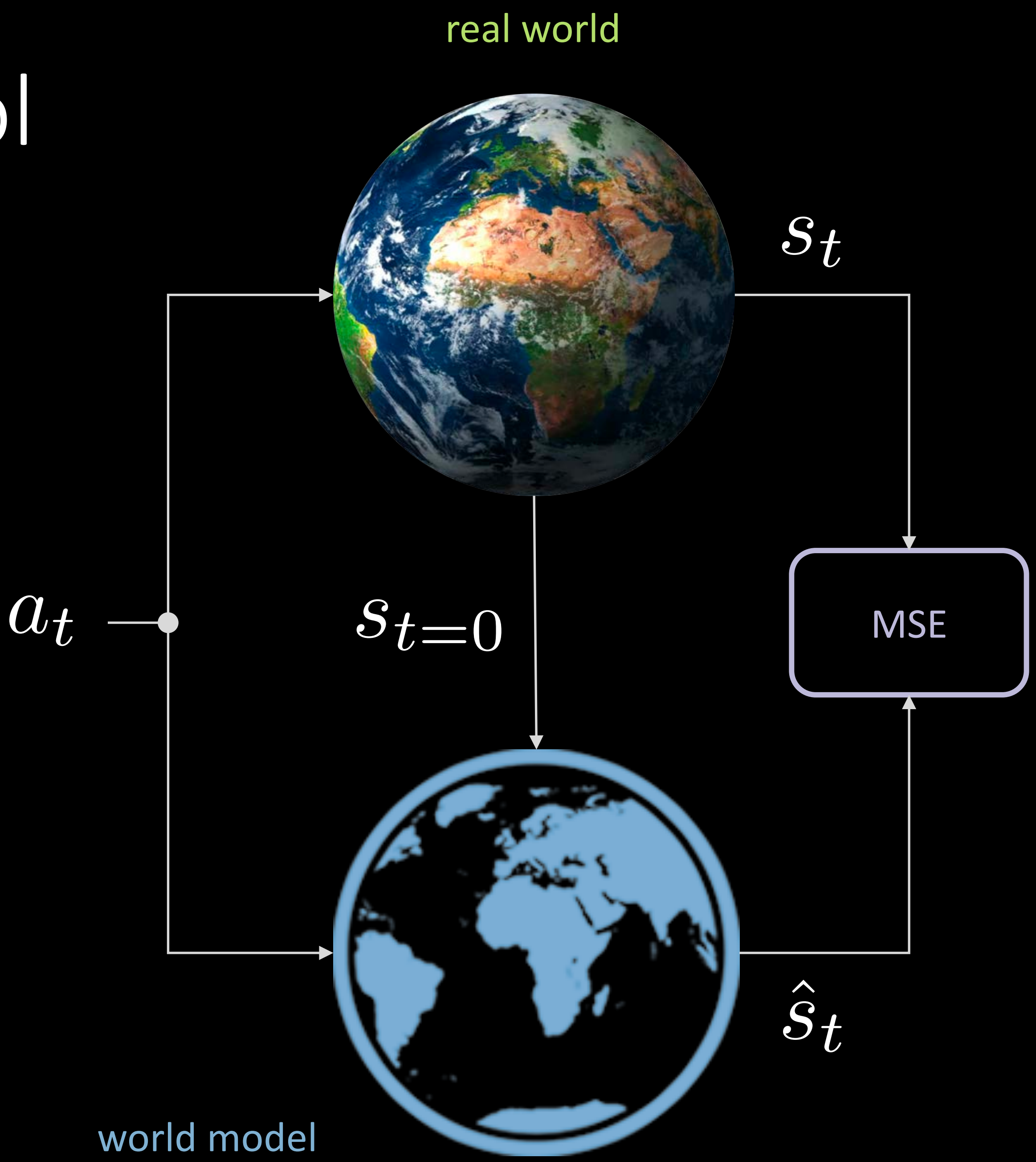

agent
state
$\hat{s}_t$
action
$a_t$
$\hat{c}_t$
cost
world model


real world
$s_t, c_t$
$a_t$
$s_{t=0}$
MSE
$\hat{s}_t, \hat{c}_t$
world model


Model based
agent
state
$\hat{s}_t$
action
$a_t$
$\hat{c}_t$
cost
world model
real world
$s_t, \cancel{c_t}$
$a_t$
$s_{t=0}$
MSE
$\hat{s}_t, \cancel{\hat{c}_t}$
world model

