Enabling Deep Learning in IoT Applications with Apache MXNet

Julien Simon
Principal AI/ML Evangelist, EMEA
@julsimon

March 2018

Agenda

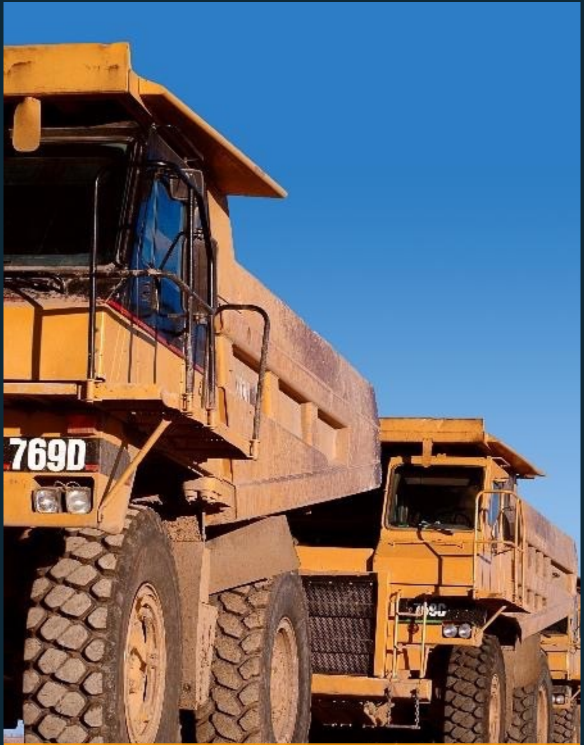
- Deep Learning at the Edge?
- Apache MXNet
- Predicting in the Cloud or at the Edge?
- AWS DeepLens
- Getting started

Services covered: Apache MXNet, Deep Learning AMI, Amazon SageMaker, AWS IoT, AWS Greengrass (ML), AWS DeepLens

Deep Learning at the Edge?

Most machine data never reaches the cloud

Medical equipment

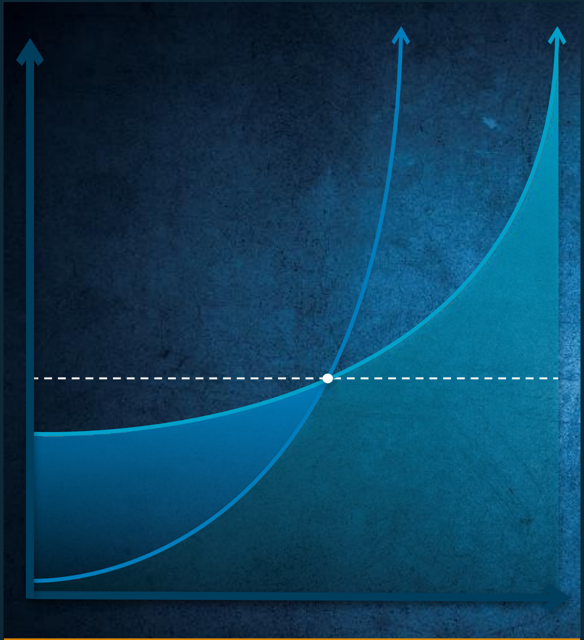


Industrial machinery

Extreme environments

Why this problem isn't going away

Law of physics



Law of economics

Law of the land

Deep Learning at the Edge

- Capturing data at the Edge and sending it to the Cloud is a good start.
- We can use it for analytics, model training, etc.
- Let’s see how we could close the loop and use predictive models at the Edge.

Deep Learning challenges at the Edge

- Resource-constrained devices
  - CPU, memory, storage, power consumption.
- Network connectivity
  - Availability, cost, bandwidth, latency.
  - On-device prediction may be the only option.
- Deployment
  - Updating code and models on a fleet of devices is not easy.

Deep Learning wishlist at the Edge

- Rely on cloud-based services for seamless training and deployment.
- Have the option to use cloud-based prediction.
- Be able to run device-based prediction with good performance.
- Support different technical environments (CPUs, languages).

Apache MXNet

Apache MXNet: Open Source library for Deep Learning

Programmable

Simple syntax,
multiple
languages

Portable

Highly efficient
models for
mobile
and IoT

High Performance

Near linear scaling
across hundreds of
GPUs

Most Open

Accepted into the
Apache Incubator

Best On AWS

Optimized for
Deep Learning on AWS

Apache MXNet for IoT

1. Flexible experimentation in the Cloud.
2. Scalable training in the Cloud.
3. Good prediction performance at the Edge.
4. Prediction in the Cloud or at the Edge.

