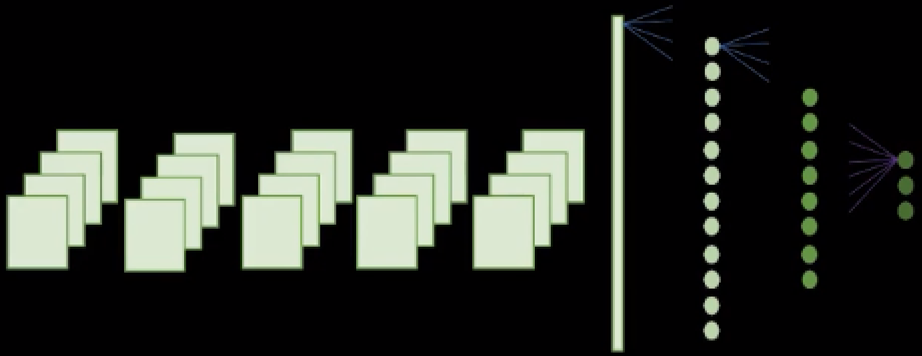
(NeurIPS 2021).

Causal Navigation by Continuous-time Neural Networks

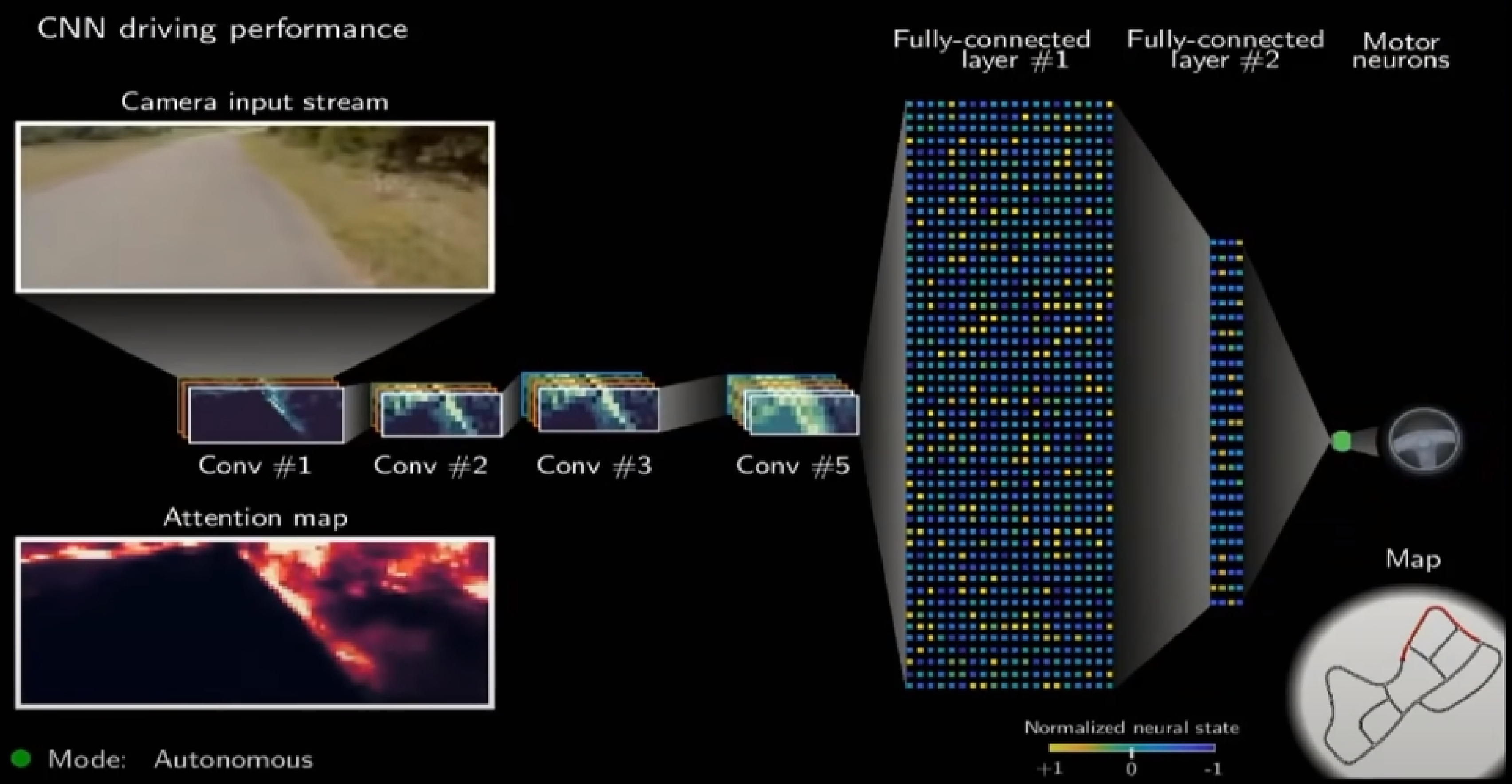
Charles Vorbach[1,*], Ramin Hasani[1,*], Alexander Amini[1], Mathias Lechner[2], Daniela Rus[1]