# Model predictive control



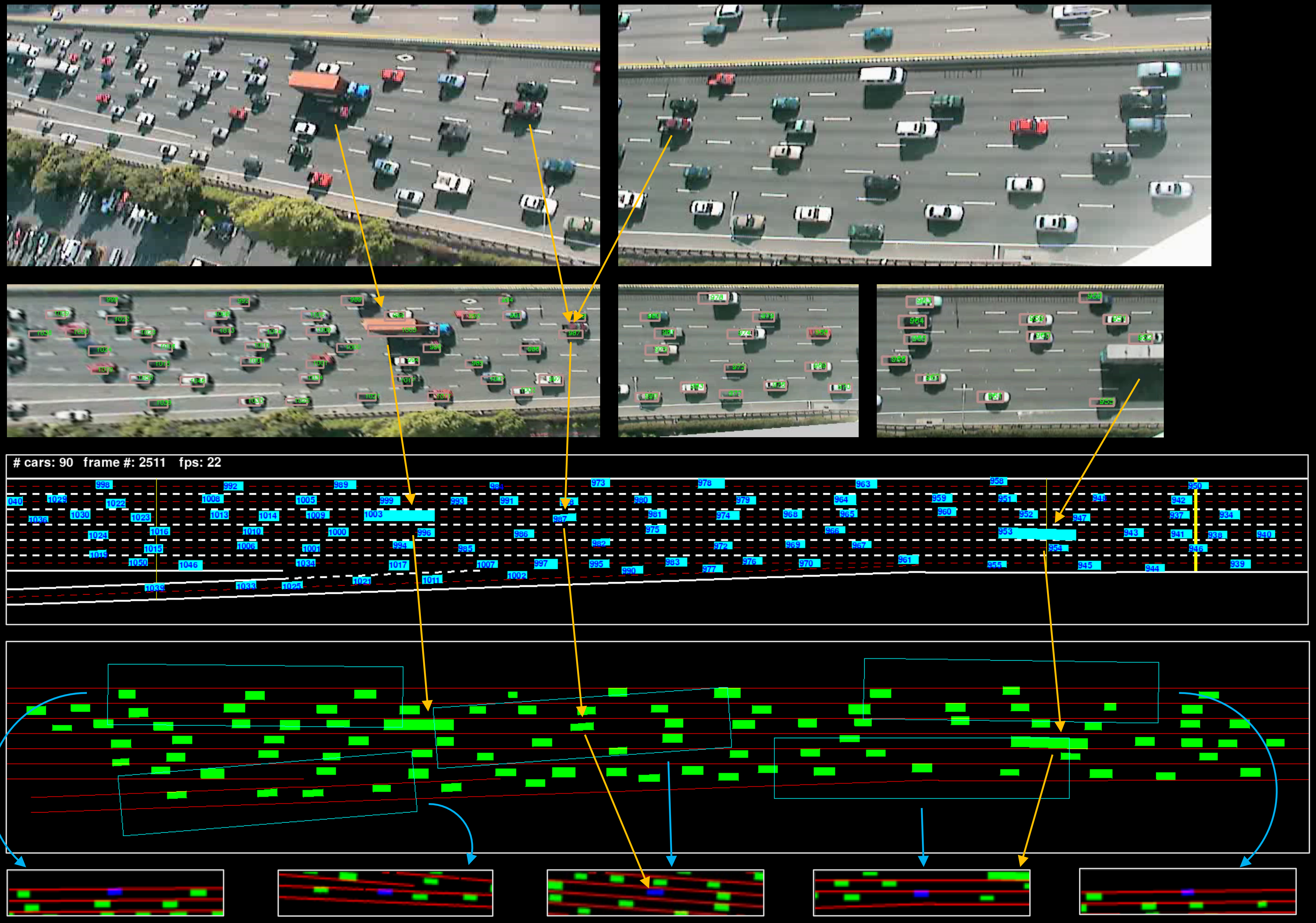
Real world

$a_t$

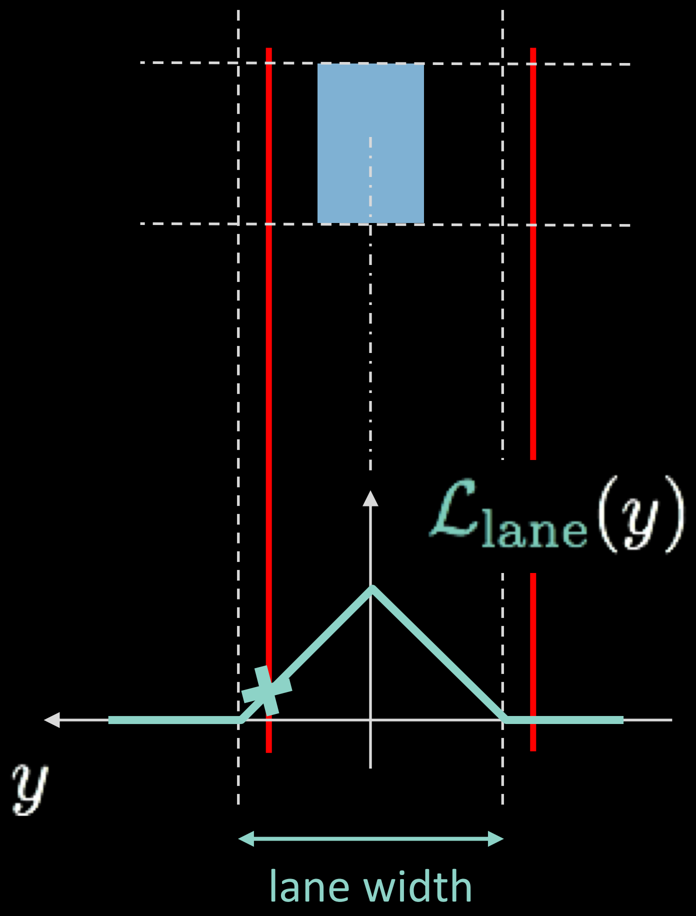
$p_t, v_t$

$s_t$

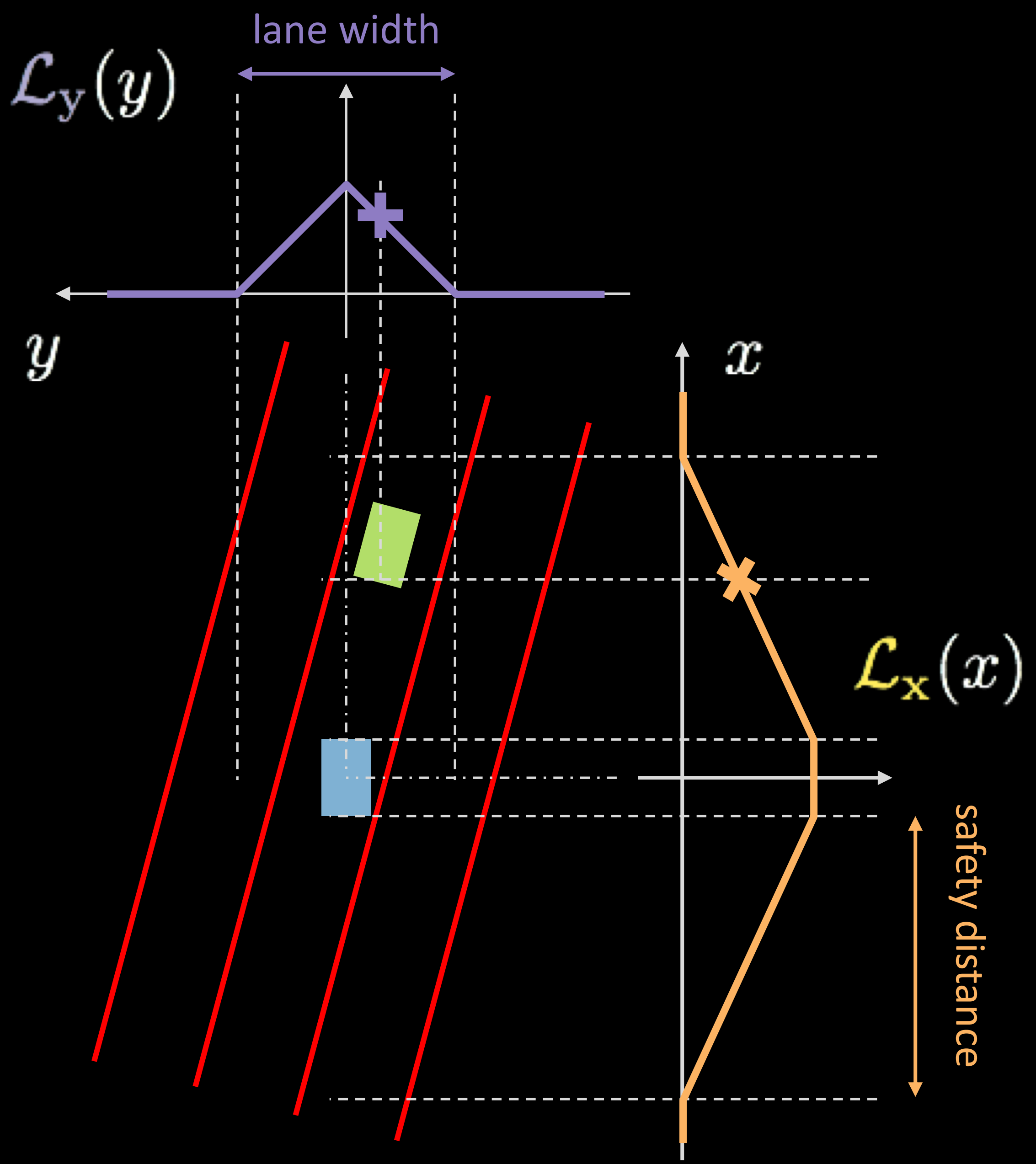
$i_t$



# Cost (I)



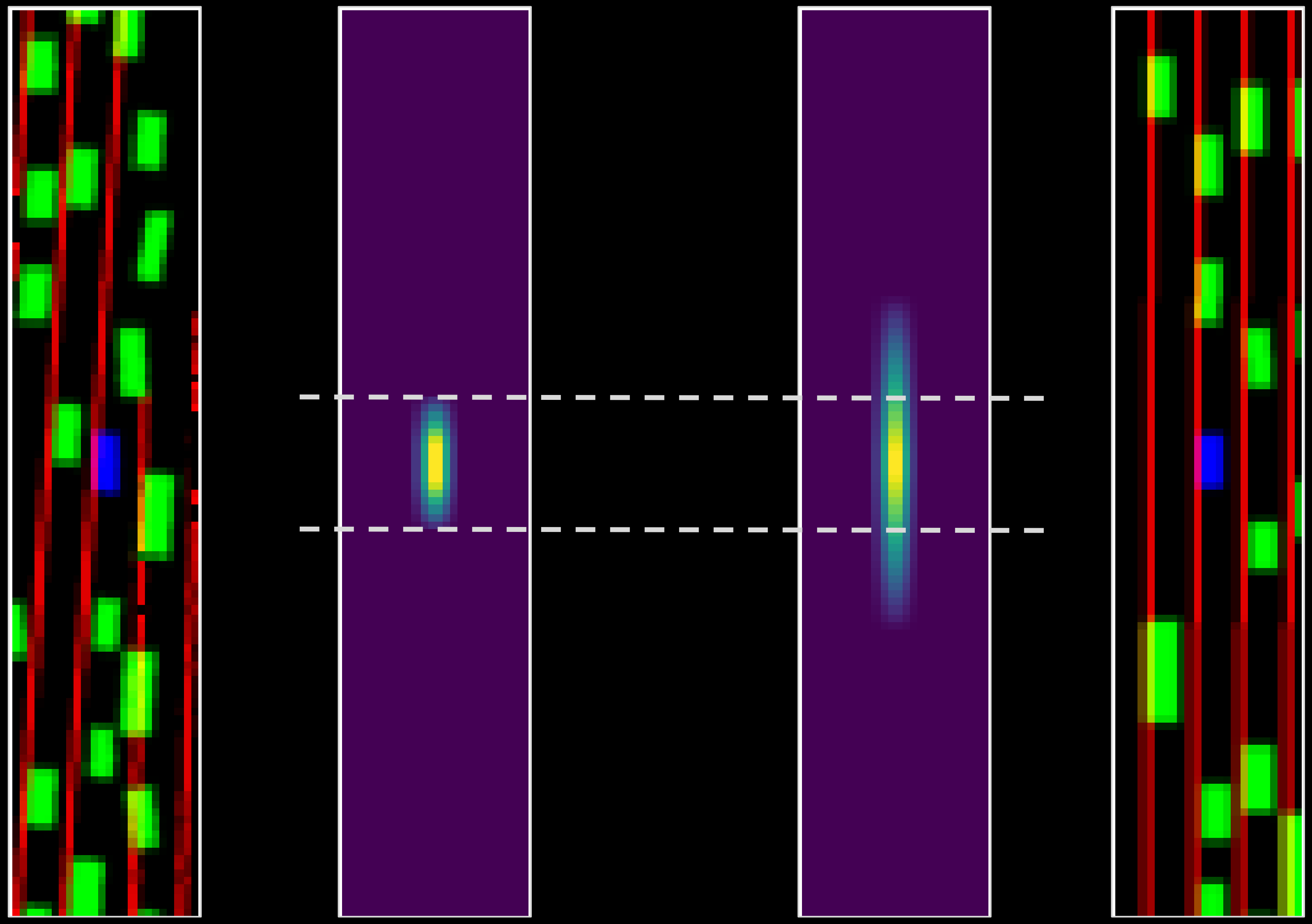

lane cost




proximity cost

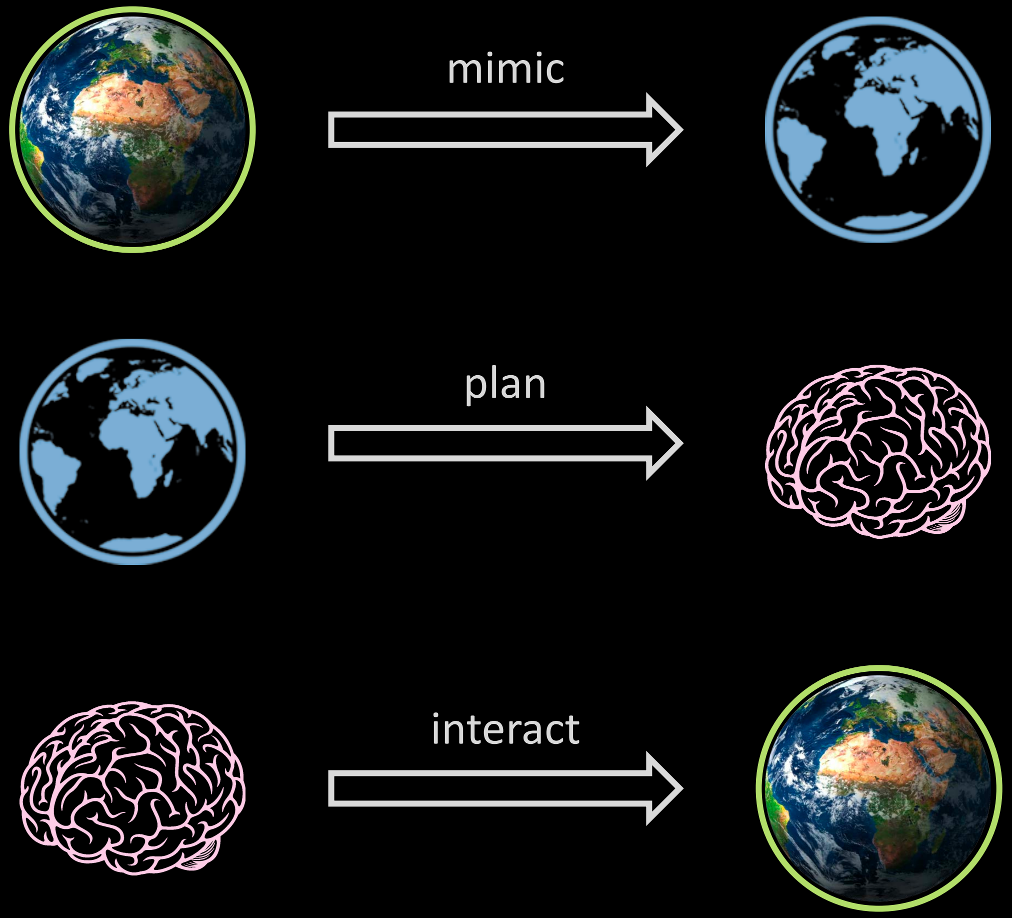
# Cost (II)

it’s differentiable!



