SEOUL

19.09.26

DEV DAY

aws

DEV DAY
SEOUL

19.09.26

모두를 위한 컴퓨터 비전 딥러닝 툴킷, GluonCV 따라하기

2-2. GluonCV Overview

강지양 딥러닝 아키텍트
Amazon Machine Learning Solutions Lab

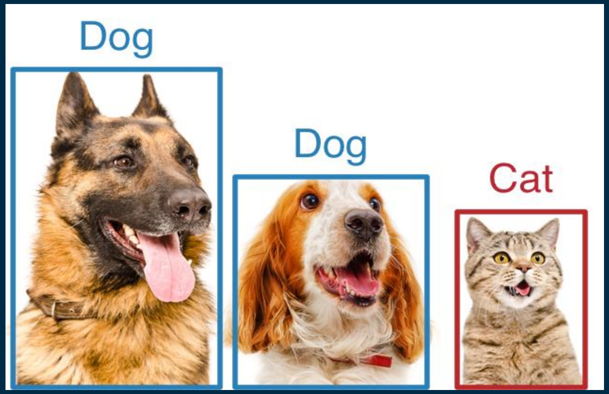
GluonCV: A Vision Toolkit

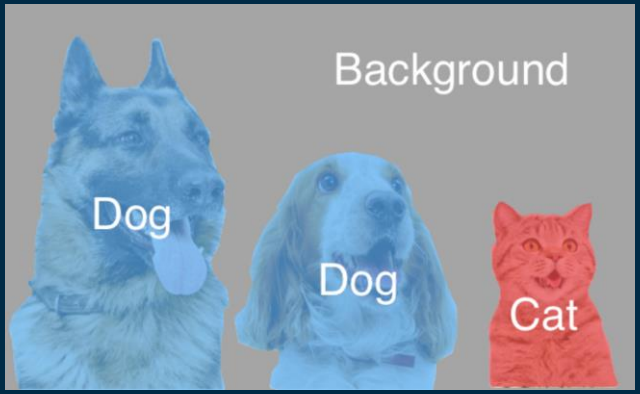
- State-of-the-Art Models
- Fast Development
- Easy Deployment
- Official Maintenance

GluonCV Demos
https://www.youtube.com/watch?v=nfpouVAzXt0

Models

- Classification
- Detection
- Segmentation

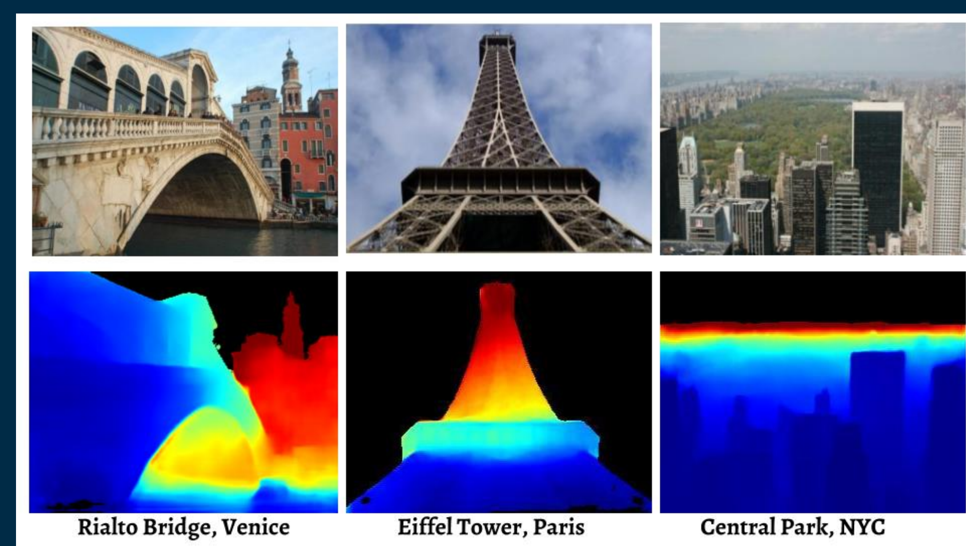




Model Zoo
https://gluon-cv.mxnet.io/model_zoo/

Models

- Available
  - Classification
  - Detection
  - Segmentation
  - Pose Estimation
  - Action Recognition
- In-Development
  - Keypoint detection
  - Depth prediction



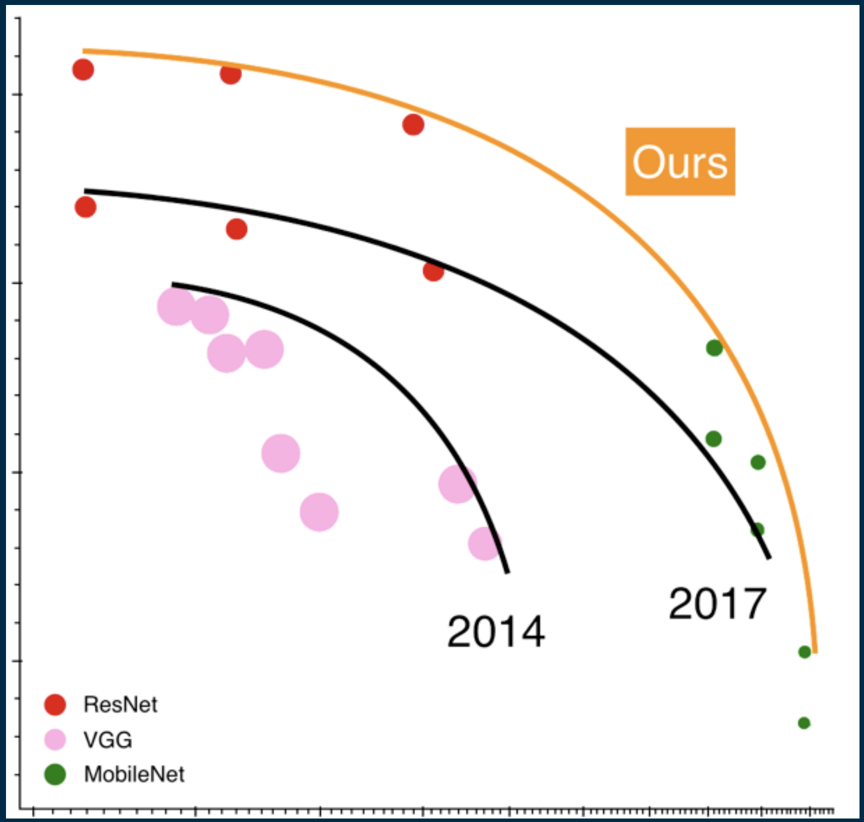
Model Zoo
https://gluon-cv.mxnet.io/model_zoo/

Classification with GluonCV

GluonCV Model Zoo

He, Tong, et al. "Bag of Tricks for Image Classification with Convolutional Neural Networks" arXiv preprint arXiv:1812.01187 (2018).