Camera input stream

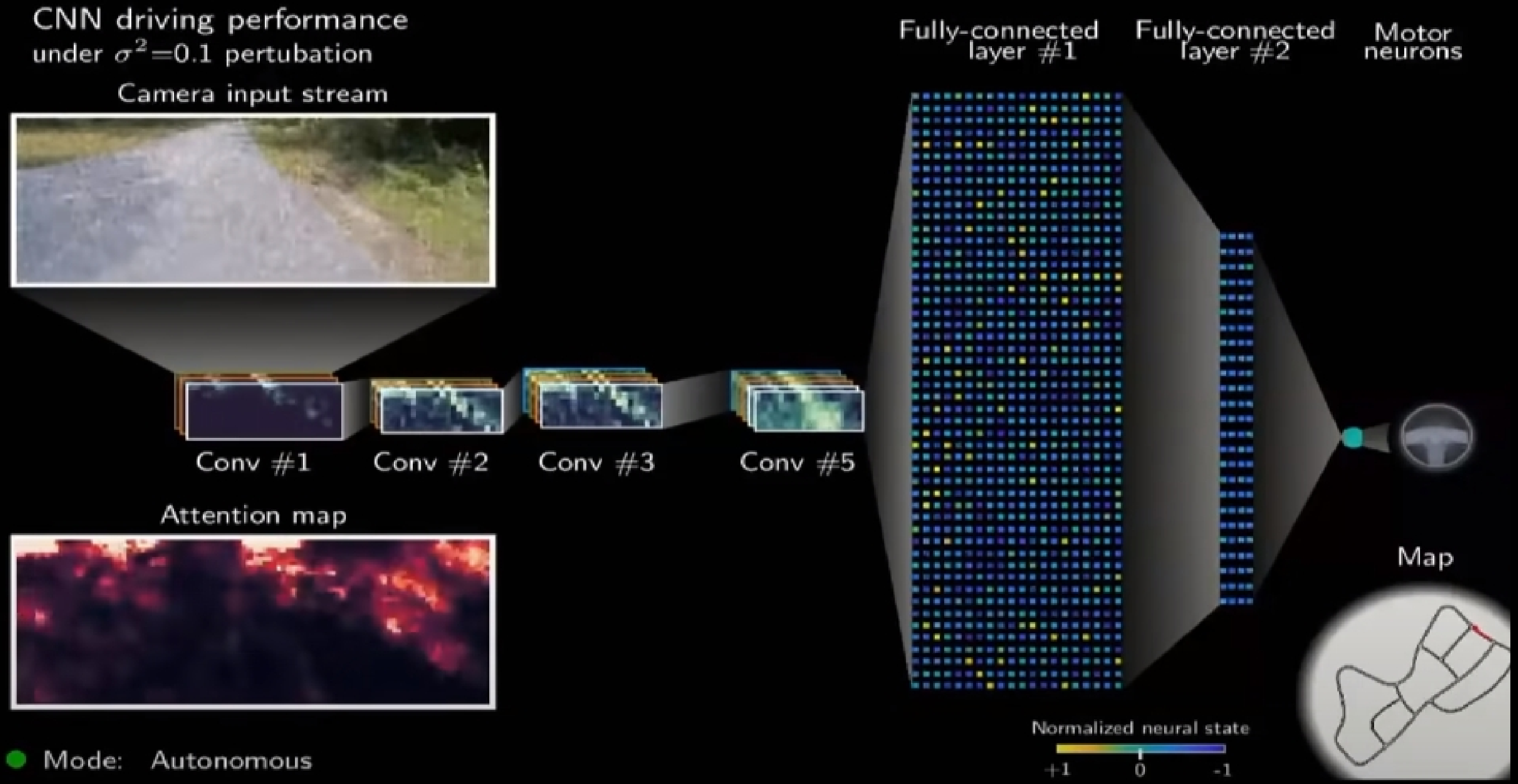
Map