# Outline

1. The world model
2. The agent
3. The evaluation

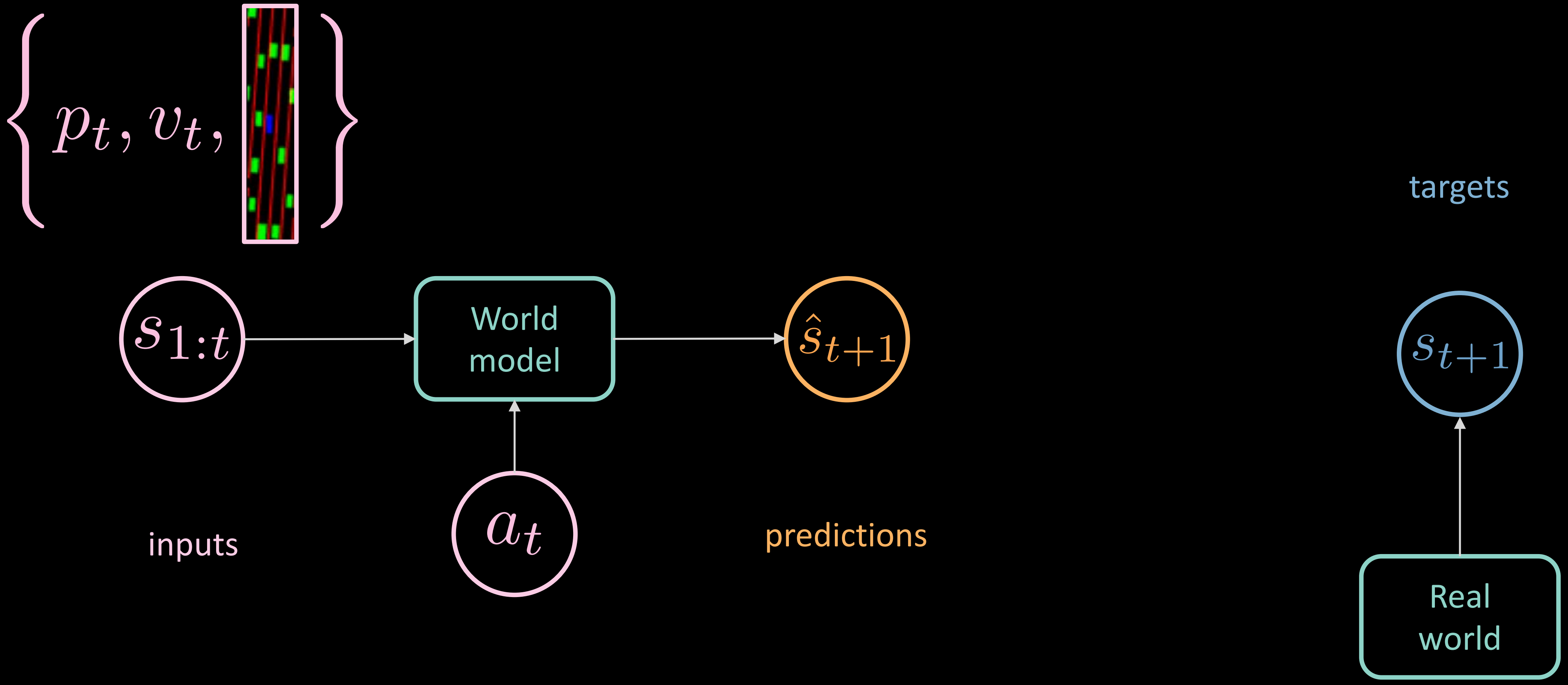


# 1. The world model

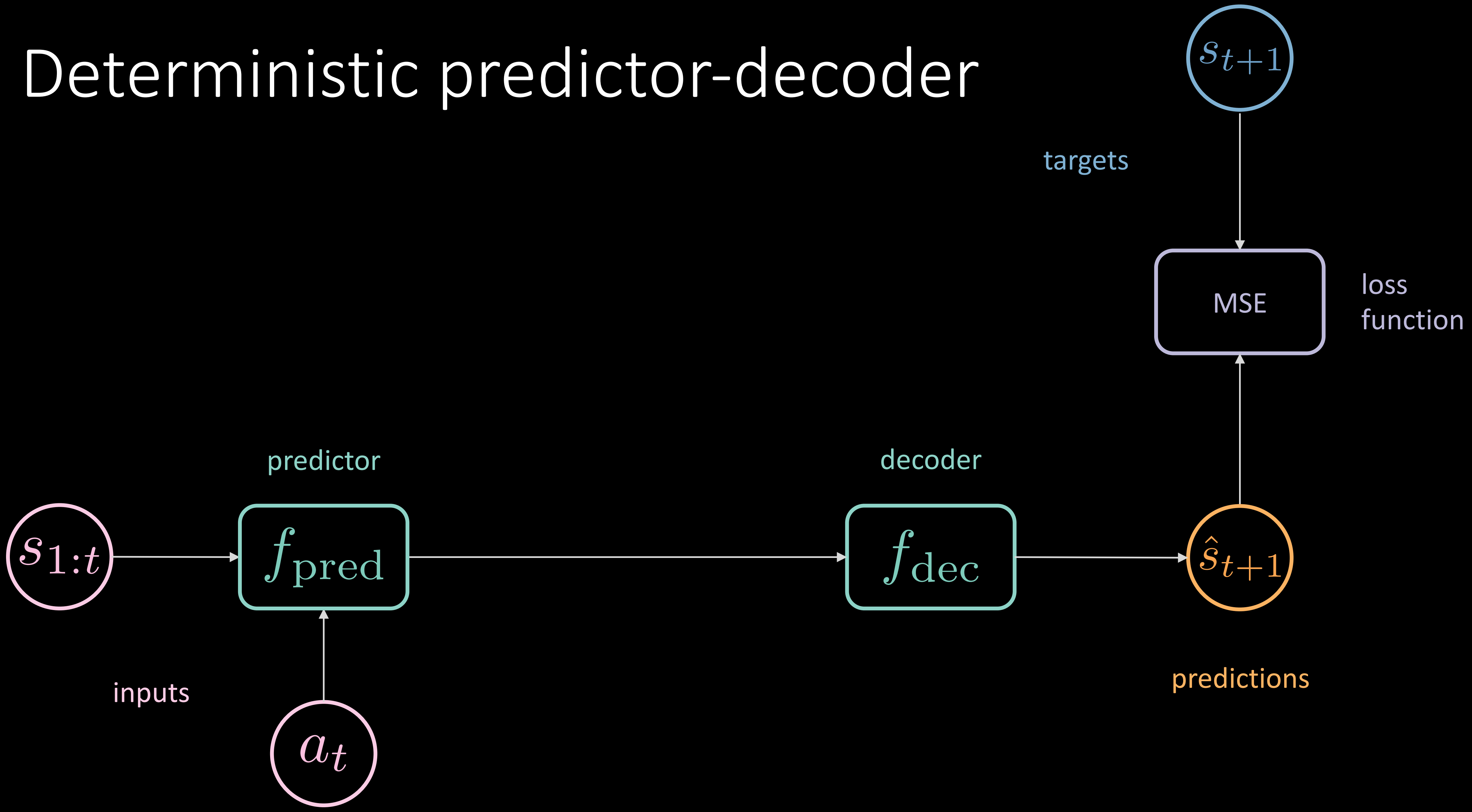
Predicting what’s next,

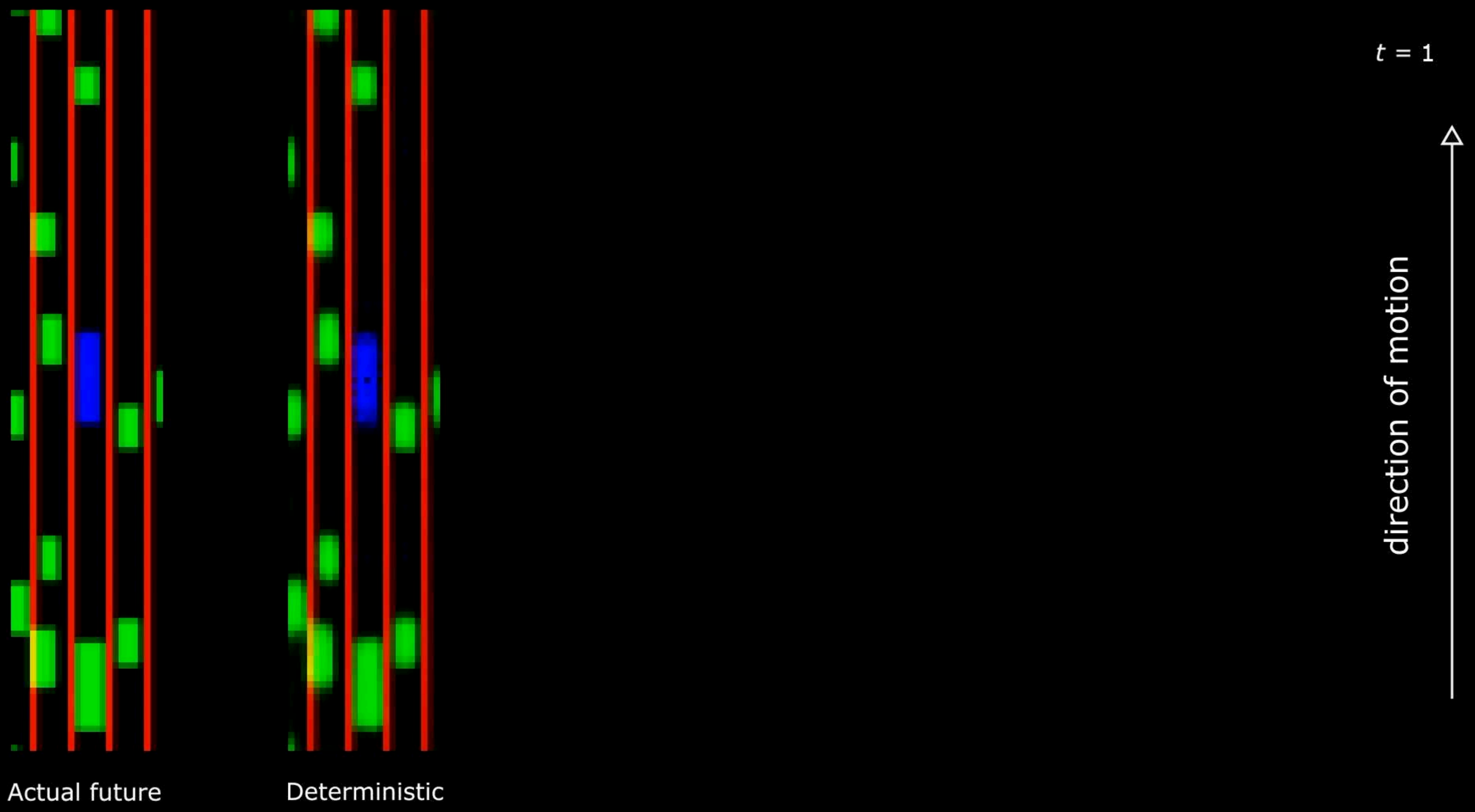
given history and action

# World model

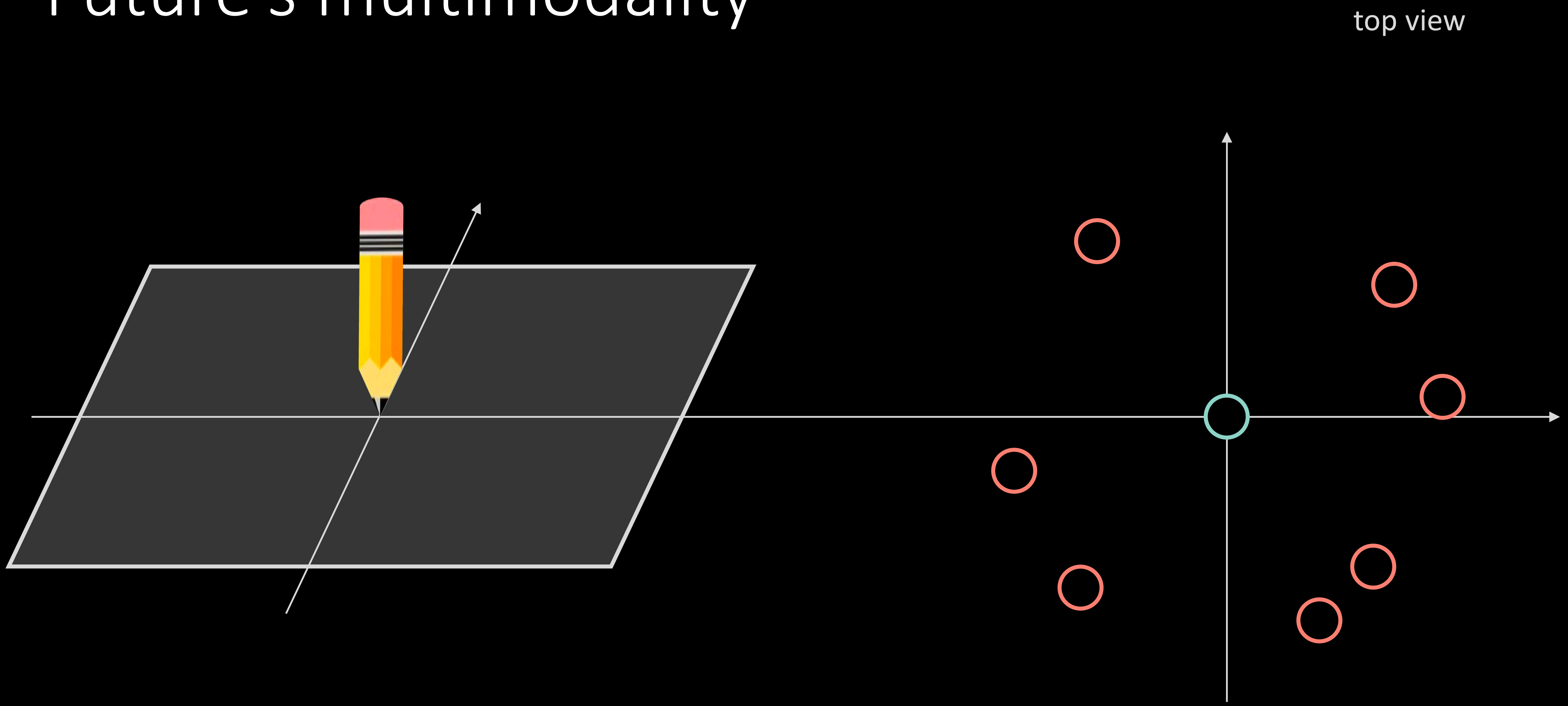

$\{p_t, v_t,$ $\}$
$s_{1:t}$
World model
$\hat{s}_{t+1}$
$a_t$
inputs
predictions
targets
$s_{t+1}$
Real world

