

Reinforcement Learning & Beyond

Remember MDPs

A Markov Decision Process (MDP) is a problem described by a tuple $M = (S, A, P, R, \gamma)$, where

- S is a finite set of states
- A is a finite set of actions
- $P : S \times A \times S \rightarrow [0, 1]$ is transition probability distribution, i.e., $P(s'|s, a)$ is the probability that the system transitions into state s' when the system is in state s and action a is taken.
- $R : S \rightarrow \mathbb{R}$ is a reward function;

and the problem is to find a policy $\pi : S \rightarrow A$ such that

$$V_{\pi}(s) := \mathbb{E} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t) \mid s_0 = s \right] \quad V_{\pi}(s) = \mathbb{E} \left[\sum_{t=0}^H \gamma^t R(s_t) \mid s_0 = s \right]$$

is maximized for all $s \in S$.

**In some formulations, Reward R may depend on the state and the action chosen at that state: $R(s,a)$.*

Value Iteration Solves the MDP: Infinite Horizon Case

$$V_{\pi}(s) := \mathbb{E} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t) \mid s_0 = s \right]$$

- How does this algorithm? Why does it solve our problem?

Value Iteration

- 1 $V_0(s) \leftarrow 0$ for all $s \in S$;
- 2 **for** $k = 1, 2, \dots$ **do**
- 3 **for all** $s \in S$ **do**
- 4 $V_{k+1}(s) \leftarrow \max_{a \in A} [R(s) + \gamma \sum_{s' \in S} P(s'|s, a) V_k(s')]$;

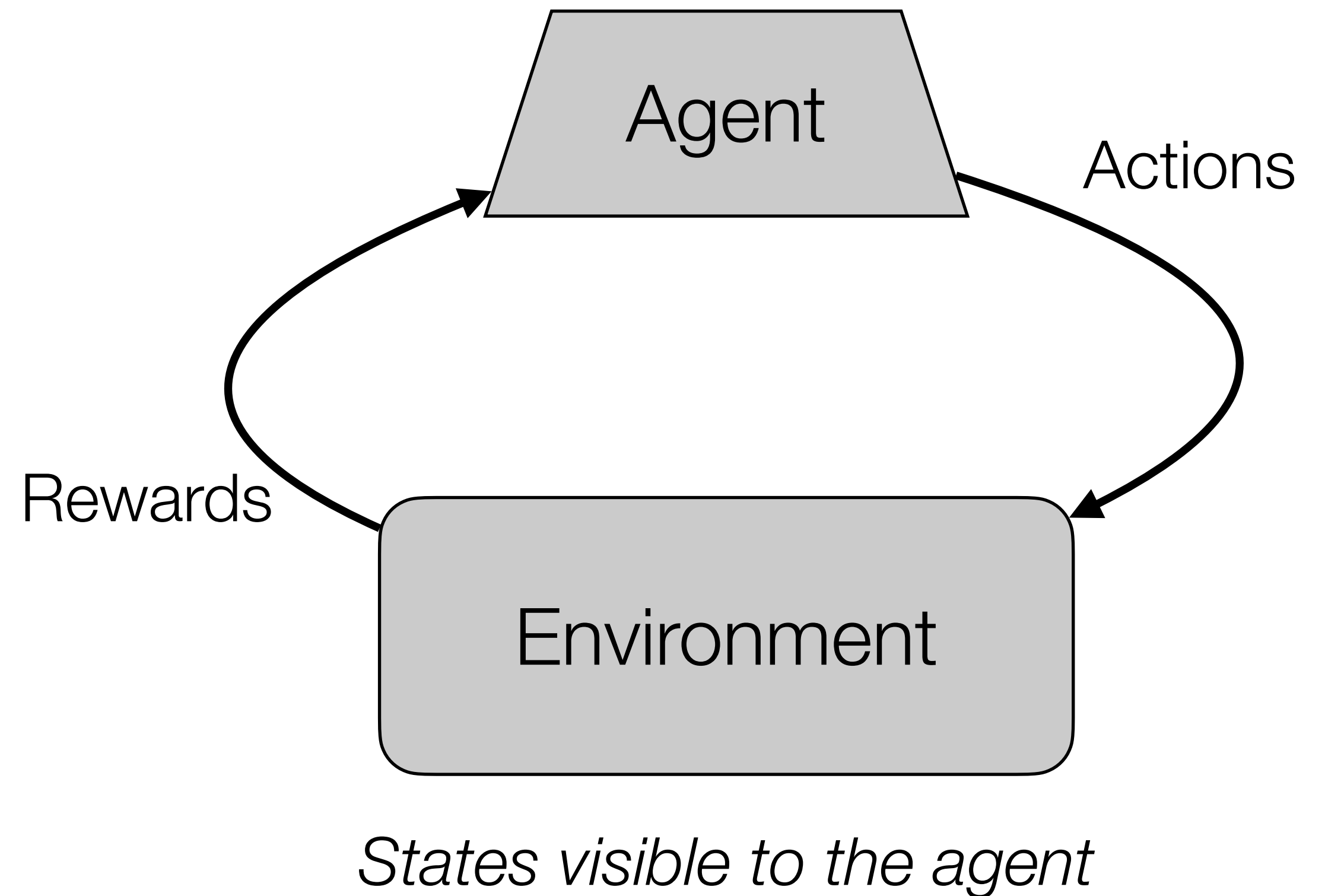
We have $\lim_{k \rightarrow \infty} V_k(s) = V^*(s)$ for all $s \in S$.

Reinforcement Learning

- Can we compute a policy simply if we did not know the transition function.
- The agent will “experiment” with the environment and construct a policy that maximizes the long term reward:

$$V_{\pi}(s) = \mathbb{E} \left[\sum_{t=0}^H \gamma^t R(s_t) \mid s_0 = s \right]$$

$$V_{\pi}(s) := \mathbb{E} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t) \mid s_0 = s \right]$$

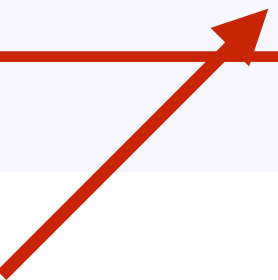


Value Iteration Update

- Why can't we use the value iteration?

Value Iteration

```
1  $V_0(s) \leftarrow 0$  for all  $s \in S$ ;  
2 for  $k = 1, 2, \dots$  do  
3   for all  $s \in S$  do  
4      $V_{k+1}(s) \leftarrow \max_{a \in A} [R(s) + \gamma \sum_{s' \in S} P(s'|s, a) V_k(s')]$ ;
```



We do not know the transition probabilities...

- Instead, let's learn an action-value function - also called the Q function:

$$Q_{\pi}(s, a) = \mathbb{E} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t) \mid s_0 = s, a_0 = a \right]$$

Q value: the long term value at state s taking action a



How do we do this?

- We update the Q value as the agent explores the environment:

$$Q(\mathbf{s}_t, \mathbf{a}_t) \leftarrow Q(\mathbf{s}_t, \mathbf{a}_t) + \alpha \underbrace{\left[r(\mathbf{s}_t, \mathbf{a}_t) + \gamma \max_{\mathbf{a}} Q(\mathbf{s}_{t+1}, \mathbf{a}) - Q(\mathbf{s}_t, \mathbf{a}_t) \right]}_{\text{Bellman error}}$$

- This gives us a policy: at state \mathbf{s} , take action \mathbf{a} that maximizes $\mathbf{Q}(\mathbf{s}, \mathbf{a})$.
- This policy constantly exploits the best actions it has found! – In addition, we must allow the agent to “explore” environment! Hence, we need a way to constantly explore the environment!