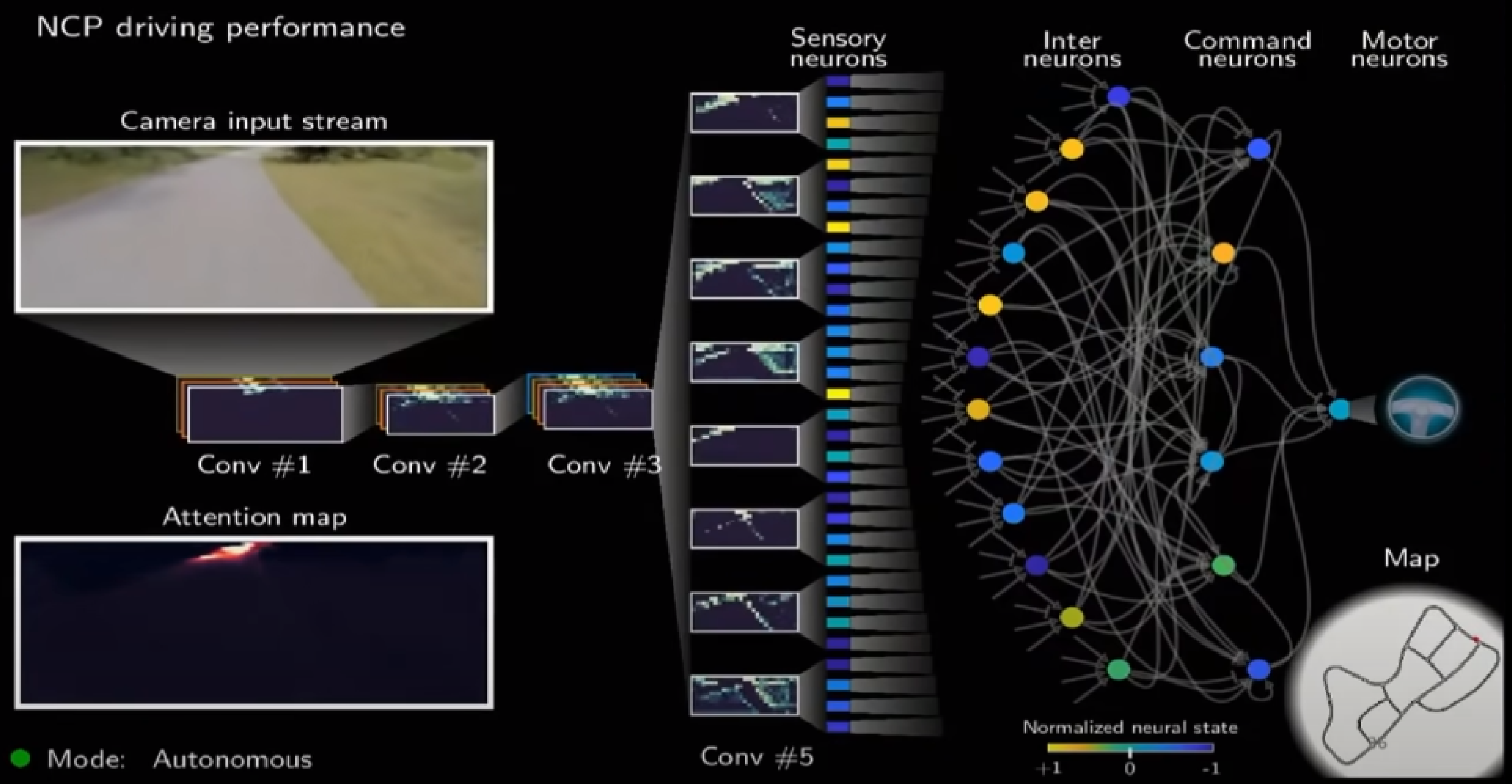
CNN+Fully connected Network



