Deploying Deep Learning for Driving

Alex Kendall @ CVPR, Long Beach, June 2019

WAYVE


20
WX16 FDY

WAYVE

Outline of talk

1. Recipe for success for representation learning
2. Understanding what we don't know
3. Strategy for training data
4. Interpretability & verification

Machine Learning for Autonomous Driving

Some Historical Background

Alex Kendall © Wayve 2019

1989 ALVINN: End-to-End Imitation Learning

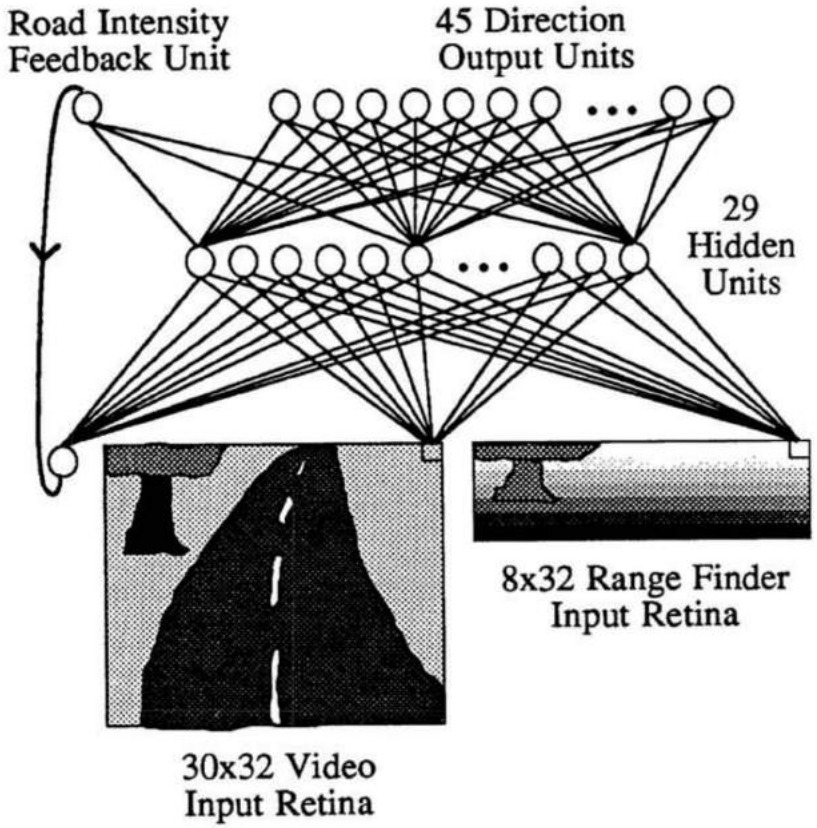


Figure 1: ALVINN Architecture

What's Hidden in the Hidden Layers?

The contents can be easy to find with a geometrical problem, but the hidden layers have yet to give up all their secrets

David S. Touretzky and Dean A. Pomerleau

tions, we fed the network road images taken under a wide variety of viewing angles and lighting conditions. It would be impractical to try to collect thousands of real road images for such a data set. Instead, we developed a synthetic road-image generator that can create as many training examples as we need.

To train the network, 1200 simulated road images are presented 40 times each, while the weights are adjusted using the back-propagation learning algorithm. This takes about 30 minutes on Carnegie Mellon's Warp systolic-array supercomputer. (This machine was designed at Carnegie Mellon and is built by General Electric. It has a peak rate of 100 million floating-point operations per second and can compute weight adjustments for back-propagation networks at a rate of 20 million connections per second.)

Once it is trained, ALVINN can accurately drive the NAVLAB vehicle at about 3½ miles per hour along a path through a wooded area adjoining the Carnegie Mellon campus, under a variety of weather and lighting conditions. This speed is nearly twice as fast as that achieved by non-neural-network algorithms running on the same vehicle. Part of the reason for this is that the forward pass of a back-propagation network can be computed quickly. It takes about 200 milliseconds on the Sun-3/160 workstation installed on the NAVLAB.

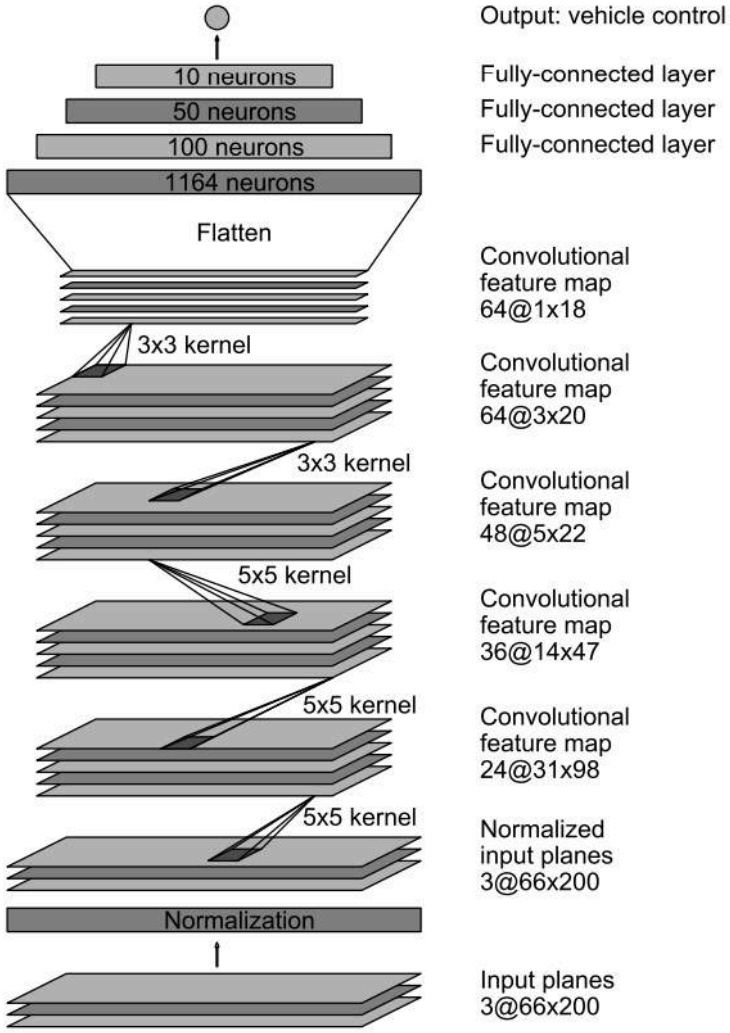
The hidden-layer representations ALVINN develops are interesting. When trained on roads of a fixed width, the network chooses a representation in which hidden units act as detectors for complete roads at various positions and orientations. When trained on roads of variable

continued

Photo 1: *The NAVLAB autonomous navigation test-bed vehicle and the road used for trial runs.*

Pomerleau, Dean A. "ALVINN: An autonomous land vehicle in a neural network." Advances in neural information processing systems. 1989.