Model	Ours	Reference
ResNet-50	79.2%	76.2%
ResNet-101	80.5%	77.4%
MobileNet	73.3%	70.9%



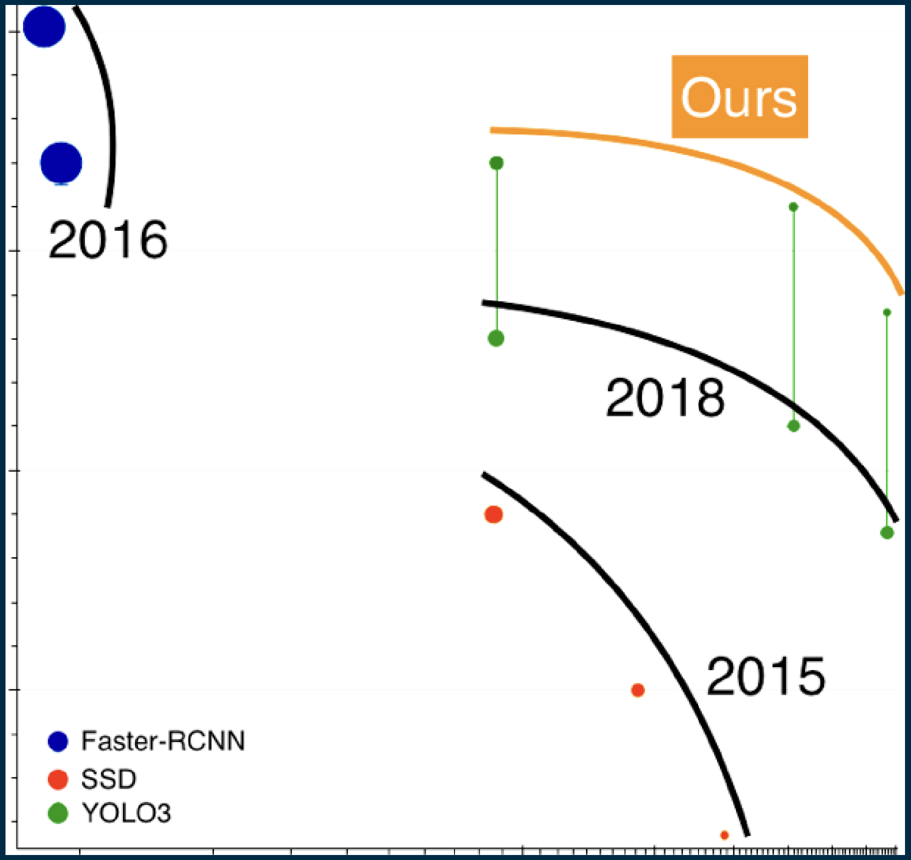
https://gluon-cv.mxnet.io/model_zoo/classification.html

Object Detection with GluonCV

GluonCV Model Zoo

Paper under review,
to be released soon

Model	Ours	Reference
Faster-RCNN	40.1%	39.6%
YOLOv3	37.0%	33.0%



https://gluon-cv.mxnet.io/model_zoo/detection.html

Bag of Tricks for Image Classification with Convolutional Neural Networks

Tong He Zhi Zhang Hang Zhang Zhongyue Zhang Junyuan Xie Mu Li

Amazon Web Services
{htong,zhiz,hzaws,zhongyue,junyuanx,mli}@amazon.com

Abstract

Much of the recent progress made in image classification research can be credited to training procedure refinements, such as changes in data augmentations and optimization methods. In the literature, however, most refinements are either briefly mentioned as implementation details or only visible in source code. In this paper, we will examine a collection of such refinements and empirically evaluate their impact on the final model accuracy through ablation study. We will show that, by combining these refinements together, we are able to improve various CNN models significantly. For example, we raise ResNet-50's top-1 validation accuracy from 75.3% to 79.29% on ImageNet. We will also demon-

Model	FLOPs	top-1	top-5
ResNet-50 [9]	3.9 G	75.3	92.2
ResNeXt-50 [27]	4.2 G	77.8	-
SE-ResNet-50 [12]	3.9 G	76.71	93.38
SE-ResNeXt-50 [12]	4.3 G	78.90	94.51
DenseNet-201 [13]	4.3 G	77.42	93.66
ResNet-50 + tricks (ours)	4.3 G	**79.29**	**94.63**

Table 1: **Computational costs and validation accuracy of various models.** ResNet, trained with our "tricks", is able to outperform newer and improved architectures trained with standard pipeline.

https://arxiv.org/abs/1812.01187

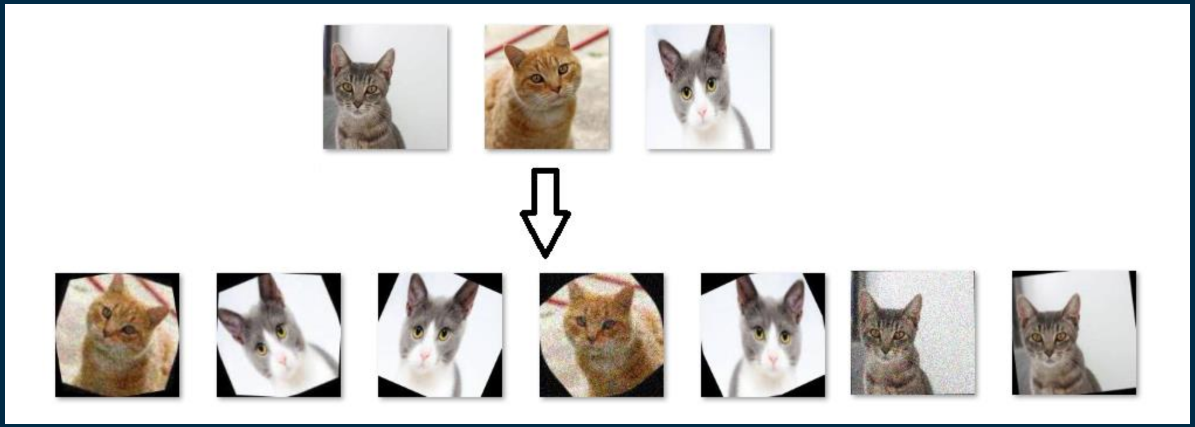
Classification with GluonCV

Training Essentials

- Data Preprocessing
- Network architecture definition
- Optimizer
- Loss
- Metric
- GPU Acceleration

Data Transformation with GluonCV

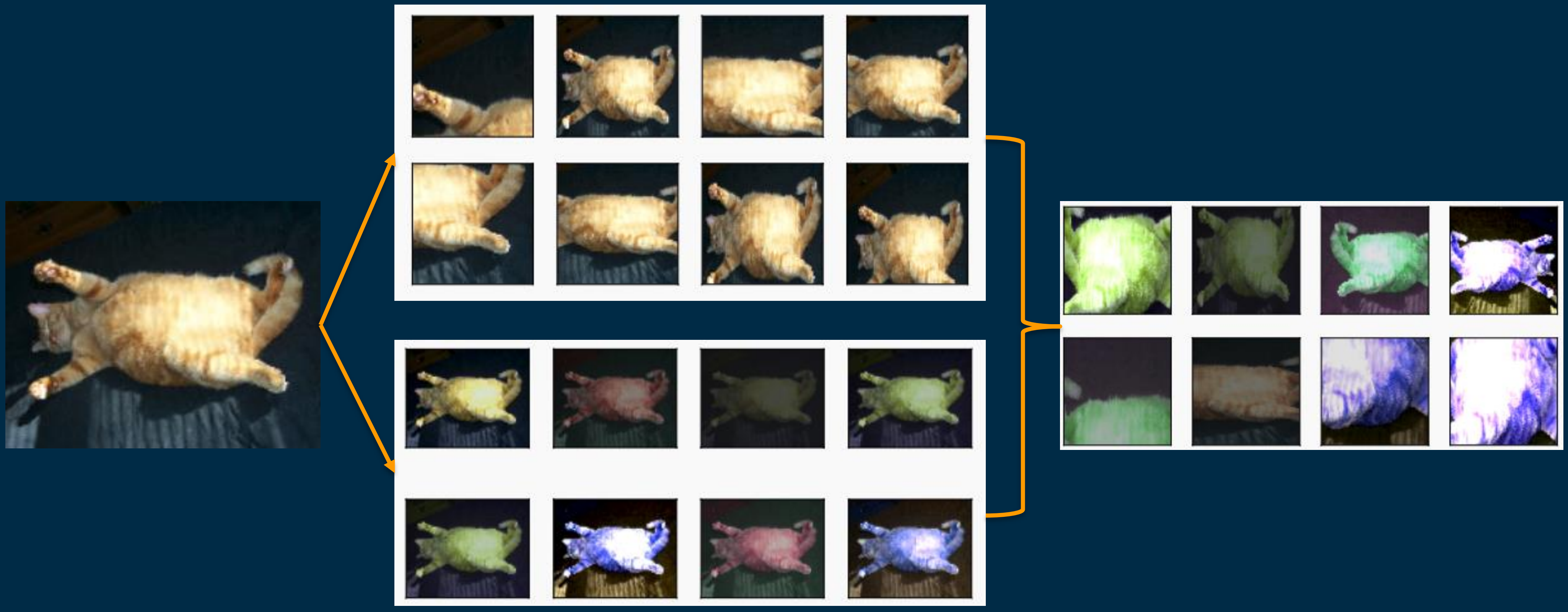
- Popular Transformation
  - Resize
  - Crop
  - Flip
  - Rotation
  - Adding Noise
  - Normalization