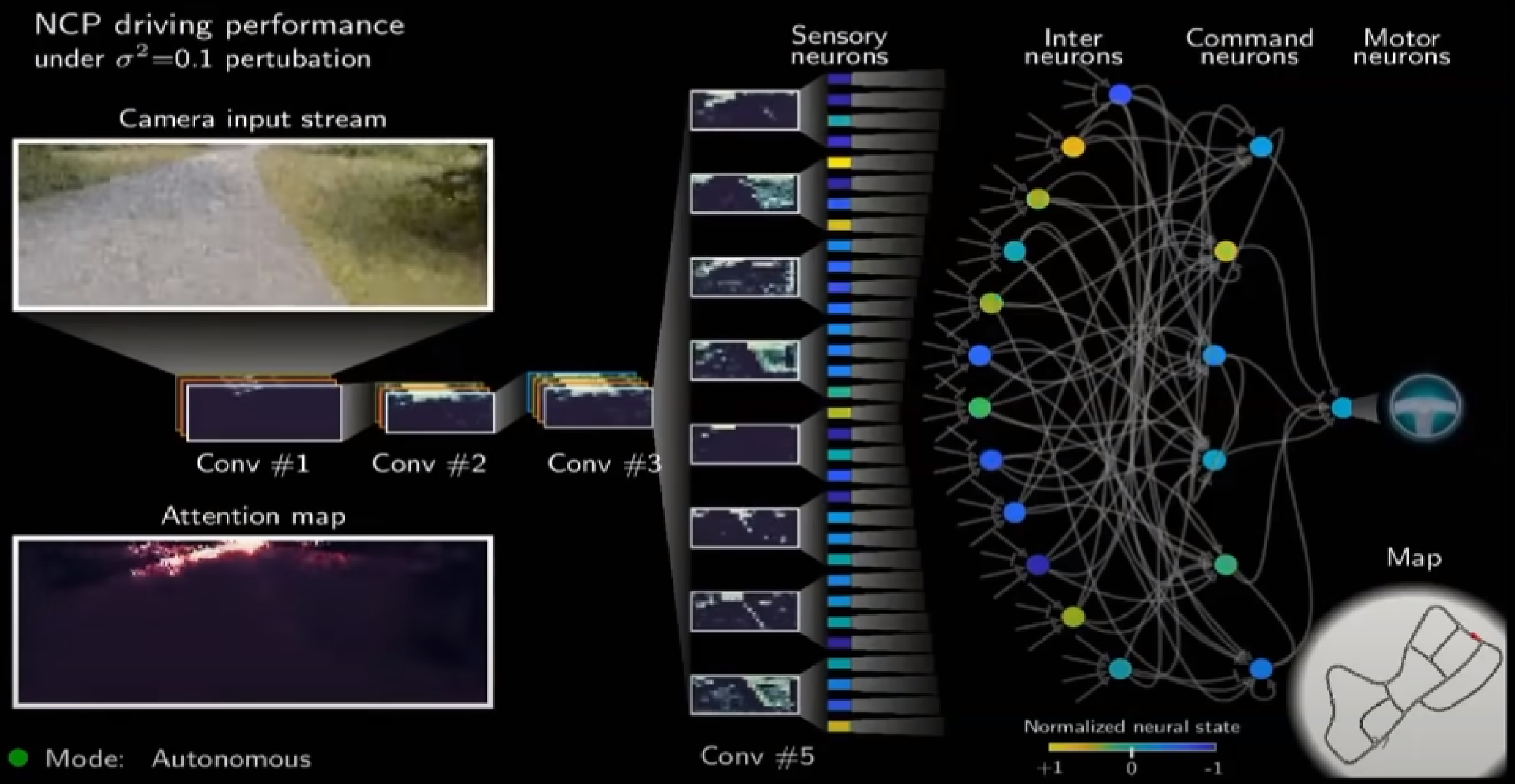
CNN+Fully connected Network + Noise!