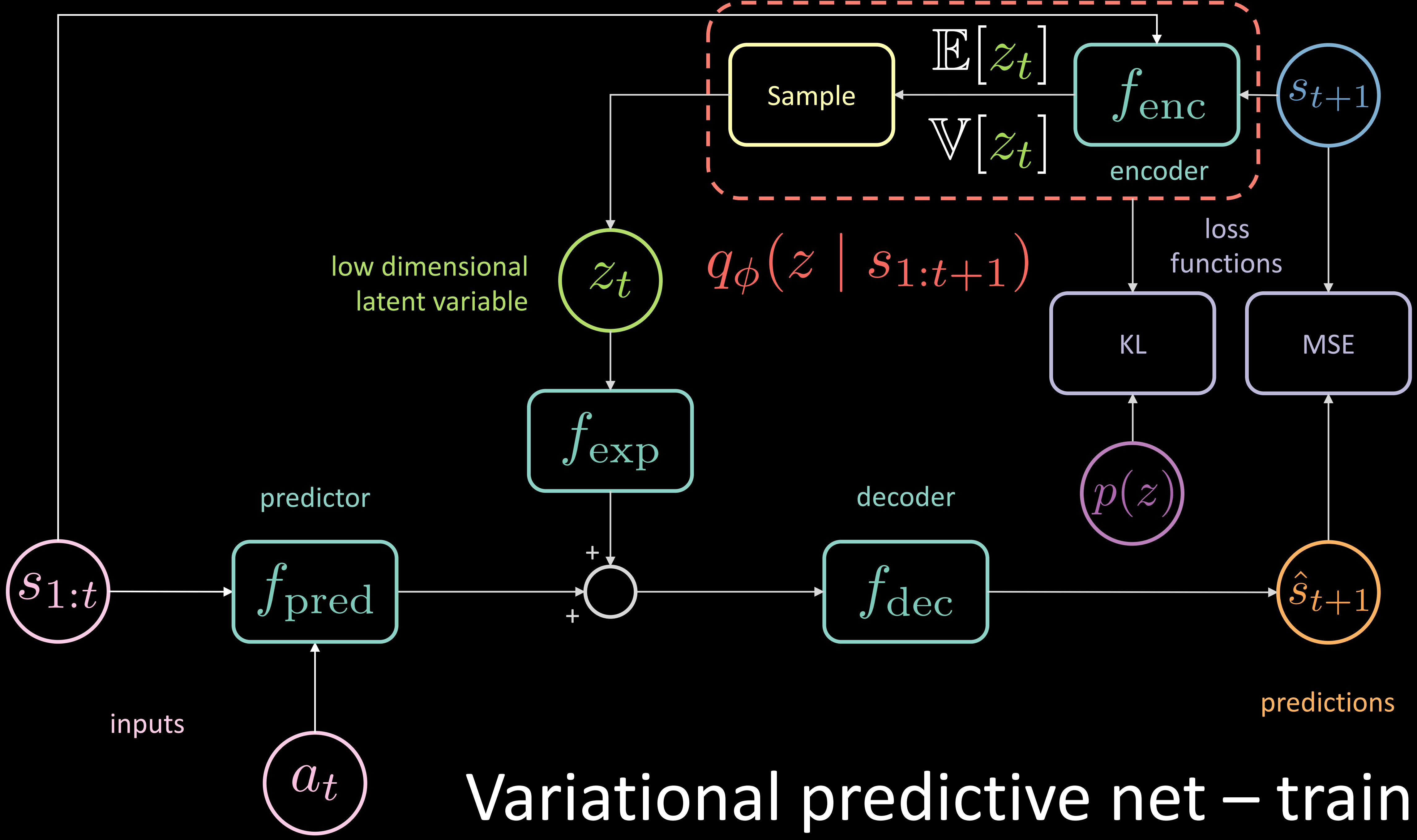
# Deterministic predictor-decoder




t = 1
direction of motion
Actual future
Deterministic

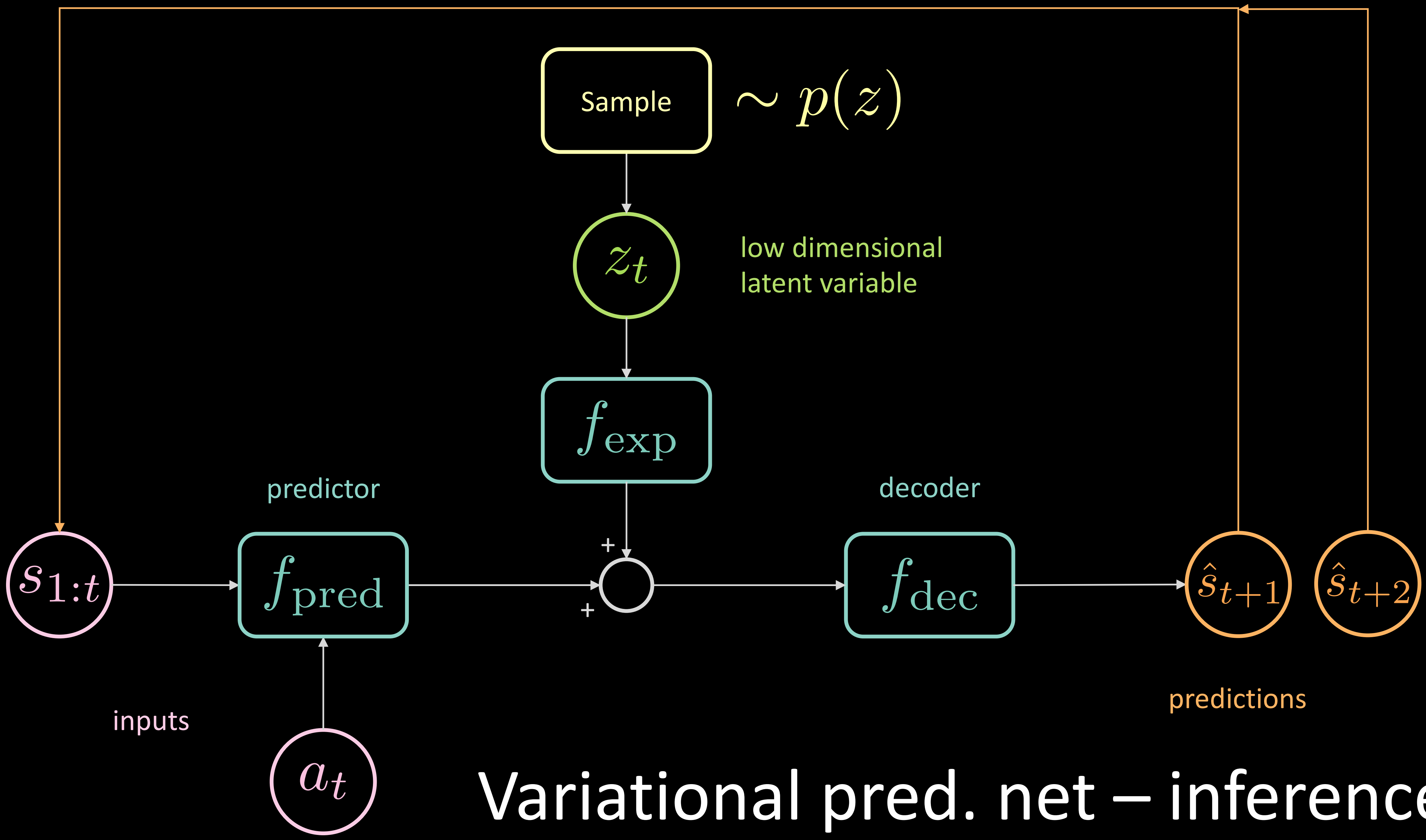
# Future’s multimodality

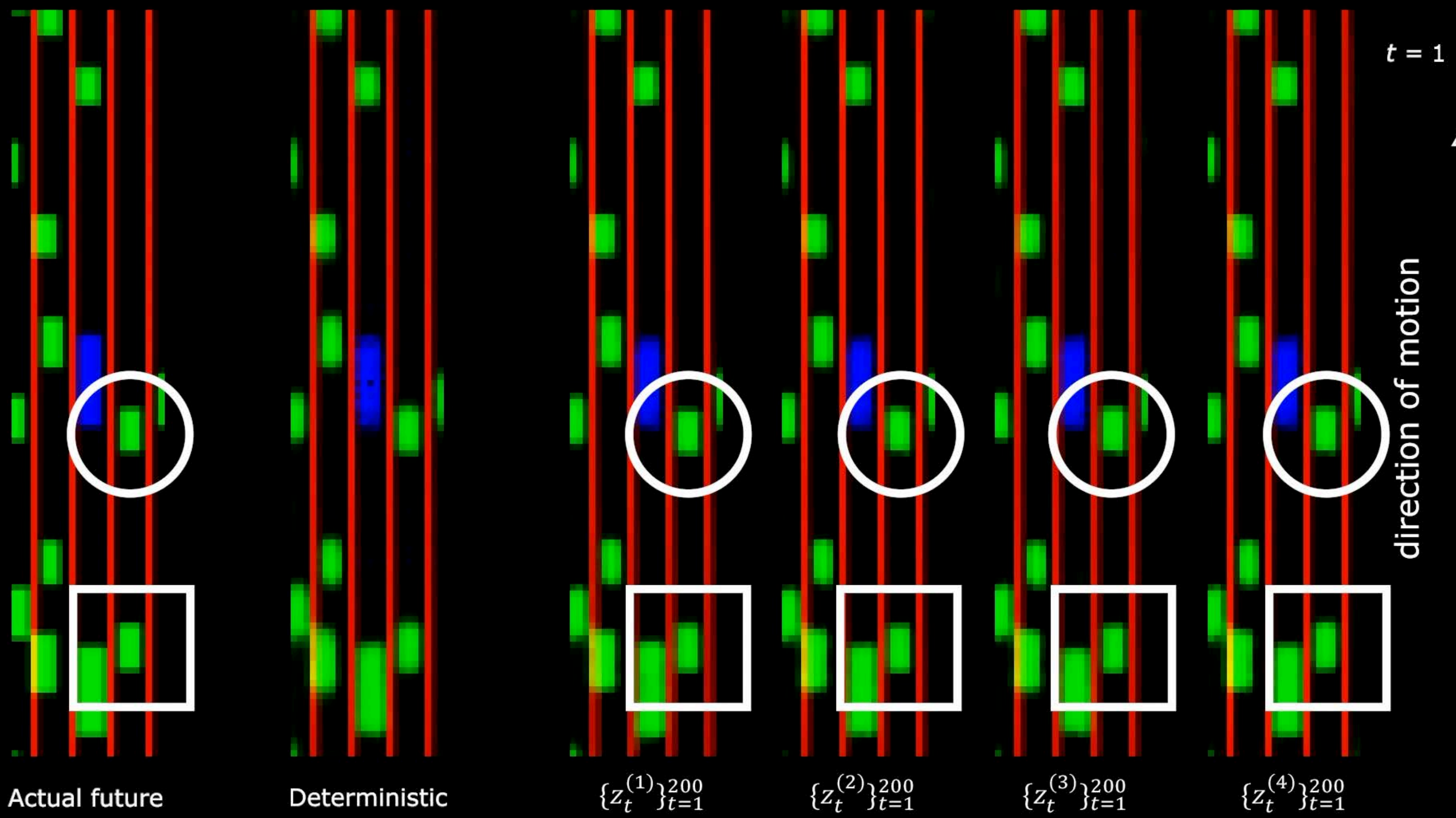




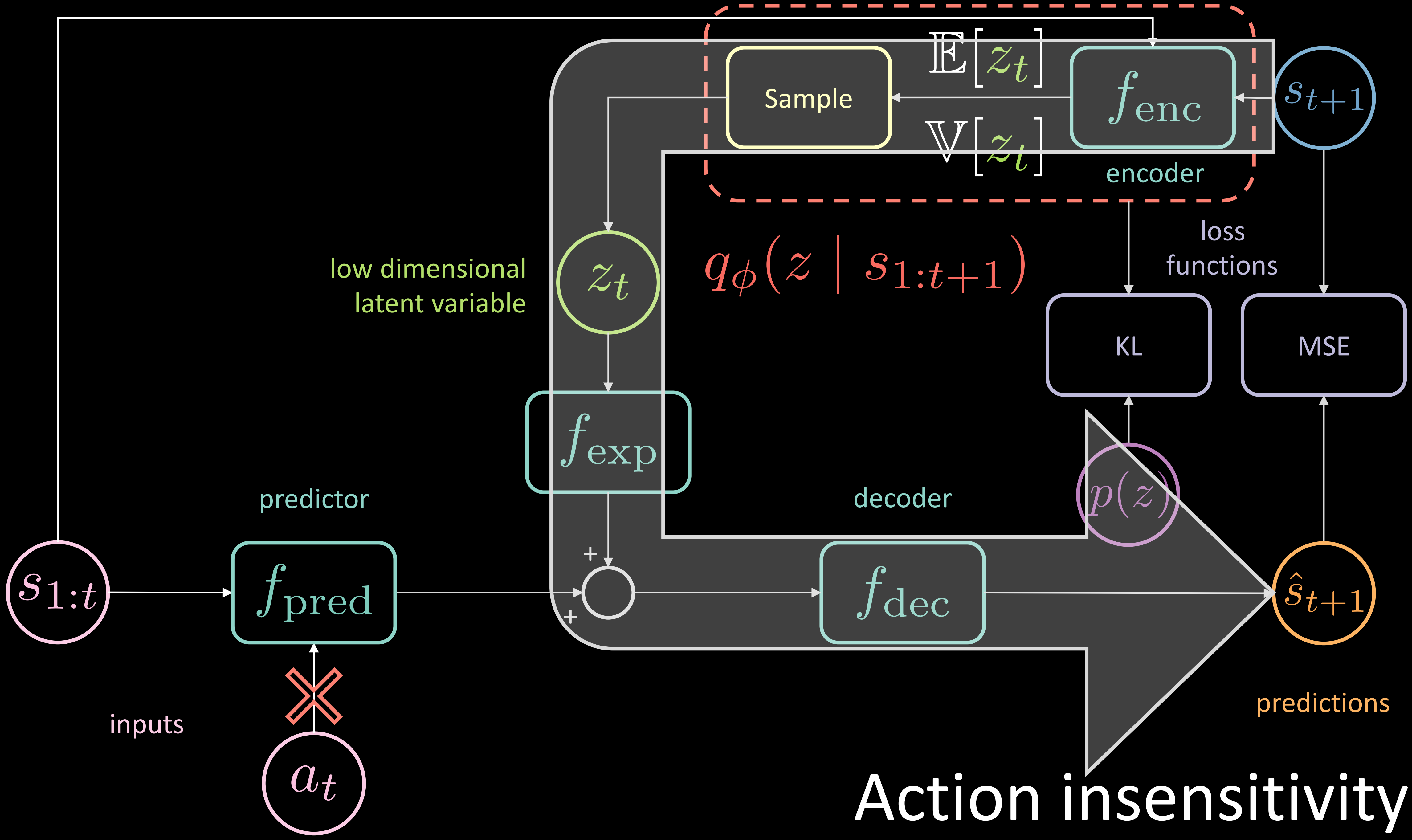

# Variational predictive net – train

# Variational pred. net – inference



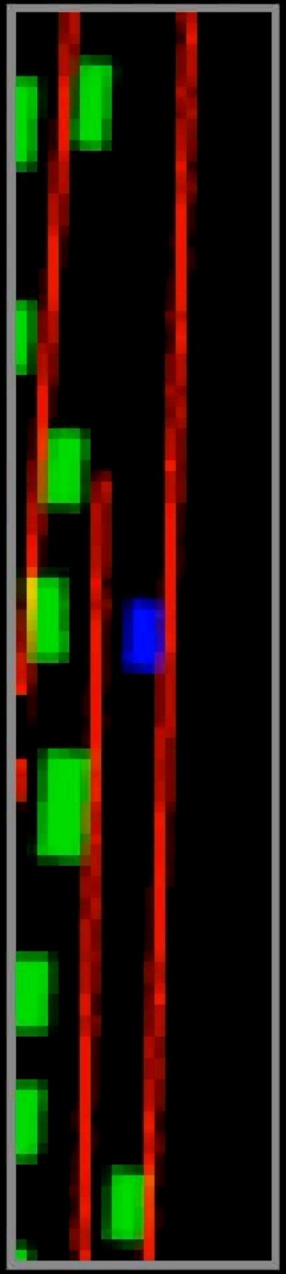


$t = 1$
direction of motion
Actual future
Deterministic
$\{z_t^{(1)}\}_{t=1}^{200}$
$\{z_t^{(2)}\}_{t=1}^{200}$
$\{z_t^{(3)}\}_{t=1}^{200}$
$\{z_t^{(4)}\}_{t=1}^{200}$