ϵ -greedy policy

- Explore with probability ϵ : take a random action
- Exploit with probability $1-\epsilon$: take the action according to the Q function

Q-Learning

Initialize $Q(s, a), \forall s \in \mathcal{S}, a \in \mathcal{A}(s)$, arbitrarily, and $Q(\text{terminal-state}, \cdot) = 0$

Repeat (for each episode):

Initialize S

Repeat (for each step of episode):

Choose A from S using policy derived from Q (e.g., ϵ -greedy)

Take action A , observe R, S'

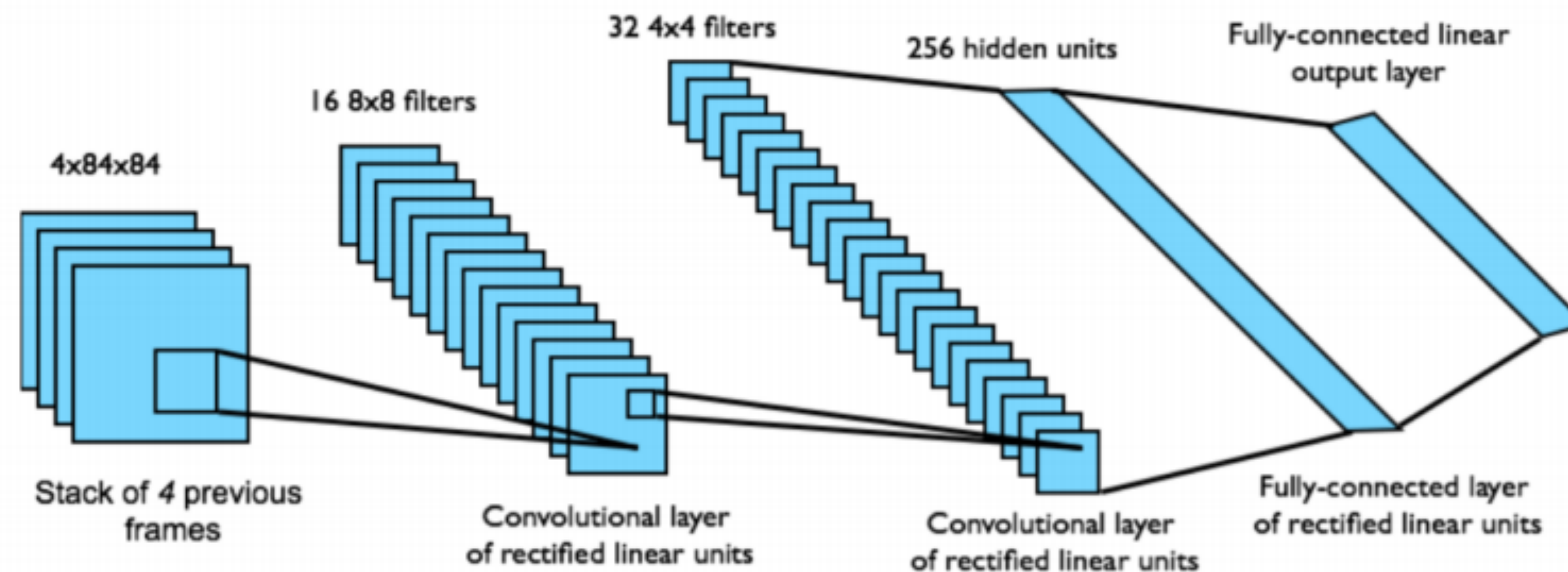
$$Q(S, A) \leftarrow Q(S, A) + \alpha [R + \gamma \max_a Q(S', a) - Q(S, A)]$$

$S \leftarrow S'$;

until S is terminal

Q representations

- We can build a tabular representation: For each (\mathbf{s}, \mathbf{a}) pair, we compute $\mathbf{Q}(\mathbf{s}, \mathbf{a})$. The agent will eventually try all states \mathbf{s} and all actions \mathbf{a} several times, and learn $\mathbf{Q}(\mathbf{s}, \mathbf{a})$ for all states \mathbf{s} and all actions \mathbf{a} .
- This may not scale well when \mathbf{S} is too large!
- Instead, we can represent Q with a neural network. This idea has been around for decades, but in 2015, DeepMind used a DNN to teach an agent how to play Atari games - now called **deep Q-learning**:



Playing Atari with Deep Reinforcement Learning

Volodymyr Mnih Koray Kavukcuoglu David Silver Alex Graves Ioannis Antonoglou

Daan Wierstra Martin Riedmiller

DeepMind Technologies

{vlad, koray, david, alex.graves, ioannis, daan, martin.riedmiller} @ deepmind.com

Abstract

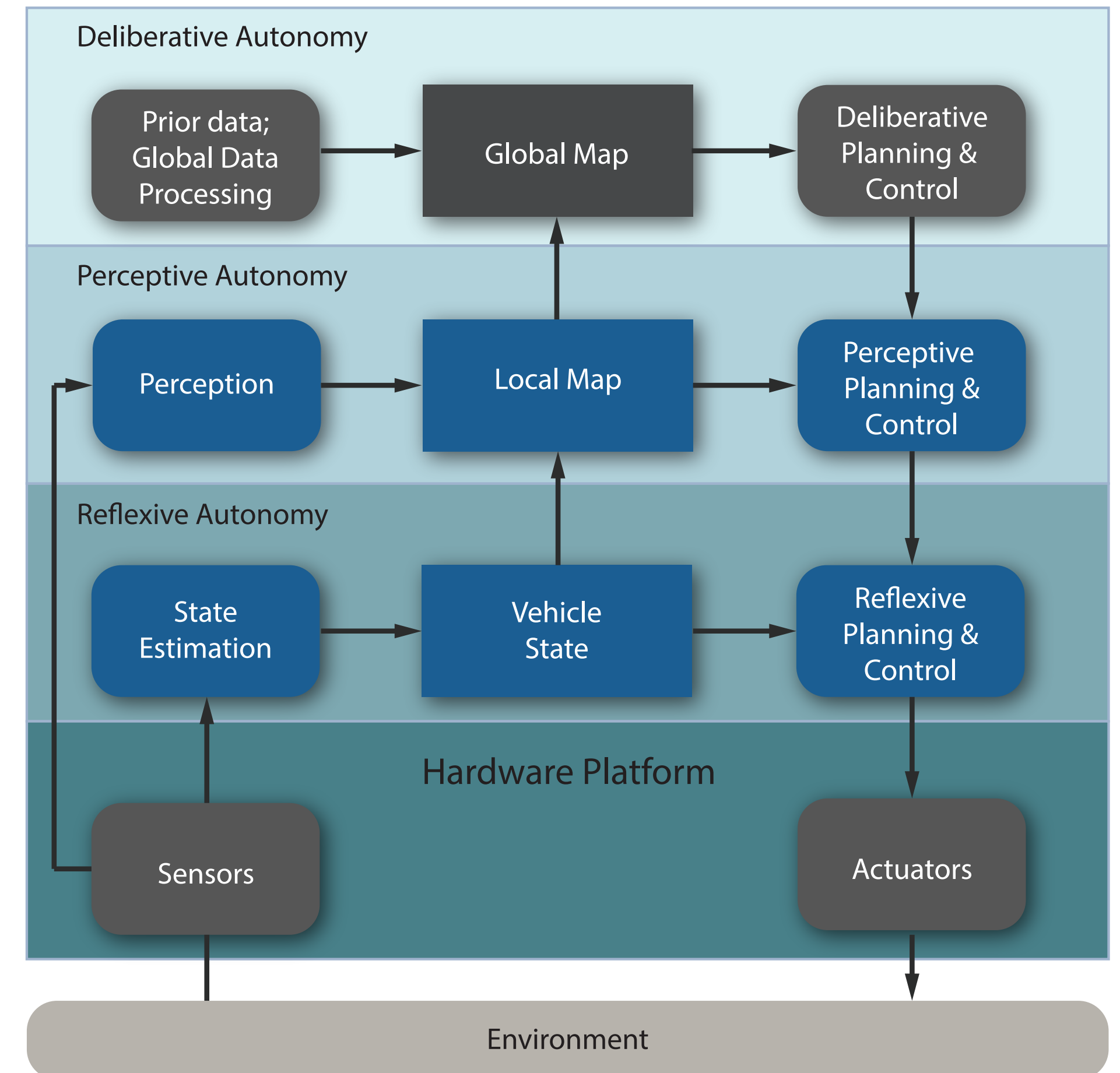
We present the first deep learning model to successfully learn control policies directly from high-dimensional sensory input using reinforcement learning. The model is a convolutional neural network, trained with a variant of Q-learning, whose input is raw pixels and whose output is a value function estimating future rewards. We apply our method to seven Atari 2600 games from the Arcade Learning Environment, with no adjustment of the architecture or learning algorithm. We find that it outperforms all previous approaches on six of the games and surpasses a human expert on three of them.

