The Importance of Uncertainty for Deep Learning in Robotics

Niko Suenderhauf

Queensland University of Technology
Australian Centre for Robotic Vision

With material by **Dimity Miller** and **Feras Dayoub**

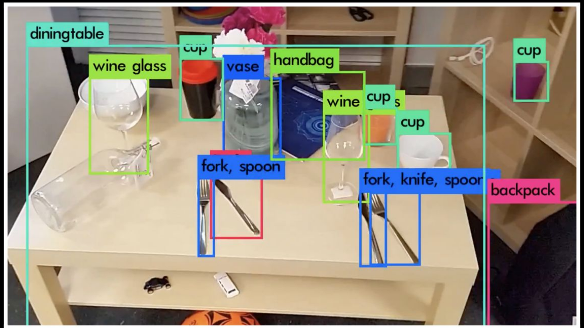
Output

Input

Input \ Output	Images	Data
Images	Image Processing	Computer Vision
Data	Computer Graphics	Data Science

Output

	Images	Data
Images	Image Processing	Computer Vision
Data	Computer Graphics	Data Science

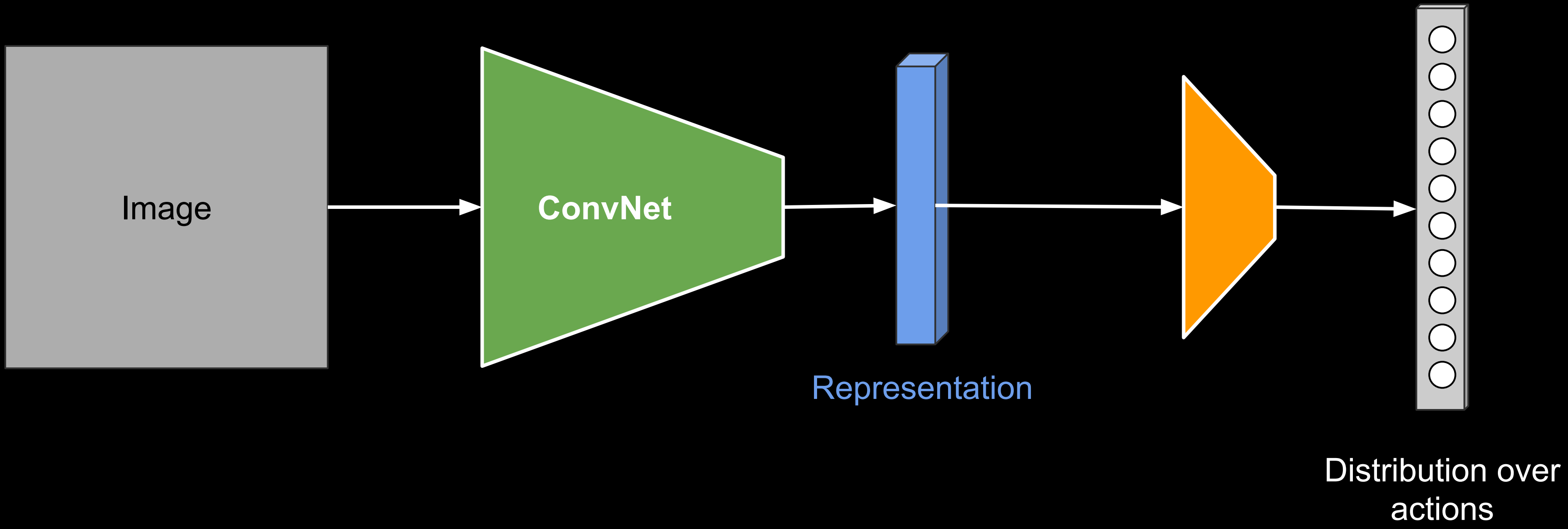


Input

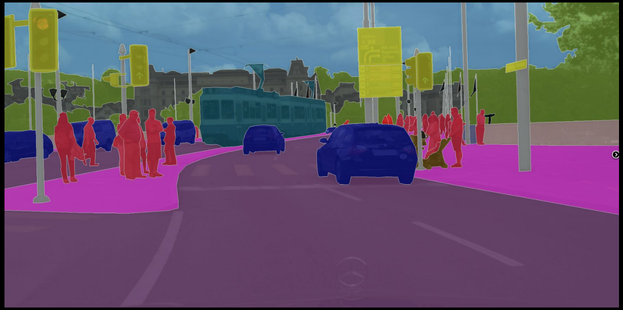
Output

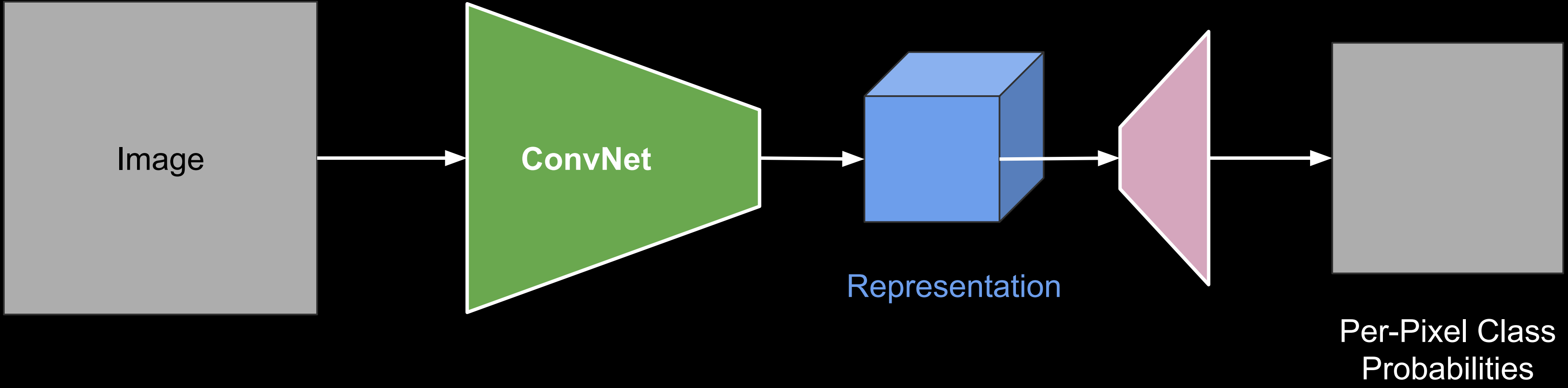
Input	Images	Data	**Actions**
Images	Image Processing	Computer Vision	Robotic Vision
Data	Computer Graphics	Data Science	

Reinforcement Learning

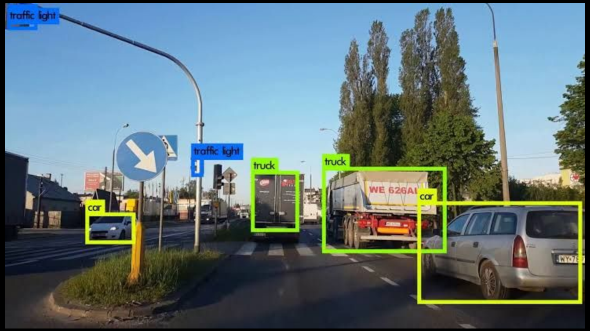


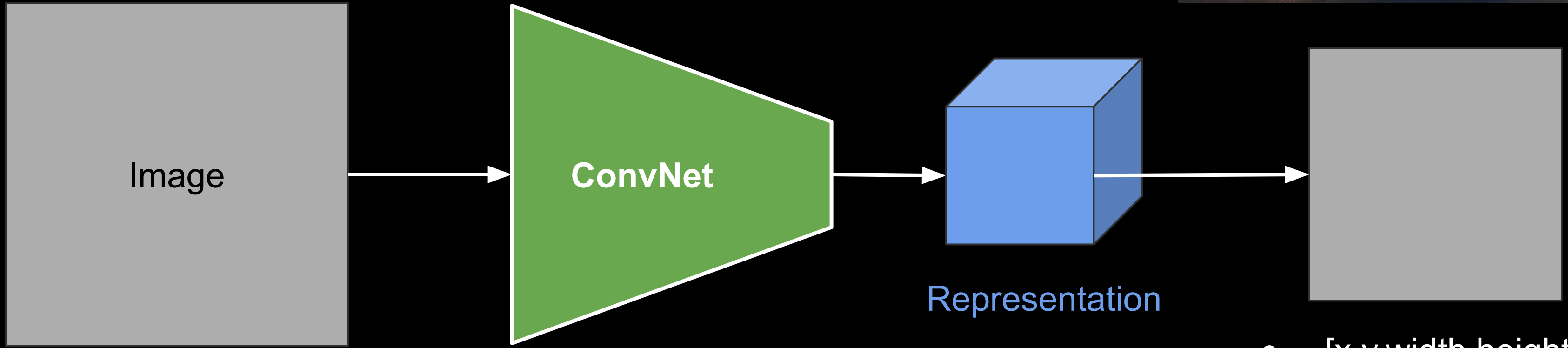
Semantic Segmentation



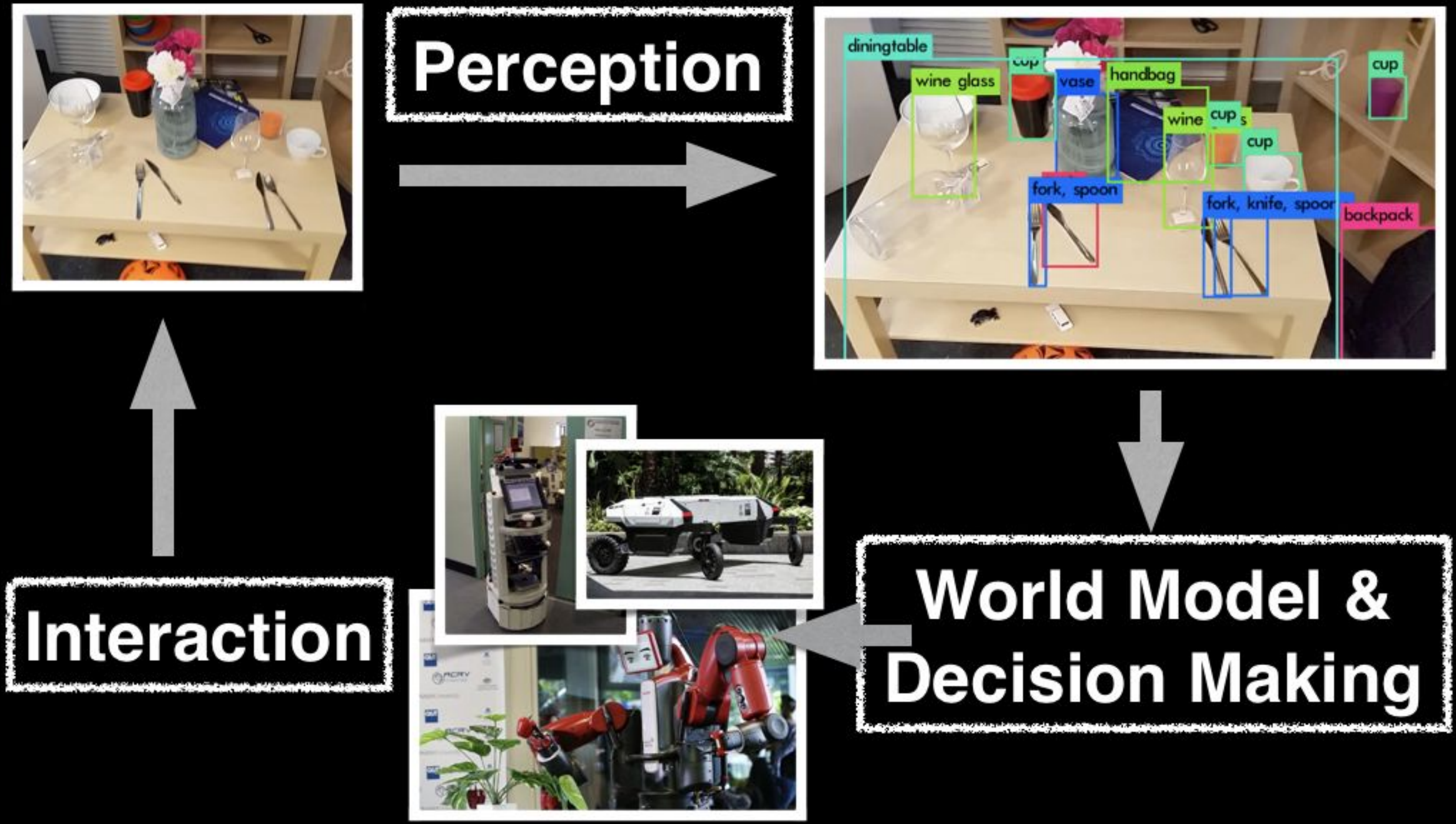


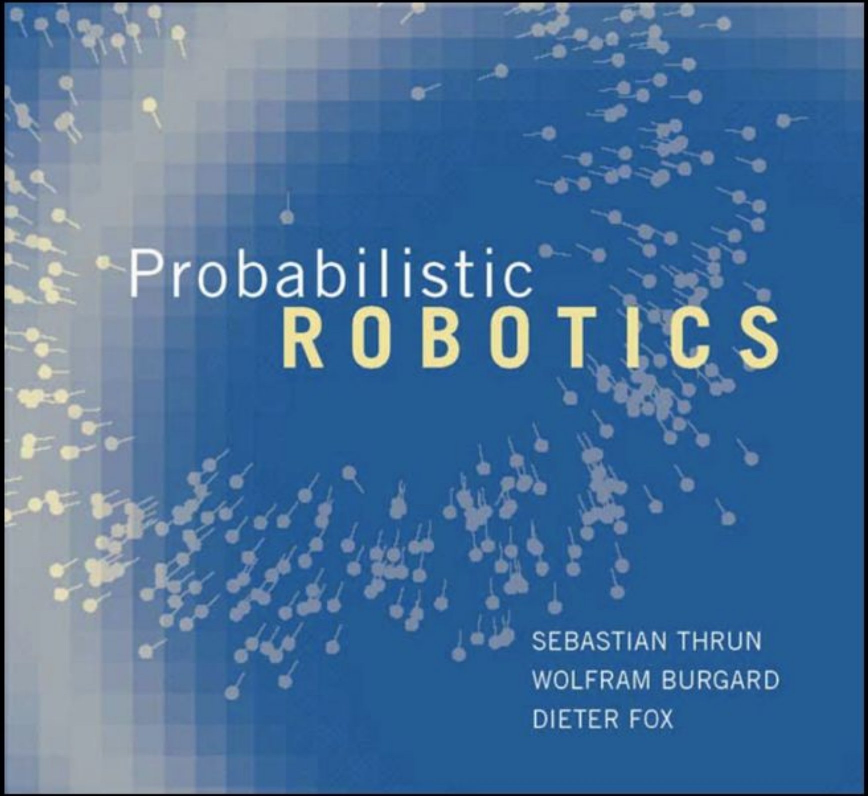
Object Detection





- [x,y,width,height]
- confidence
- class label


Perception
diningtable
wine glass
cup
vase
handbag
wine
cup
cup
cup
fork, spoon
backpack
World Model &
Decision Making
Interaction


Probabilistic
ROBOTICS
SEBASTIAN THRUN
WOLFRAM BURGARD
DIETER FOX

Aleatoric and Epistemic Uncertainty

Aleatoric Uncertainty

- Due to noise inherent in the observations
  - E.g. over-exposure, motion blur
- Can **not** be reduced with more data.
- From Latin “alea” = “dice”

Epistemic Uncertainty

- Due to lack of knowledge
- Can be reduced by more data.