# Action insensitivity

$t = 1$

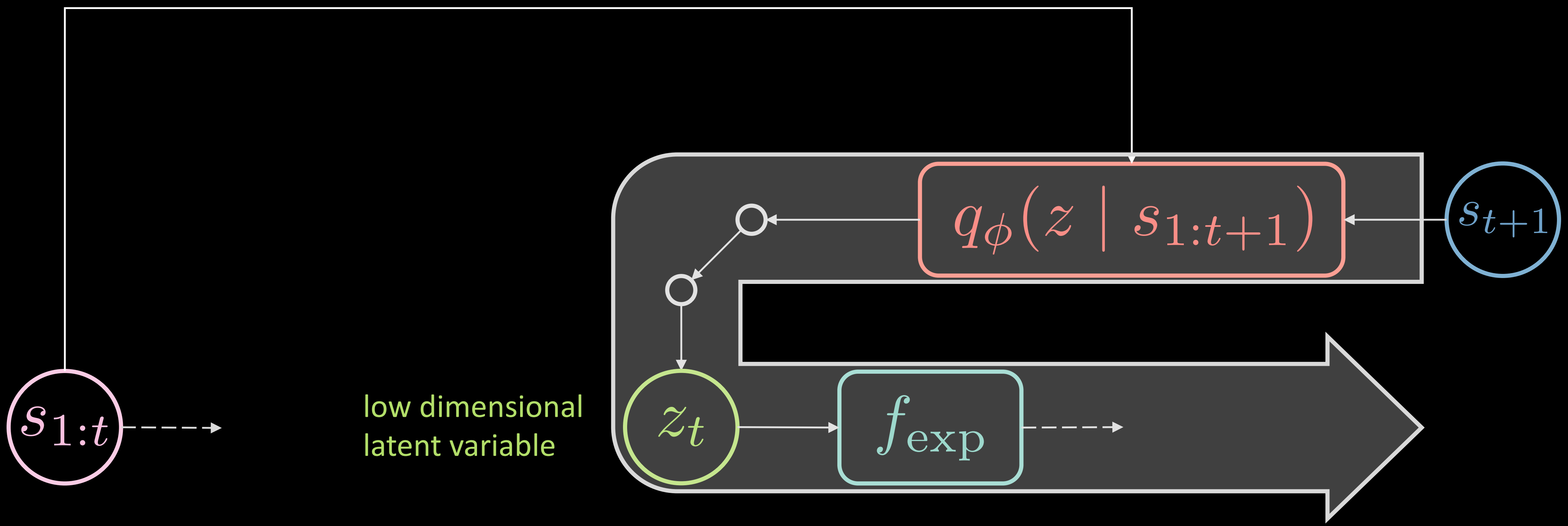


$\mathbb{R}\{z\} \wedge \mathbb{R}\{a\}$

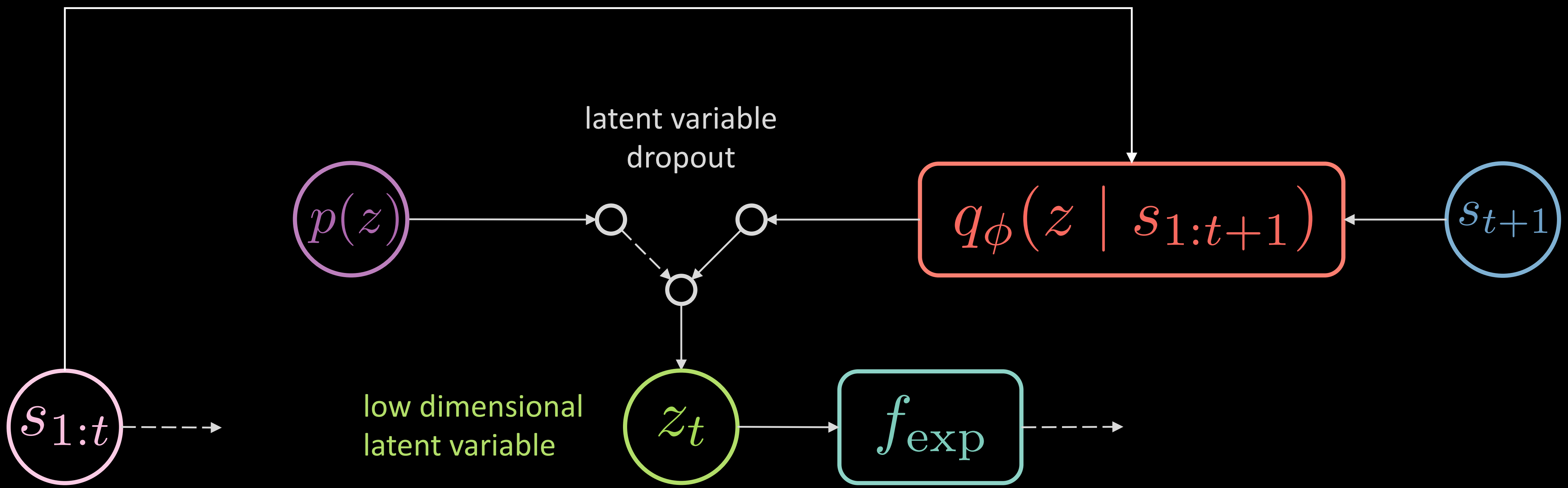
direction of motion

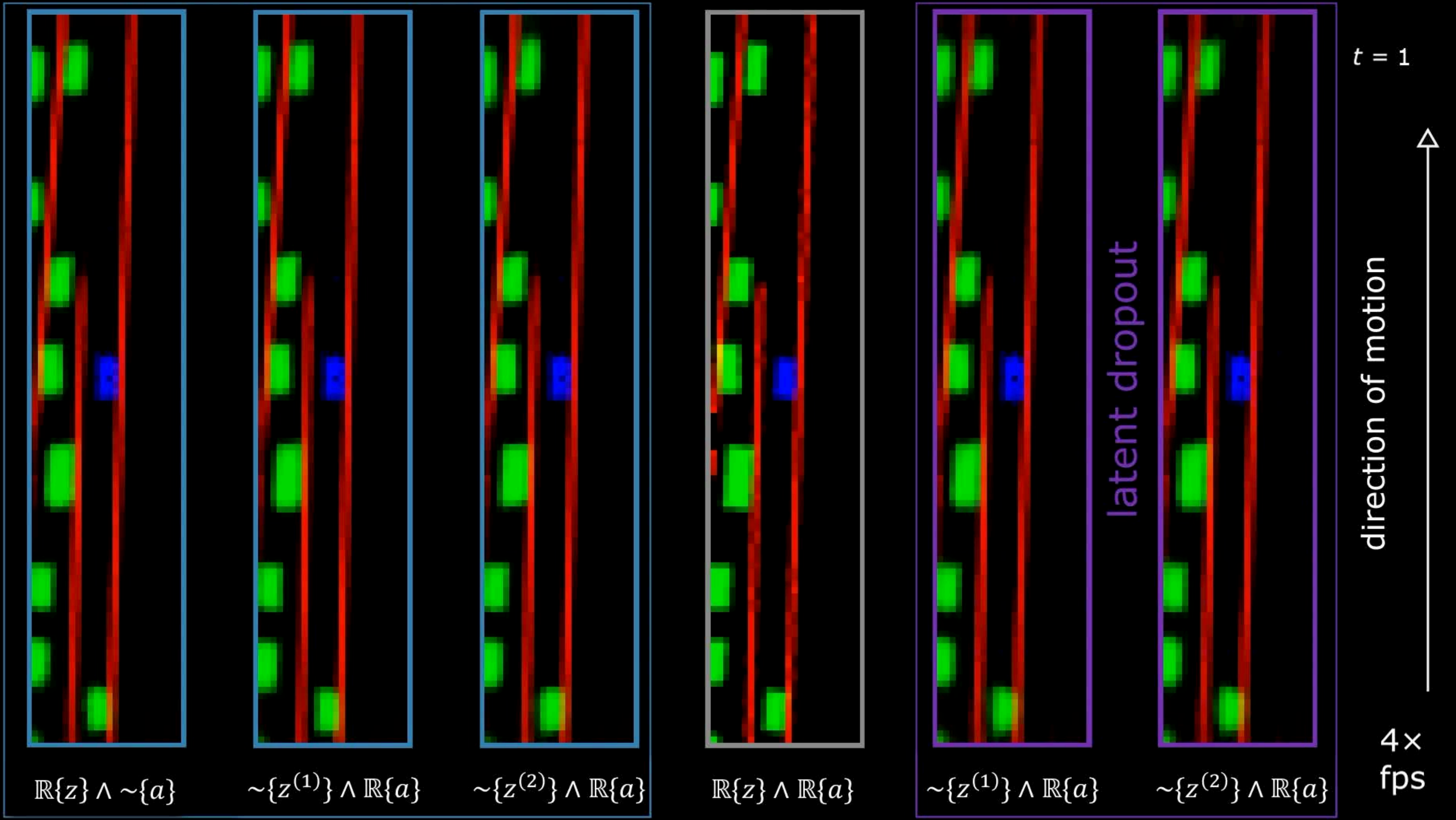
4×
fps

# Latent dropout



# Latent dropout



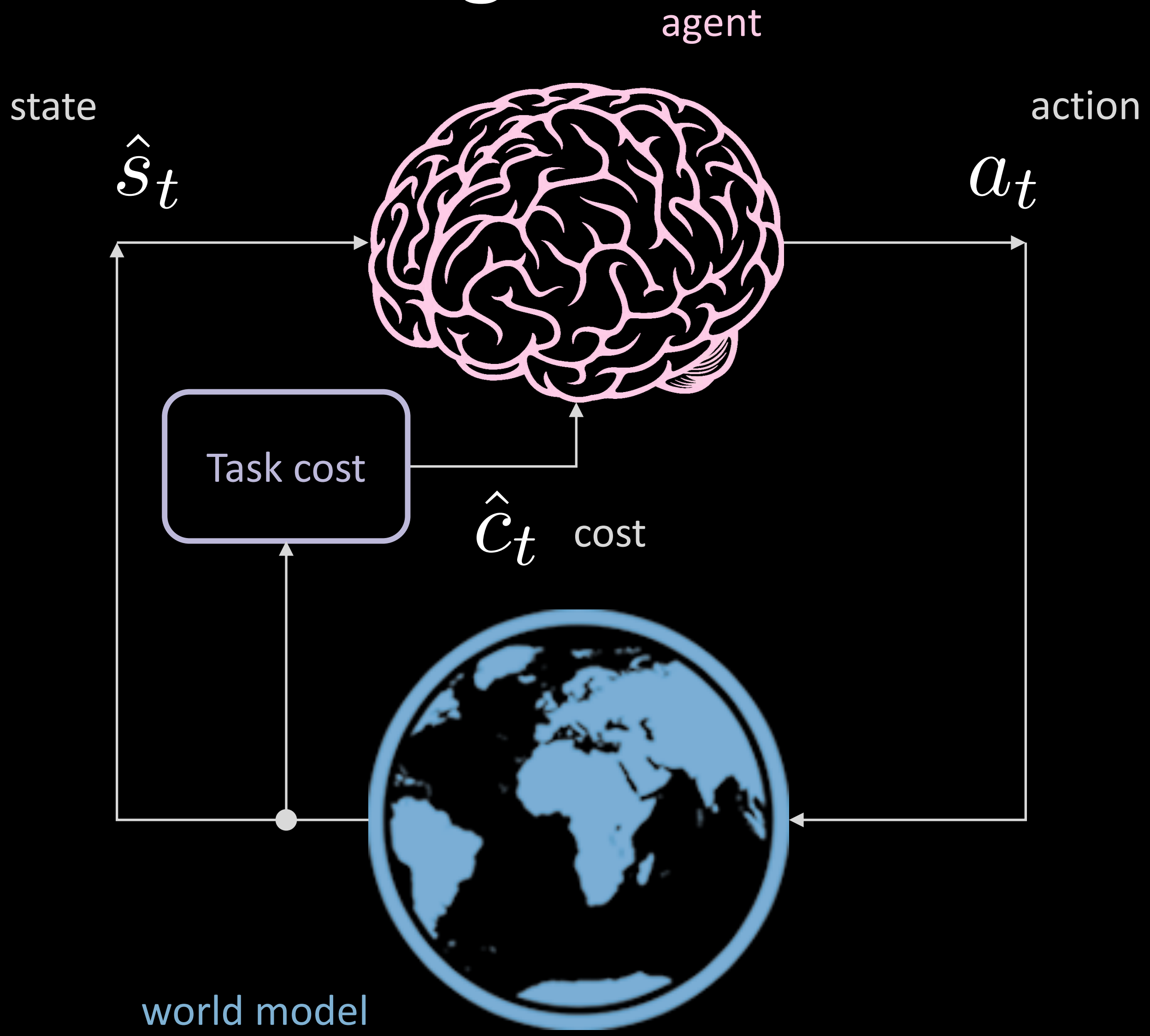


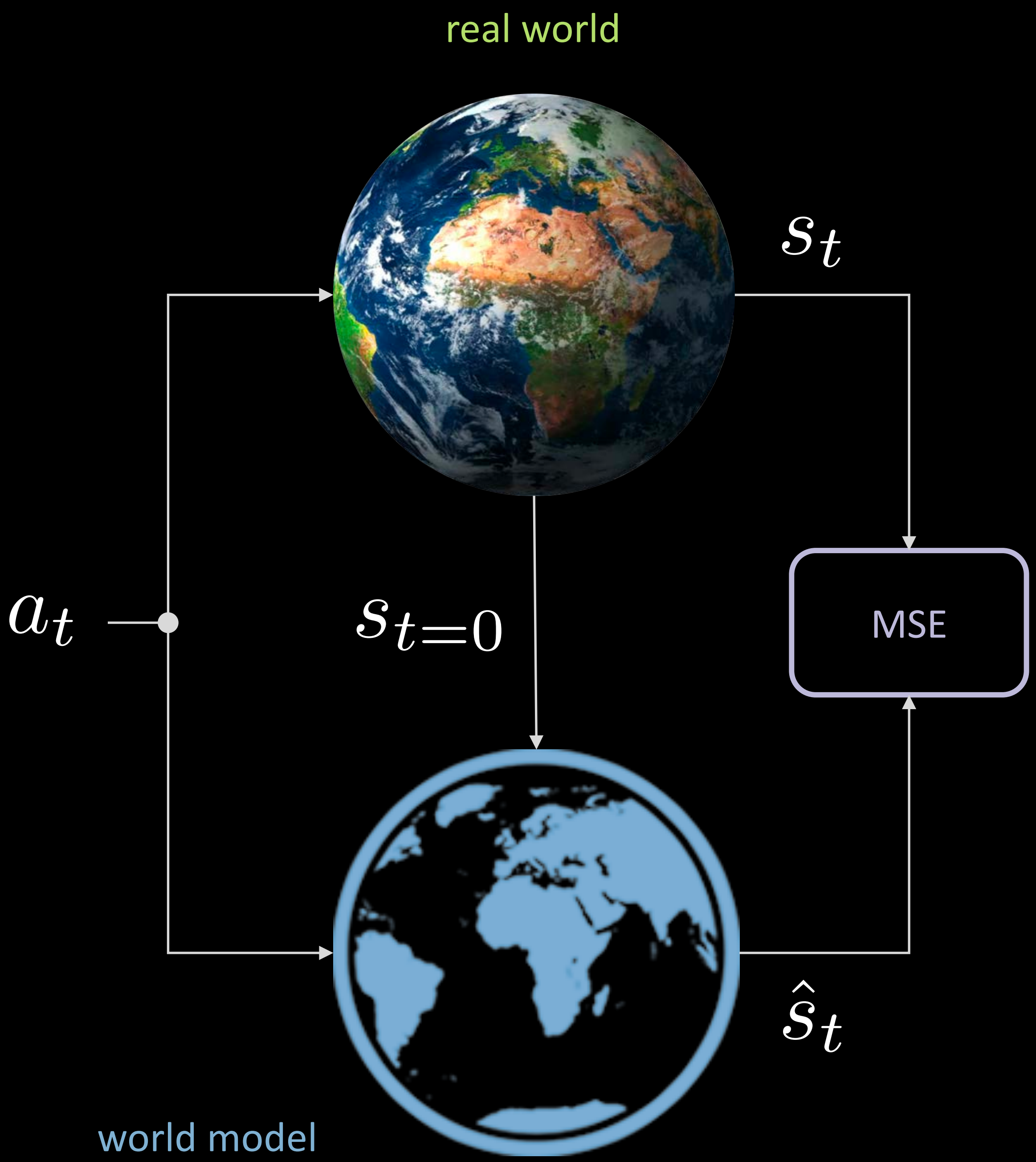
$t = 1$
latent dropout
direction of motion
$\mathbb{R}\{z\} \wedge \sim\{a\}$
$\sim\{z^{(1)}\} \wedge \mathbb{R}\{a\}$
$\sim\{z^{(2)}\} \wedge \mathbb{R}\{a\}$
$\mathbb{R}\{z\} \wedge \mathbb{R}\{a\}$
$\sim\{z^{(1)}\} \wedge \mathbb{R}\{a\}$
$\sim\{z^{(2)}\} \wedge \mathbb{R}\{a\}$
4× fps