Deep RL has taken off!

- Human performance has been suppressed in pretty much every game!
 - AI has more precision
 - AI does better on strategy
- We are entering a new “Machine Data” Era - where machines learn from experience (rather than from human demonstration / human-generated data) and ultimately are better than humans

What did we cover? What else is missing?

- We have only scratched the surface on autonomy, covering parts of:
 - Deliberative planning & control
 - Perceptive Planning & control
 - State estimation
- The reminder is equally exciting and deep!



 **OPTIMUS**
RIDE



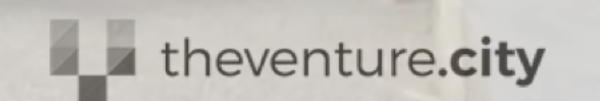
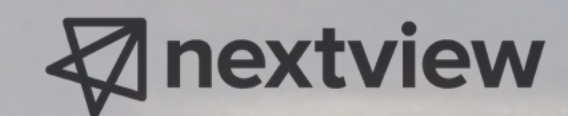
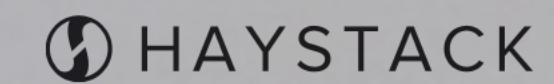
REIMAGINE MOBILITY

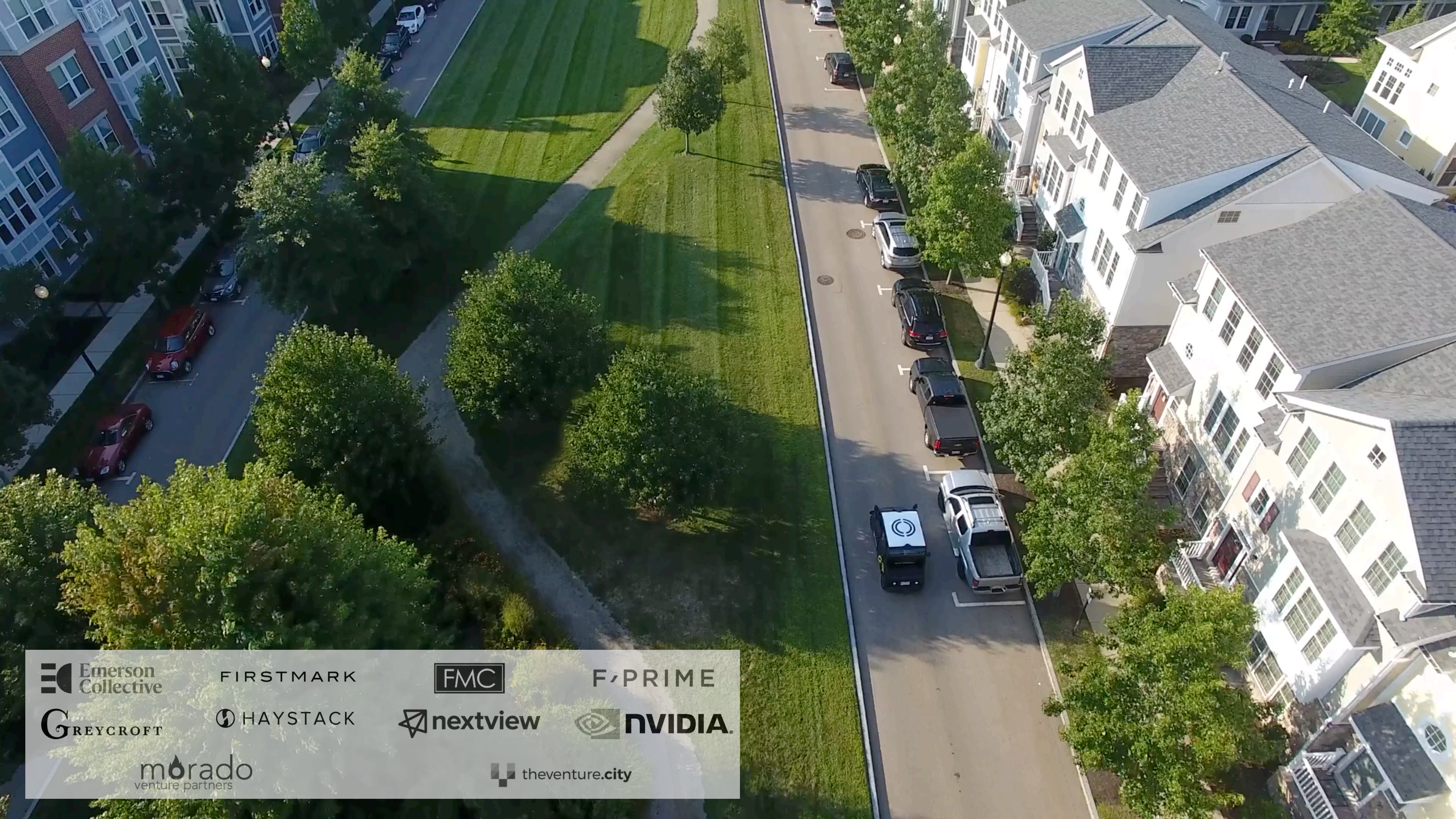


FIRSTMARK



F/PRIME






 Emerson
Collective


FIRSTMARK

 FMC

F/PRIME


 GREYCROFT

 HAYSTACK

 nextview

 nVIDIA

 morado
venture partners

 theventure.city

BOSTON HEADQUARTERS

BOSTON'S SELF-DRIVING
VEHICLE ZONE

OPTIMUS RIDE COMPLEX

HALLEY RISE

Reston, VA



Geofenced
Community



Large Service
Area

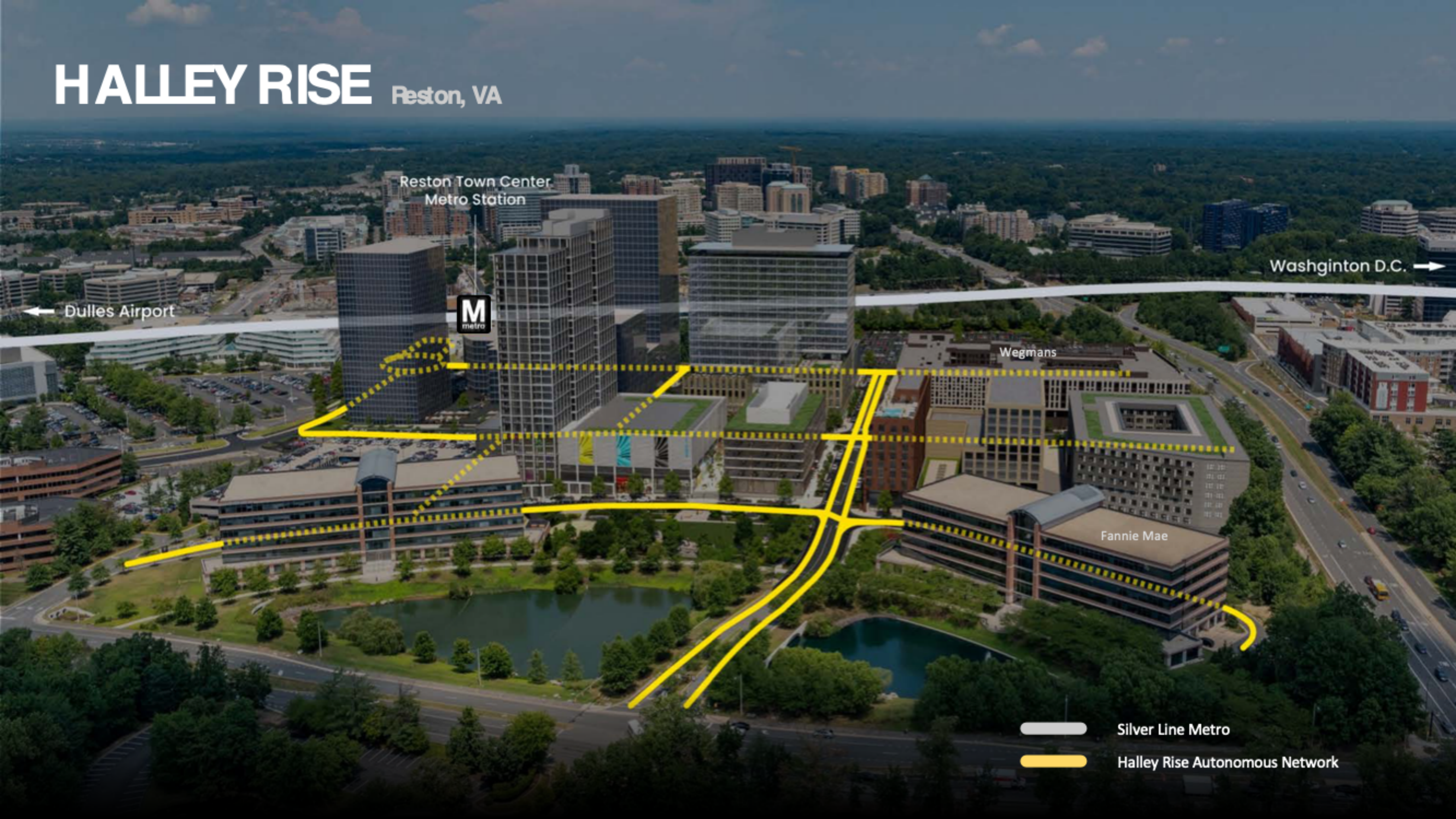


Low Speed
Roads



Moderate
Weather

HALLEY RISE Reston, VA



← Dulles Airport

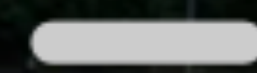
Reston Town Center
Metro Station



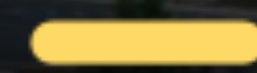
Washington D.C. →

Wegmans

Fannie Mae



Silver Line Metro



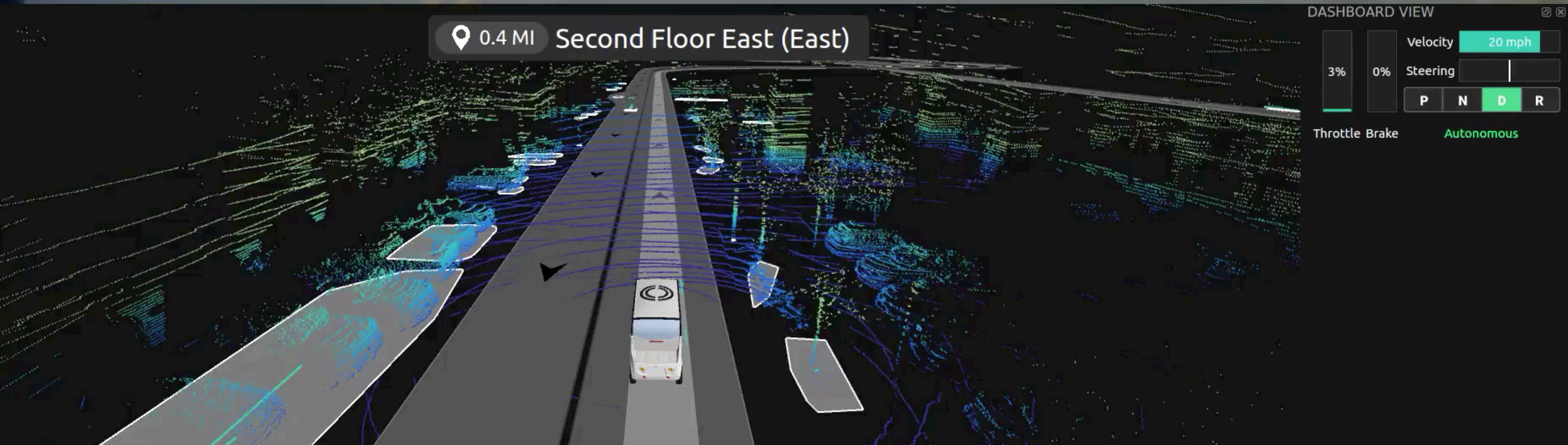
Halley Rise Autonomous Network



Brooklyn Navy Yard