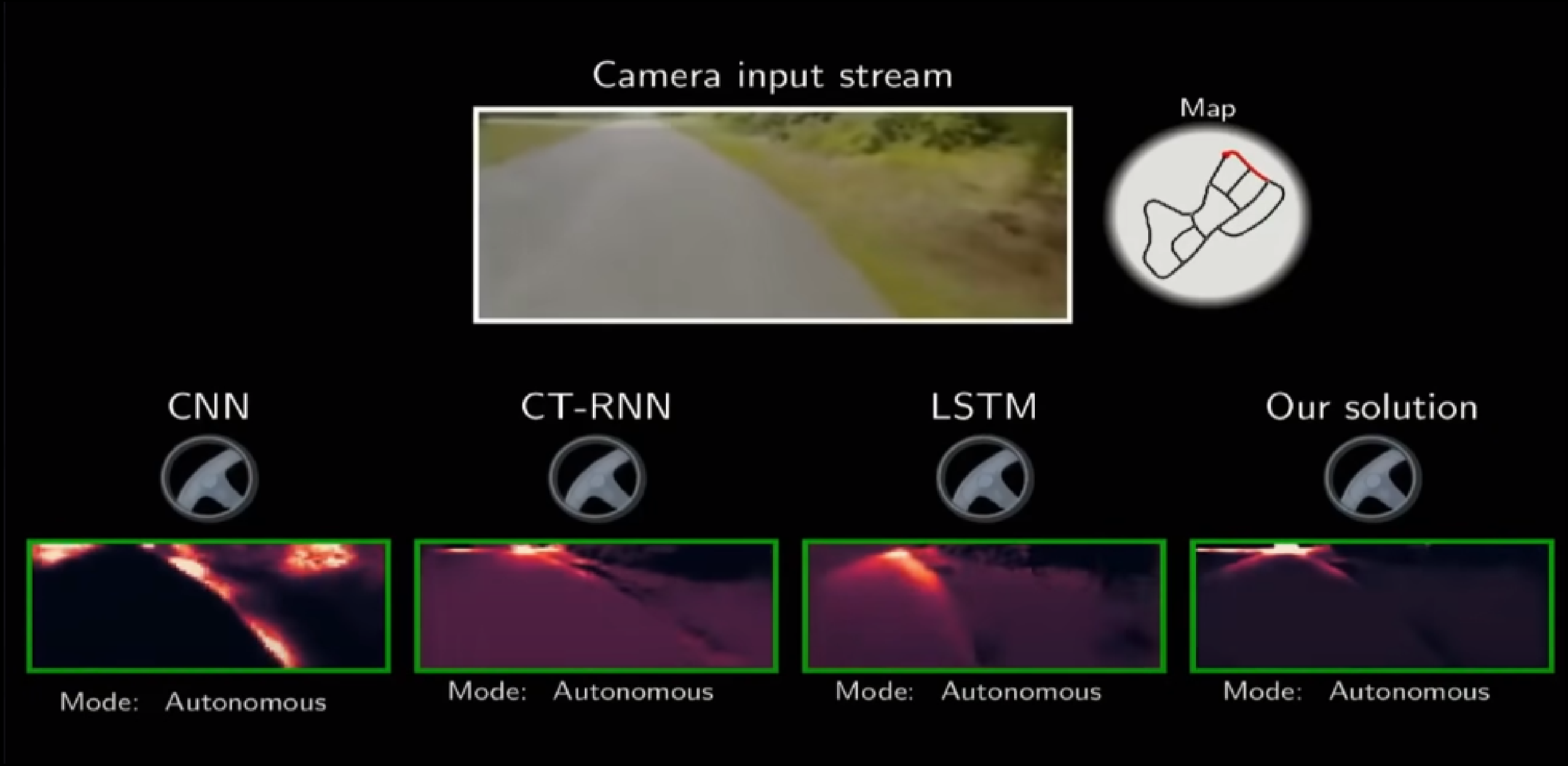


CNN + Neural Circuit Policy



CNN + Neural Circuit Policy + Noise




Camera input stream
Map
CNN
CT-RNN
LSTM
Our solution
Mode: Autonomous
Mode: Autonomous
Mode: Autonomous
Mode: Autonomous