1 - Flexible experimentation in the Cloud

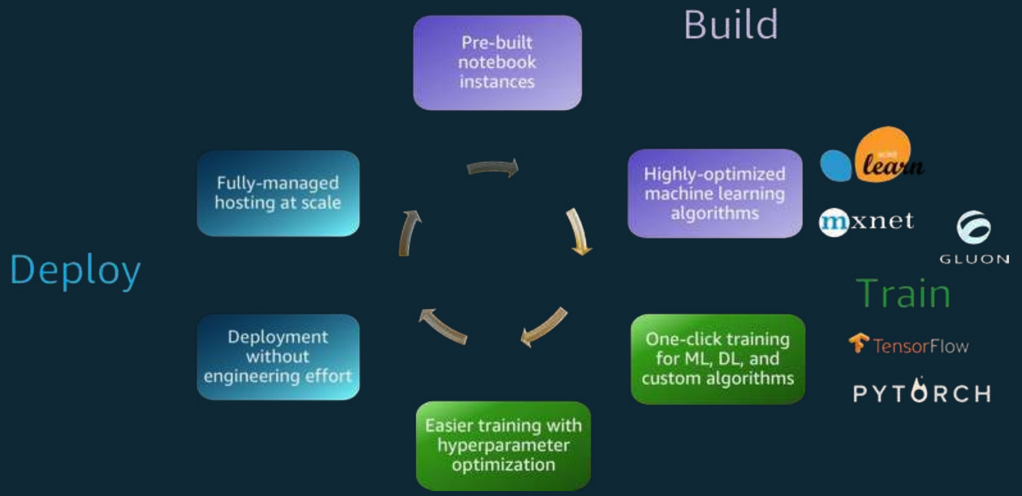
- API for Python, R, Perl, Matlab, Scala, C++.

- Gluon
  - Imperative programming aka 'define-by-run'.
  - Inspect, debug and modify models during training.

- Extensive model zoo
  - Pre-trained computer vision models.
  - DenseNet, SqueezeNet for resource-constrained devices.

2 - Scalable training in the Cloud

Amazon SageMaker



Build
Pre-built notebook instances
Highly-optimized machine learning algorithms
Train
One-click training for ML, DL, and custom algorithms
Easier training with hyperparameter optimization
Deploy
Deployment without engineering effort
Fully-managed hosting at scale
learn
mxnet
GLUON
TensorFlow
PYTORCH

AWS Deep Learning AMI

TensorFlow

K

Microsoft Cognitive Toolkit

mxnet

GLUON

Caffe2

PYTORCH

Amazon EC2

c5

p3

aws

3 - Good prediction performance at the Edge

- MXNet is written in C++.
- Gluon networks can be 'hybridized' for additional speed.
- Two libraries boost performance on CPU-only devices
    - Fast implementation of math primitives
    - Hardware-specific instructions, e.g. Intel AVX or ARM NEON
    - Intel Math Kernel Library https://software.intel.com/en-us/mkl
    - NNPACK https://github.com/Maratyszcza/NNPACK
- Mixed precision training on GPUs
    - Use float16 instead of float32 for weights and activations
    - Almost 2x reduction in model size, no loss of accuracy
    - https://devblogs.nvidia.com/parallelforall/mixed-precision-training-deep-neural-networks/

4 - Predicting in the Cloud or at the Edge

- Cloud-based: invoke a Lambda function with AWS IoT.
- Cloud-based: invoke a SageMaker endpoint with HTTP.
- Device-based: bring your own code and model.
- Device-based: deploy your code and model with AWS Greengrass.

Invoking a Lambda function with AWS IoT

- Train a model in SageMaker (or bring your own).
- Host it in S3 (or embed it in a Lambda function).
- Write a Lambda function performing prediction.
- Invoke it through AWS IoT.

Best when
Devices can support neither HTTP nor local inference (e.g. Arduino).
Costs must be kept as low as possible.

Requirements
Network is available and reliable (MQTT is less demanding than HTTP).
Devices are provisioned in AWS IoT (certificate, keys).

https://aws.amazon.com/blogs/compute/seamlessly-scale-predictions-with-aws-lambda-and-mxnet/

Invoking a SageMaker endpoint with HTTP

- Train a model in SageMaker (or bring your own).
- Deploy it to a prediction endpoint.
- Invoke the HTTP endpoint from your devices.

Best when
Devices are not powerful enough for local inference.
Models can’t be easily deployed to devices.
Additional cloud-based data is required for prediction.
Prediction activity must be centralized.

Requirements
Network is available and reliable.
Devices support HTTP.

Bring your own code and model

- Train a model in SageMaker (or bring your own).
- Bring your own application code.
- Provision devices at manufacturing time (or use your own update mechanism).