Brooklyn Navy Yard



AI Deployment at Optimus Ride: Hardware Considerations

- **Computing hardware resources:**

- Edge computing on the vehicles
- Servers on premises
- Cloud computing

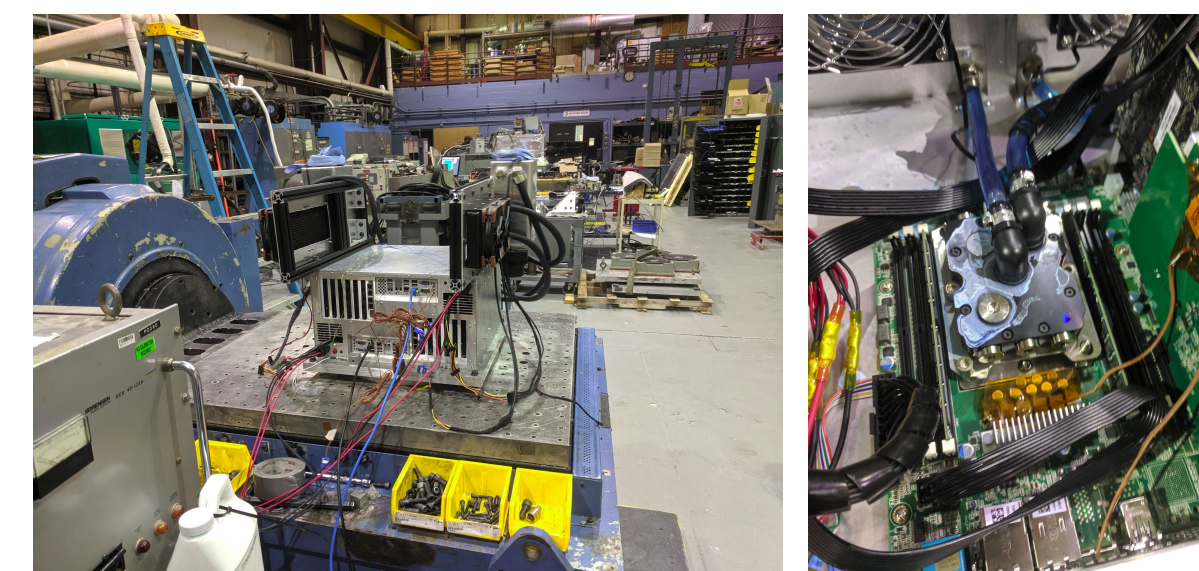
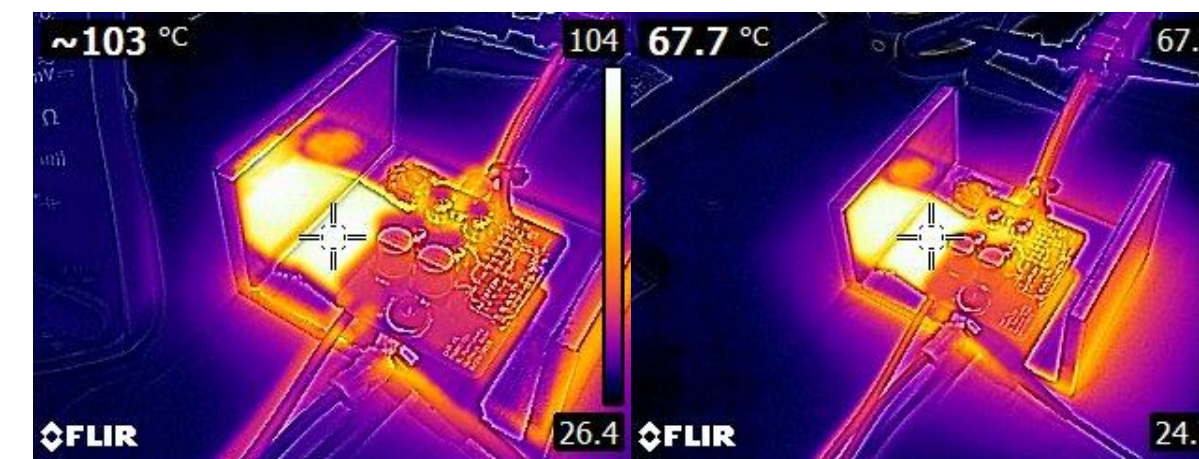
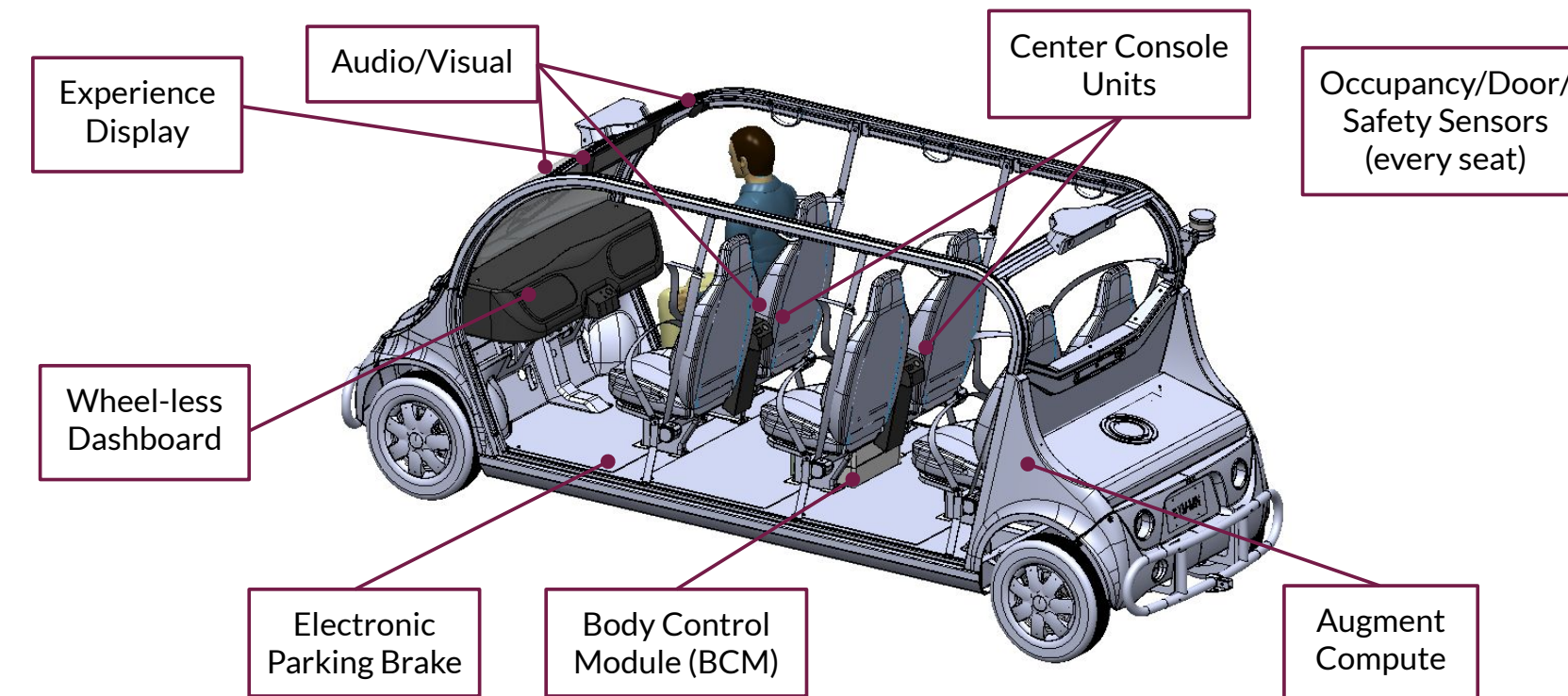
- **Relevant resources:**

- Communication is broadly available at high rates. But, may not be reliable.
- Can augment infrastructure in cost effective, not-too-invasive ways.

- **Requirement:** Vehicles must have a reasonable level of autonomy, should work safely even if the communication is completely cutoff.

- **ML pieces:** Trains on cloud; updates on premises with that days data; Runs on the edge computer!

- **The edge computer** implementation is optimized to produce predictions for many outputs of interest, sharing resources.



Edge computing



Servers on premises included 10s of powerful computers for AI training

AI Deployment at Optimus Ride: Human involvement

- **The entire product is a human-in-the-loop AI system, which tons of machine learning sprinkled in!**
- Humans monitor the system at different scales:
 - **Fleet monitors** have view of an area (much like air traffic controllers)
 - **Vehicles monitors** connect to different vehicles throughout the day
- The key business case is to have:
number of operators << number of vehicles

(Unlike popular belief, the products are not “zero-operators” fully autonomous. No one is doing that in “driverless cars”...)