2016 NVIDIA: Lane Following on Highways



Bojarski, Mariusz, et al. "End to end learning for self-driving cars." arXiv preprint arXiv:1604.07316 (2016).

Urban driving with end-to-end machine learning

Inputs: camera video and a sat-nav

End-to-end deep learning

Uncertainty propagation from *sensing* to *action*

Outputs: driving commands

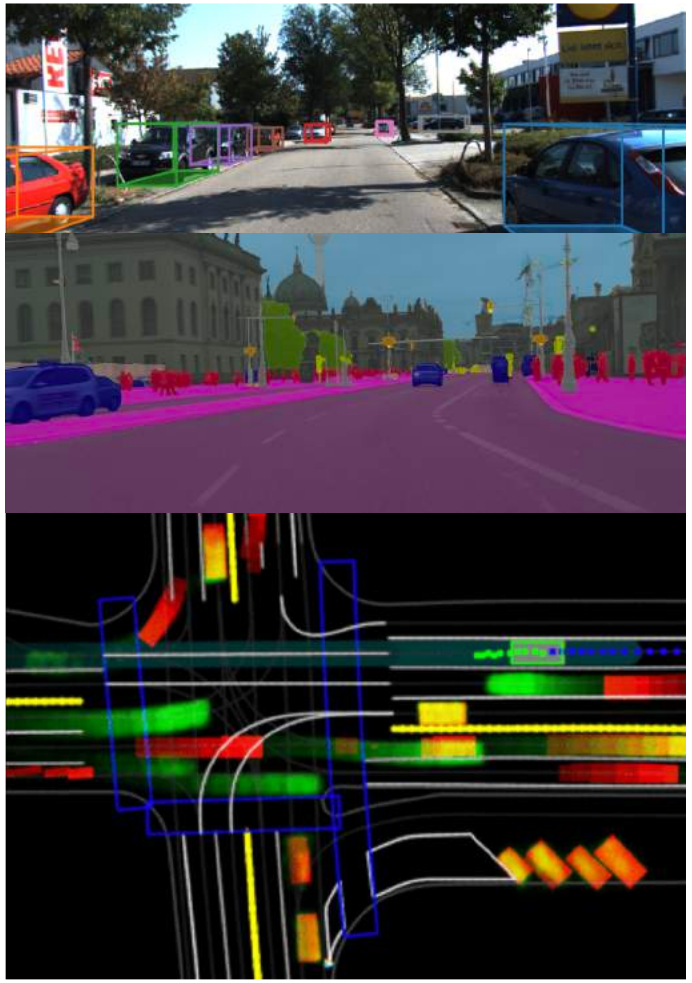
WAYVE

Alex Kendall et al. Learning to Drive in a Day. ICRA, 2019
Wayve, https://wayve.ai/blog/driving-like-human (2019)

A recipe for representing driving

Alex Kendall © Wayve 2019

The Self-Driving State Representation Today



3D Object Detection

Semantic Segmentation

Agent Prediction

Turning indicator detector

HD Map

Driving Affordability Prediction

Traffic sign detection

.....

Autonomous Driving Representation

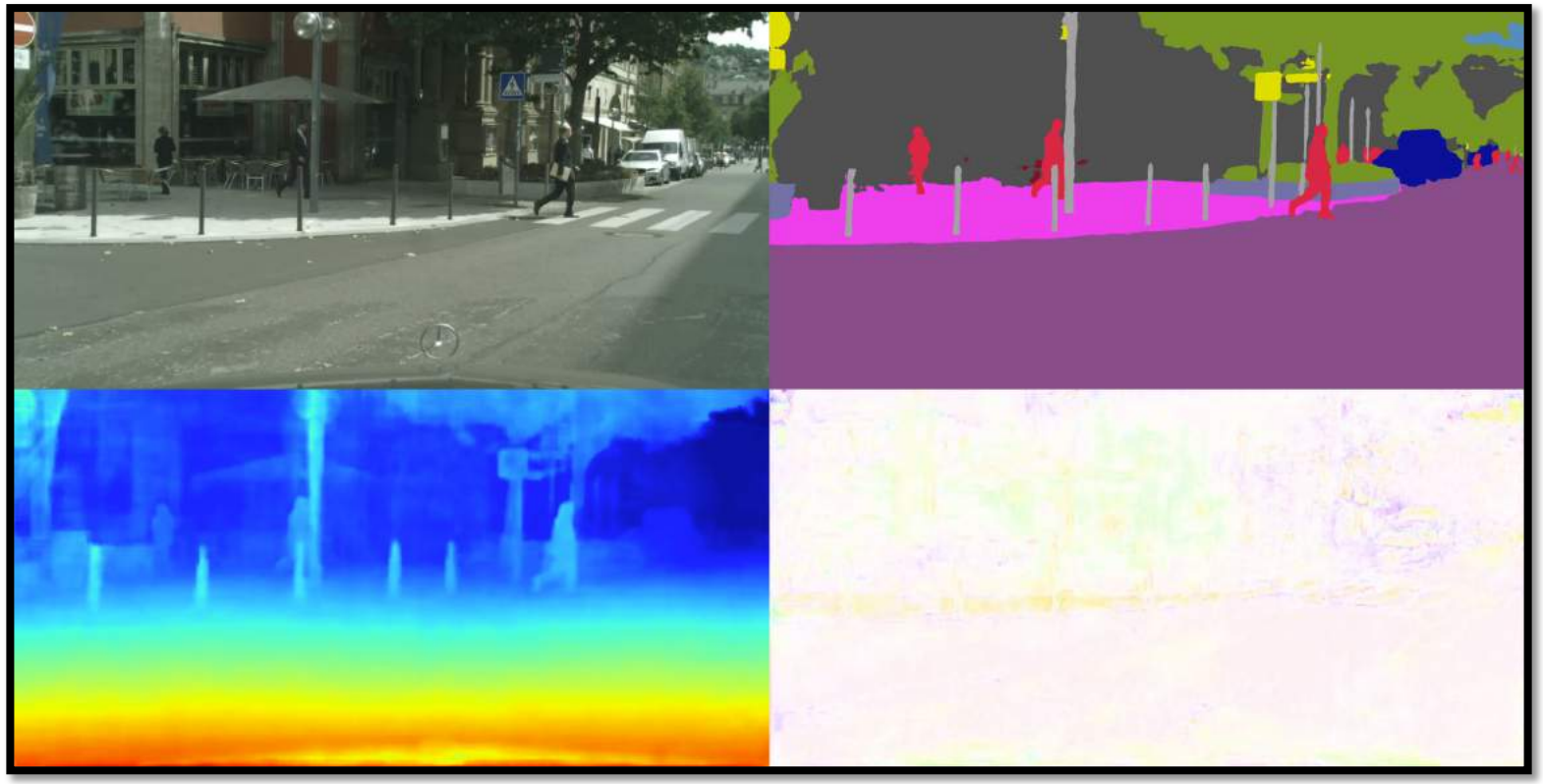
A recipe for a good representation

1. Needs to encode information that we believe is necessary (but not sufficient) for the task
   - For driving, this includes semantics, motion and geometry