Aleatoric Uncertainty

Aleatoric Uncertainty

$$\mathcal{L}_{\text{NN}}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{2\sigma(\mathbf{x}_i)^2}||\mathbf{y}_i - \mathbf{f}(\mathbf{x}_i)||^2 + \frac{1}{2}\log\sigma(\mathbf{x}_i)^2$$

Heteroscedastic Noise Term

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

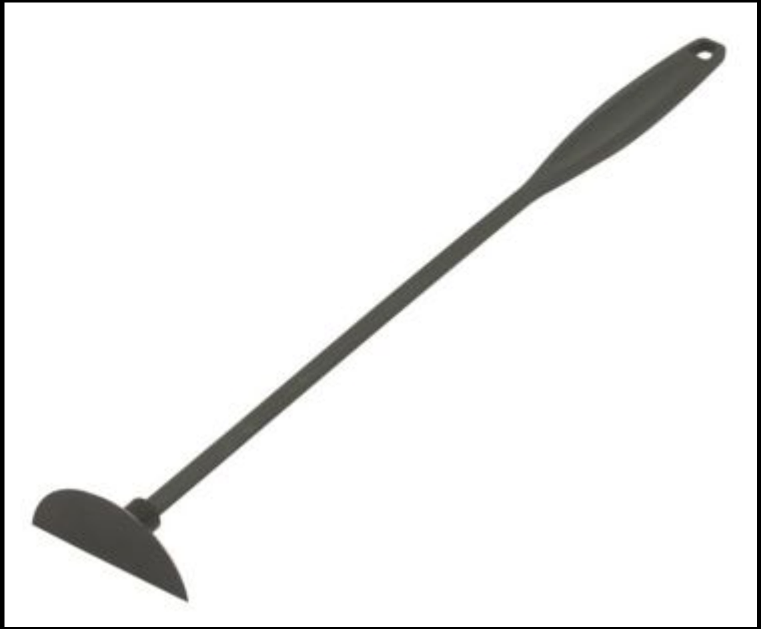
Epistemic Uncertainty

Epistemic Uncertainty

Okapi

Rambutan

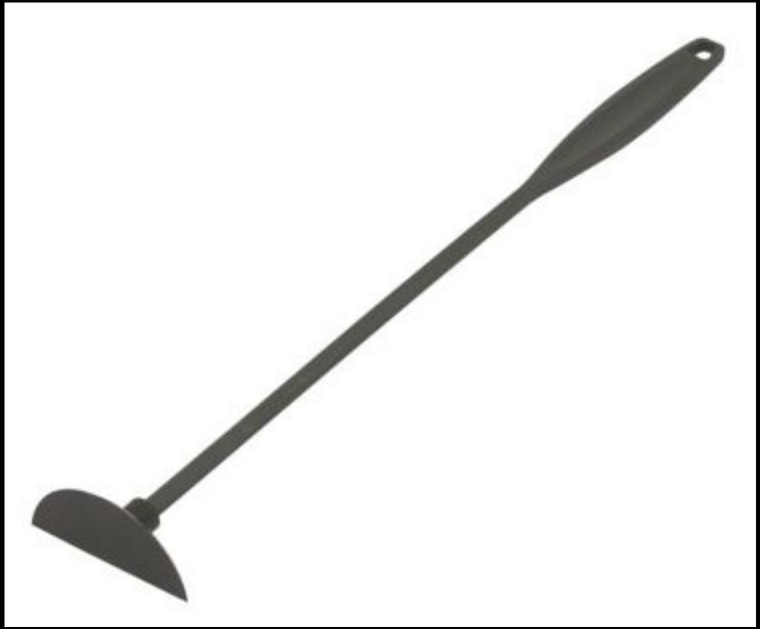
Epistemic Uncertainty



Epistemic Uncertainty

Flessenlikker

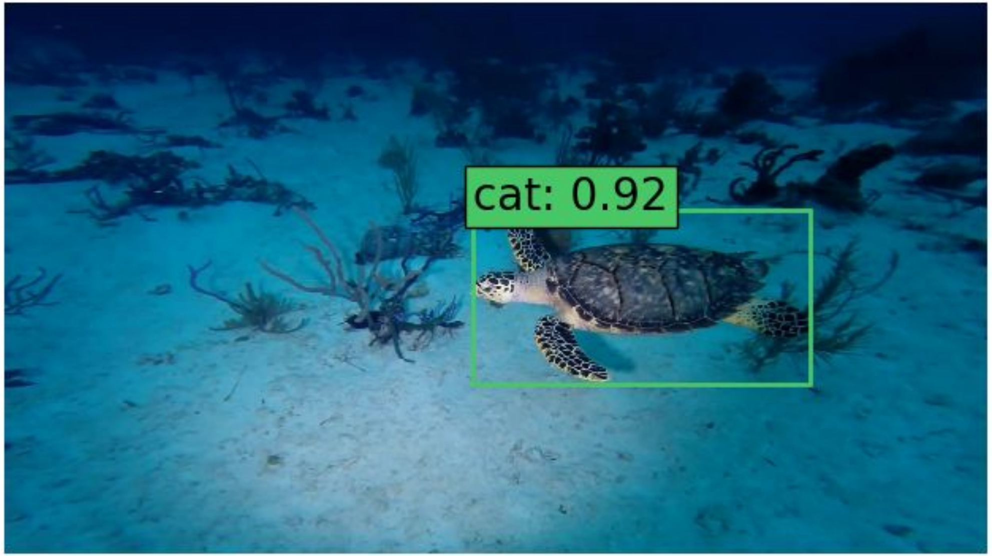
Epistemic Uncertainty



Flessenlikker


chair: 0.76
tv: 0.74
bench: 0.71
refrigerator: 0.86
refrigerator: 0.95


dog: 0.99


cat: 0.92

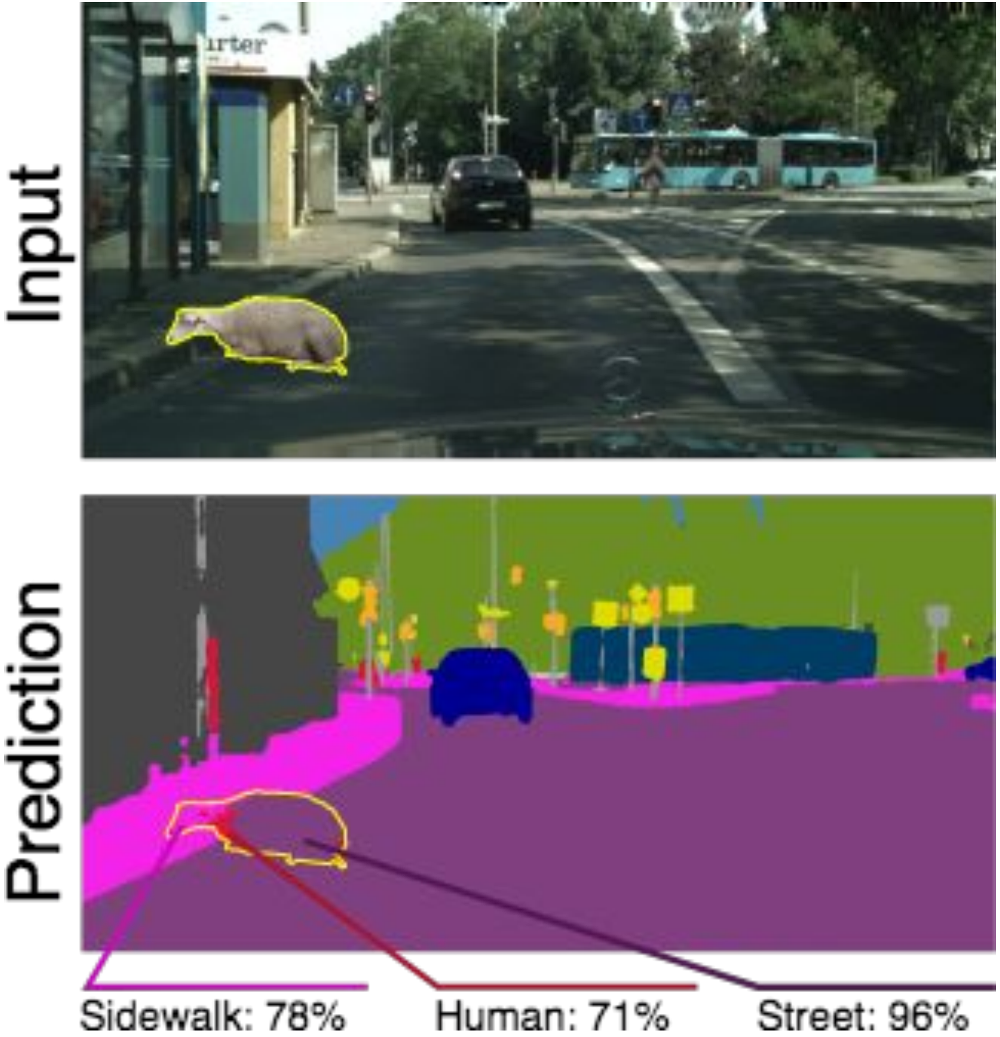
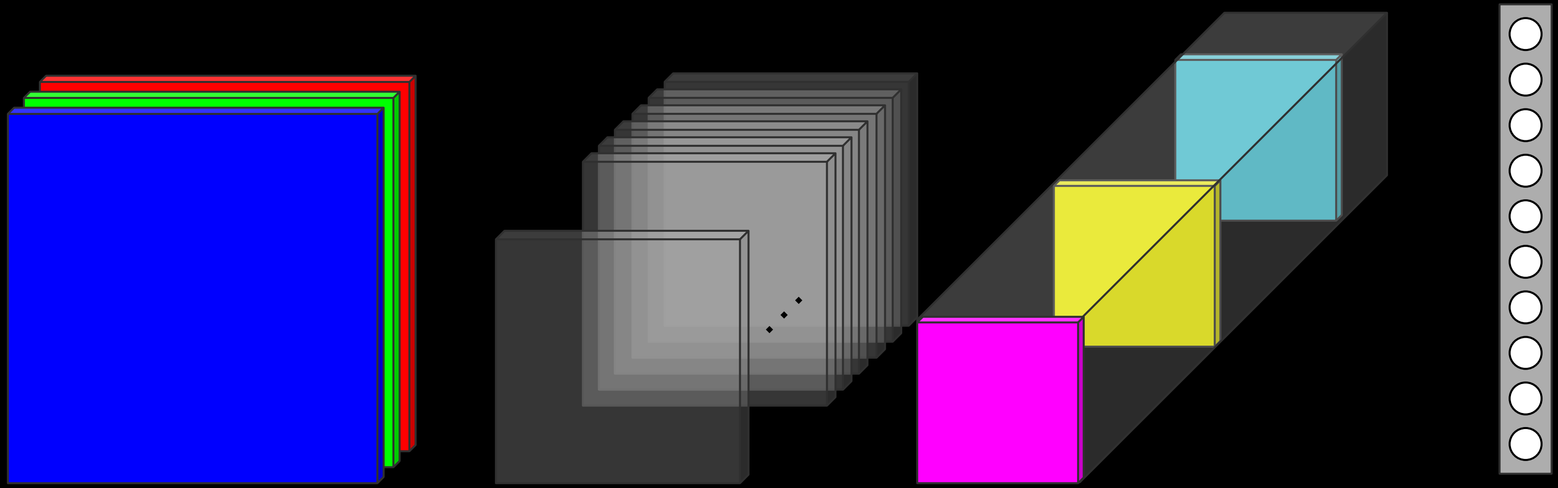
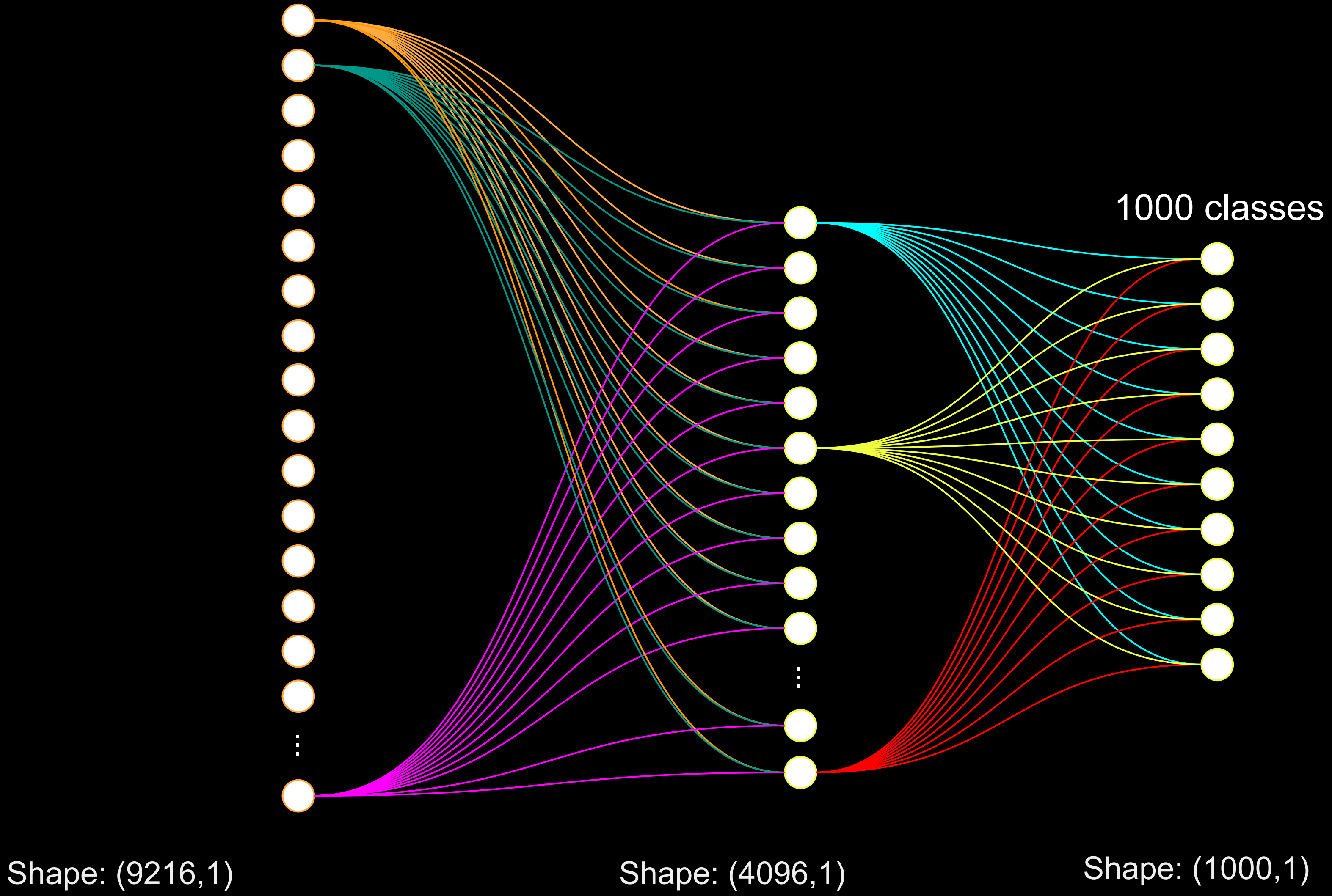
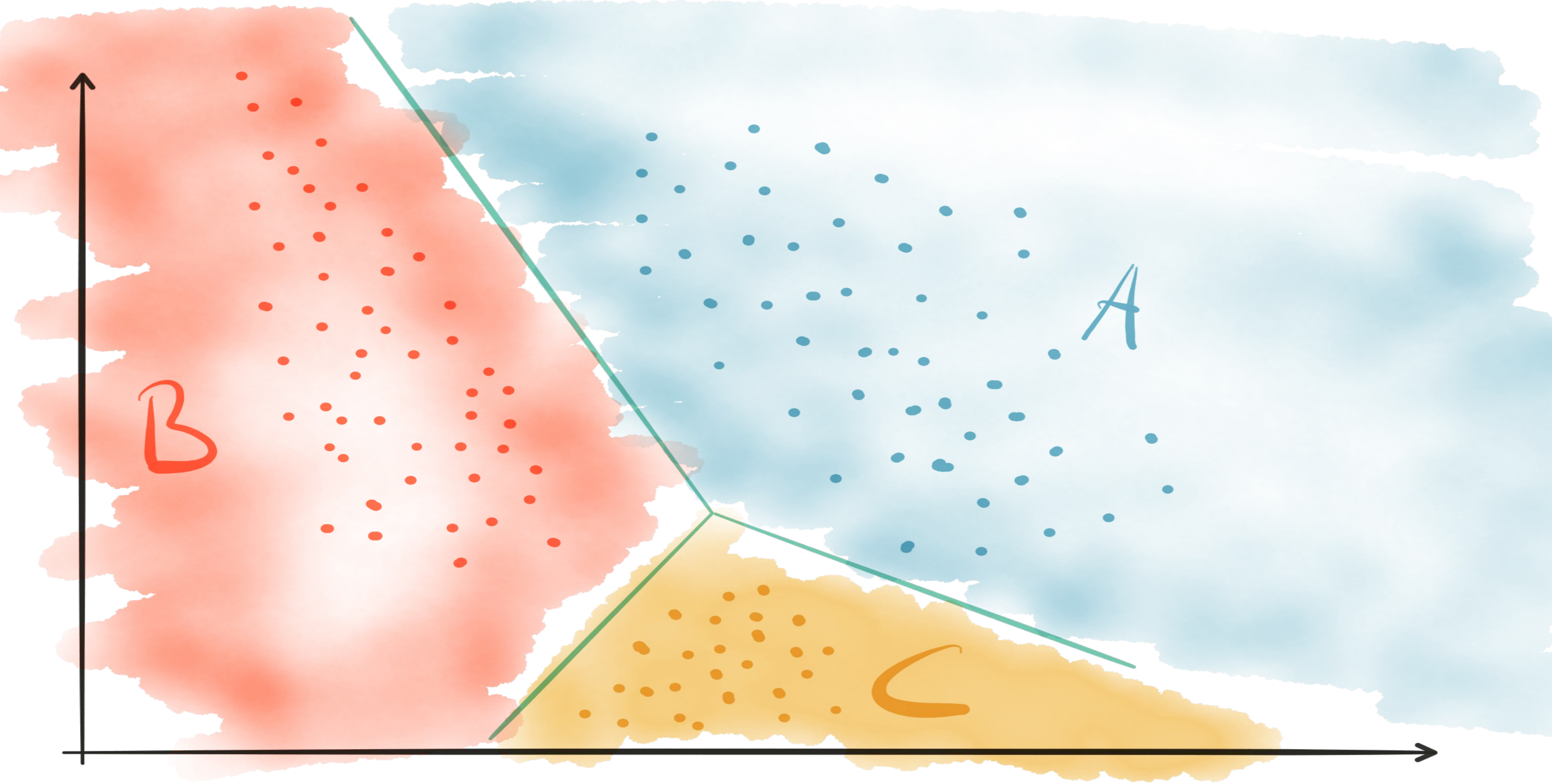