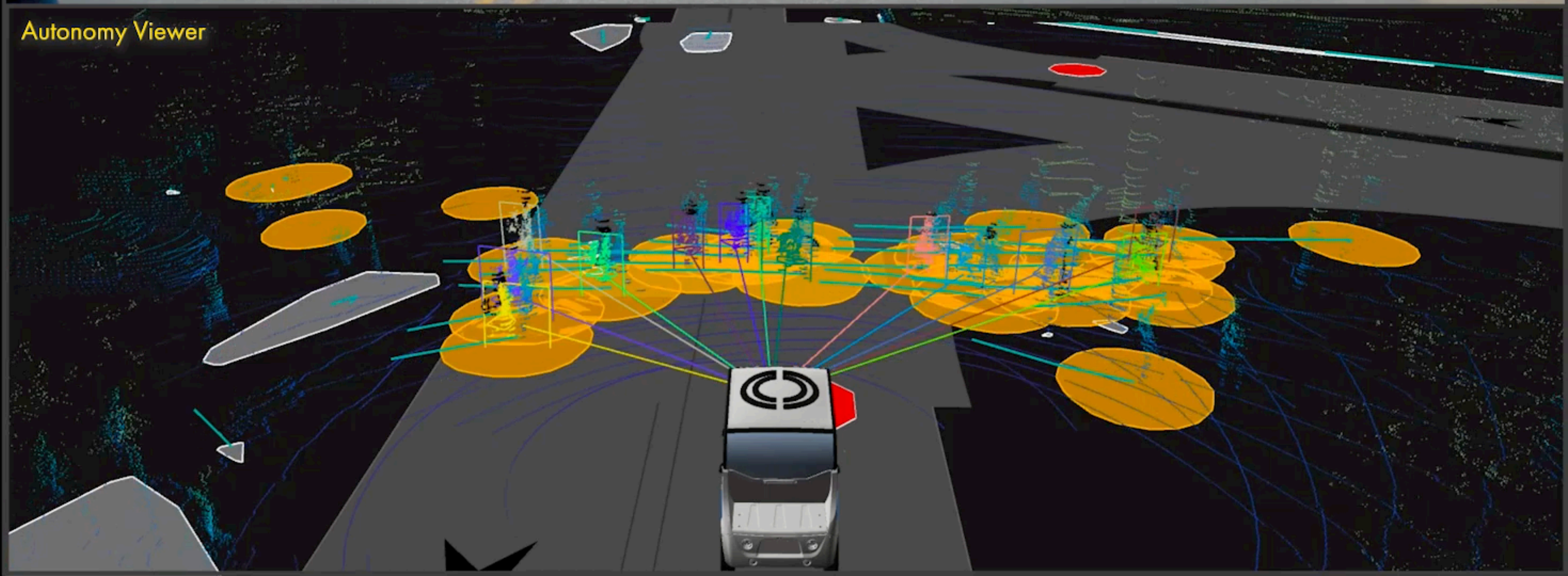
Fleet View



Vehicle View

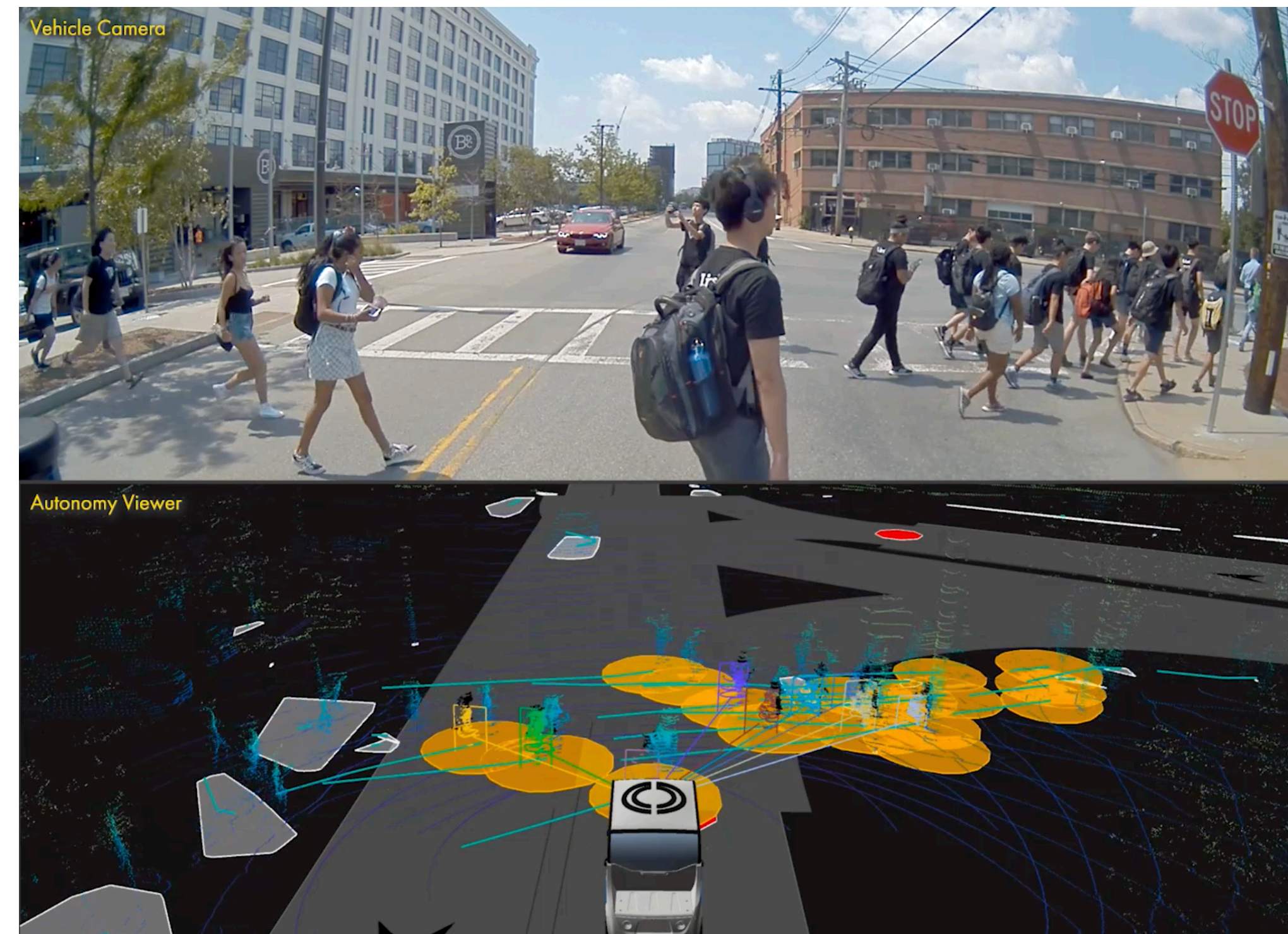
OPTIMUS RIDE TECHNOLOGY

PEDESTRIAN
SEGEMENTATION



The Impact of Machine Learning

- ***(Supervised) Machine Learning rapidly took over in 2015 and matured by 2020***
 - Many tasks could not be done without ML
 - Bigger models trained with more data seemed to do a lot better.




Pedestrian detection and tracking

The Impact of Machine Learning

- ***Unsupervised Large-scale ML arrived in ~2022 and will likely mature by 2027+***
 - Models seem to understand most edge cases.
 - The value seems to increase with bigger models (trained on more data, requiring more compute resources)

You



how should i drive in this scene?

ChatGPT

In the scene depicted, where there is a car on fire on the side of the road, it's important to prioritize safety for yourself and others. Here's how you should drive:

1. **Slow Down:** Reduce your speed significantly to ensure you have full control of your vehicle and to respond to any unforeseen circumstances.
2. **Keep Distance:** Make sure to keep a safe distance from the burning vehicle. There could be a risk of explosion or the fire could spread quickly.