# 2. The agent

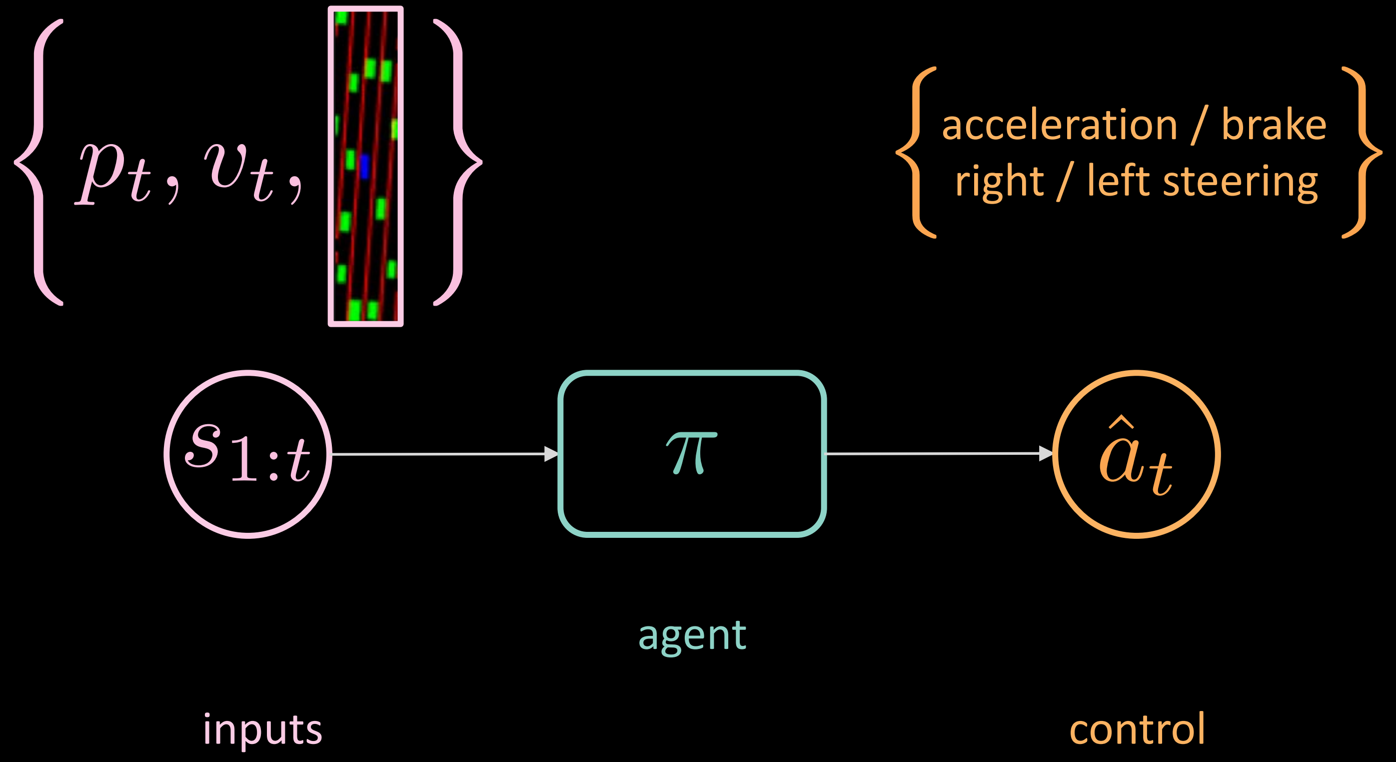
Learning to act

# Planning with task cost



# Agent model

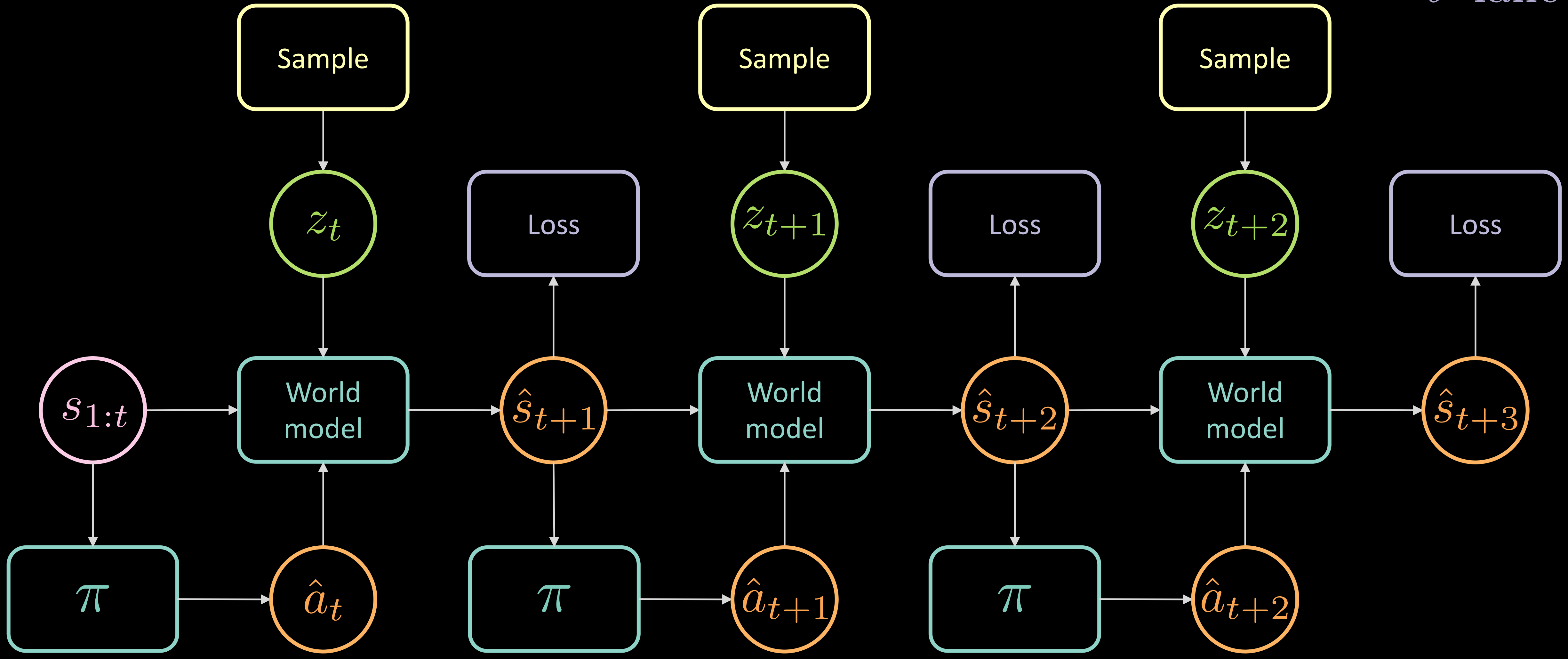


$\{ p_t, v_t,$ $\}$
acceleration / brake
right / left steering
$s_{1:t}$
$\pi$
$\hat{a}_t$
agent
inputs
control

# Training the agent (I)

Loss

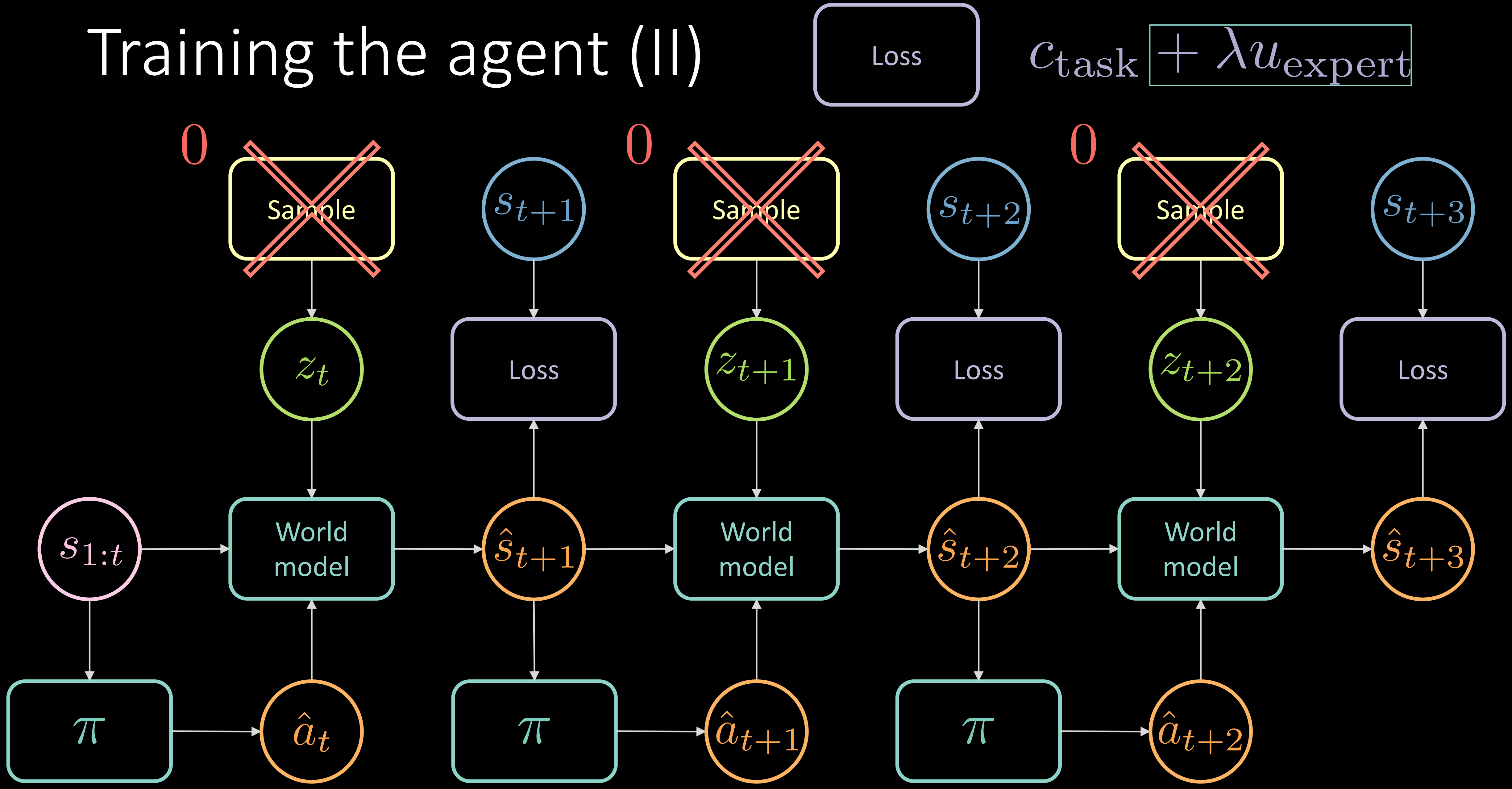
$c_{\text{task}} = c_{\text{proximity}} + \lambda_\ell c_{\text{lane}}$