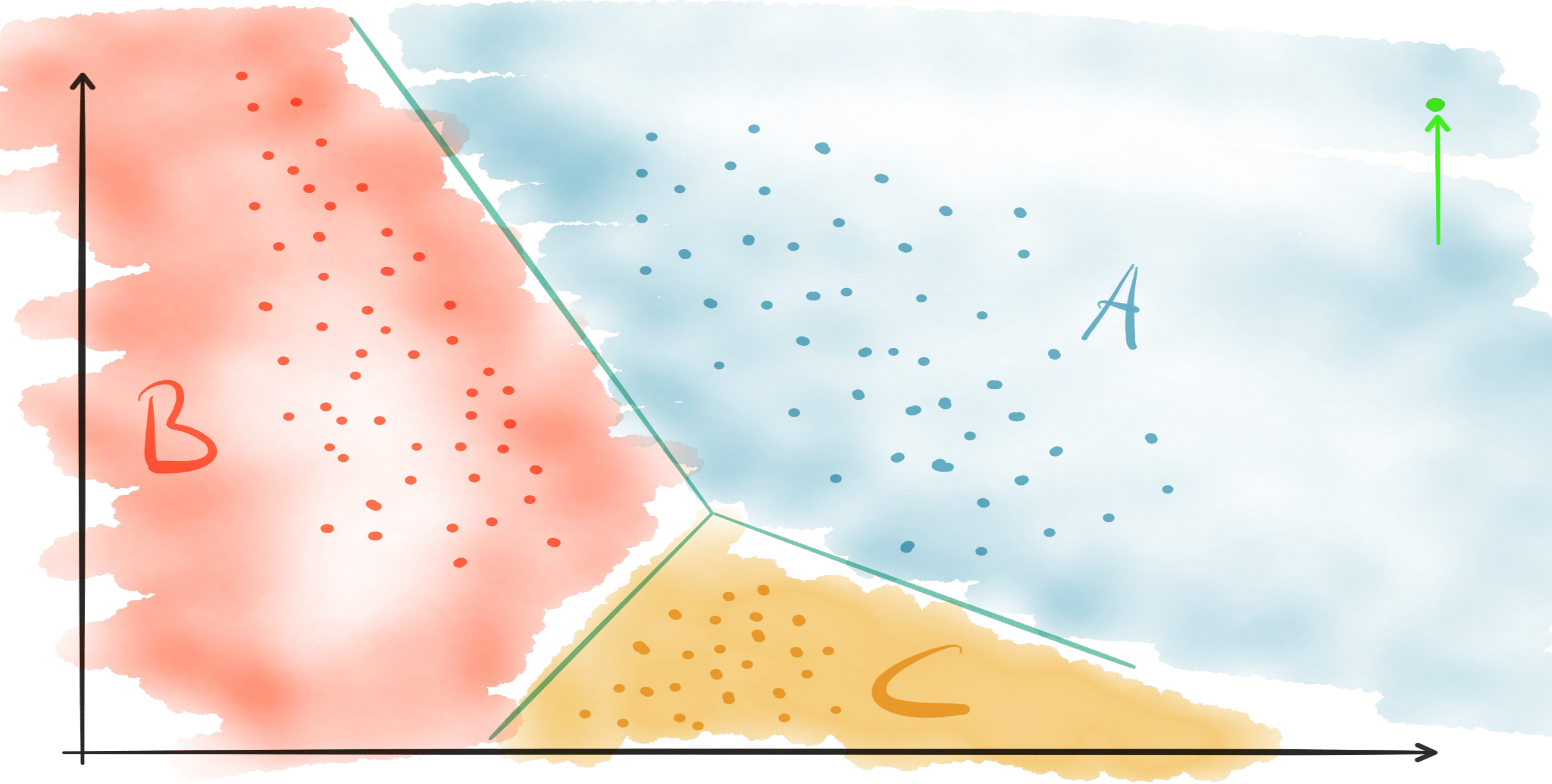
Image credit: Hermann Blum et al.
https://fishyscapes.com/

The Fishyscapes Benchmark: Measuring Blind Spots in Semantic Segmentation.
Blum, Hermann and Sarlin, Paul-Edouard and Nieto, Juan and Siegwart, Roland and Cadena, Cesar. https://arxiv.org/pdf/1904.03215.pdf

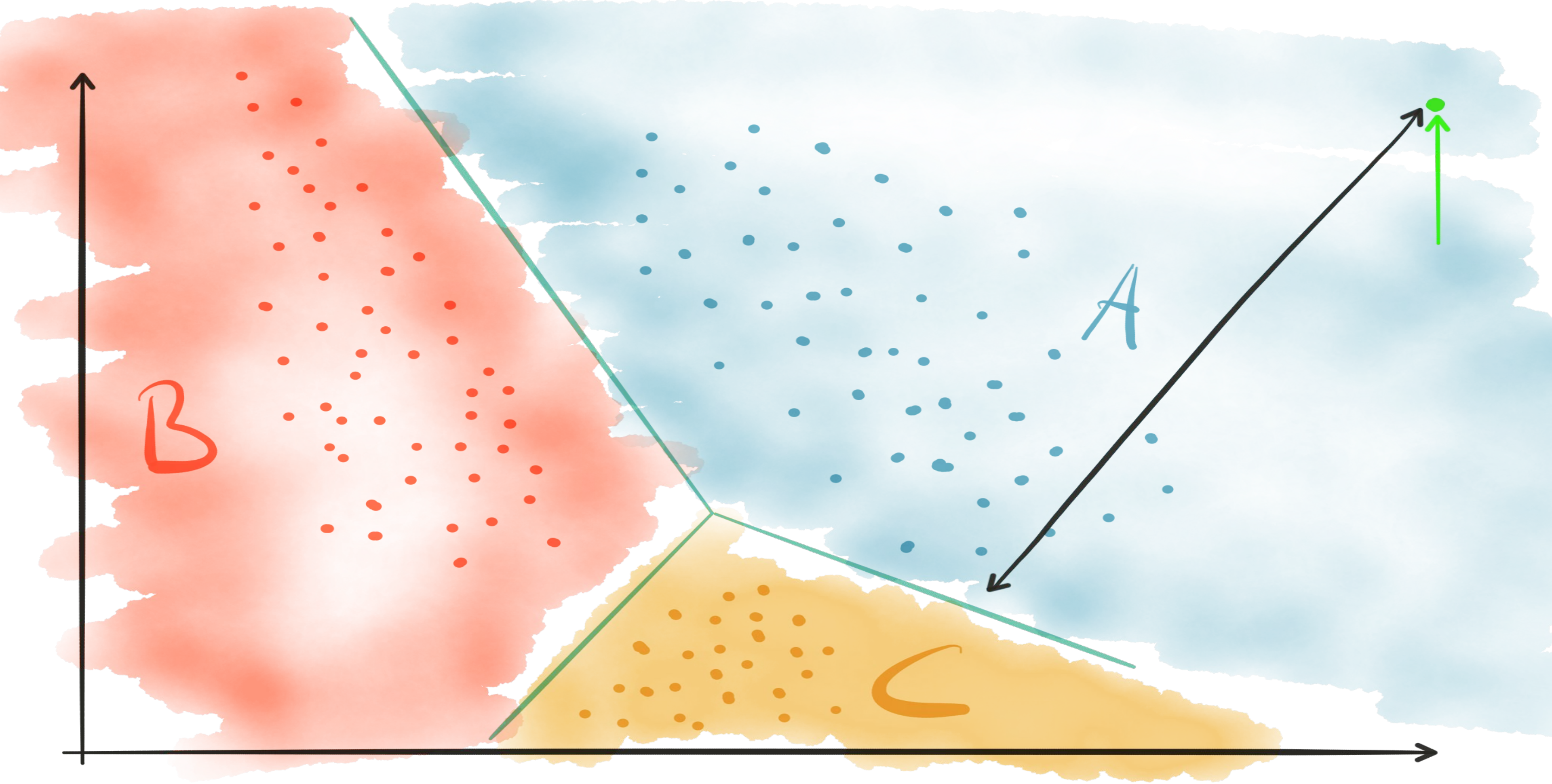



1000 classes
Shape: (9216,1)
Shape: (4096,1)
Shape: (1000,1)


A
B
C



A
B
C


A
B
C

The Open-Set Problem

Training under **Closed-Set** conditions. Deployment under **Open-Set** conditions.

- Carefully curated training (and test) datasets vs. the real world.
- Relevant for perception and action.

The Open-Set Problem

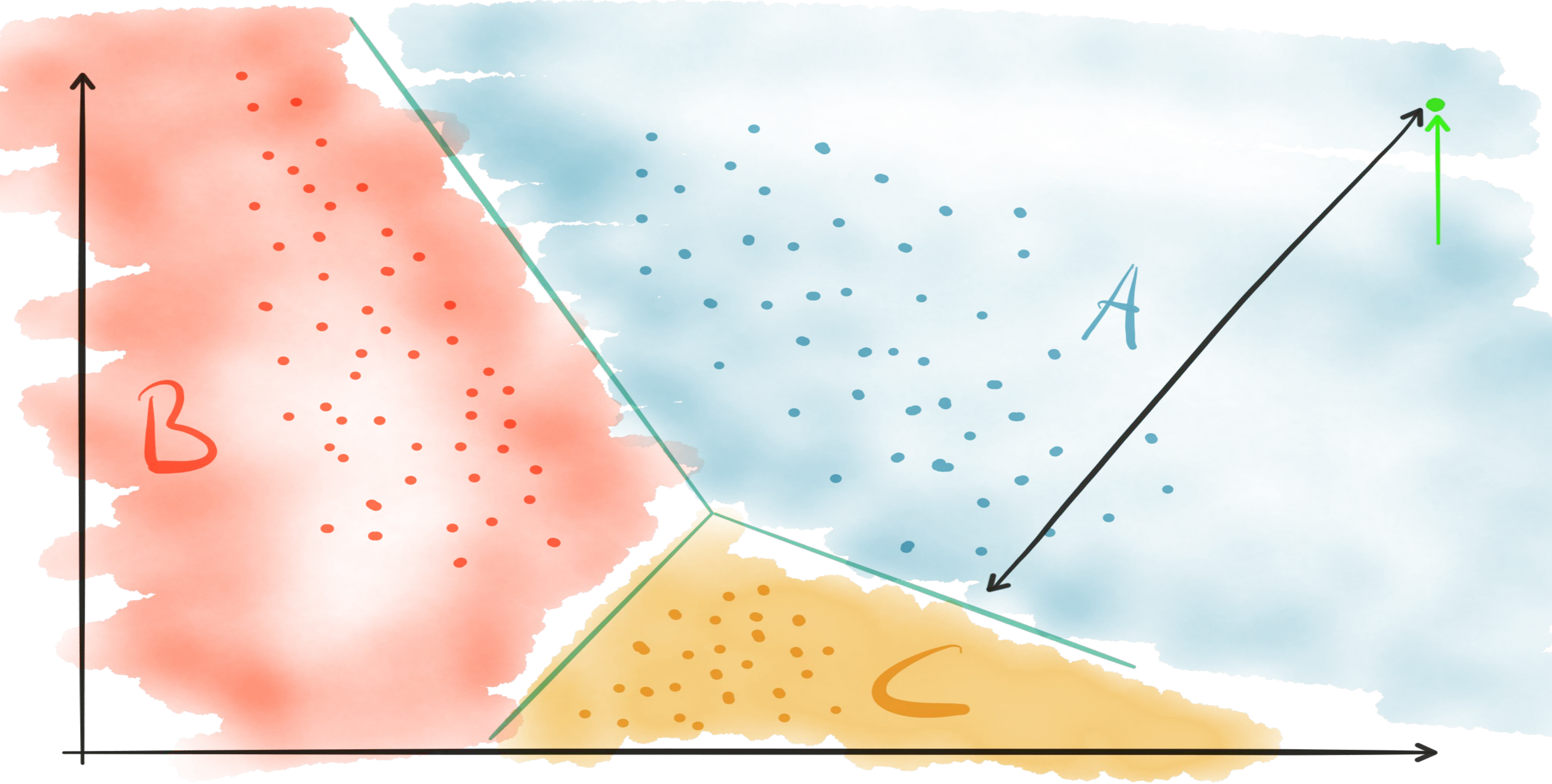
Training under **Closed-Set** conditions. Deployment under **Open-Set** conditions.

- Distribution of classes, conditions, appearance, imaging conditions (viewpoint, motion blur, focus, arrangement, …), noise, system dynamics, ... differs between training and deployment.

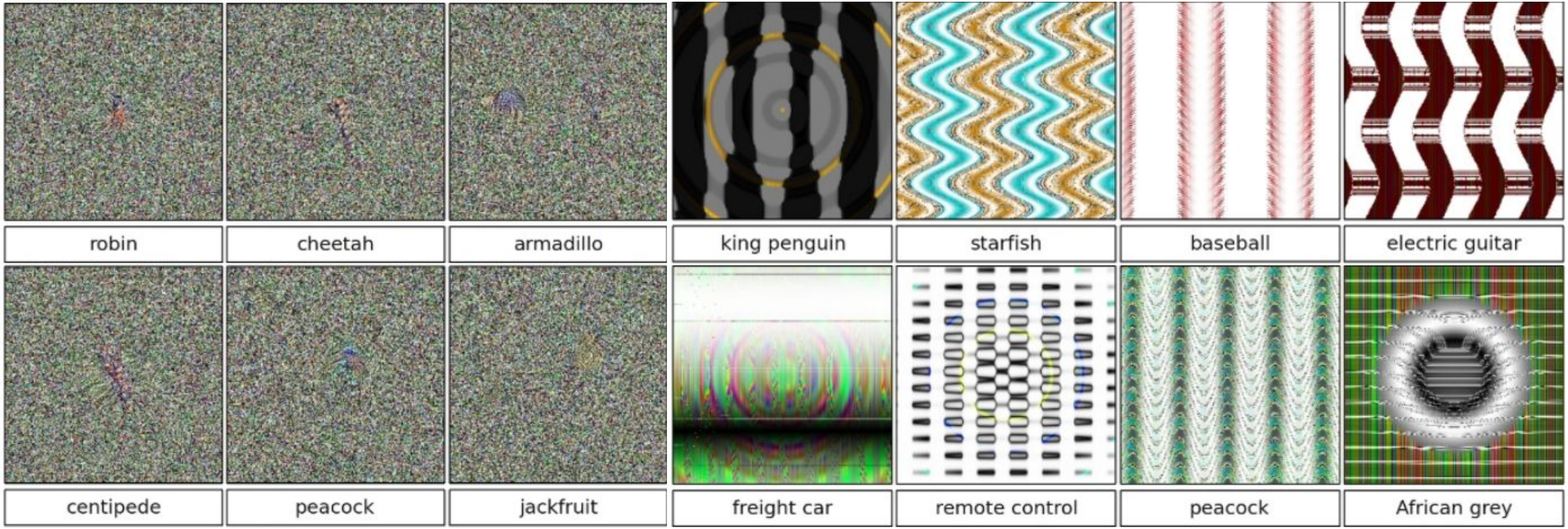
The Open-Set Problem

Training under **Closed-Set** conditions. Deployment under **Open-Set** conditions.