Causal navigation of a drone towards the red box

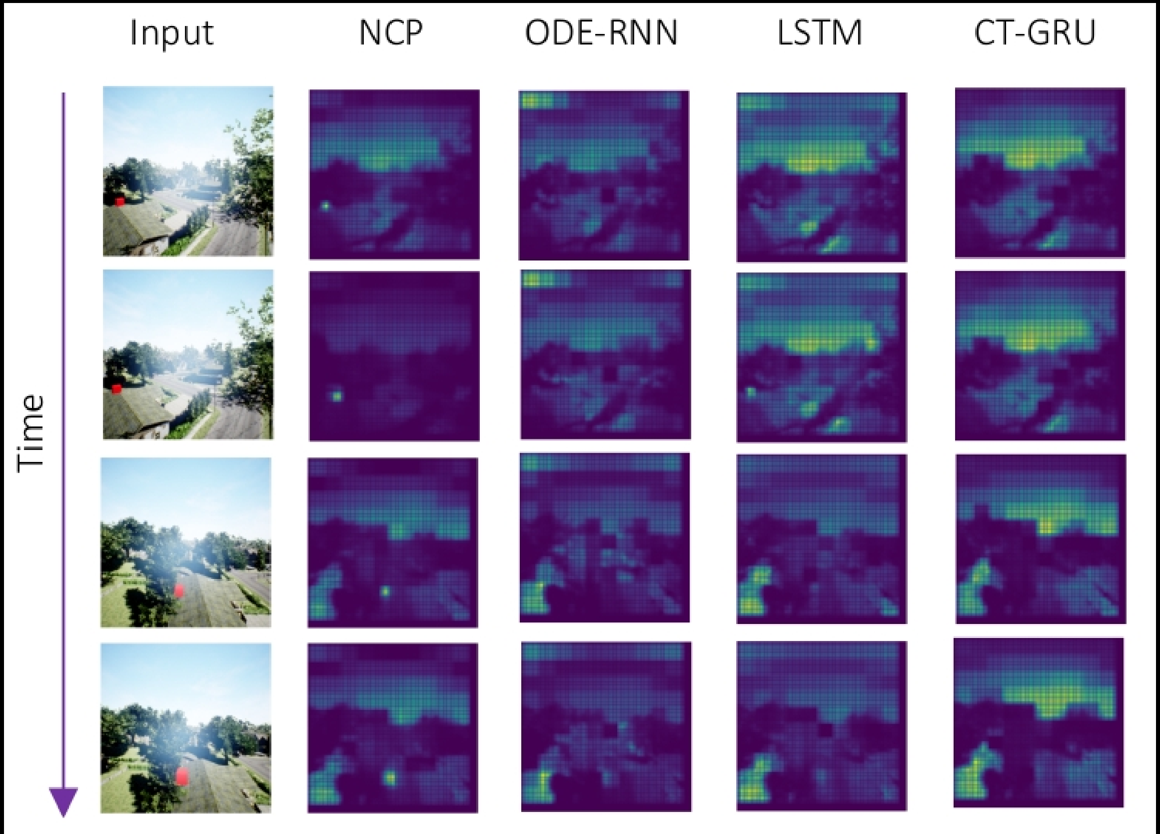


Figure 1: **Causal navigation from raw visual inputs**. Given a sequence of raw RGB inputs (left) a drone is trained to navigate towards the red-cube target. We visualize the saliency maps (right) for each model. Neural circuit policies (Lechner et al., 2020a) (a specific representation of CT-RNNs) can learn causal relationships (i.e., attend to the red-cube) directly from data while other models fail to do so. ODE-RNNs (Rubanova et al., 2019b), LSTM (Hochreiter and Schmidhuber, 1997) and CT- Gated Recurrent Units (Mozer et al., 2017). Saliency maps are computed by the visual backprop algorithm (Bojarski et al., 2016).