# Falling from the manifold

imitate expert

# Training the agent (II)

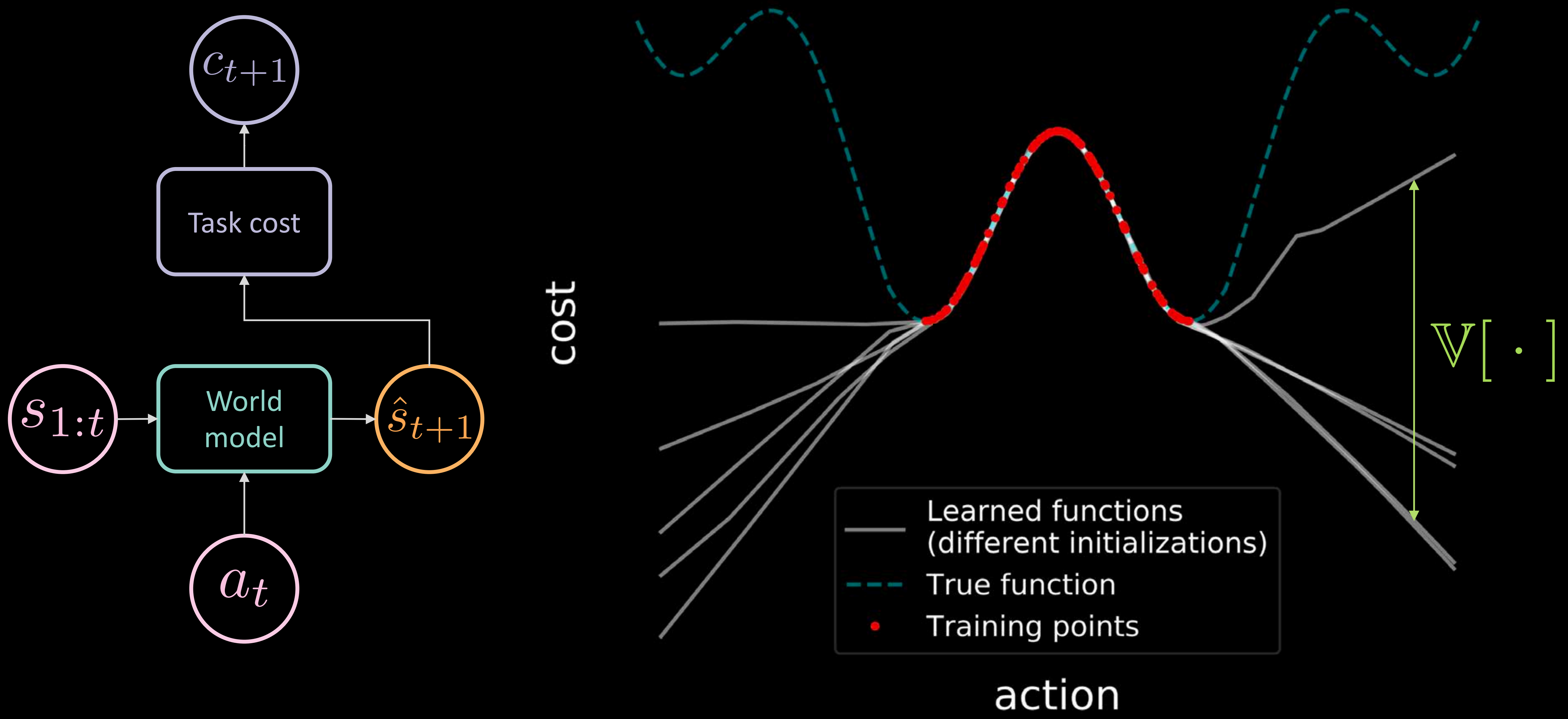


# ~~Falling from the manifold~~ Imitating the experts

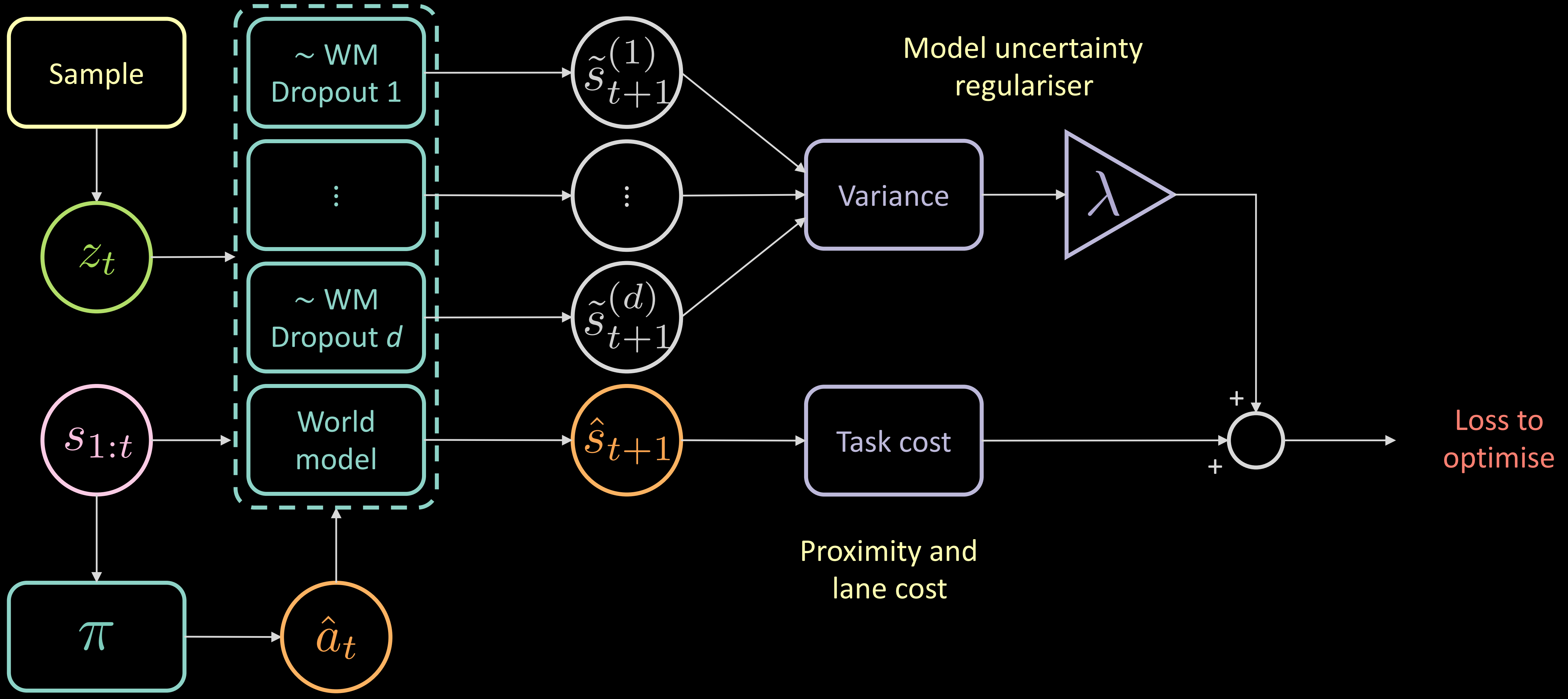
add different manifold attractor

# Forward model uncertainty (I)

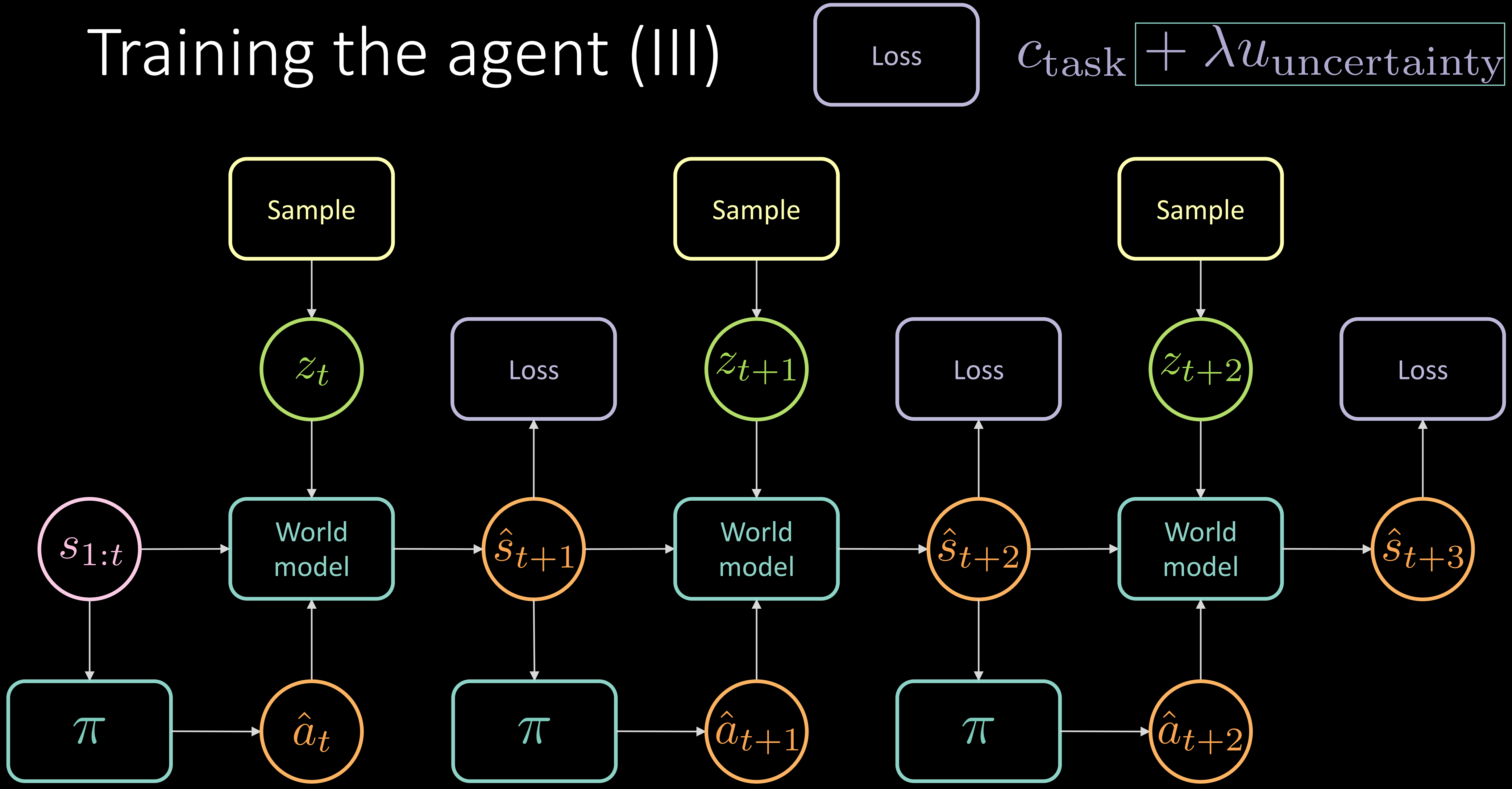
$u_{\text{uncertainty}}$



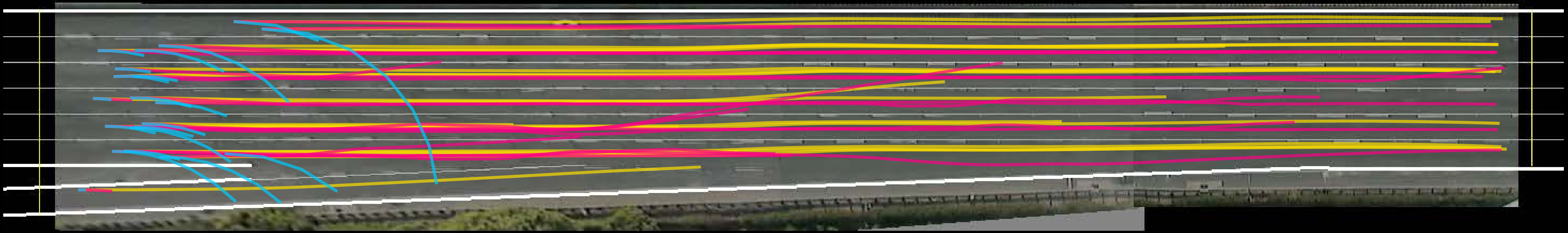
# Forward model uncertainty (II)

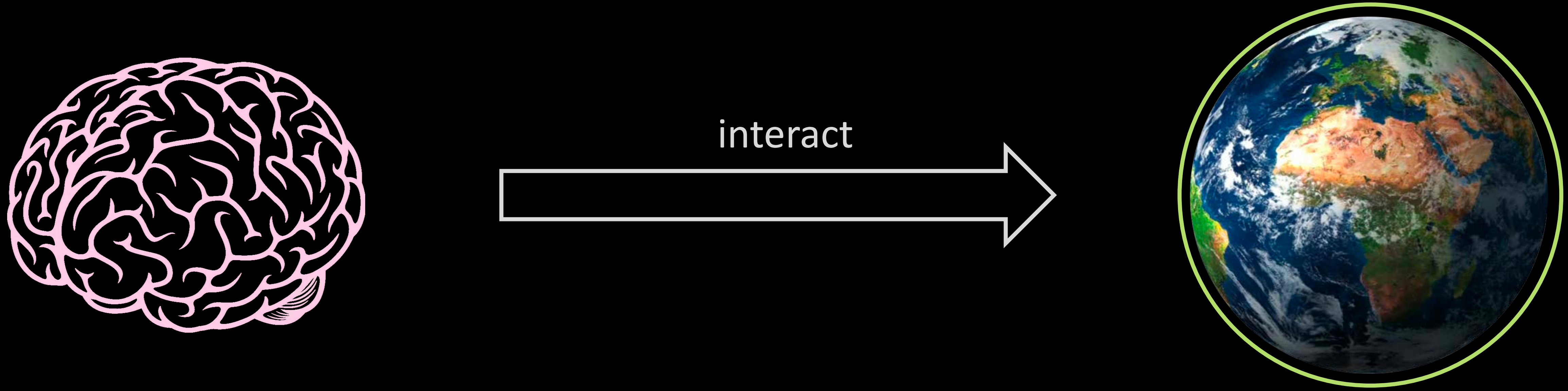


# Training the agent (III)



# ~~Falling from the manifold~~ ~~Imitating the experts~~ Minimising the uncertainty

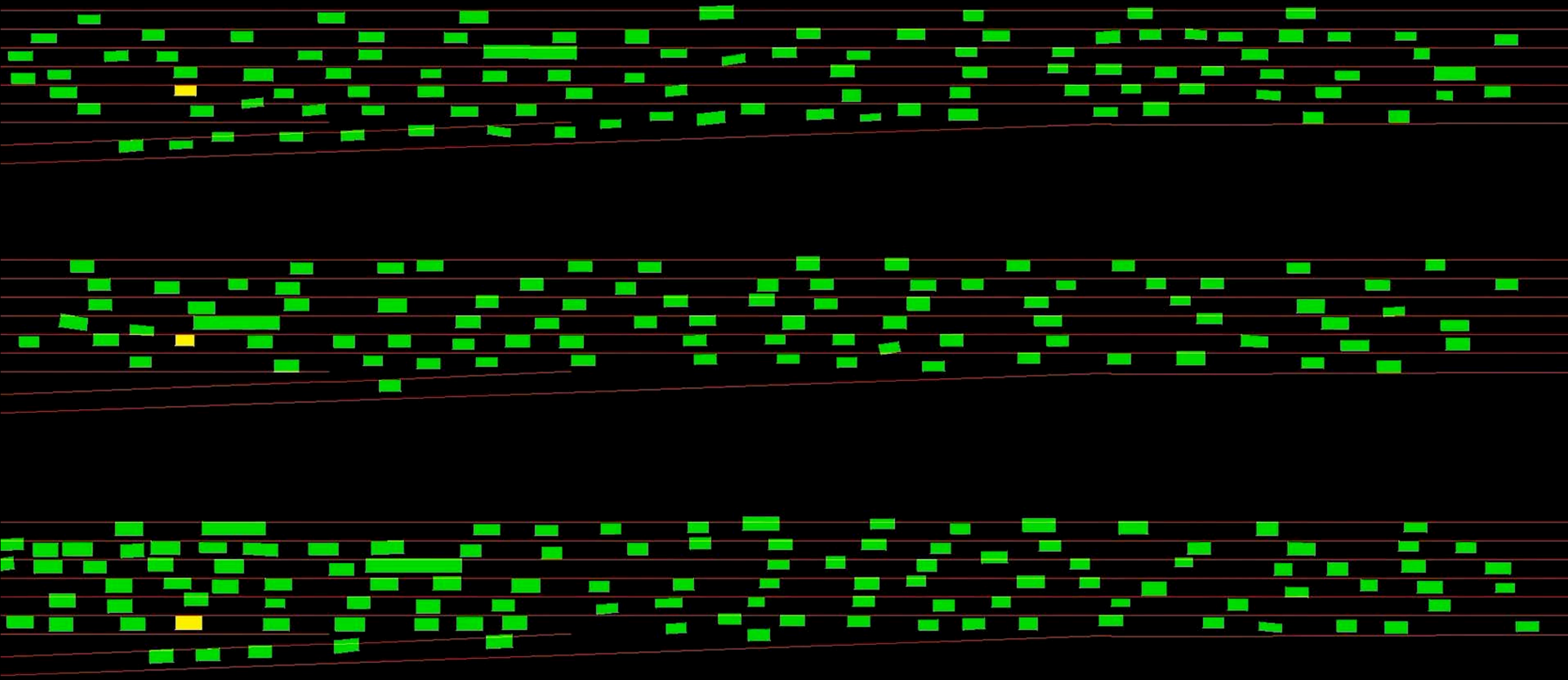





# 3. The evaluation

Computing a performance metric

ghost car / original driver (yellow) our controller / policy (blue) others, blind to us (green)



3 × fps

# Model-predictive policy learning with uncertainty regularisation for driving in dense traffic

**M**odel-predictive **P**olicy learning with **U**ncertainty **R**egularisation

- Uncertainty regularisation
- Latent dropout for improving action sensitivity
- Large-scale data set of driving behaviour from traffic camera
- Additionally, we can "copy the past" with MPExpertR

# Info

- Title: Prediction and Policy-learning Under Uncertainty (PPUU)
- Speaker: Alfredo Canziani @alfcnz
- Collaborators:
  - Mikael Henaff @HenaffMikael
  - Yann LeCun @ylecun
- Slides: `bit.ly/PPUU-slides`
- Article: `bit.ly/PPUU-article`
- Code: available in PyTorch on `bit.ly/PPUU-code`
- Website: `bit.ly/PPUU-web`
- Poster: `bit.ly/PPUU-poster`