- out-of-distribution detection, anomaly detection, novelty detection


B
A
C

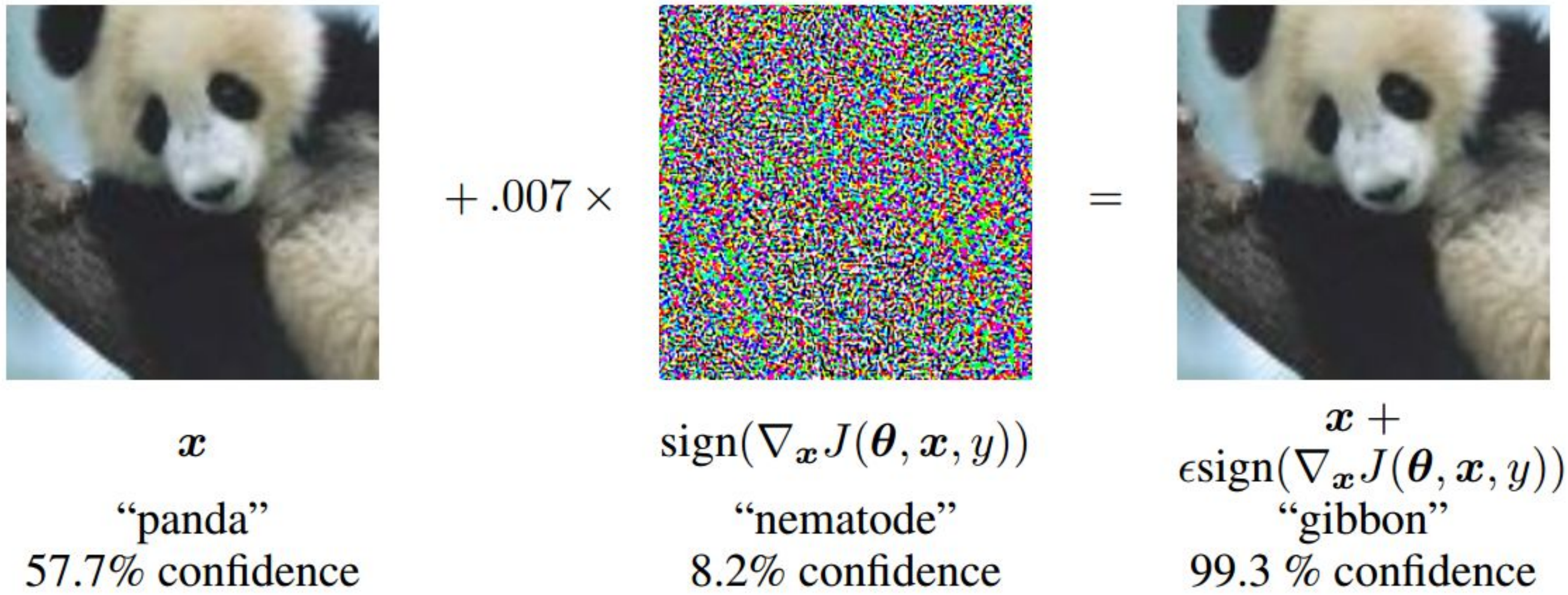
Fooling Networks



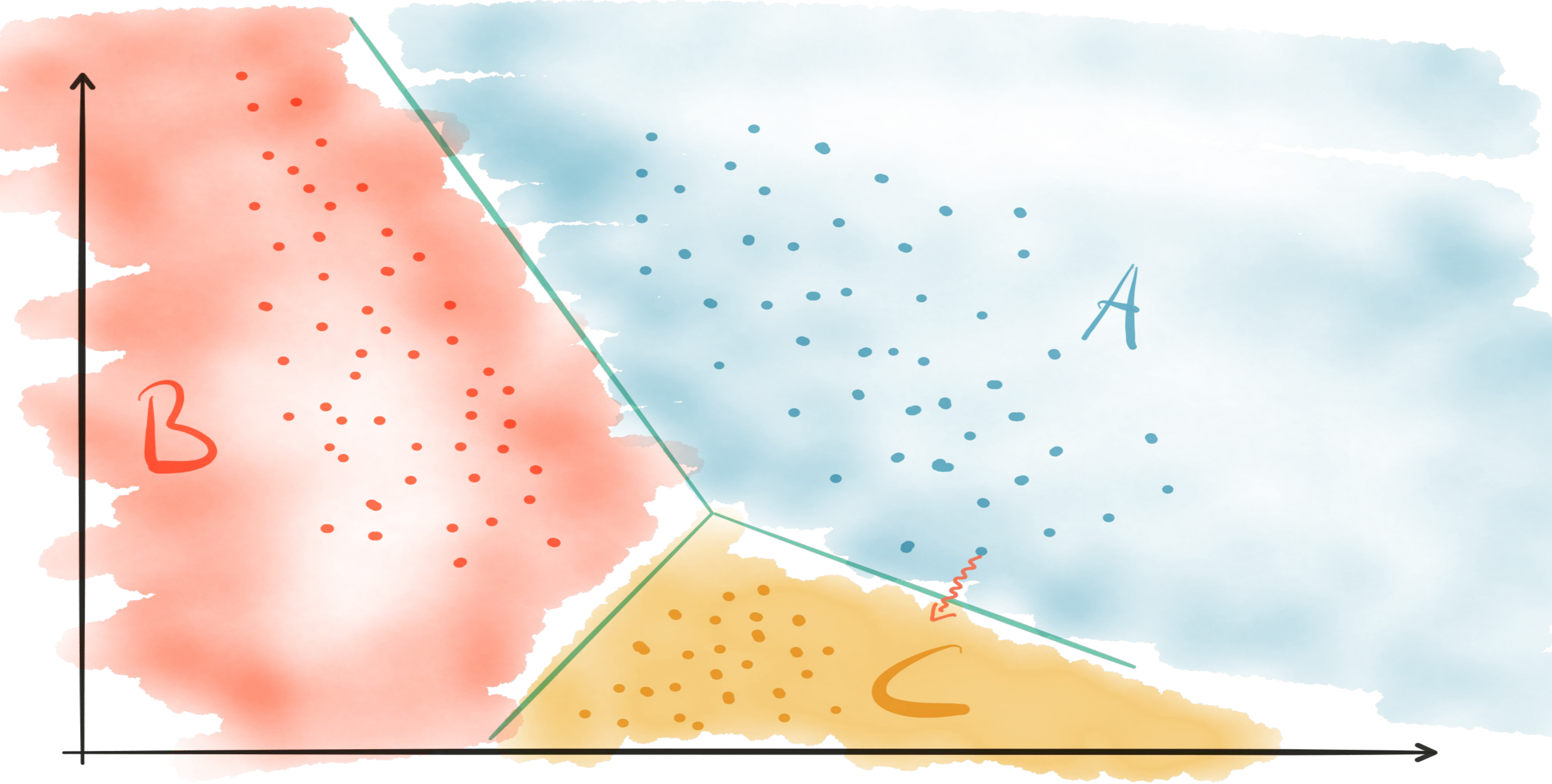
Training on ImageNet, confidence > 99.6%

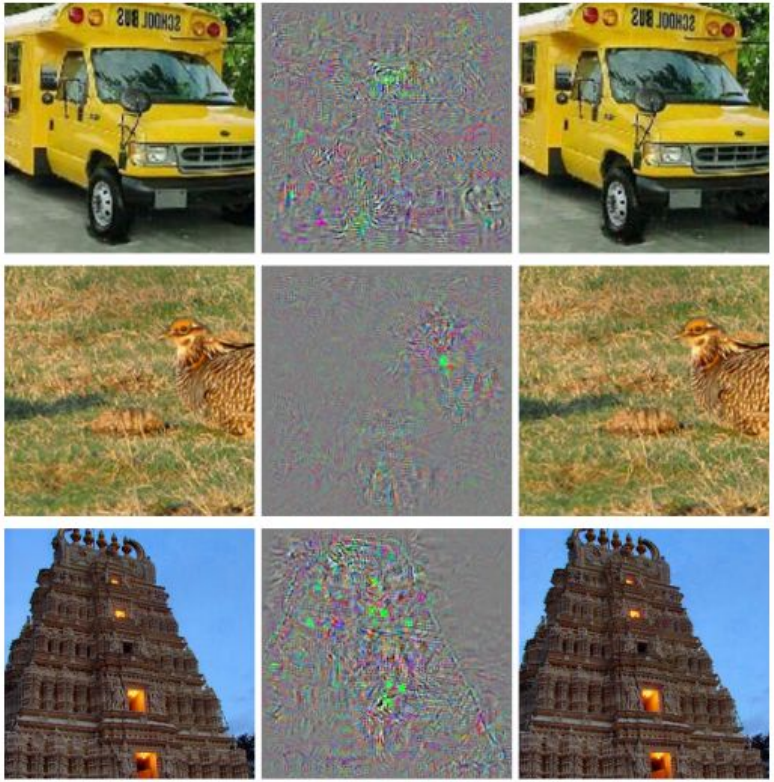
Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images (Nguyen et al., CVPR 2015)

Adversarial Examples

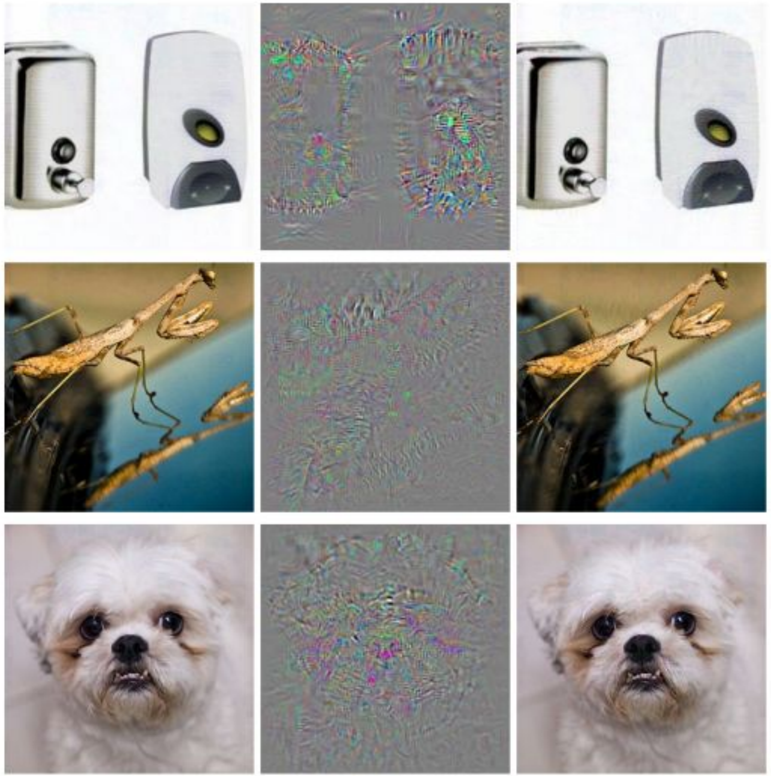


Explaining and Harnessing Adversarial Examples (Goodfellow et al., ICLR 2015)


A
B
C



correctly classified + distortion = “ostrich”



correctly classified + distortion = “ostrich”

Intriguing properties of neural networks (Szegedy et al., 2013)

Why should we care about uncertainty?

- Reliability, Safety, Trust
  - Know when the network does not know.
  - (and take appropriate action)
- Bayesian Fusion
  - Treat deep networks like any other sensor: fuse predictions with other sensors or prior knowledge in a Bayesian way.
- Active Learning
  - When uncertain, ask for help!
- Interpretability
  - More insights into the training process?

Article

The International Journal of Robotics Research
2018, Vol. 37(4–5) 405–420
© The Author(s) 2018
Reprints and permissions:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/0278364918770733
journals.sagepub.com/home/ijr
SAGE

The limits and potentials of deep learning for robotics

Niko Sünderhauf[1], Oliver Brock[2], Walter Scheirer[3], Raia Hadsell[4], Dieter Fox[5], Jürgen Leitner[1], Ben Upcroft[6], Pieter Abbeel[7], Wolfram Burgard[8], Michael Milford[1] and Peter Corke[1]

Abstract

The application of deep learning in robotics leads to very specific problems and research questions that are typically not addressed by the computer vision and machine learning communities. In this paper we discuss a number of robotics-specific learning, reasoning, and embodiment challenges for deep learning. We explain the need for better evaluation metrics, highlight the importance and unique challenges for deep robotic learning in simulation, and explore the spectrum between purely data-driven and model-driven approaches. We hope this paper provides a motivating overview of important research directions to overcome the current limitations, and helps to fulfill the promising potentials of deep learning in robotics.

Keywords

Robotics, deep learning, machine learning, robotic vision

1. Introduction

A robot is an inherently *active* agent that interacts with the real world, and often operates in uncontrolled or detrimental conditions. Robots have to perceive, decide, plan, and execute actions, all based on incomplete and uncertain knowledge. Mistakes can lead to potentially catastrophic results that will not only endanger the success of the robot's mission, but can even put human lives at risk, e.g. if the robot is a driverless car.

The application of deep learning in robotics therefore motivates research questions that differ from those typically addressed in computer vision: How much trust can we put in the predictions of a deep learning system when misclassifications can have catastrophic consequences? How can we estimate the uncertainty in a deep network's predictions and how can we fuse these predictions with prior knowledge and other sensors in a probabilistic framework? How well does deep learning perform in realistic unconstrained open-set scenarios where objects of unknown class and appearance are regularly encountered?

If we want to use data-driven learning approaches to generate motor commands for robots to move and act in the world, we are faced with additional challenging questions: How can we generate enough high-quality training data? Do we rely on data solely collected on robots in real-world scenarios or do we require data augmentation through simulation? How can we ensure the learned policies transfer well to different situations, from simulation to reality, or between different robots?

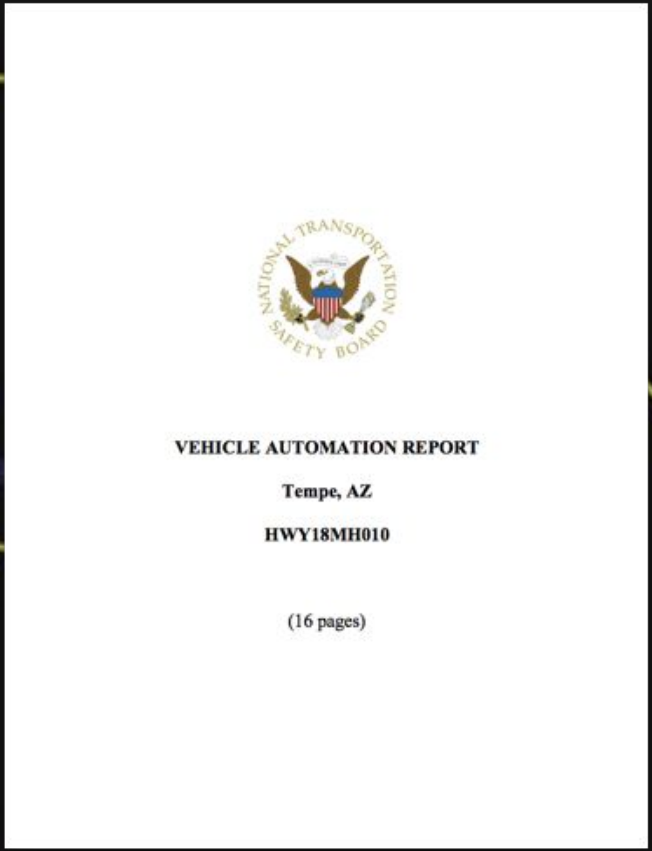
This leads to further fundamental questions: How can the structure, the constraints, and the physical laws that govern robotic tasks in the real world be leveraged and exploited by a deep learning system? Is there a fundamental difference between model-driven and data-driven problem solving, or are these rather two ends of a spectrum?

This paper explores some of the challenges, limits, and potentials for deep learning in robotics. The invited speakers and organizers of the workshop on *The Limits and*

[1]Australian Centre for Robotic Vision, Queensland University of Technology (QUT), Brisbane, Australia
[2]Robotics and Biology Laboratory, Technische Universität Berlin, Germany
[3]Department of Computer Science and Engineering, University of Notre Dame, IN, USA
[4]DeepMind, London, UK
[5]Paul G. Allen School of Computer Science & Engineering, University of Washington, WA, USA
[6]Oxbotica Ltd., Oxford, UK
[7]UC Berkeley, Department of Electrical Engineering and Computer Sciences, CA, USA
[8]Department of Computer Science, University of Freiburg, Germany

Corresponding author:
Niko Sünderhauf, Queensland University of Technology (QUT), 2 George Street, Brisbane 4000 QLD, Australia.
Email: niko.suenderhauf@roboticvision.org

The Limits and Potentials of Deep Learning for Robotics. Sünderhauf, Brock, Scheirer, Hadsell, Fox, Leitner, Upcroft, Abbeel, Burgard, Milford, Corke. IJRR 2018.