2. Should also be optimised w.r.t. the end task
   - Therefore we need an end to end learning signal

3. The decision must be observable in the input data
   - We need the right sensor type and configuration

4. Our representation must have a very good signal to noise ratio
   - We must transform the signal into a compressed, nuisance free & invariant representation

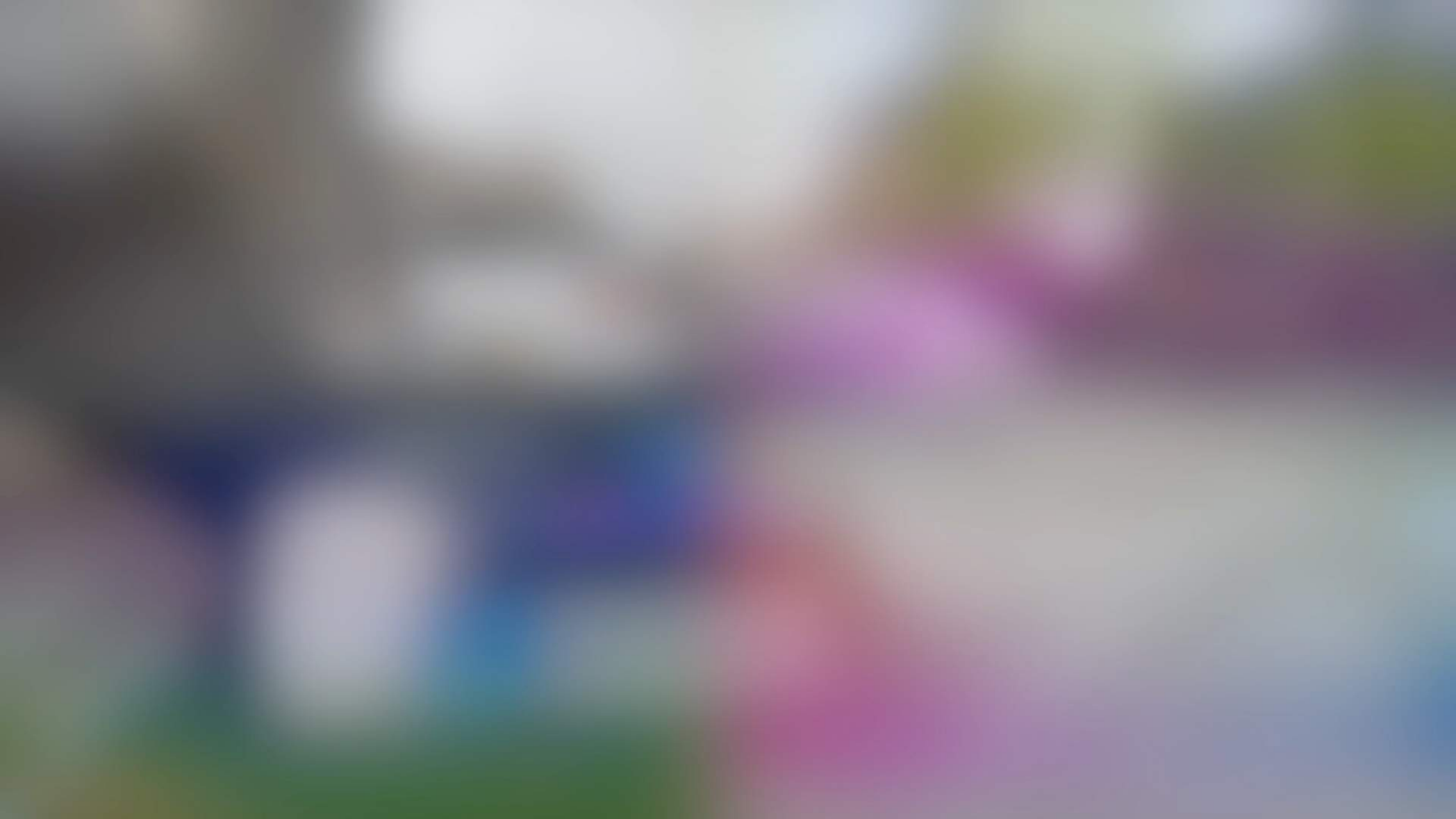


Progression of computer vision from

2015

… to 2018

Badrinarayanan, Kendall, Cipolla. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. PAMI, 2017.
Kendall, Gal and Cipolla. Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. CVPR, 2018.



Modelling Uncertainty

Understanding what we don't know

Alex Kendall © Wayve 2019

What kind of uncertainty can we model?

Epistemic uncertainty

- Measures what you're model doesn't know
- Can be explained away by unlimited data

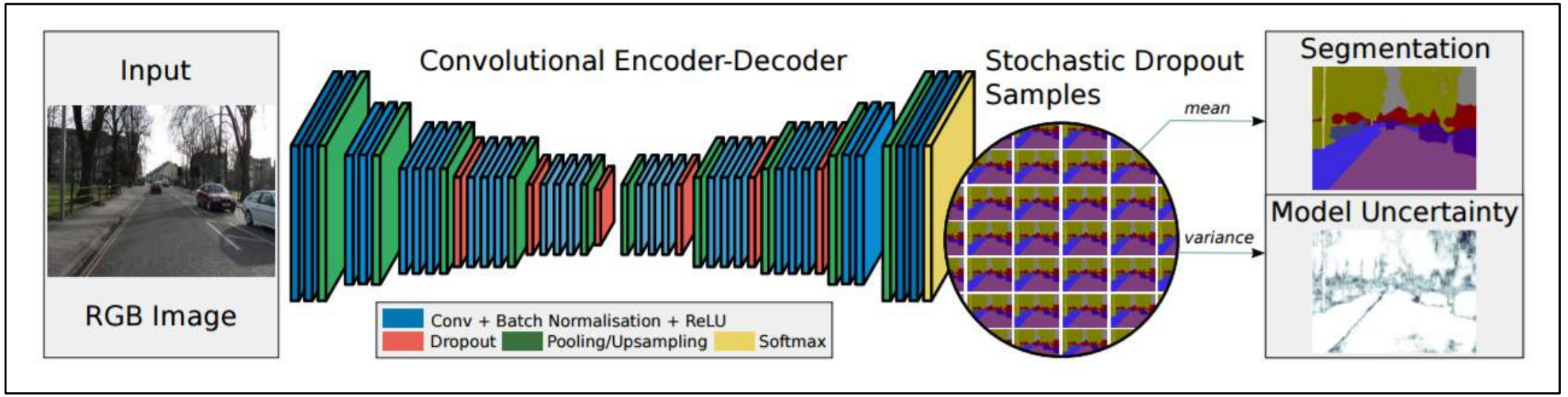
Aleatoric uncertainty

- Measures what you can't understand from the data
- Can be explained away by unlimited sensing

Der Kiureghian, A. and Ditlevsen, O. (2009). Aleatory or epistemic? does it matter? Structural Safety, 31(2):105–112.