3. **Do Not Stop:** Unless you are involved in the incident or you have the ability to assist safely, do not stop near the scene. It could create additional hazards or block emergency services.
4. **Move Over:** If it is safe and you are on a multi-lane road, move over to a lane further away from the fire to give maximum space for emergency vehicles and personnel.
5. **Be Alert:** Pay attention to the environment and any possible instructions from emergency services or road signs that might be directing traffic due to the incident.
6. **Do Not Disturb:** Avoid the temptation to take pictures or videos. This can distract you and others from driving safely.
7. **Follow Directions:** If there are emergency responders or traffic control in place, follow their directions immediately.
8. **Report the Incident:** If it appears that emergency services are not yet on the scene, you can report the incident to the authorities once you have passed the scene safely and can stop in a secure location.

Remember, your safety and the safety of others are the most important considerations in an emergency situation like this.

The Impact of Machine Learning

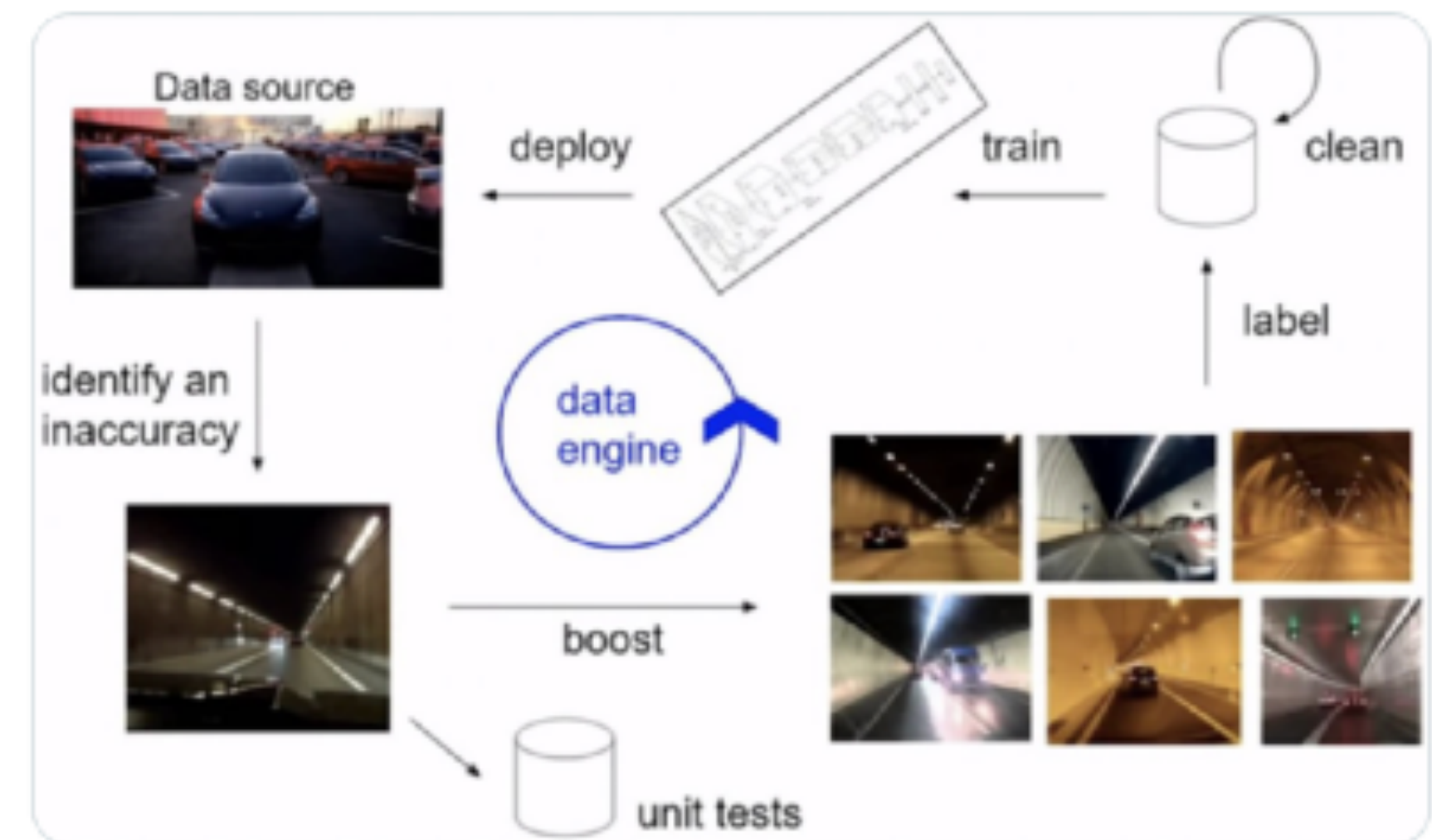
- Managing data pipelines seem critical
- ***The ultimate goal:*** Develop a fully-automated pipeline that leads to better capabilities with more driving