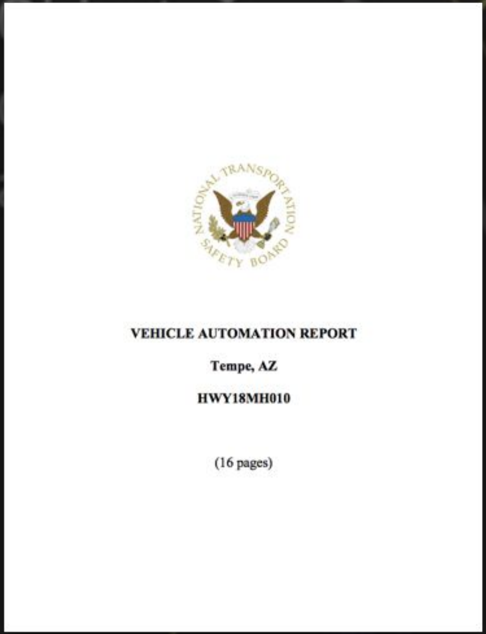

NATIONAL TRANSPORTATION SAFETY BOARD

VEHICLE AUTOMATION REPORT

Tempe, AZ

HWY18MH010

(16 pages)


VEHICLE AUTOMATION REPORT

Tempe, AZ

HWY18MH010

(16 pages)

According to data obtained from the self-driving system, the system first registered radar and LIDAR observations of the pedestrian about 6 seconds before impact, when the vehicle was traveling at 43 mph.

As the vehicle and pedestrian paths converged, the self-driving system software **classified the pedestrian as an unknown object, as a vehicle, and then as a bicycle with varying expectations of future travel path**.

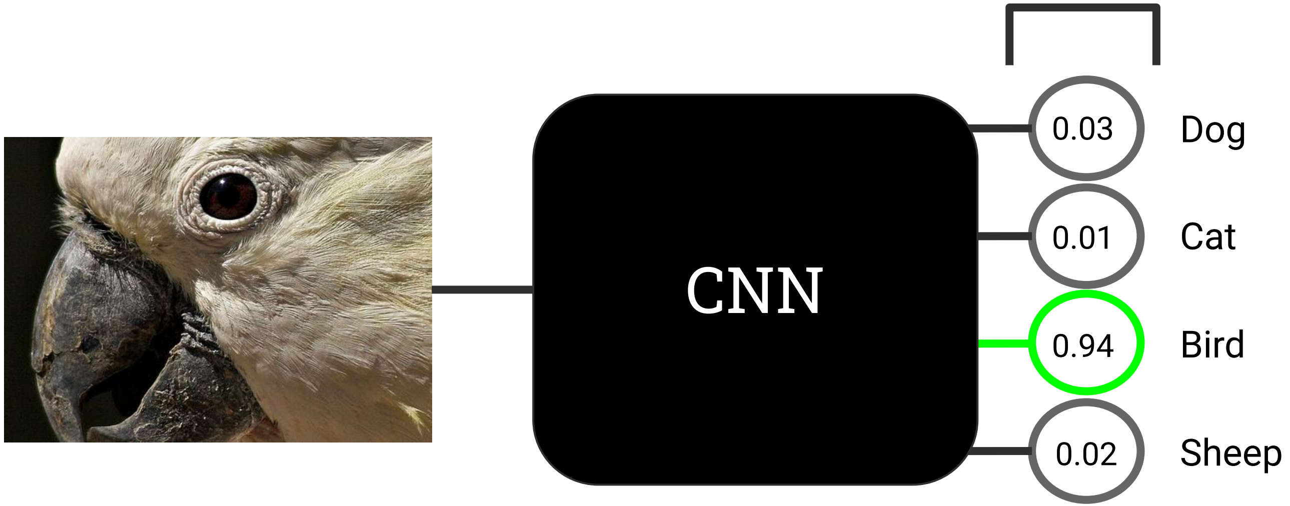
At 1.3 seconds before impact, the self-driving system determined that an emergency braking maneuver was needed …


LOST
CONFUSED
UNSURE
UNCLEAR
PERPLEXED
DISORIENTED
BEWILDERED

Softmax-based Uncertainty

Baseline (Hendrycks et al., 2016)

Softmax "probabilities"



< θ, uncertain
> , certain

Closed-set Performance (Accuracy, mAP, etc.) ✓

Open-set Performance (Uncertainty) ✗

Robotic Vision (Object Detection/Instance Segmentation) ✓

Slide courtesy of Dimity Miller

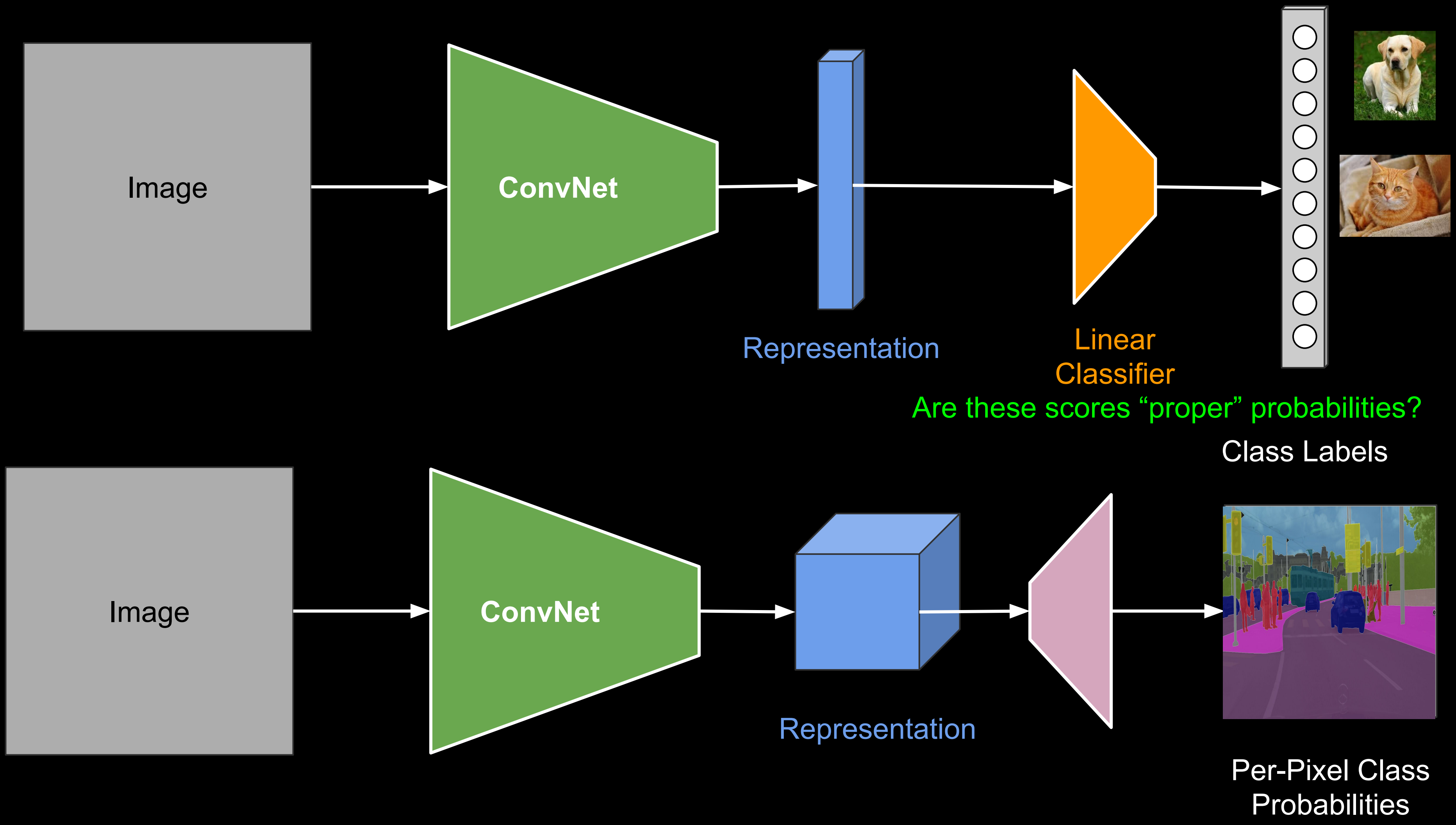
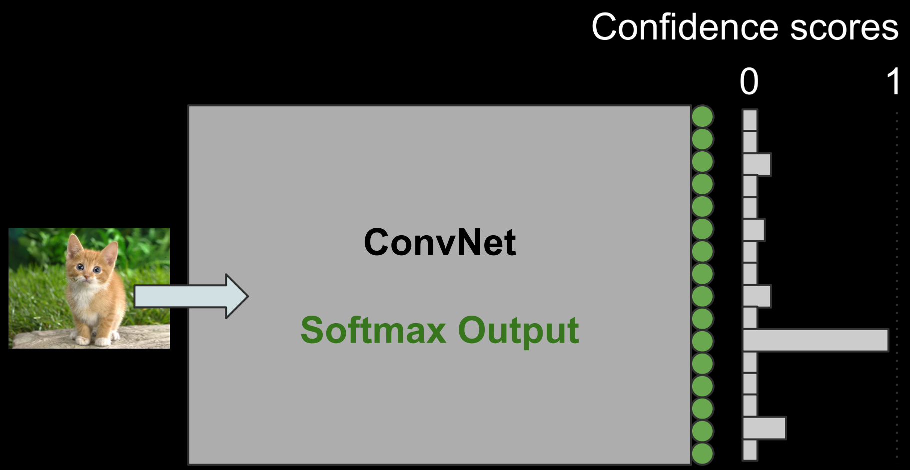
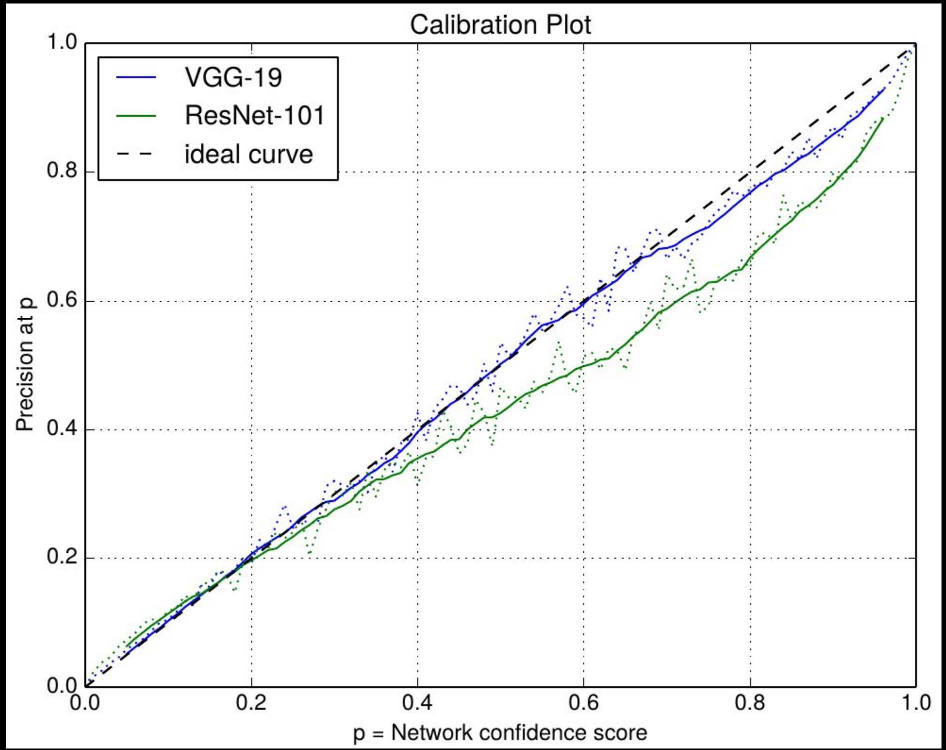


Image
ConvNet
Representation
Linear
Classifier
Are these scores “proper” probabilities?
Class Labels
Image
ConvNet
Representation
Per-Pixel Class
Probabilities

Confidence = Probability?





Softmax-based Uncertainty

Out-of-DIstribution detector for Neural networks (ODIN) (Liang et al., 2018)

1. Perturbations to input

CNN

Softmax "probabilities"

2. Temperature Scaling

$$S_i(\boldsymbol{x};T) = \frac{\exp\left(f_i(\boldsymbol{x})/T\right)}{\sum_{j=1}^{N}\exp\left(f_j(\boldsymbol{x})/T\right)}$$