https://gluon-cv.mxnet.io/api/data.transforms.html

Classification with GluonCV

Data Preprocessing

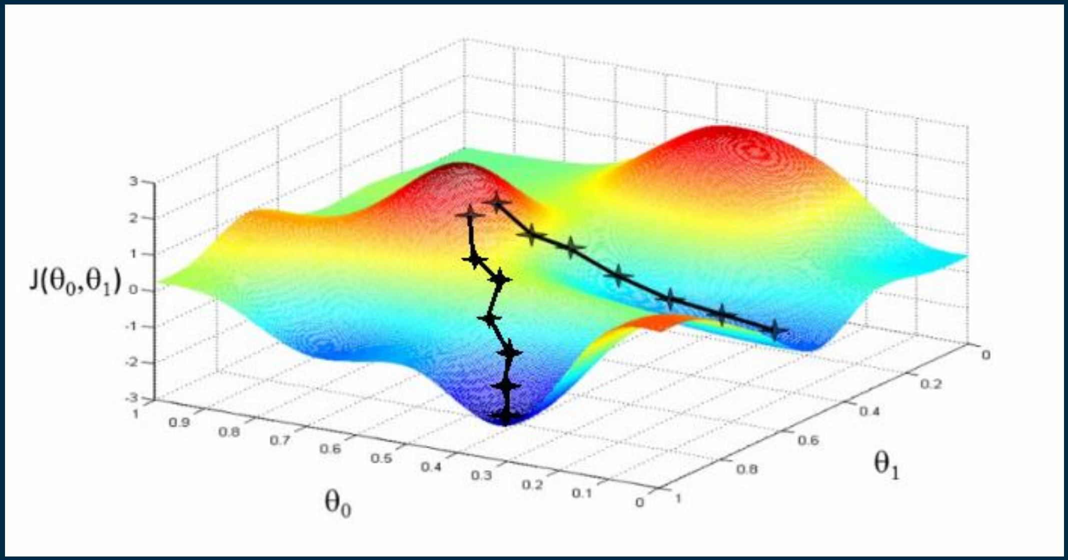


https://gluon-cv.mxnet.io/api/data.transforms.html

Classification with GluonCV

Optimizers

- SGD
- Adam
- RMSProp
- …



https://beta.mxnet.io/api/gluon-related/mxnet.optimizer.html

Model Tweaks

Minor Adjustment to the ResNet-50 Network

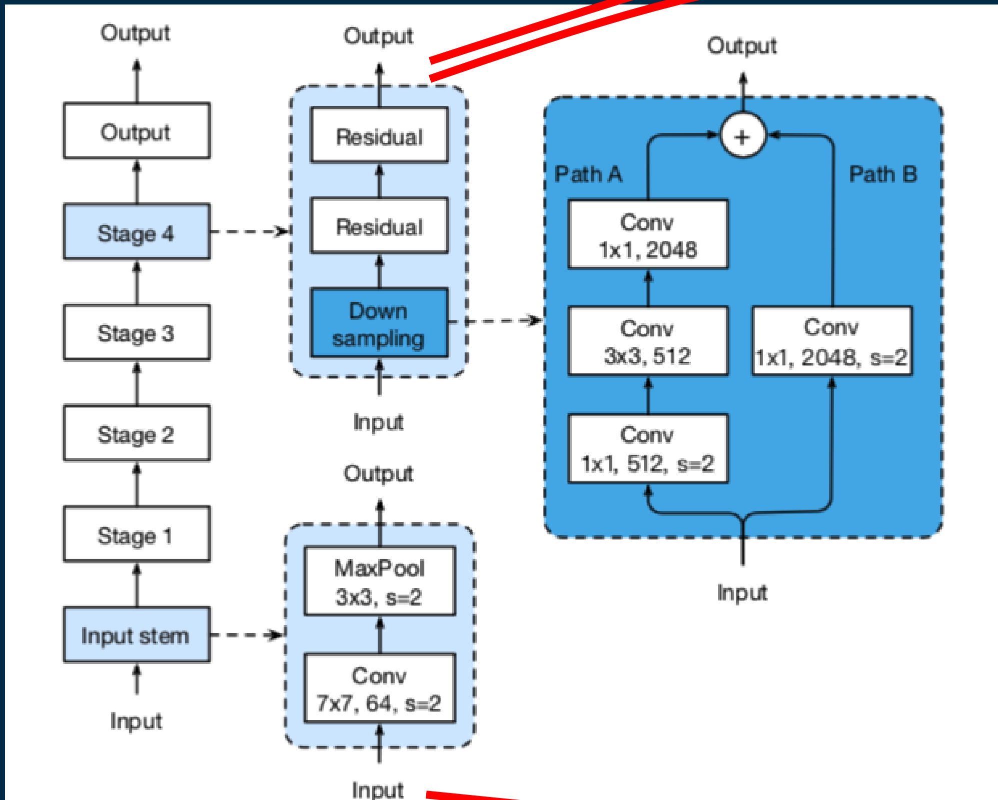


Figure 1: The architecture of ResNet-50. The convolution kernel size, output channel size and stride size (default is 1) are illustrated, similar for pooling layers.