Modeling Epistemic Uncertainty with Bayesian Deep Learning

- We can model epistemic uncertainty in deep learning models using Monte Carlo dropout sampling at test time.
- Dropout sampling can be interpreted as sampling from a distribution over models.



Alex Kendall and Yarin Gal. **What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?** NeurIPS, 2017.

Aleatoric Uncertainty with Probabilistic Deep Learning

	Deterministic Deep Learning	Probabilistic Deep Learning
Model	$[\hat{y}\,] = f(x)$	$[\hat{y}, \hat{\sigma}^2] = f(x)$
Regression	$Loss = \|y - \hat{y}\|^2$	$Loss = \frac{\|y - \hat{y}\|^2}{2\hat{\sigma}^2} + \log \hat{\sigma}$
Classification	$Loss = SoftmaxCrossEntropy(\hat{y}_t)$	$\hat{y}_t = \hat{y} + \epsilon_t \quad \epsilon_t \sim N(0, \hat{\sigma}^2)$ $Loss = \frac{1}{T}\sum_t SoftmaxCrossEntropy(\hat{y}_t)$

Alex Kendall and Yarin Gal. **What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?** NeurIPS, 2017.

Train/Test Distribution Shift

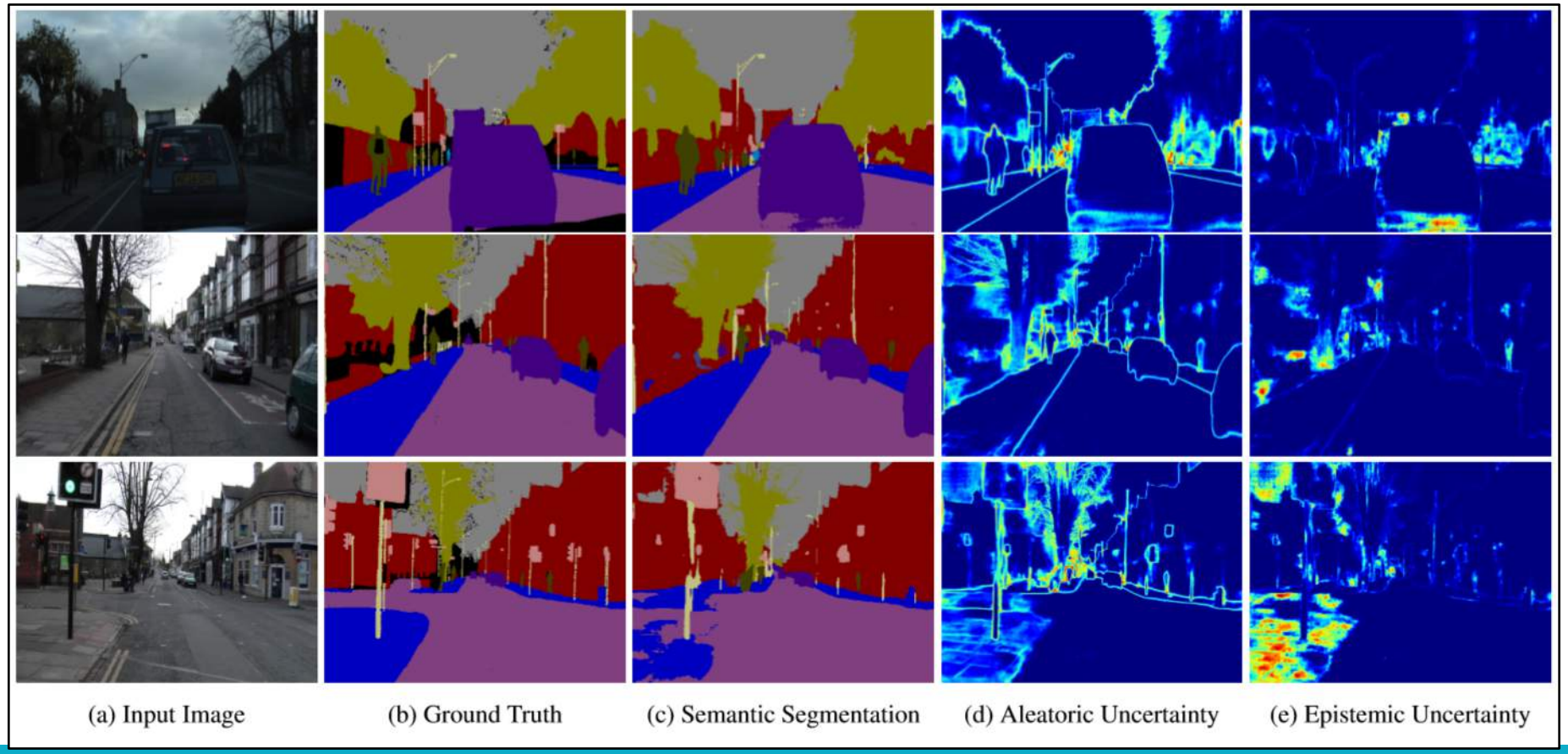
- Aleatoric uncertainty remains constant while epistemic uncertainty increases for out of dataset examples!