0.03	Dog
0.01	Cat
0.94	Bird
0.02	Sheep

< θ, uncertain

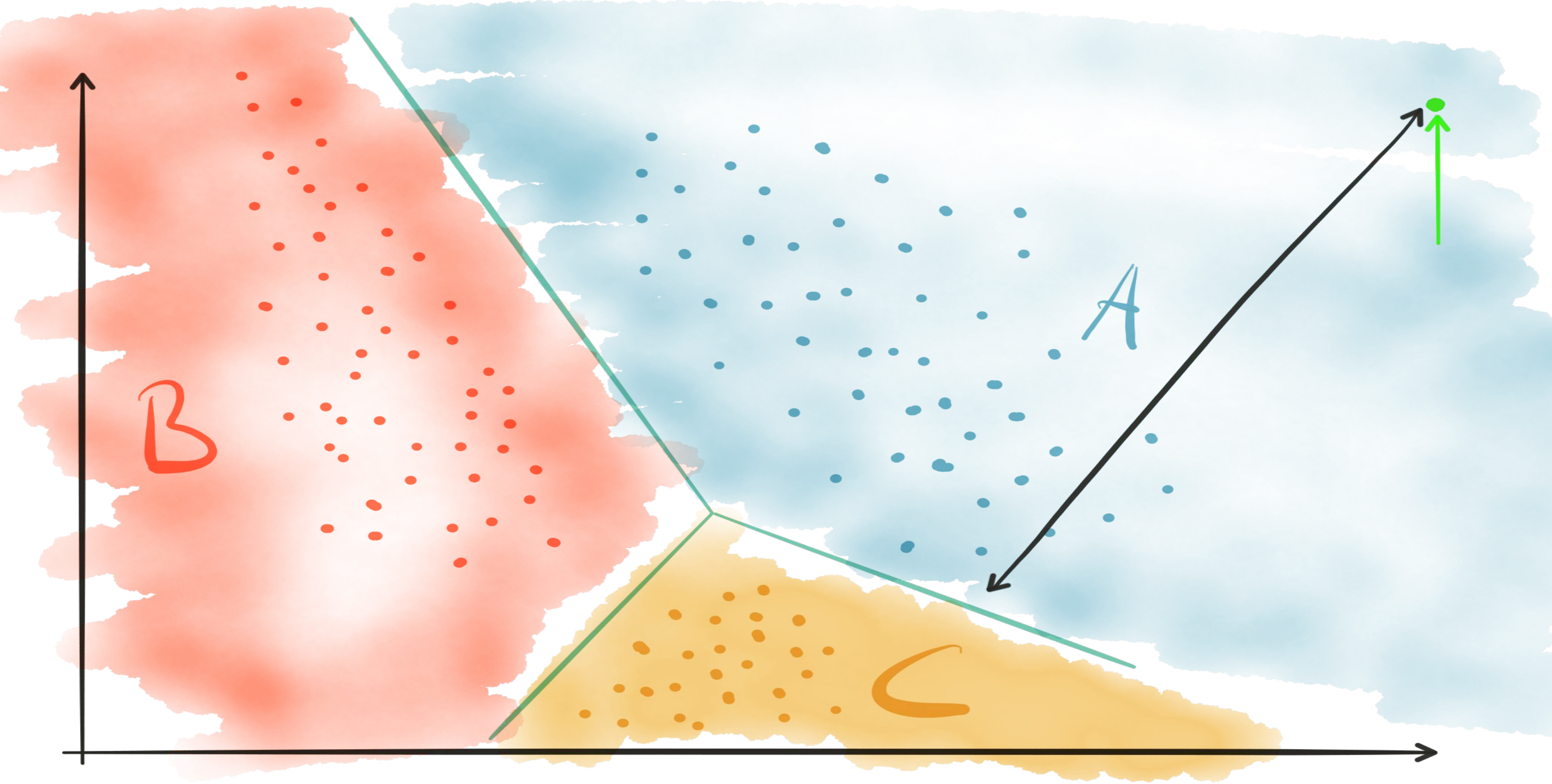
> , certain

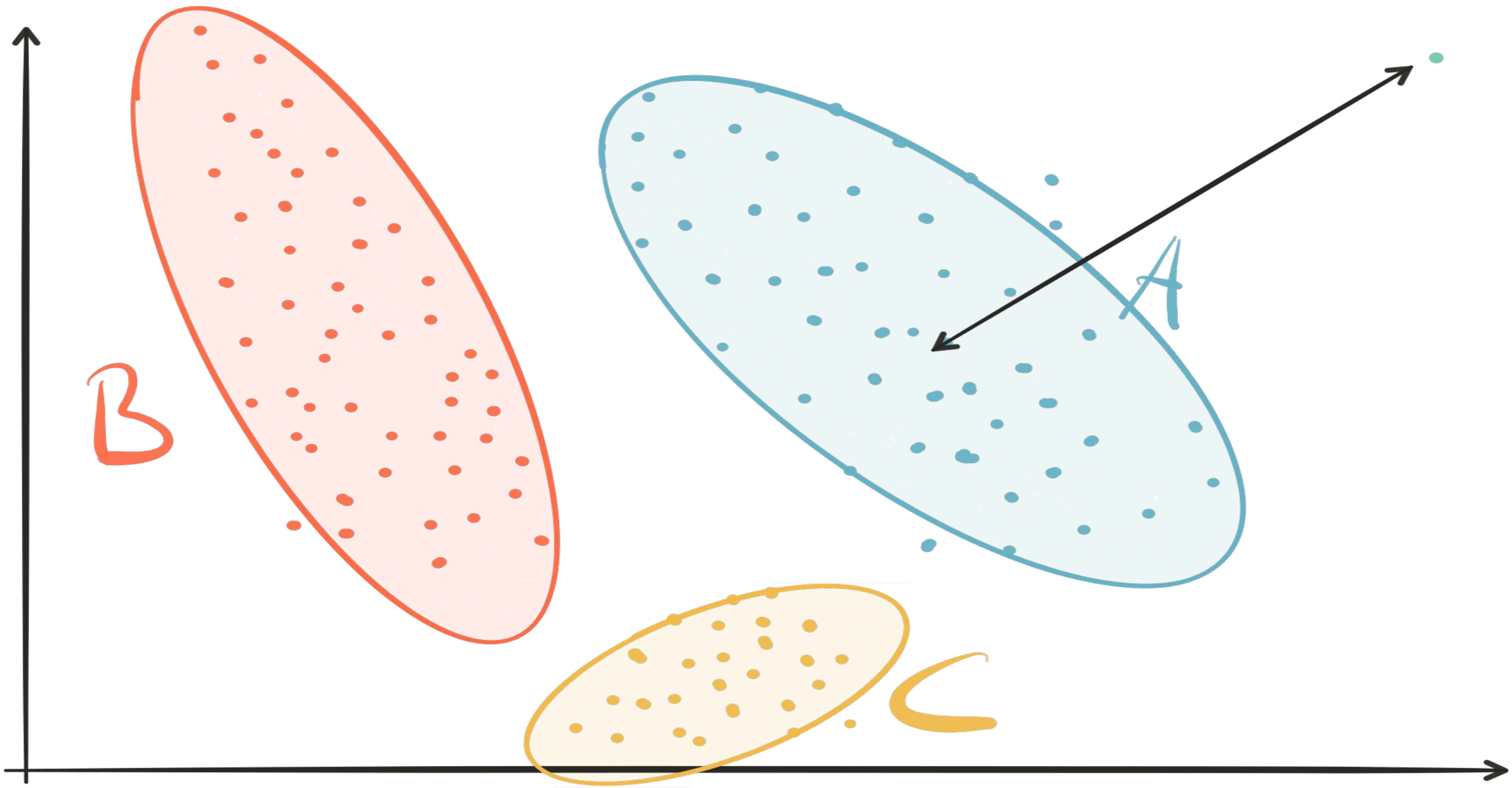
Closed-set Performance ✓
(Accuracy, mAP, etc.)

Open-set Performance ✗
(Uncertainty)

Robotic Vision ✓
(Object Detection/Instance Segmentation)

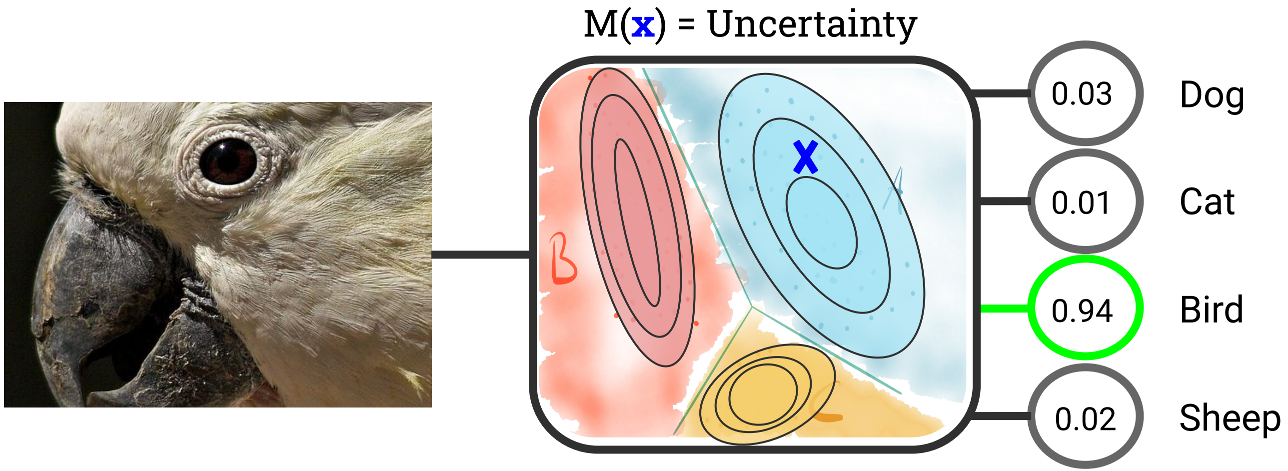
Slide courtesy of Dimity Miller


A
B
C


A
B
C

Distance-based Uncertainty with Cross-Entropy Loss

Multivariate Gaussians and Mahalanobis Distance (Lee et al., 2018)



Closed-set Performance
(Accuracy, mAP, etc.) ✓

Open-set Performance
(Uncertainty) ✓

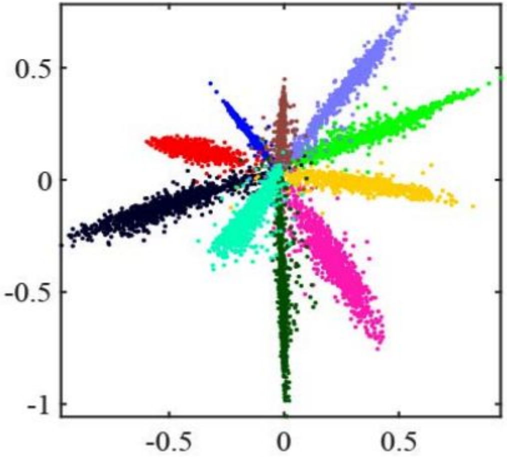
Robotic Vision
(Object Detection/Instance Segmentation) ✗

Slide courtesy of Dimity Miller

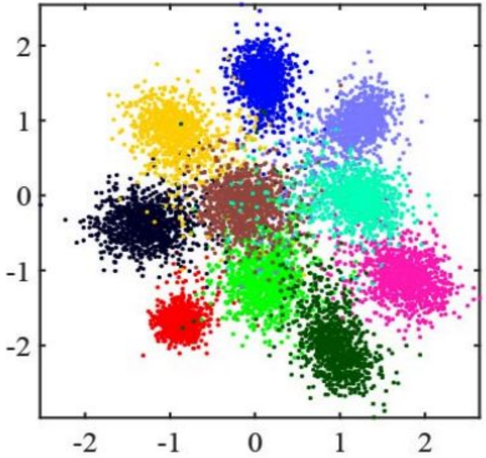
Distance-based Uncertainty with Metric Learning Losses

Contrastive Loss (Masana et al., 2018)

Gaussian Kernel Loss (Meyer et al., 2019)



Cross-entropy Loss
(softmax CNNs)



Contrastive Loss
(metric learning loss)

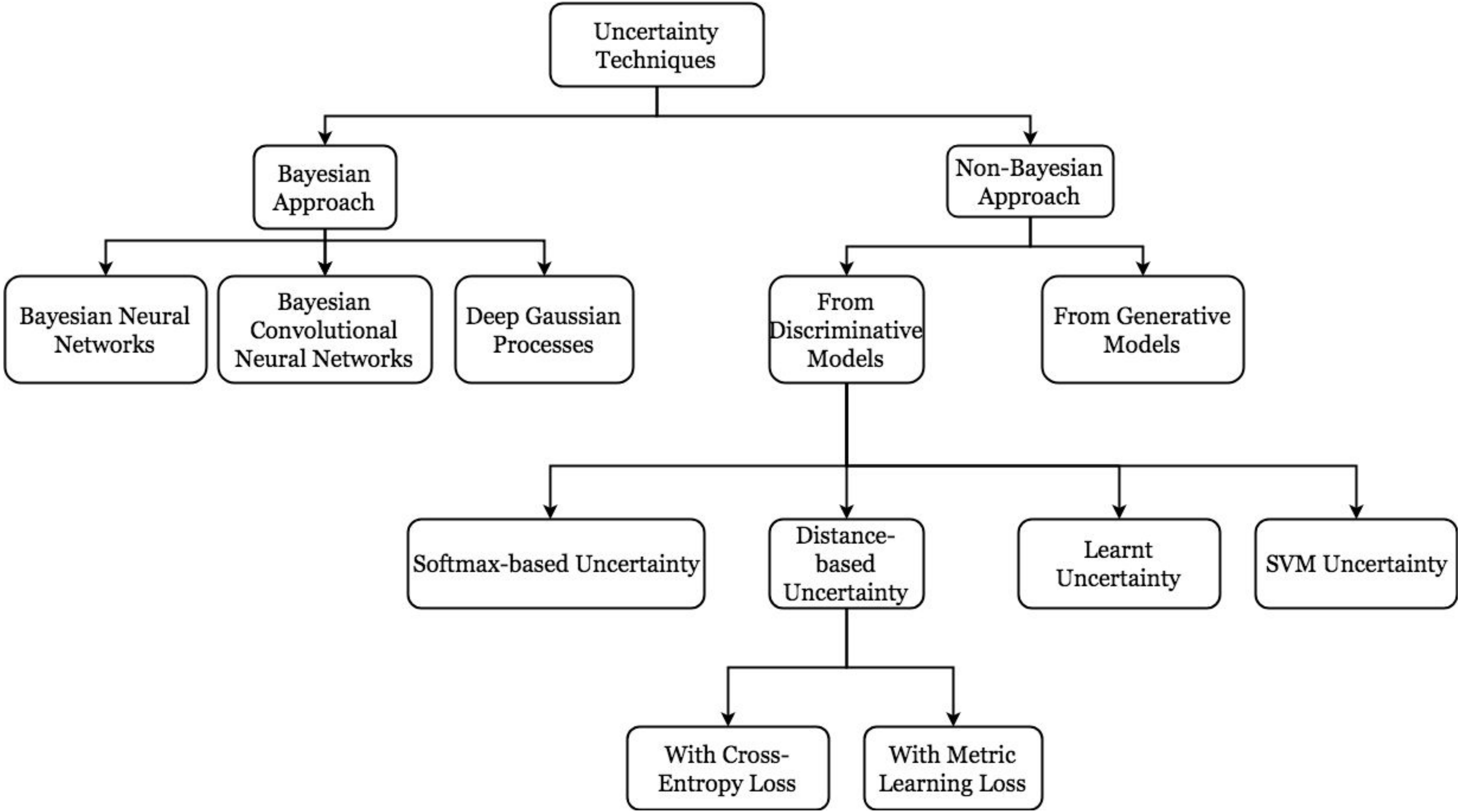
(Image: Horiguchi et al., 2017)

Closed-set Performance
(Accuracy, mAP, etc.)

Open-set Performance
(Uncertainty)

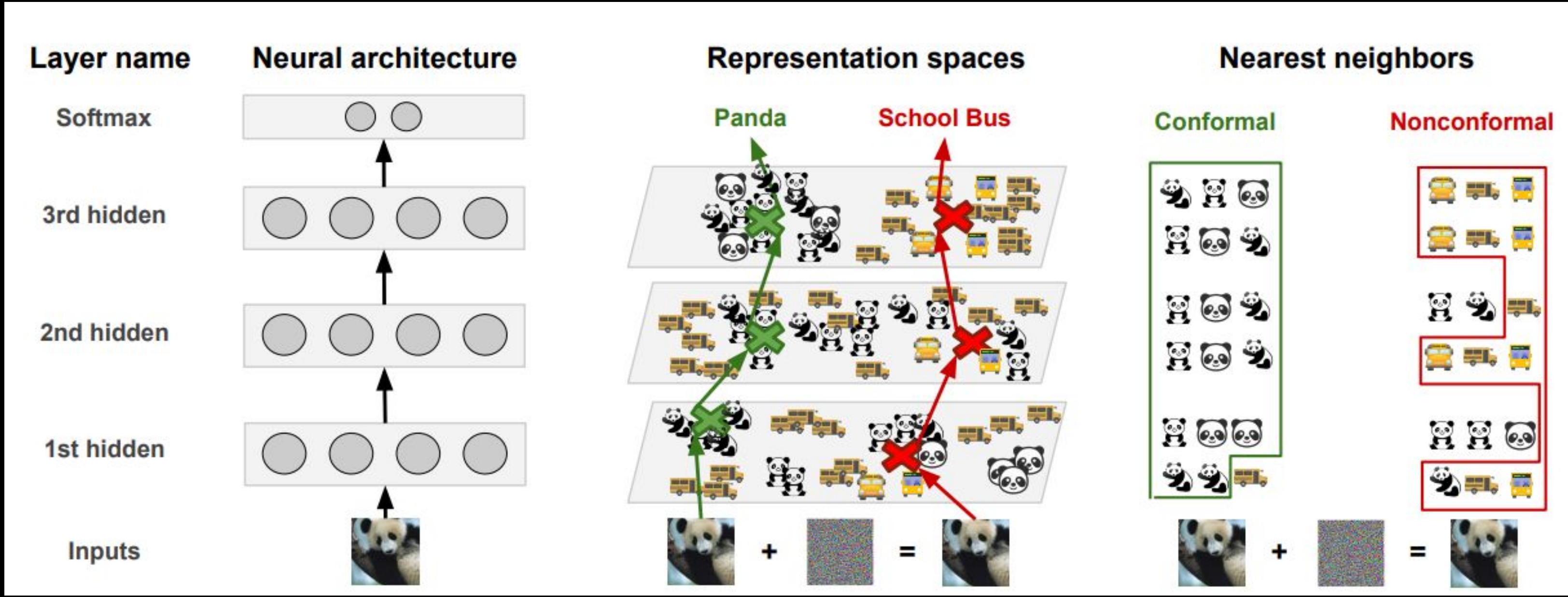
Robotic Vision
(Object Detection/Instance Segmentation)

Slide courtesy of Dimity Miller



Slide courtesy of Dimity Miller

Deep k-Nearest Neighbors



“Deep k-Nearest Neighbors: Towards Confident, Interpretable and Robust Deep Learning”. Nicolas Papernot and Patrick McDaniel