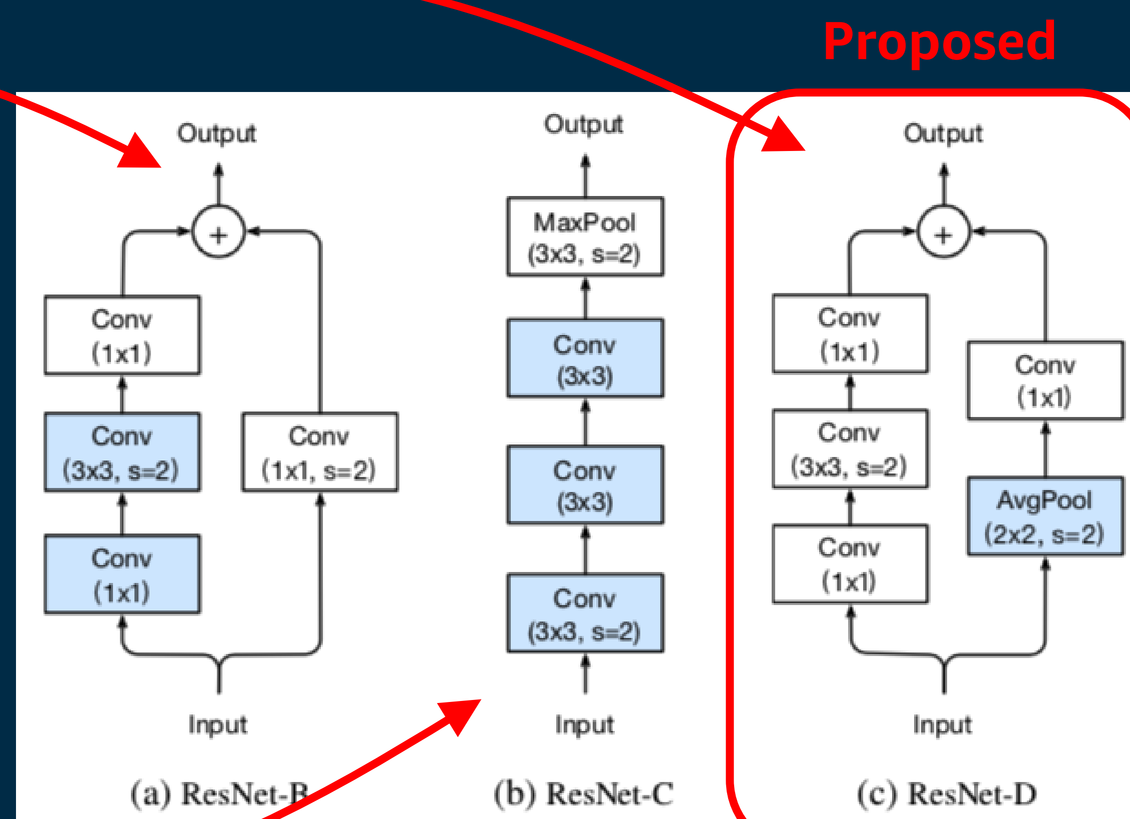


Figure 2: Three ResNet tweaks. ResNet-B modifies the downsampling block of Resnet. ResNet-C further modifies the input stem. On top of that, ResNet-D again modifies the downsampling block.

Training Refinements

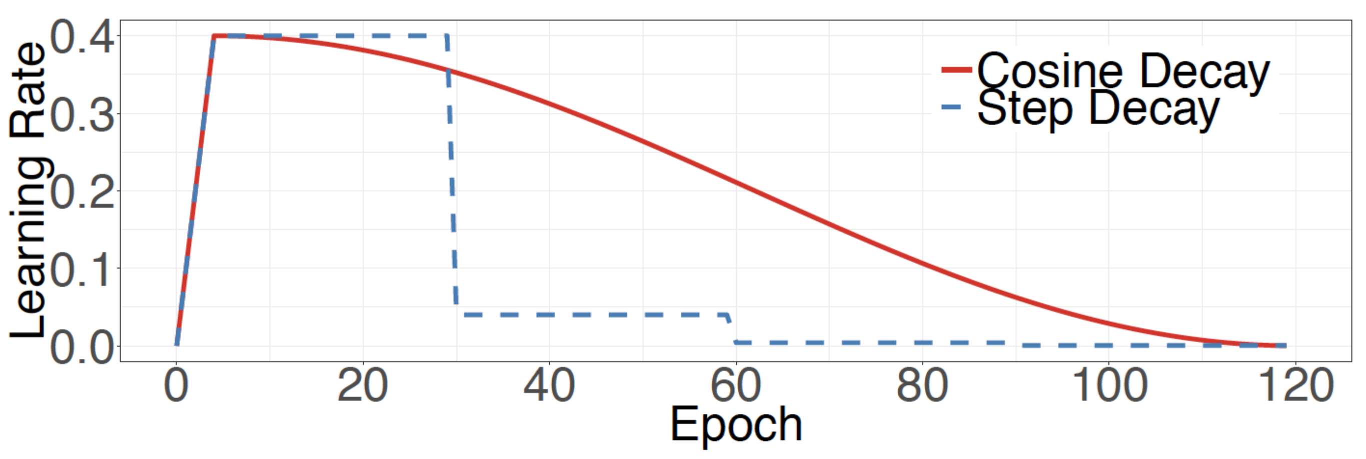
Advanced Tricks

- Label smoothing
- Learning rate schedule
- Mix-Up
- Knowledge Distillation

Training Refinements

Cosine Learning Rate Decay

$$\eta_t = \frac{1}{2}\left(1 + \cos\left(\frac{t\pi}{T}\right)\right)\eta$$



(a) Learning Rate Schedule

SGDR: Stochastic Gradient Descent with Warm Restarts

https://arxiv.org/abs/1608.03983

Training Refinements

Label Smoothing

- One hot: (0, 1, 0, 0, 0)
- Smoothed: (0.01, 0.96, 0.01, 0.01, 0.01)
- Prevent overfitting!

$$q_i = \begin{cases} 1 - \varepsilon & \text{if } i = y, \\ \varepsilon/(K-1) & \text{otherwise,} \end{cases}$$

Rethinking the Inception Architecture for Computer Vision
https://arxiv.org/abs/1512.00567

Training Refinements

Knowledge Distillation

- Dark Knowledge
    - Dog vs Cat
    - Dog vs Car

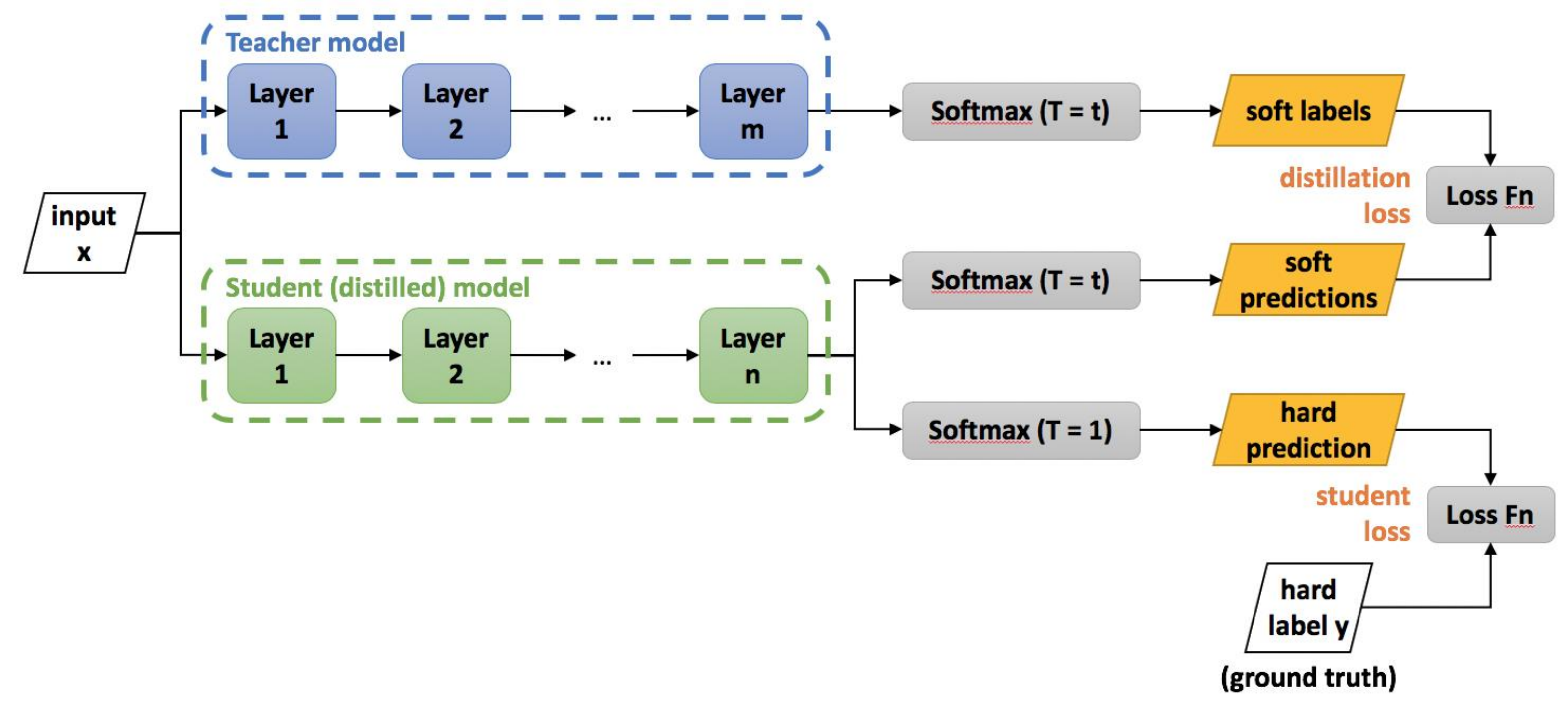
cow	dog	cat	car	
0	1	0	0	original hard targets
10^{-6}	.9	.1	10^{-9}	output of geometric ensemble
.05	.3	.2	.005	softened output of ensemble