Main Message

- Differential Equations can form causal structures for navigation tasks
- Continuous-time neural networks (our method) can learn the causal relationships between the agent and the environment, while RNNs can't.

Continuous-Time Neural Networks

Most of dynamical system, like this navigation task, ideally can be described by differential equations

State

$$\frac{dx}{dt} = \underbrace{f(\mathbf{x}(t), t, \theta),}_{\text{Neural Network}} \quad \mathbf{x} \in \mathbb{R}^d,$$

Continuous-time recurrent neural networks (CT-RNNs)
Achieving stability with an extra term,
τ: time-constant

$$\frac{d\mathbf{x}(t)}{dt} = -\frac{\mathbf{x}(t)}{\tau} + f(\mathbf{x}(t), t, \theta),$$

liquid time-constant networks (LTCs)

$$\frac{d\mathbf{x}(t)}{dt} = -\Big[\frac{1}{\tau} + f(\mathbf{x}(t), \mathbf{I}(t), t, \theta)\Big] \odot \mathbf{x}(t) + f(\mathbf{x}(t), \mathbf{I}(t), t, \theta) \odot A.$$

(Hasani et al., 2021b)

Methodology

Inputs to the system

Hidden states

Assume the Dynamic Causal Model to be:

$$d\mathbf{x}/dt = (A + \mathbf{I}(t)B)\mathbf{x}(t) + C\mathbf{I}(t)$$

$$A = \frac{\partial F}{\partial \mathbf{x}(t)}\Big|_{I=0}, \quad B = \frac{\partial^2 F}{\partial \mathbf{x}(t)\partial \mathbf{I}(t)}, \quad C = \frac{\partial F}{\partial \mathbf{I}(t)}\Big|_{x=0},$$

- Matrix A is a fixed internal coupling of the system
- Matrix B controls the impact of the inputs on the the coupling sensitivity
- Matrix C embodies the external inputs' influence on the state of the system

Let f be an activation function such as tanh:

$$f(\mathbf{x}(t), \mathbf{I}(t), t, \theta) = tanh(W_r\mathbf{x} + W\mathbf{I} + b)$$

$$\begin{cases} \dfrac{\partial F}{\partial \mathbf{I}}\Big|_{x=0} = \boxed{W}(1 - f^2) \odot A \\[2ex] \dfrac{\partial^2 F}{\partial \mathbf{x}(t)\partial \mathbf{I}(t)} = W(f^2 - 1) \odot [2\boxed{W_r} f \odot (\boxed{A} - x) + 1] \end{cases}$$

Weights to be optimized!

Table 4: Closed-loop evaluation of trained policies on various navigation and interaction tasks. Agents and policies are reinitialized randomly at the beginning of each trial (n=50). Values correspond to success rates (higher is better).

	Static Target					Chasing				Hiking
Model	**Clear**	**Fog**	**Light Rain**	**Heavy Rain**	**Occlusion**	**Clear**	**Fog**	**Light Rain**	**Heavy Rain**	**Clear**
CNN	36%	6%	32%	2%	4%	50%	42%	54%	28%	0%
LSTM	24%	22%	22%	4%	20%	66%	**62%**	56%	44%	2%
ODE-RNN	18%	10%	18%	2%	24%	52%	42%	62%	44%	4%
CT-GRU	40%	8%	**60%**	32%	28%	38%	36%	48%	42%	0%
NCP (ours)	**48%**	**40%**	52%	**60%**	**32%**	**78%**	52%	**84%**	**54%**	**30%**

Reference links

1. Causal Navigation by Continuous-time Neural Networks
2. CAUSALITY FOR MACHINE LEARNING
3. Dynamic causal modelling
4. Liquid Time-constant Networks
5. https://slideslive.com/38968213/causal-navigation-by-continuoustime-neural-networks?ref=recommended
6. https://www.youtube.com/watch?v=IlliqYiRhMU&list=WL&index=3&ab_channel=MITCBMM