Evaluating Uncertainty Techniques for Robotic Vision

Low Resolution Datasets

Non-diverse Datasets

Unrealistic Open-set Conditions

Dataset	# Classes
CIFAR-10	10
SVHN	10
LSUN	10
MNIST	10
CIFAR-100	100

Shafaei et al., 2018

Slide courtesy of Dimity Miller

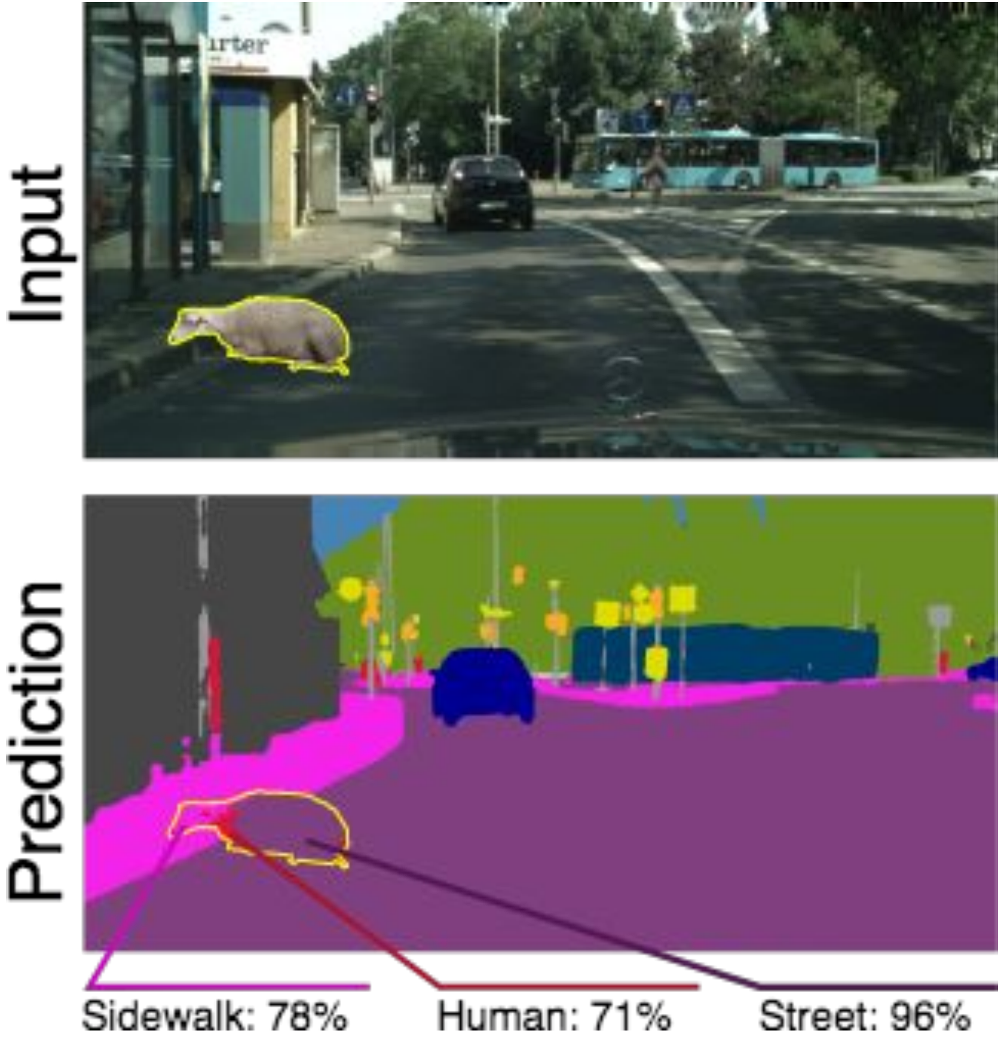


Image credit: Hermann Blum et al.
https://fishyscapes.com/

The Fishyscapes Benchmark: Measuring Blind Spots in Semantic Segmentation.
Blum, Hermann and Sarlin, Paul-Edouard and Nieto, Juan and Siegwart, Roland and Cadena, Cesar. https://arxiv.org/pdf/1904.03215.pdf


LOST
CONFUSED
UNSURE
UNCLEAR
PERPLEXED
DISORIENTED
BEWILDERED

Bayesian Deep Learning

"Normal" Deep Learning

- CNN is a function f with parameters w
- f(x) generates labels y
- we seek the optimal parameters w (via stochastic gradient descent etc)

$$\text{training inputs } \mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$$
$$\text{outputs } \mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$$
$$\mathbf{y} = \mathbf{f}^{\boldsymbol{\omega}}(\mathbf{x})$$

Bayesian Deep Learning

- Use a prior $p(\boldsymbol{\omega})$ on the network parameters
- Learning is finding the posterior over parameters $p(\boldsymbol{\omega}|\mathbf{X}, \mathbf{Y})$
- not just one CNN, but a distribution over CNNs!

Bayesian Deep Learning

Classify a new input image x = **inference**:

intractable!

$$p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{X}, \mathbf{Y}) = \int p(\mathbf{y}^*|\mathbf{x}^*, \boldsymbol{\omega})p(\boldsymbol{\omega}|\mathbf{X}, \mathbf{Y})\mathrm{d}\boldsymbol{\omega}$$

Given the training data X,Y, and a new image x, obtain the distribution over labels y by ...

… averaging over the individual predictions of ALL possible network parameters w!

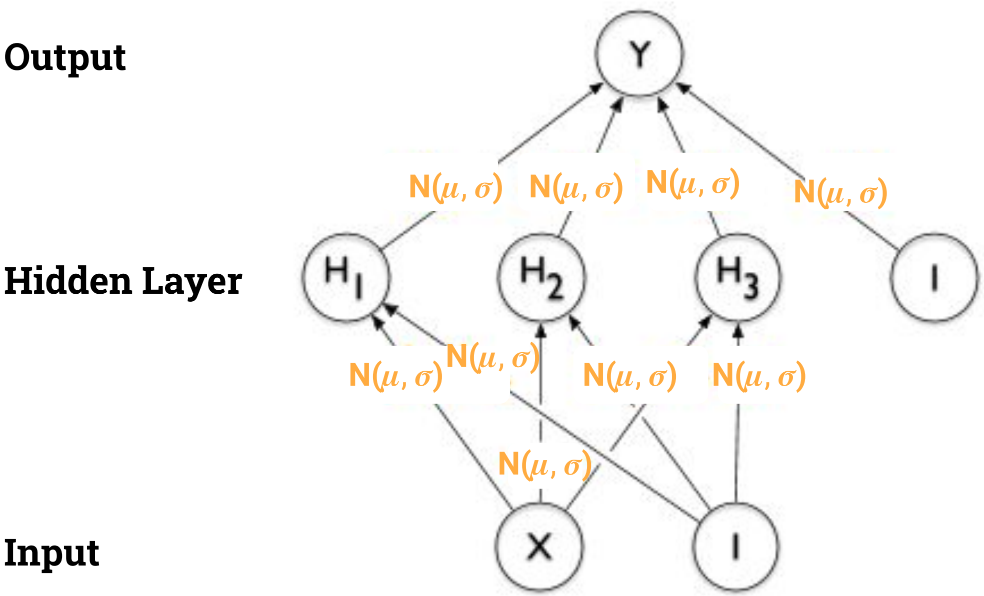
intractable!

intractable!

learning: $p(\boldsymbol{\omega}|\mathbf{X}, \mathbf{Y}) = \dfrac{p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\omega})p(\boldsymbol{\omega})}{p(\mathbf{Y}|\mathbf{X})}$ $\qquad p(\mathbf{Y}|\mathbf{X}) = \int p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\omega})p(\boldsymbol{\omega})\mathrm{d}\boldsymbol{\omega}$

We need approximations!

Bayesian Neural Networks



Approximate Bayesian Neural Network

(Image: Blundell et al., 2015)

Posterior

Intractable

$$p(\boldsymbol{\omega}|\mathbf{X},\mathbf{Y}) = \frac{p(\mathbf{Y}|\mathbf{X},\boldsymbol{\omega})p(\boldsymbol{\omega})}{p(\mathbf{Y}|\mathbf{X})}$$

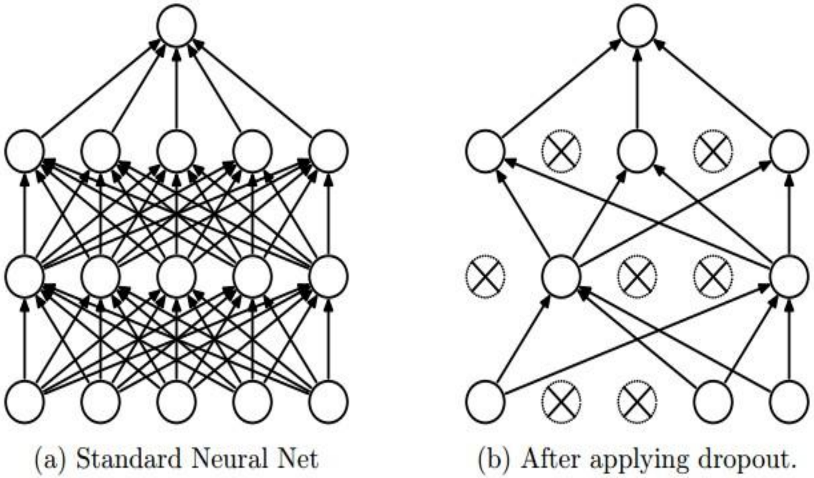
Approximate Posterior

Variational Inference

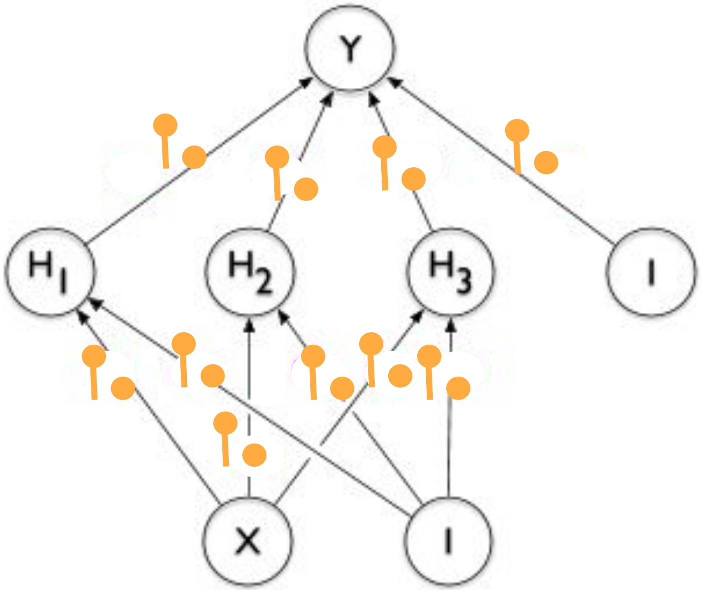
Slide courtesy of Dimity Miller

Bayesian Convolutional Neural Networks

Monte Carlo (MC) Dropout (Gal et al., 2017)



(Image: Srivastava et al., 2014)



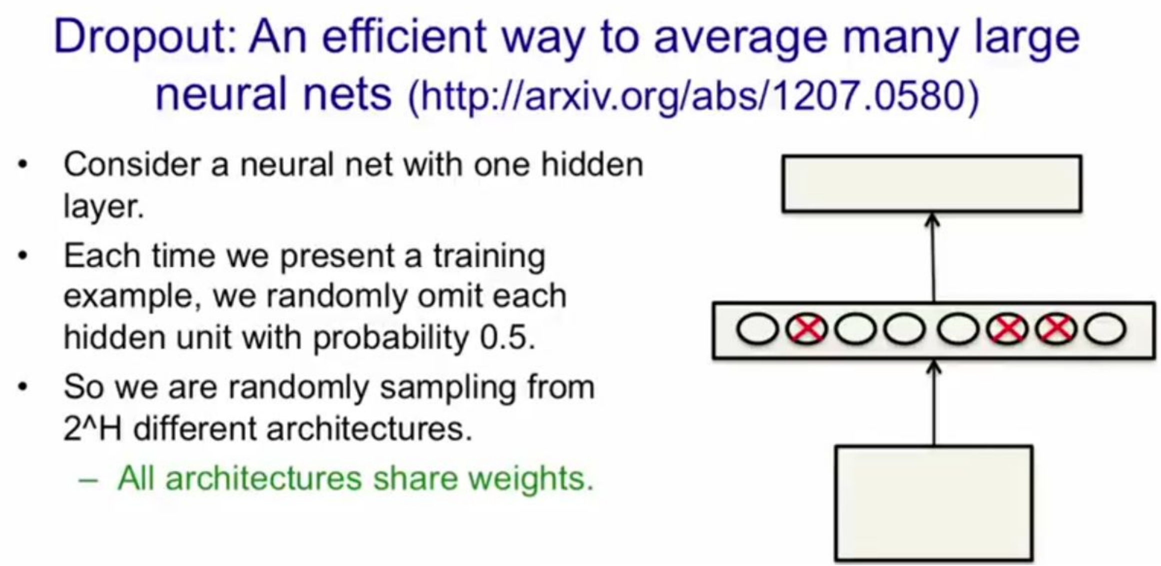
Closed-set Performance
(Accuracy, mAP, etc.)

Open-set Performance
(Uncertainty)

Robotic Vision
(Object Detection/Instance Segmentation)

Slide courtesy of Dimity Miller

Dropout to the Rescue (again)



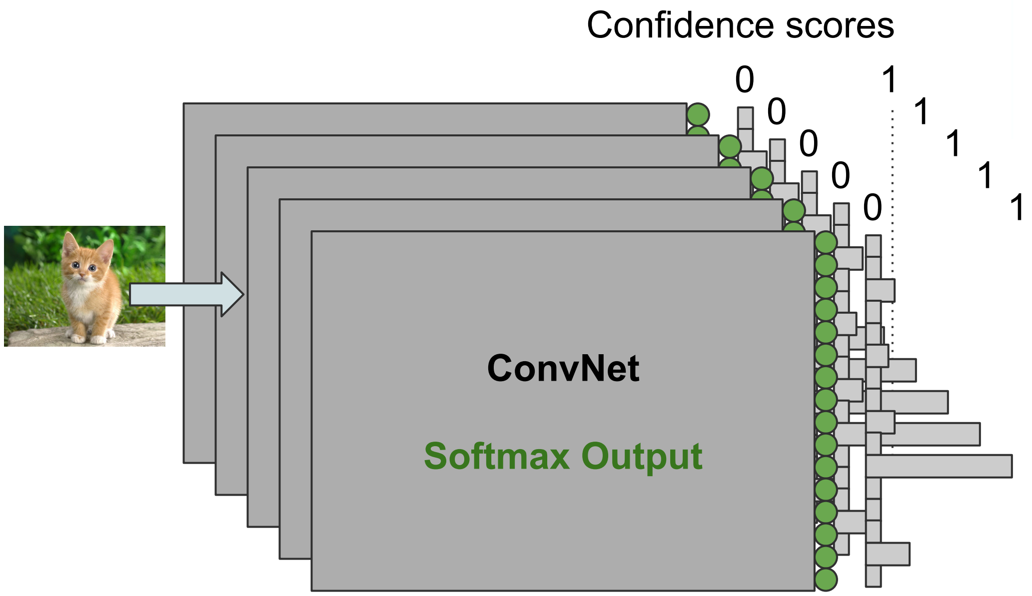
Neural Networks for Machine Learning, Geoffrey Hinton on Coursera in 2012

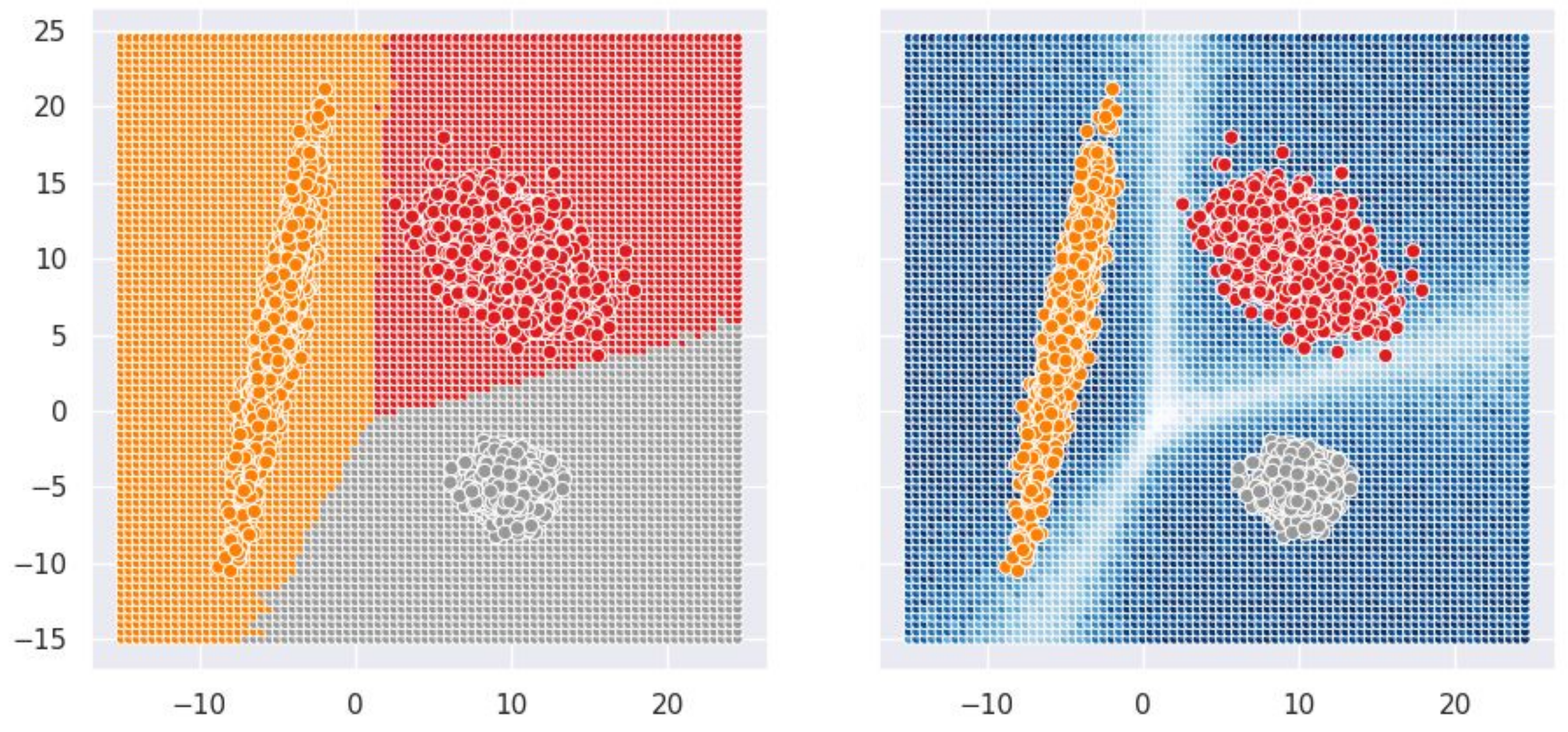
- Dropout as a Bayesian Approximation (Gal and Ghahramani, ICLR 2015)
- Yarin Gal's PhD thesis
- NIPS 2016 workshop (www.bayesiandeeplearning.org)