Train dataset	Test dataset	RMS	Aleatoric variance	Epistemic variance
Make3D / 4	Make3D	5.76	0.506	7.73
Make3D / 2	Make3D	4.62	0.521	4.38
Make3D	Make3D	3.87	0.485	2.78
Make3D / 4	NYUv2	-	0.388	15.0
Make3D	NYUv2	-	0.461	4.87

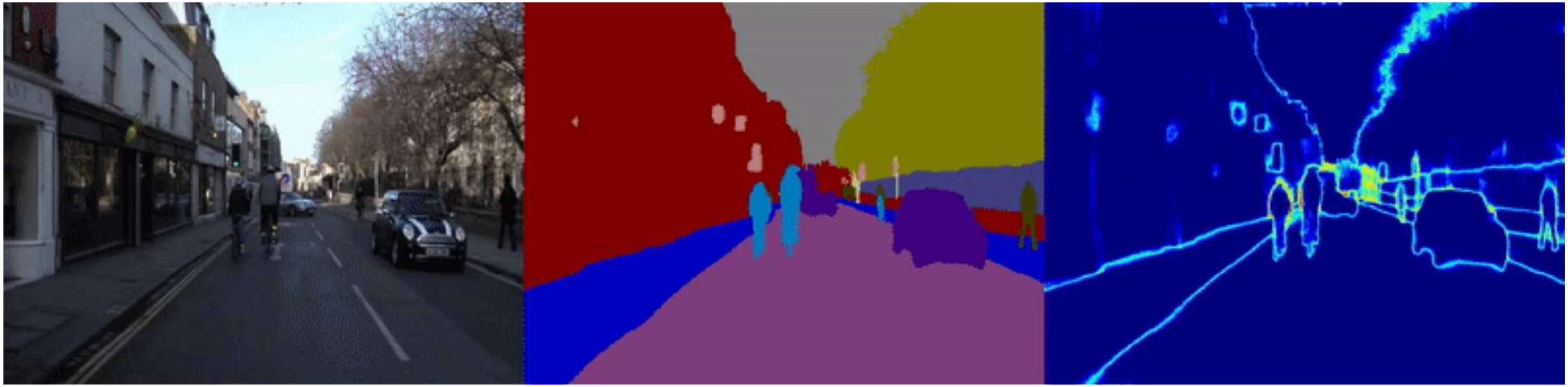
Alex Kendall and Yarin Gal. **What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?** NeurIPS, 2017.

Qualitative comparison

- Epistemic uncertainty is modeling uncertainty
- Aleatoric uncertainty is sensing uncertainty



Bayesian Deep Learning for Segmentation



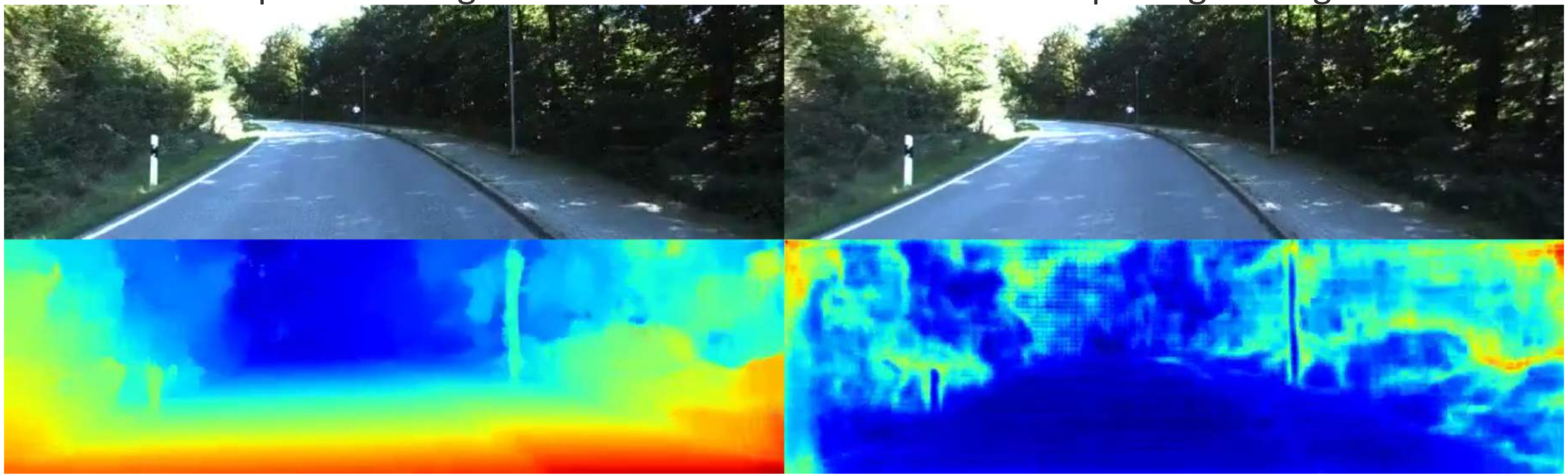
Input Image | Semantic Segmentation | Uncertainty

Alex Kendall et al. Bayesian SegNet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. BMVC 2017

Bayesian Deep Learning for Stereo Vision

Input Left Image

Input Right Image



Depth Prediction

Depth Prediction Uncertainty

Alex Kendall et al. **End-to-End Learning of Geometry and Context for Deep Stereo Regression.** ICCV, 2017.