$$q_i = \frac{exp(z_i/T)}{\sum_j exp(z_j/T)}$$

Distilling the Knowledge in a Neural Network
https://arxiv.org/abs/1503.02531

Training Refinements

Knowledge Distillation



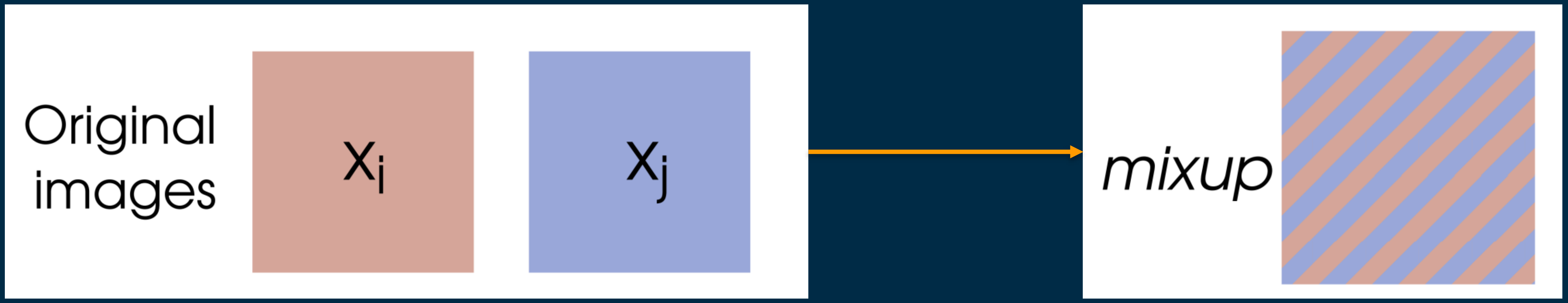
$$\ell(p, \mathrm{softmax}(z)) + T^2 \ell(\mathrm{softmax}(r/T), \mathrm{softmax}(z/T))$$