Confidence = Probability?

```
class Net(nn.Module):
    def __init__(self):
        super(Net, self).__init__()
        self.fc1 = nn.Linear(2, 64)
        self.fc2 = nn.Linear(64, 3)

    def forward(self, x):
        x = self.fc1(x)
        x = nn.functional.relu(x)
        x = nn.functional.dropout(x, training=True)
        x = self.fc2(x)
        return x
```



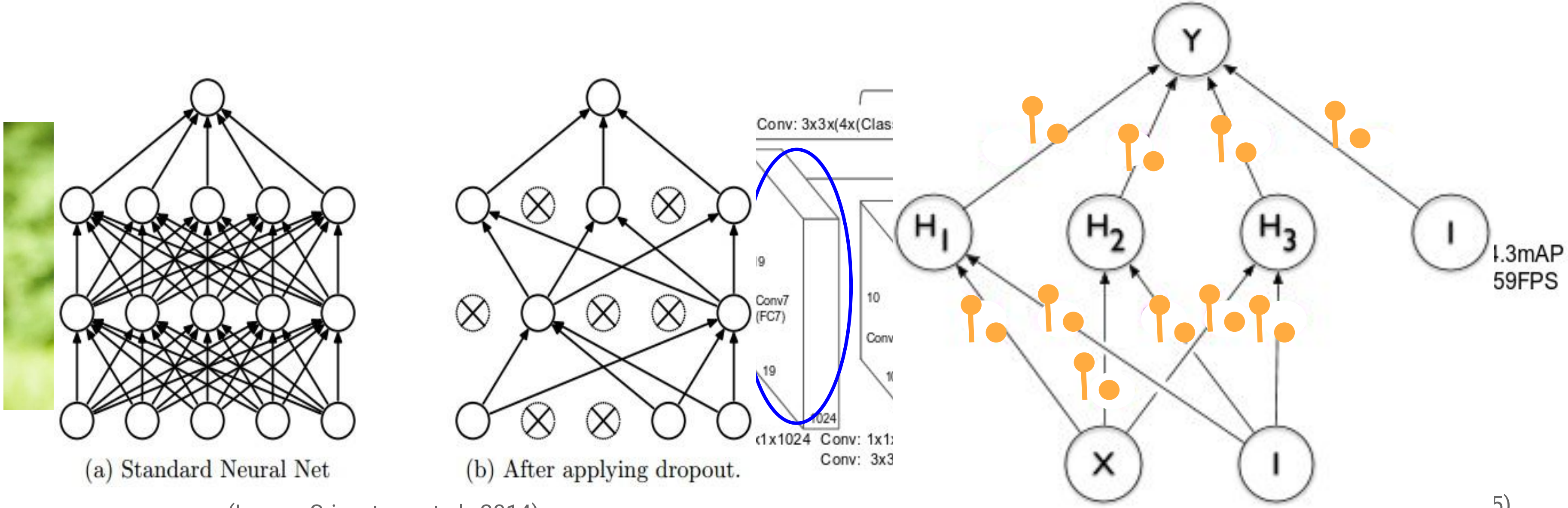

25
20
15
10
5
0
−5
−10
−15
−10
0
10
20
−10
0
10
20

Uncertainty from Object Detection

Single Shot MultiBox Detector (SSD)
(Liu et al., 2015)

MC Dropout SSD
(Miller et al., 2018)

MC Dropout
(Gal et al., 2017)

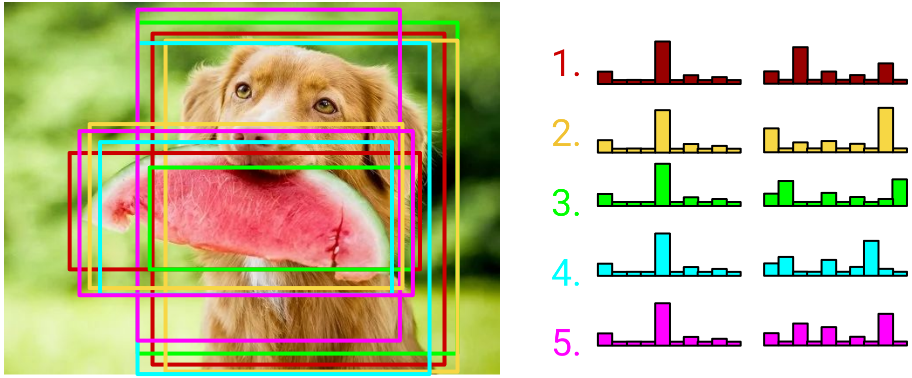


(Image: Srivastava et al., 2014)

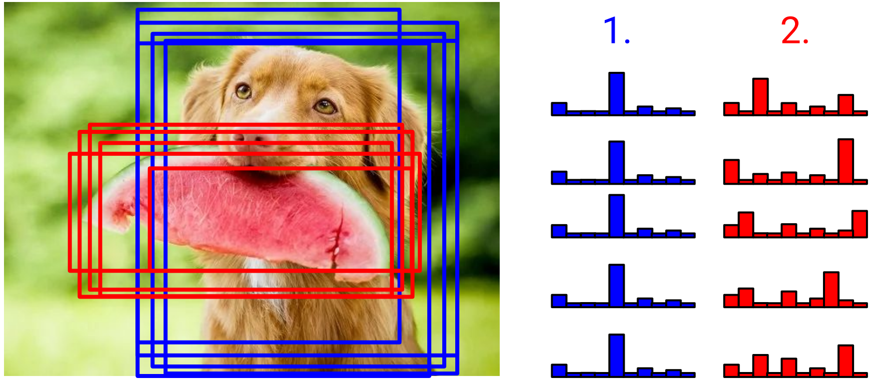
Slide courtesy of Dimity Miller

Uncertainty from Object Detection

MC Dropout SSD (Dropout Sampling for Robust Object Detection in Open-Set Conditions. Miller et al., ICRA 2018)



1. Sample from MC Dropout SSD

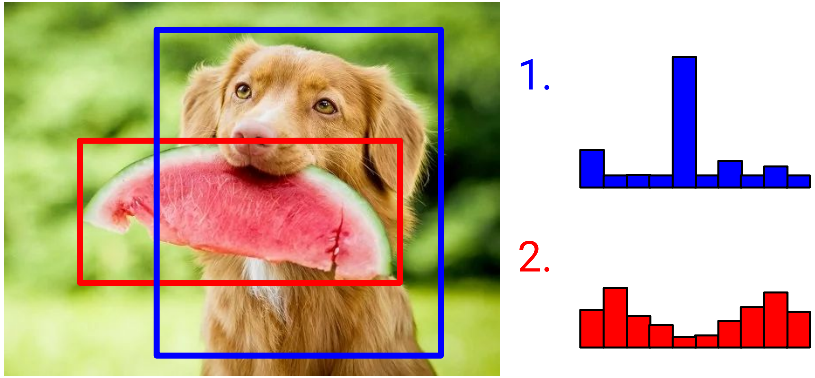


2. Group samples into observations

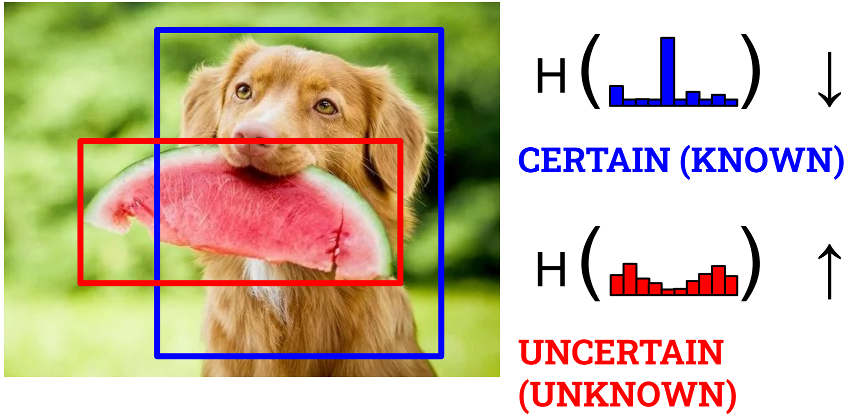
Slide courtesy of Dimity Miller

Uncertainty from Object Detection

MC Dropout SSD (Dropout Sampling for Robust Object Detection in Open-Set Conditions. Miller et al., ICRA 2018)



3. Form final detections



4. Obtain class uncertainty for detections

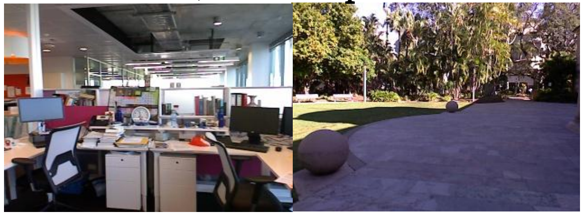
Slide courtesy of Dimity Miller

Uncertainty from Object Detection

MC Dropout SSD (Dropout Sampling for Robust Object Detection in Open-Set Conditions. Miller et al., ICRA 2018)

SceneNet RGB-D

QUT Campus

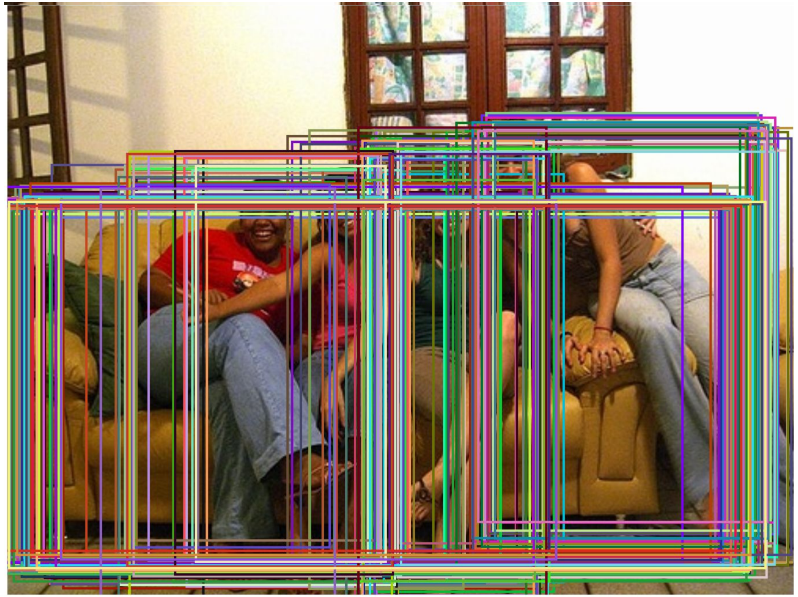


Uncertainty from MC Dropout SSD reduces open-set errors.

Slide courtesy of Dimity Miller

Evaluating Uncertainty from Object Detection

Evaluating Merging Strategies for Sampling-based Uncertainty Techniques in Object Detection
(Miller et al., 2019)



Measure affinity between samples

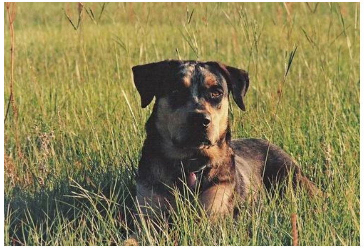
+

Clustering algorithm

Slide courtesy of Dimity Miller

Evaluating Uncertainty from Object Detection

Evaluating Merging Strategies for Sampling-based Uncertainty Techniques in Object Detection
(Miller et al., 2019)

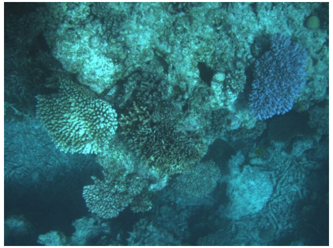


Closed-set Conditions

PASCAL VOC Dataset

Near Open-set Conditions

COCO Dataset



Distant Open-set Conditions

Underwater Dataset

Slide courtesy of Dimity Miller

Evaluating Uncertainty from Object Detection

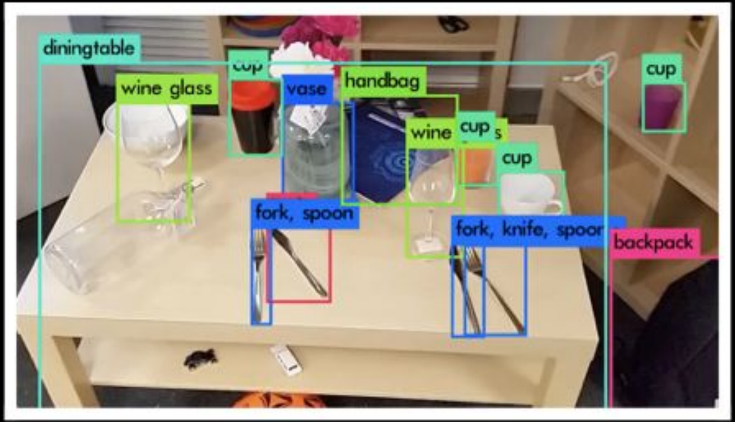
Evaluating Merging Strategies for Sampling-based Uncertainty Techniques in Object Detection (Miller et al., 2019)

Error represented by:	**Closed-Set Dataset** (Correct Detections & Closed-Set Error)				**Distant Open-Set Dataset** (Correct Detections & Distant OSE)				**Near Open-Set Dataset** (Correct Detections & Near OSE)				**All Datasets** (All detections)			
	UE ↓ (maP) (↑)	AUROC ↑	AUPR In ↑	AUPR Out ↑	UE ↓ (maP) (↑)	AUROC ↑	AUPR In ↑	AUPR Out ↑	UE ↓ (maP) (↑)	AUROC ↑	AUPR In ↑	AUPR Out ↑	UE ↓ (maP) (↑)	AUROC ↑	AUPR In ↑	AUPR Out ↑
Standard SSD	22.7(50.4)	84.1	**96.8**	48.4	16.2(61.7)	91.3	98.8	70.6	23.5(50.4)	85.1	98.0	52.5	21.6(53.0)	86.5	94.0	75.5
BSAS IoU 0.95	22.2(54.2)	84.8	96.7	51.0	10.5(59.6)	95.2	**99.2**	83.8	19.5(56.6)	88.5	**98.5**	58.8	18.6(56.6)	89.4	**94.8**	82.0
HDBScan Corner	21.3(53.7)	84.8	96.5	51.5	12.7(59.6)	94.0	99.0	79.6	22.7(56.2)	85.5	98.0	54.8	19.8(56.2)	88.0	94.0	79.4
Hungarian Exponential & SL	**20.1(55.1)**	**86.5**	96.5	**58.2**	10.8(60.4)	95.0	99.1	84.1	21.1(60.4)	7.4	98.1	59.5	18.3(56.7)	89.7	94.2	83.7
BSAS IoU 0.95 & SL	21.6(54.2)	85.5	96.6	55.0	9.9(59.6)	**95.4**	**99.2**	**86.4**	17.8(56.6)	**90.0**	98.4	**66.7**	**17.5(56.6)**	**90.3**	94.6	**85.1**
BSAS excl. IoU 0.9 & SL	20.7(55.9)	86.2	96.6	57.9	**10.3(61.8)**	95.2	99.1	85.7	20.2(61.8)	87.9	98.1	62.7	18.2(58.0)	89.9	94.2	84.7

Basic Sequential Algorithmic Scheme (BSAS) clustering using Intersection over Union (IoU) and winning label (SL) as affinity measures.

Slide courtesy of Dimity Miller

Perception



World Model & Decision Making

Interaction

Propagate uncertainty from Perception through the world model into decision making and actions?

Probabilistic Object Detection