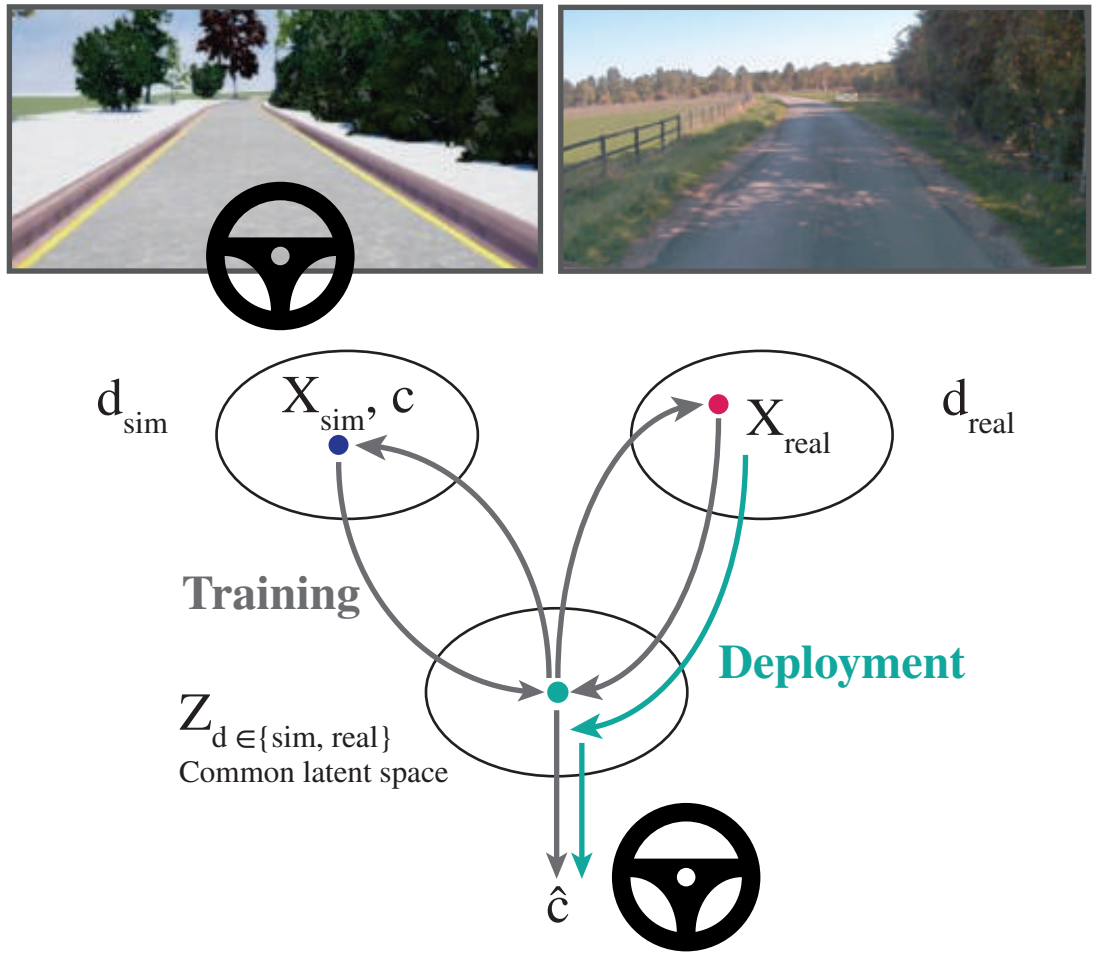
Training Data

How much and what type do we need?

Alex Kendall © Wayve 2019

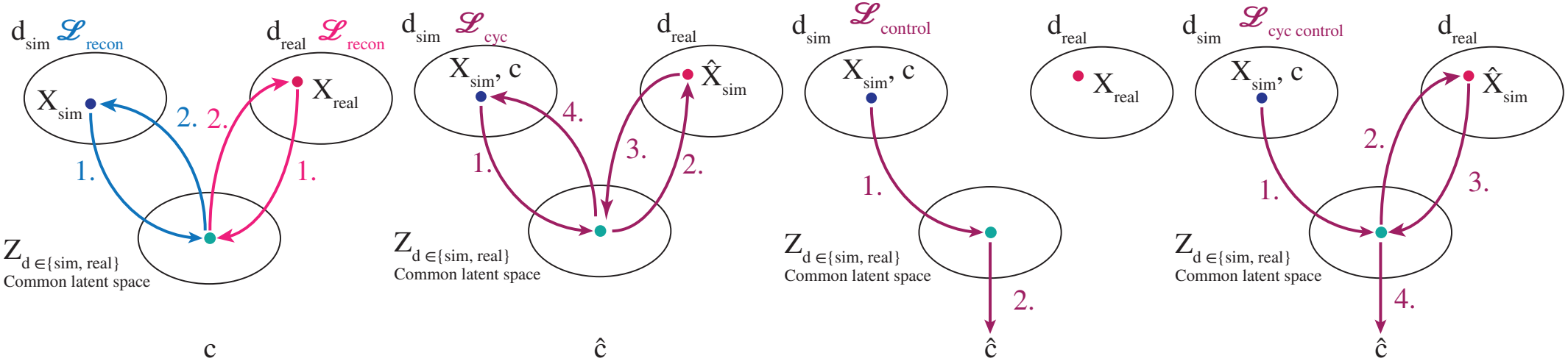
Can we train real-world models in simulated worlds?

- Zero shot sim2real
- Learn to project to a latent space for domain translation and control jointly
- Demonstrate this method can drive 3km+ on public UK roads



Alex Bewley et al. Learning to Drive from Simulation without Real World Labels. ICRA, 2019.

Learning to Drive from Simulation without Real World Labels



(a) Reconstruction loss (b) Cyclic recon. loss (c) Control loss (d) Cyclic control loss

Reconstruction Loss

$$X_d^{recon} = G_d(E_d(X_d))$$

Cyclic Reconstruction Loss

$$X_d^{cyc} = G_d(E_{d'}(G_{d'}(E_d(X_d))))$$

Control Loss

$$\hat{c} = C(E_d(X_d))$$

Cyclic Control Loss

$$\hat{c}^{cyc} = C(E_{d'}(G_{d'}(E_d(X_d))))$$

Not shown: adversarial LSGAN loss, latent reconstruction loss, perceptual loss.

Alex Bewley et al. Learning to Drive from Simulation without Real World Labels. ICRA, 2019.

Comparison to Baseline Methods

	Simulation		Real		
	MAE	Bal-MAE	MAE	Bal-MAE	DPI (metres)
Drive-Straight	0.043	0.087	**0.019**	0.093	$23^{\dagger}$
Simple Transfer	0.055	0.056	0.265	0.272	$9^{\dagger}$
Real-to-Sim Translation	-	-	0.261	0.234	$10^{\dagger}$
Sim-to-Real Translation	-	-	0.059	**0.045**	$28^{\dagger}$
Latent Feature ADA [3]	0.040	0.047	0.032	0.071	$15^{\dagger}$
Ours	**0.017**	**0.018**	0.081	0.087	$>$**3000**

Alex Bewley et al. Learning to Drive from Simulation without Real World Labels. ICRA, 2019.


B&B
Building Contractors
GB

Interpreting & Understanding Deep Learning Representations

Alex Kendall © Wayve 2019

Model-Based Saliency

Suppose $f(\cdot)$ is our driving model and $m(\cdot)$ is our saliency model and $L(\cdot)$ is our loss function for the driving model and the operator $x \cdot m$ degrades the image with noise.