Andrej Karpathy
@karpathy



Potentially nitpicky but competitive advantage in AI goes not so much to those with data but those with a data engine: iterated data acquisition, re-training, evaluation, deployment, telemetry. And whoever can spin it fastest. Slide from Tesla to ~illustrate but concept is general



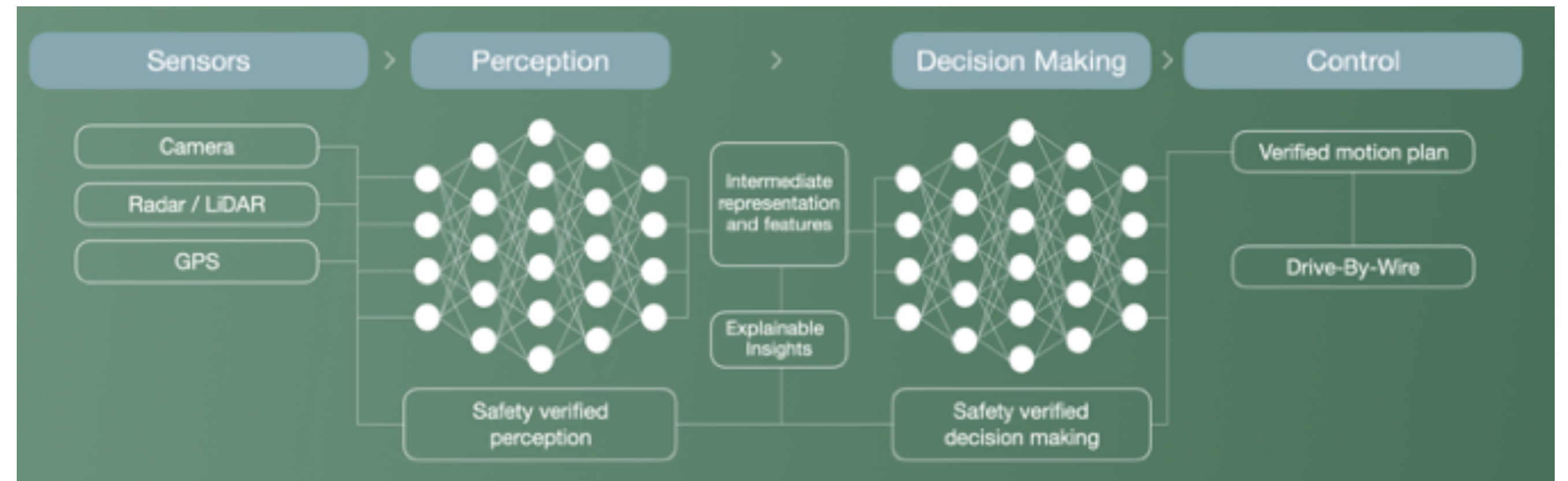
2:47 PM · Dec 5, 2022

The Impact of Machine Learning

- There is big excitement around “end-to-end” ML systems.



- However, this excitement may turn out to not fit transportation, even though the impact of ML is clearly here to stay.



Data & Compute: The Fuel of AI

Scaling Laws!

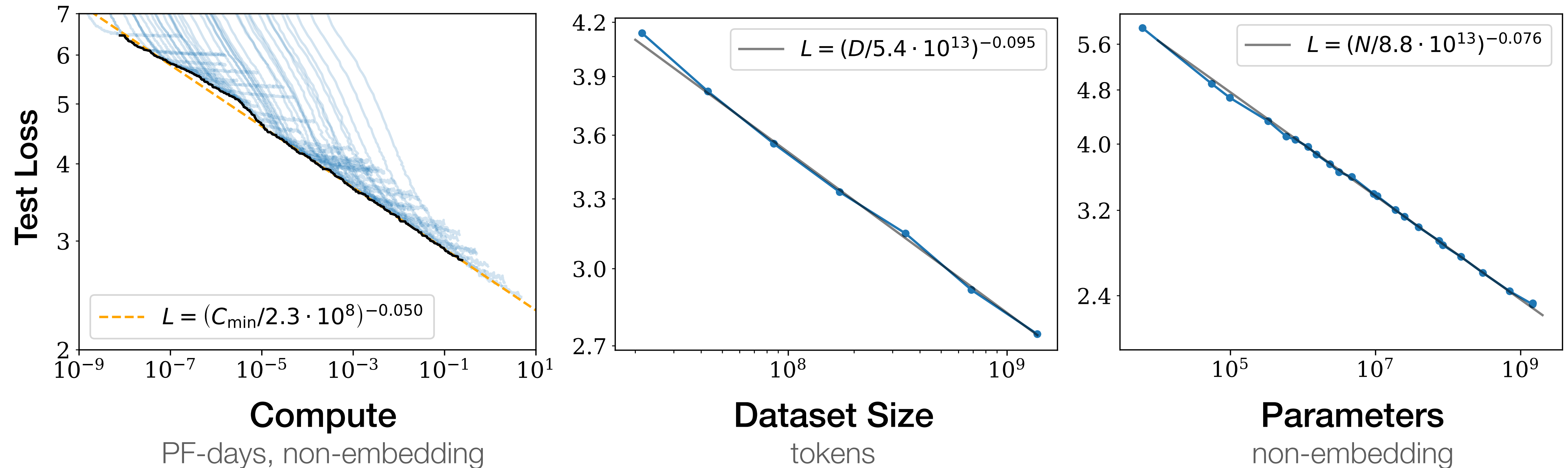


Figure 1 Language modeling performance improves smoothly as we increase the model size, dataset size, and amount of compute² used for training. For optimal performance all three factors must be scaled up in tandem. Empirical performance has a power-law relationship with each individual factor when not bottlenecked by the other two.

*From an early 2020 OpenAI paper

Data & Compute: The Fuel of AI

Emergence of Domain-specific Intelligence with Scale?