Distilling the Knowledge in a Neural Network
https://arxiv.org/abs/1503.02531

Training Refinements

Mix-Up

- Linear mapping
- $f(ax\_i+bx\_j\ )=af(x\_i\ )+bf(x\_j)$

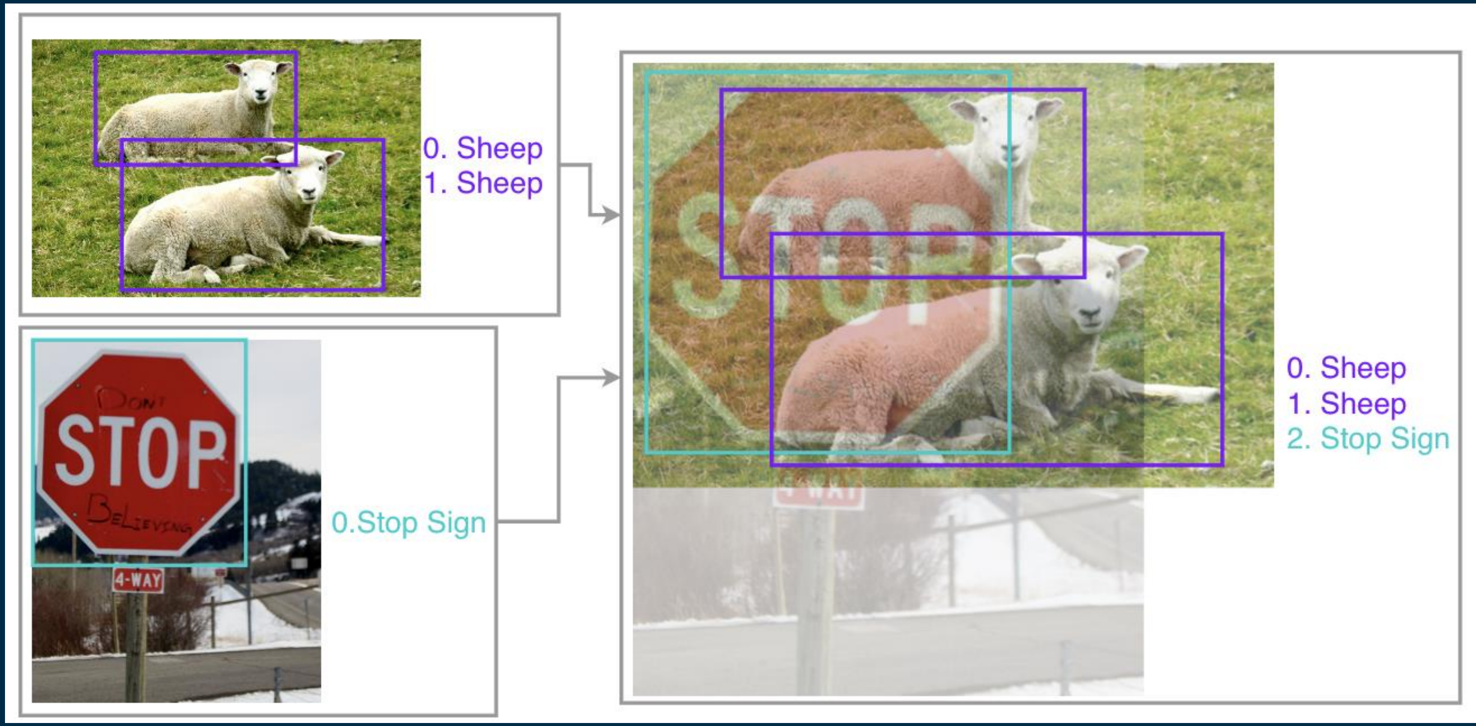


mixup: Beyond Empirical Risk Minimization
https://arxiv.org/abs/1710.09412

Training Refinements

Mix-Up

mixup: Beyond Empirical Risk Minimization
https://arxiv.org/abs/1710.09412

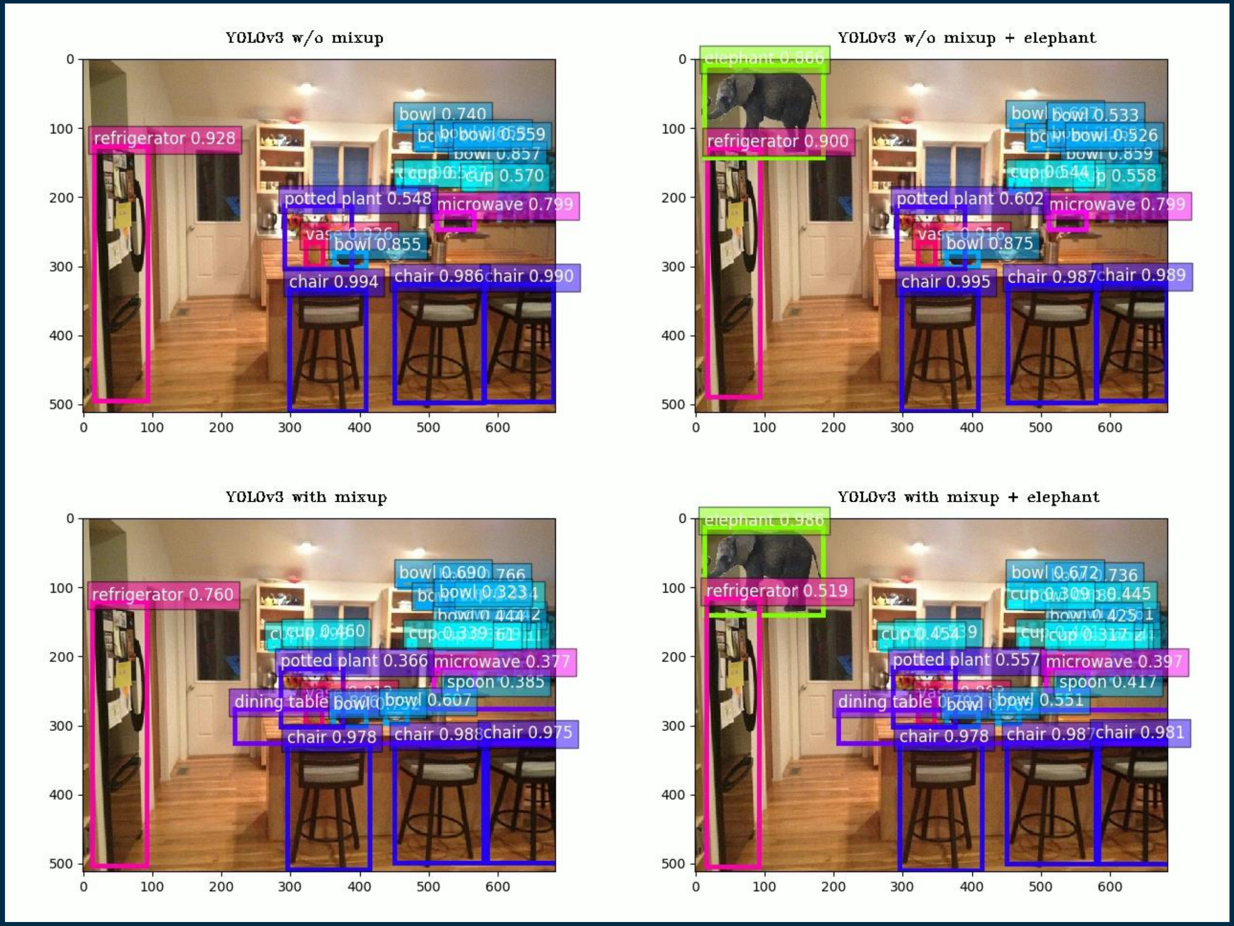


$$
\begin{aligned}
\hat{x} &= \lambda x_i + (1-\lambda) x_j, \\
\hat{y} &= \lambda y_i + (1-\lambda) y_j,
\end{aligned}
$$

Training Refinements

Elephant-in-the-Room

common failures of state-of-the art object detectors
https://arxiv.org/abs/1808.03305



https://www.youtube.com/watch?v=qcm3lL4PCC4

Refinements	ResNet-50-D		Inception-V3		MobileNet	
	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
Efficient	77.16	93.52	77.50	93.60	71.90	90.53
+ cosine decay	77.91	93.81	78.19	94.06	72.83	91.00
+ label smoothing	78.31	94.09	78.40	94.13	72.93	91.14
+ distill w/o mixup	78.67	94.36	78.26	94.01	71.97	90.89
+ mixup w/o distill	79.15	94.58	**78.77**	**94.39**	**73.28**	**91.30**
+ distill w/ mixup	**79.29**	**94.63**	78.34	94.16	72.51	91.02

Table 6: The validation accuracies on ImageNet for stacking training refinements one by one.