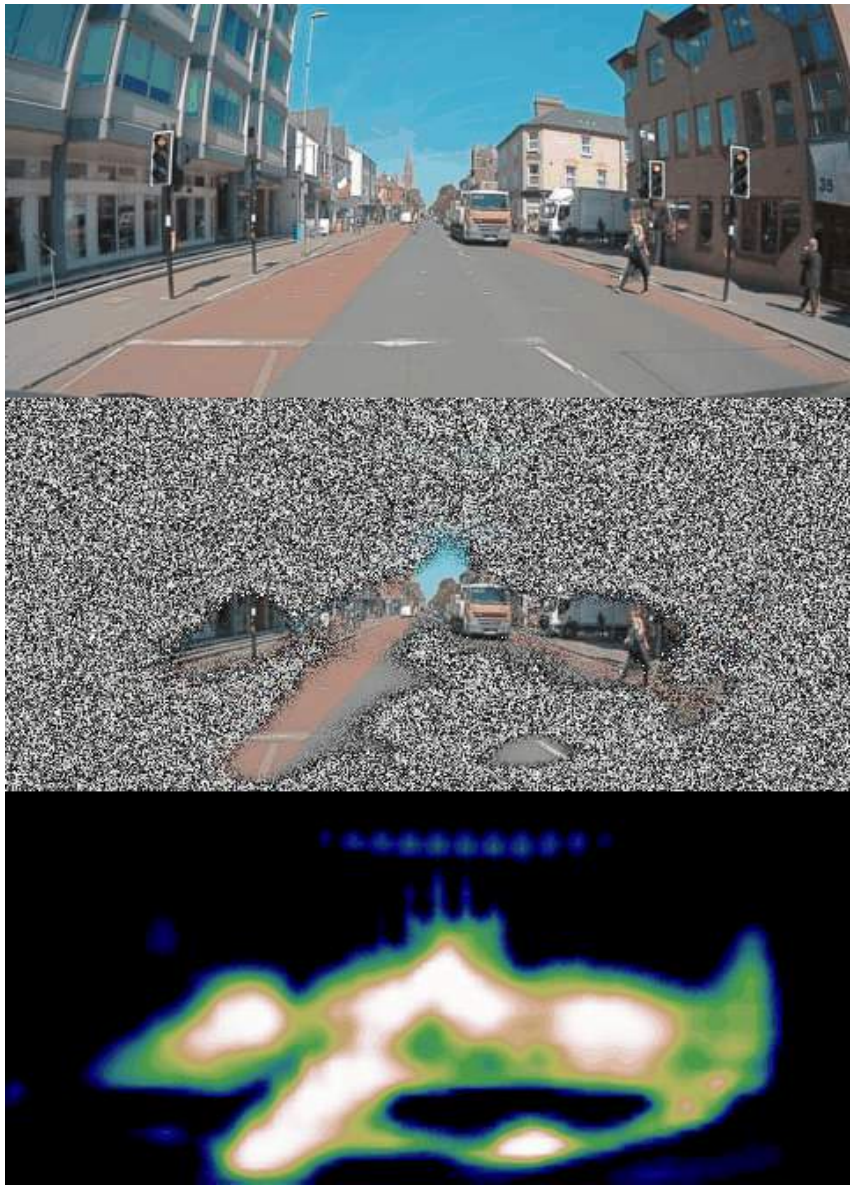
$$L = \lambda_1 |m(x)| + \lambda_2 |\nabla m(x)| + \lambda_3 L_0\left(f\left(x \cdot m(x)\right)\right) + \lambda_4 L_0\left(f\left(x \cdot \left(1 - m(x)\right)\right)\right)^{-\lambda_5}$$

Sparse saliency mask

Smooth saliency mask

Informative saliency mask

Uninformative inverse saliency mask



Dabkowski and Gal. "Real time image saliency for black box classifiers." NeurIPS. 2017.
Fong and Vedaldi. "Interpretable explanations of black boxes by meaningful perturbation." ICCV. 2017.

Model-Based Saliency

Suppose $f(\cdot)$ is our driving model and $m(\cdot)$ is our saliency model and $L(\cdot)$ is our loss function for the driving model and the operator $x \cdot m$ degrades the image with noise.

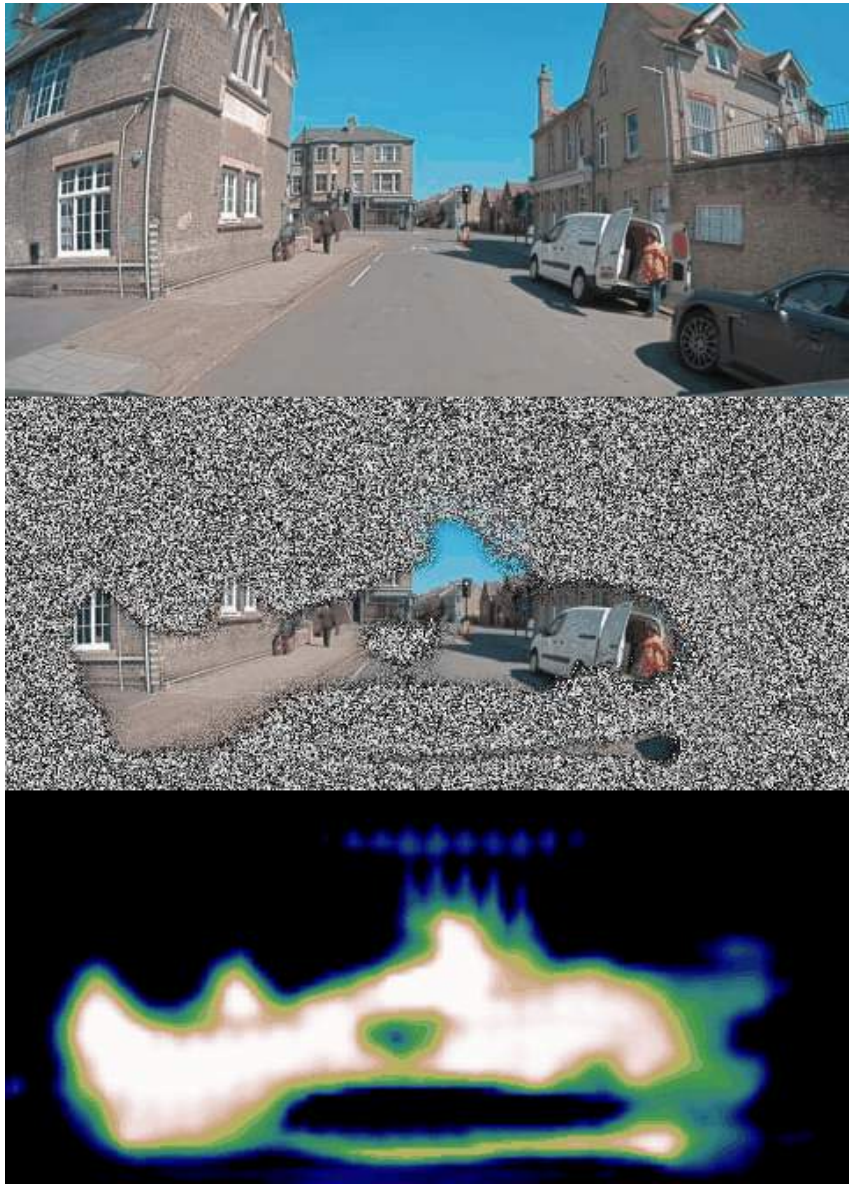
$$L = \lambda_1 |m(x)| + \lambda_2 |\nabla m(x)| + \lambda_3 L_0\left(f\left(x \cdot m(x)\right)\right) + \lambda_4 L_0\left(f\left(x \cdot \left(1 - m(x)\right)\right)\right)^{-\lambda_5}$$