- Complex “reasoning” behavior emerges rapidly as the scale increases

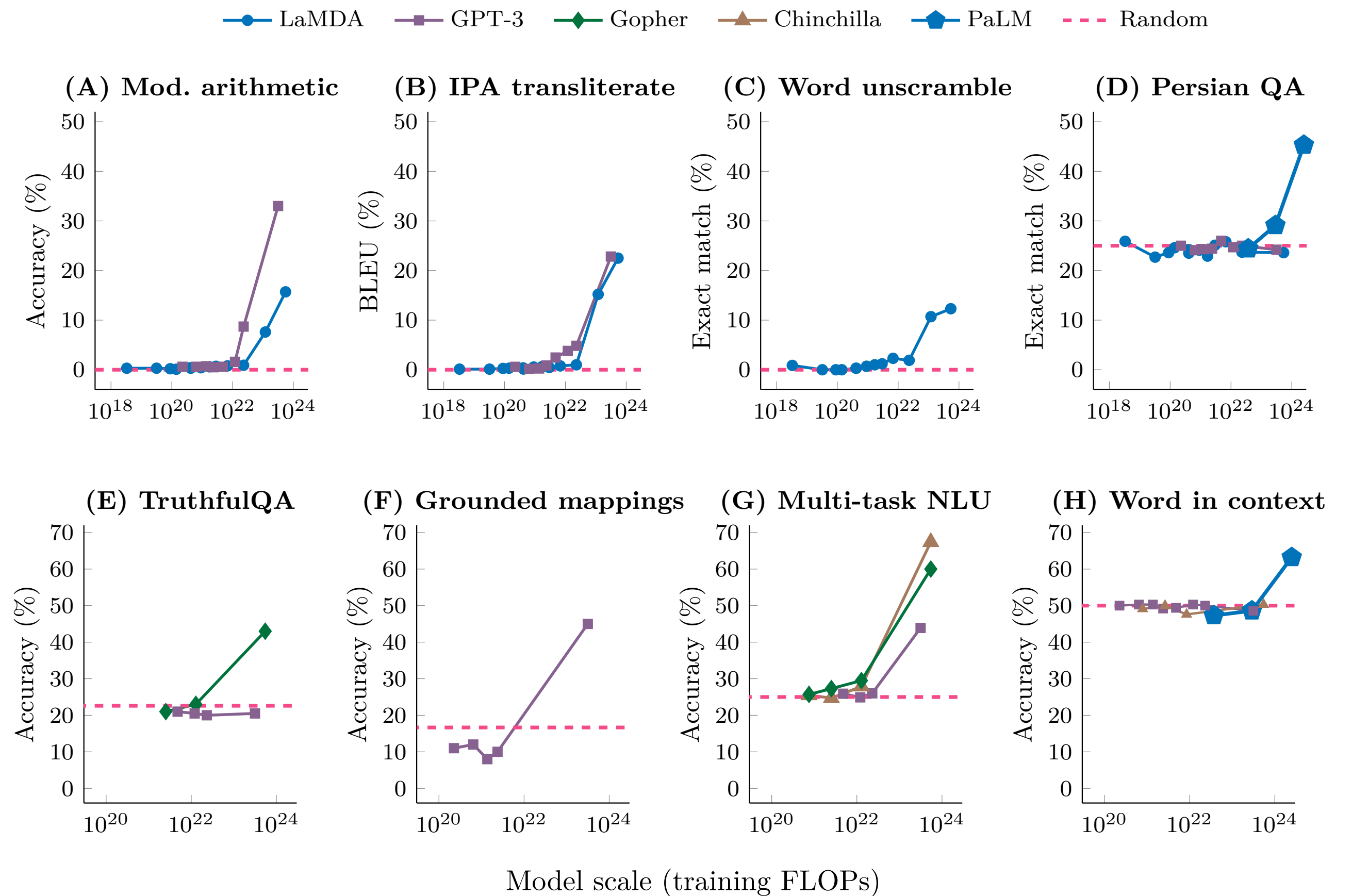


Figure 2: Eight examples of emergence in the few-shot prompting setting. Each point is a separate model. The ability to perform a task via few-shot prompting is emergent when a language model achieves random performance until a certain scale, after which performance significantly increases to well-above random. ...

**From an October 2022 Google Research paper*

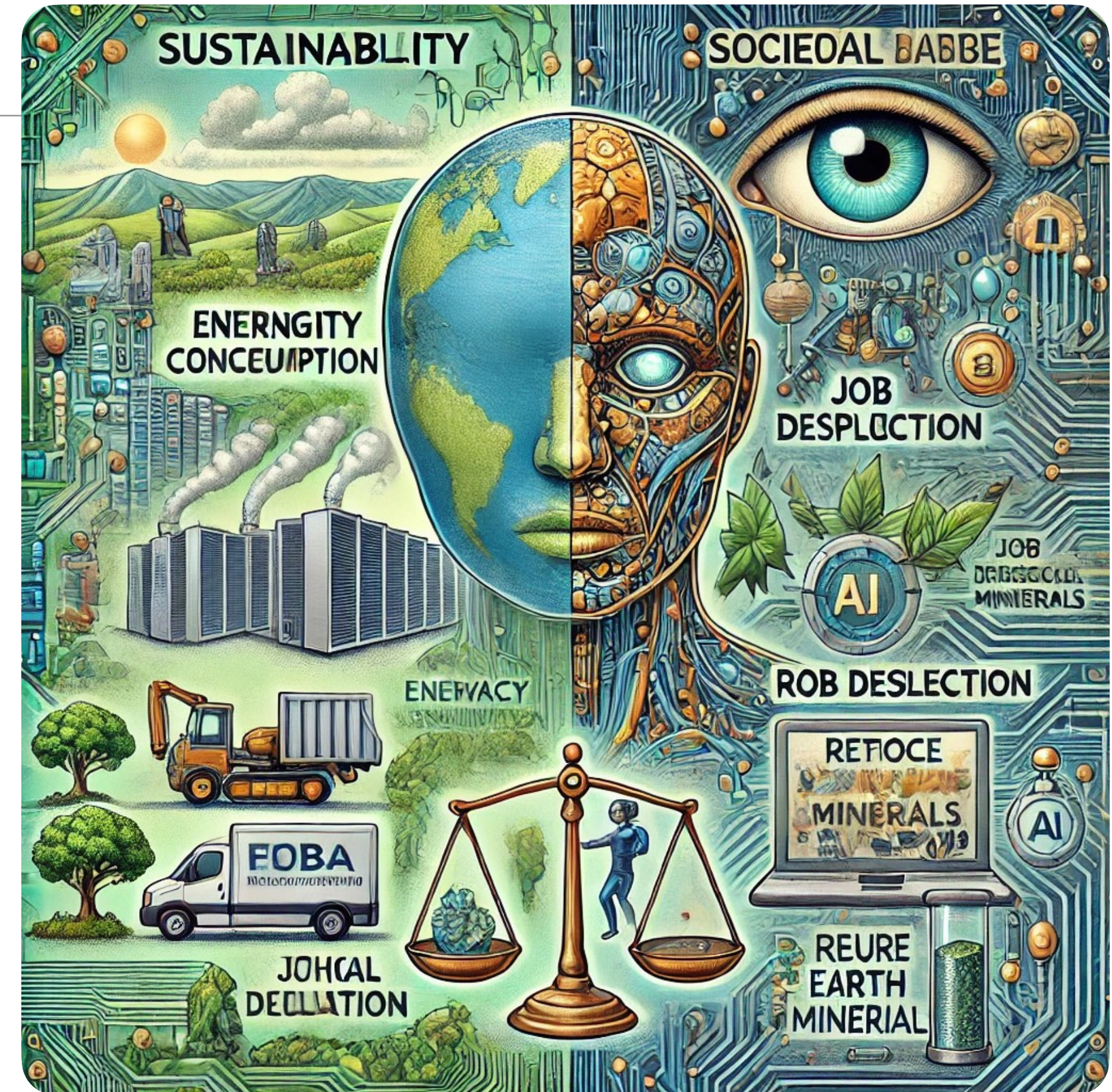
What are some of the challenges?

- **Societal Challenges**

- Job transition, training
- Privacy and security
- Biases, safety and ethics
- ...

- **Sustainability Challenges**

- Energy consumption
- Built environments
- ...



ChatGPT: Here is the illustration of AI's societal and sustainability challenges.

What is Exciting?

- **AI enables new “simulations”**
 - Video generation
 - Social/behavioral simulations
 - Physics/chemistry/biology simulations
 - ...
- **Autonomous AI and Decision Making**
 - Self-driving cars, drones, ...
 - Hiring, HR, ...
 - ...



ChatGPT: Here is the illustration depicting AI becoming more autonomous.