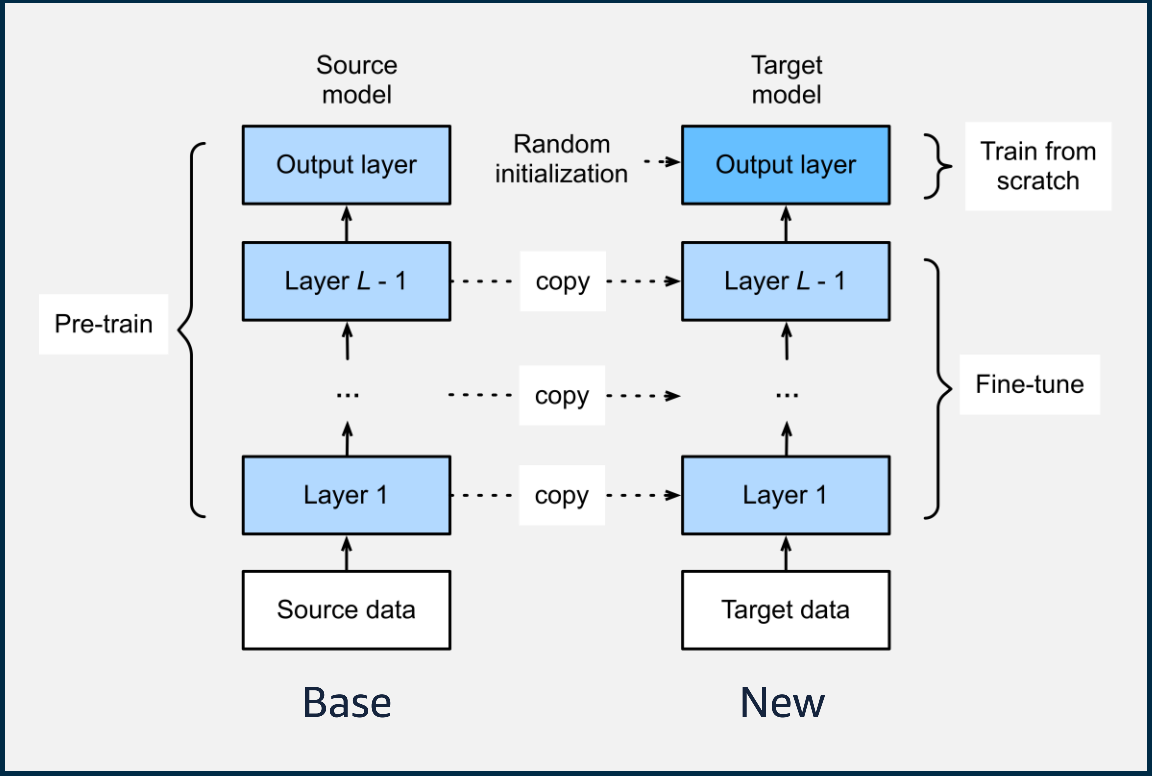
https://arxiv.org/abs/1812.01187

Transfer Learning

Three major scenarios

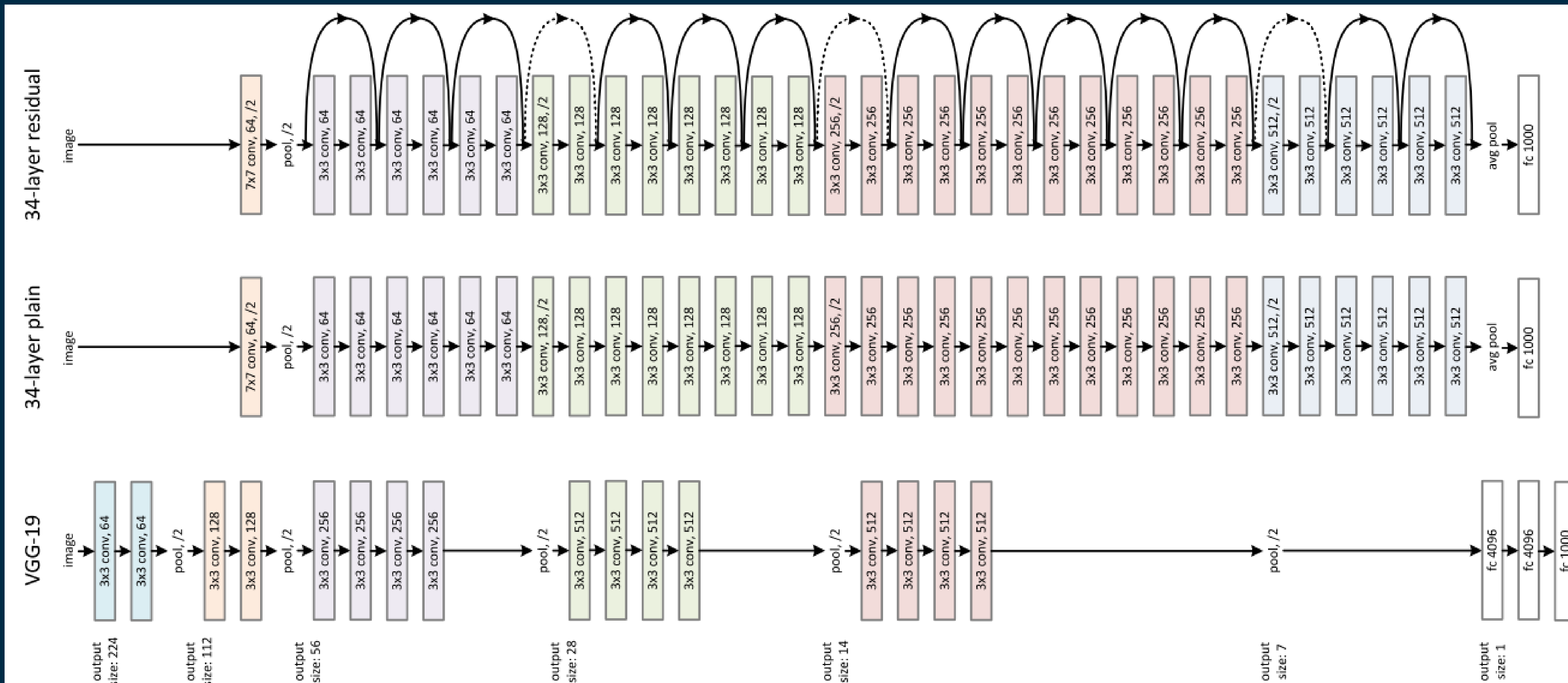
http://cs231n.github.io/transfer-learning/

- Pretrained models: Model Zoo
- Fixed feature extractor
- Fine-tuning



https://gluon-cv.mxnet.io/build/examples_classification/transfer_learning_minc.html

ResNet

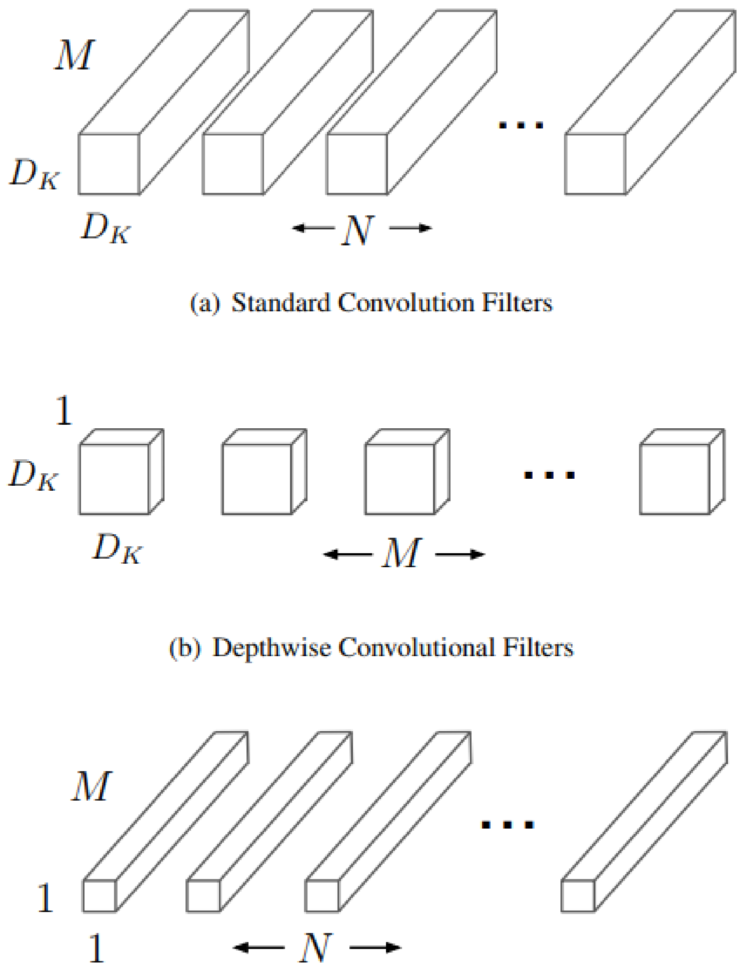


This result won the 1st place in the ImageNet localization task in *ILSVRC 2015*.

Deep Residual Learning for Image Recognition

https://arxiv.org/abs/1512.03385

MobileNet



(a) Standard Convolution Filters

(b) Depthwise Convolutional Filters

(c) 1×1 Convolutional Filters called Pointwise Convolution in the context of Depthwise Separable Convolution

Table 1. MobileNet Body Architecture

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
$5\times$ Conv dw / s1 Conv / s1	$3 \times 3 \times 512$ dw $1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$ $14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