Sparse saliency mask

Smooth saliency mask

Informative saliency mask

Uninformative inverse saliency mask



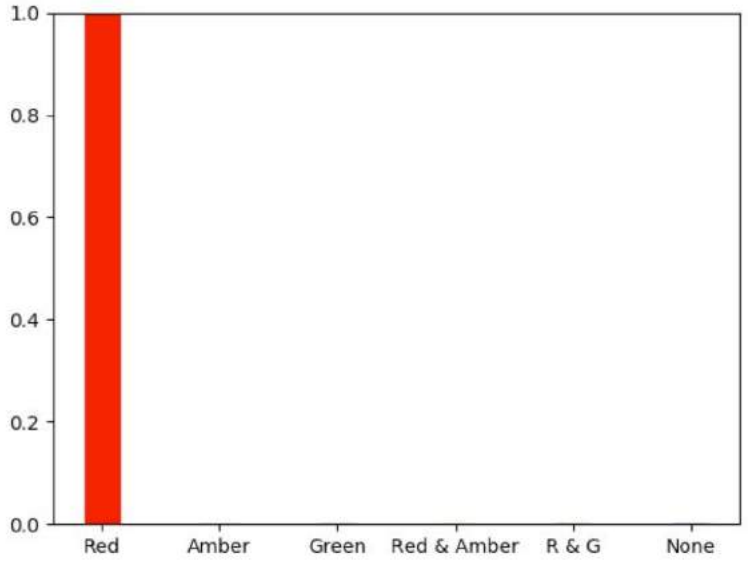
Dabkowski and Gal. "Real time image saliency for black box classifiers." NeurIPS. 2017.
Fong and Vedaldi. "Interpretable explanations of black boxes by meaningful perturbation." ICCV. 2017.

Inspecting the state for traffic light signal

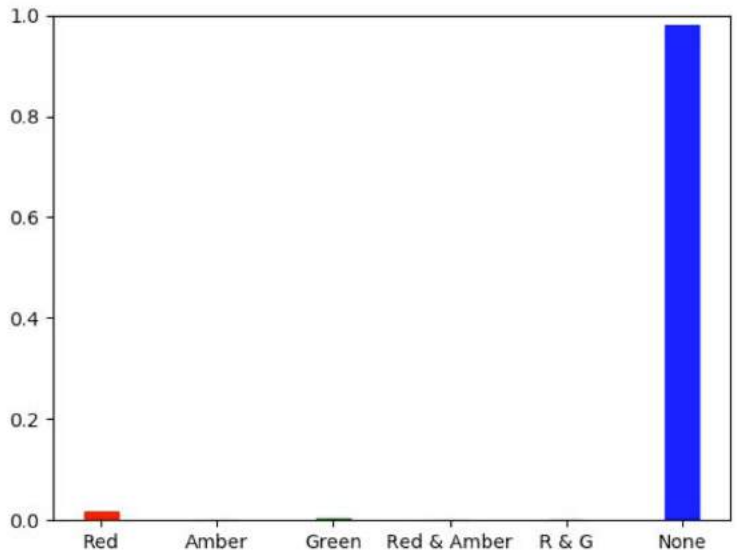
$x_{input} \rightarrow z \rightarrow y_{control}$

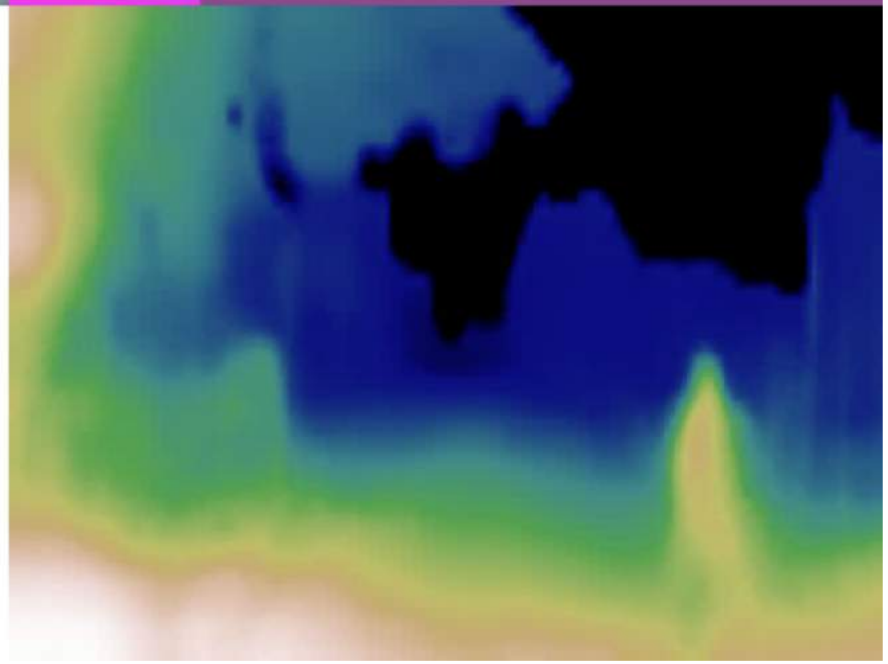
$z \downarrow y_{aux}$

Learn to decode the high dimensional state



Inspecting the state for traffic light signal, semantics and depth

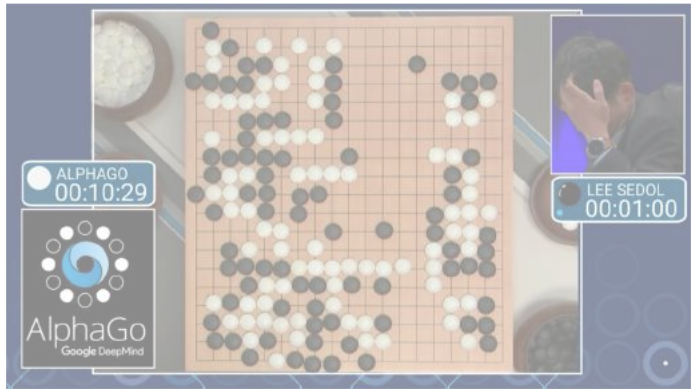




Conclusions

Alex Kendall © Wayve 2019

Games like Go & DOTA



DeepMind OpenAI

- Incredibly difficult action space: long term strategy, cooperation
- Very basic state space, often discrete, fully observable and noise-free

Autonomous Driving

- Quite easy action space: stop, go, left, right motion primitives
- Super challenging state space: manifold of natural images!

This needs to be solved by the computer vision community!

A complete paradigm shift for AVs

- Low vehicle compute and sensor requirements
- Large training compute and data requirements
- Increased vehicle intelligence
- No reliance on HD-maps
- Ability to leverage simulation for training
- Abundance of open and interesting research questions!

Come work with our team wayve.ai/careers

WAYVE