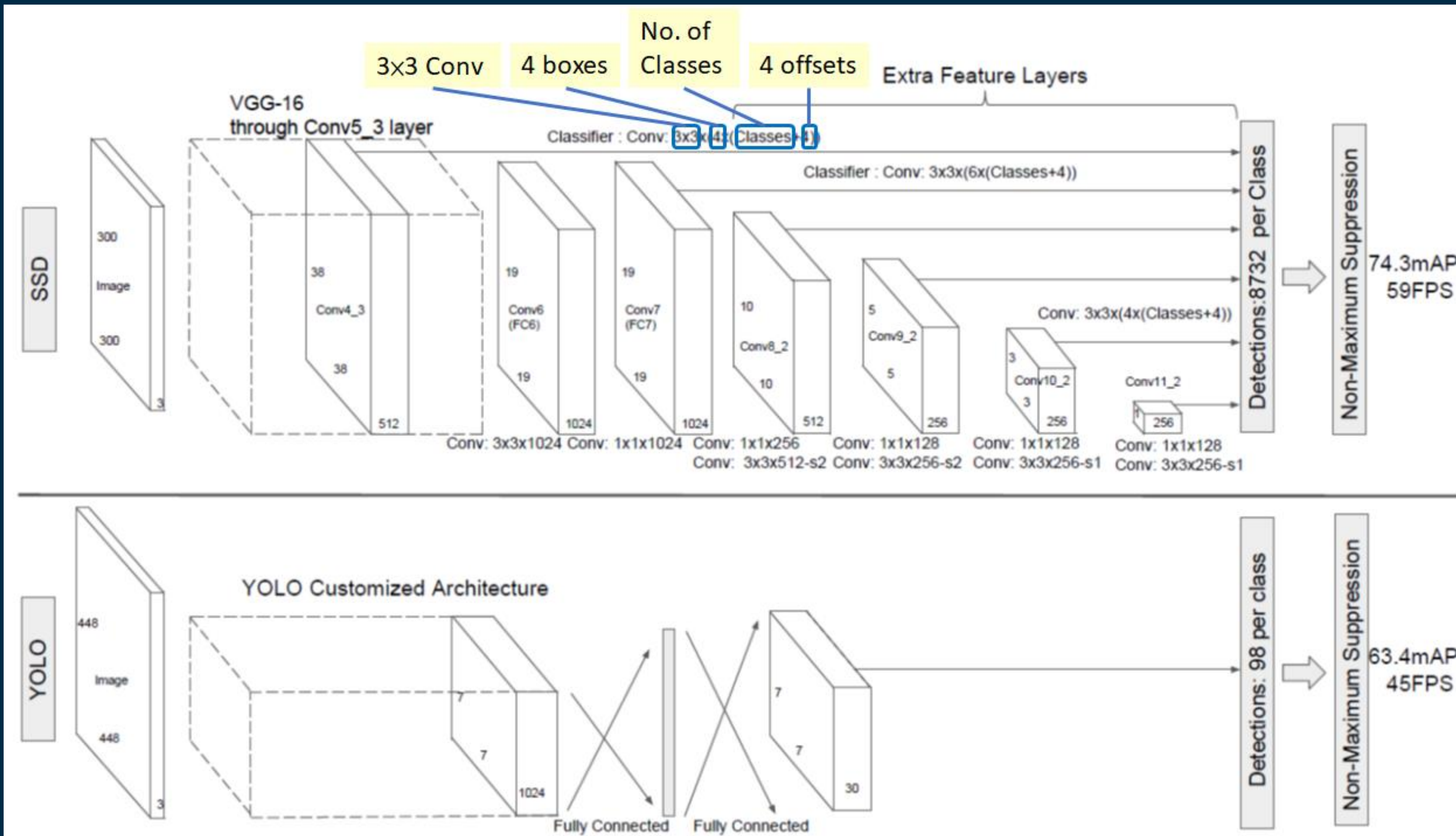
MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications

https://arxiv.org/abs/1704.04861

SSD

SSD: Single Shot MultiBox Detector

https://arxiv.org/abs/1512.02325



YOLO

You Only Look Once: Unified, Real-Time Object Detection

https://arxiv.org/abs/1506.02640

GluonC: The Best Open-Source Choice

- Pretrained Models with the Best Accuracy
- Most Comprehensive Model Zoo

GluonCV
https://gluon-cv.mxnet.io/

GluonCV GitHub Repo
https://github.com/dmlc/gluon-cv

Getting Started

- Gluon CV

  https://gluon-cv.mxnet.io

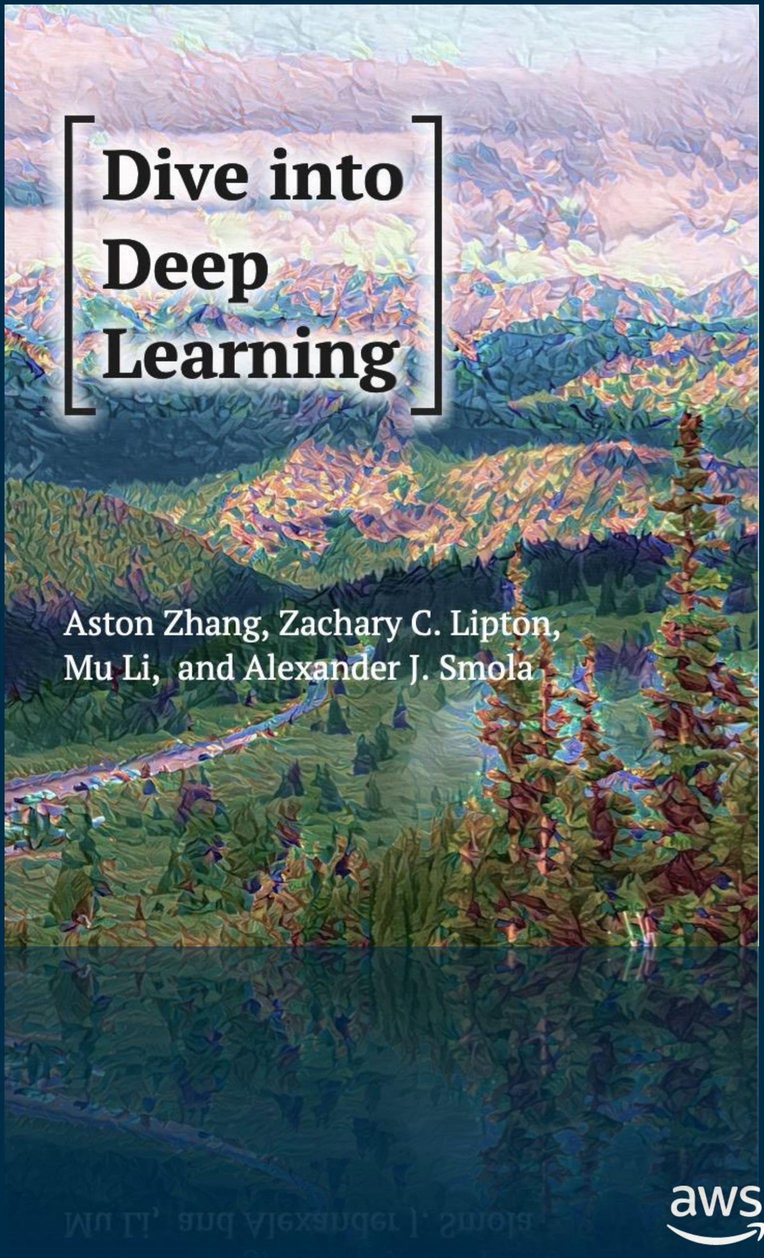
- MXNet

  http://beta.mxnet.io/

- Dive into Deep Learning

  http://d2l.ai/

  https://github.com/d2l-ai/d2l-ko



Thank you!

DEV DAY
SEOUL

19.09.26

여러분의 피드백을 기다립니다!

강연 평가 및 설문 조사
QR 코드를 통해 AWS DEV DAY SEOUL에 대한 여러분의 의견을 공유해주세요.
강연 평가 및 설문 조사에 참여해 주신 분께는 등록데스크에서 특별한 기념품을 드립니다.

강연 영상
AWS DEV DAY SEOUL 강연 영상은 행사 종료 후 메일로 공유드릴 예정입니다.

#AWSDEVDAYSEOUL
소셜미디어에 행사 참여 소감을 공유해주세요!

aws