Best when
You don't want to or can't rely on cloud services (no network connectivity?)

Requirements
Devices are powerful enough for local inference.
Models don't need to be updated, if ever.
DIY!

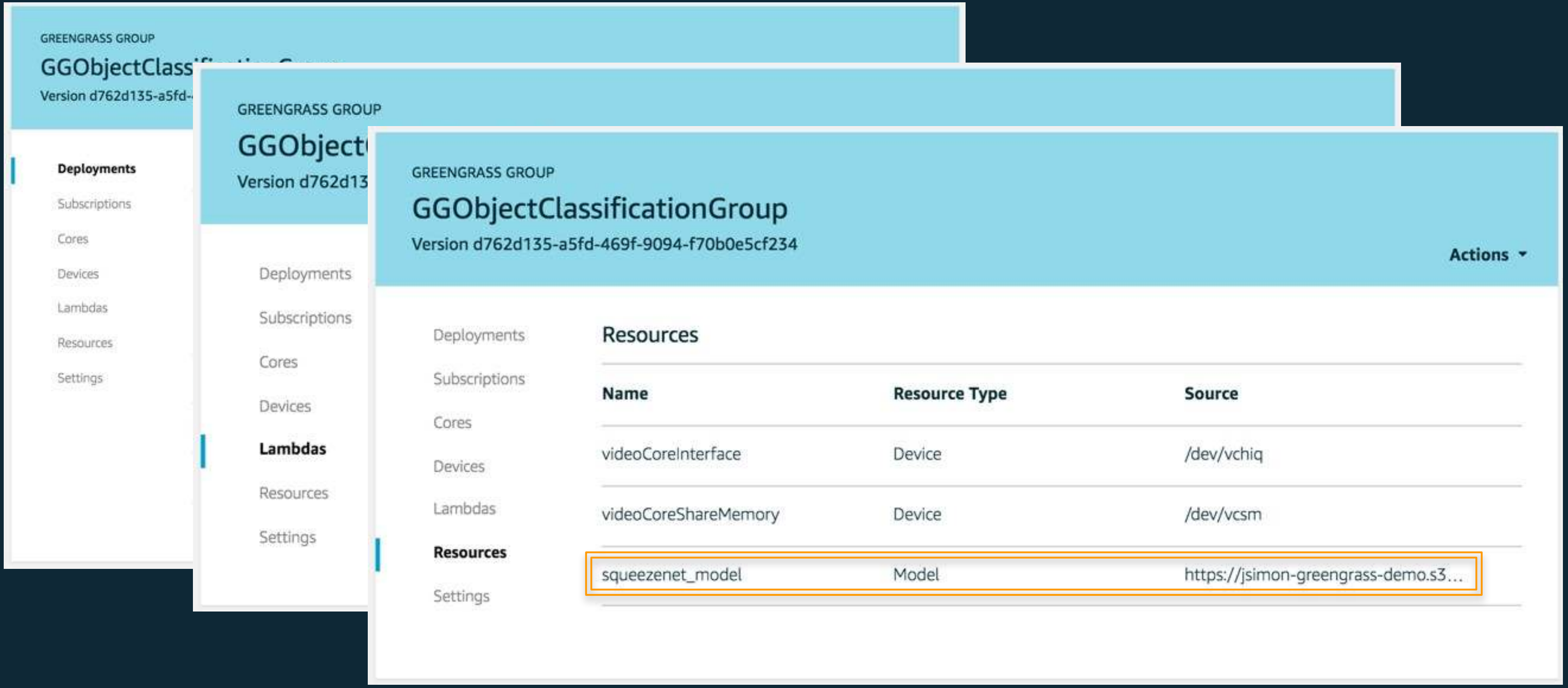
Deploy your code and model with AWS Greengrass

- Train a model in SageMaker (or bring your own).
- Write a Lambda function performing prediction.
- Add both as resources in your Greengrass group.
- Let Greengrass handle deployment and updates.

Best when
You want the same programming model in the Cloud and at the Edge.
Code and models need to be updated, even if network connectivity is infrequent or unreliable.
One device in the group should be able to perform prediction on behalf on other devices.

Requirements
Devices are powerful enough to run Greengrass (XXX HW requirements)
Devices are provisioned in AWS IoT (certificate, keys).

AWS Greengrass ML



GREENGRASS GROUP

GGObjectClassificationGroup

Version d762d135-a5fd-469f-9094-f70b0e5cf234

Resources

Name	Resource Type	Source
videoCoreInterface	Device	/dev/vchiq
videoCoreShareMemory	Device	/dev/vcsm
squeezenet_model	Model	https://jsimon-greengrass-demo.s3…

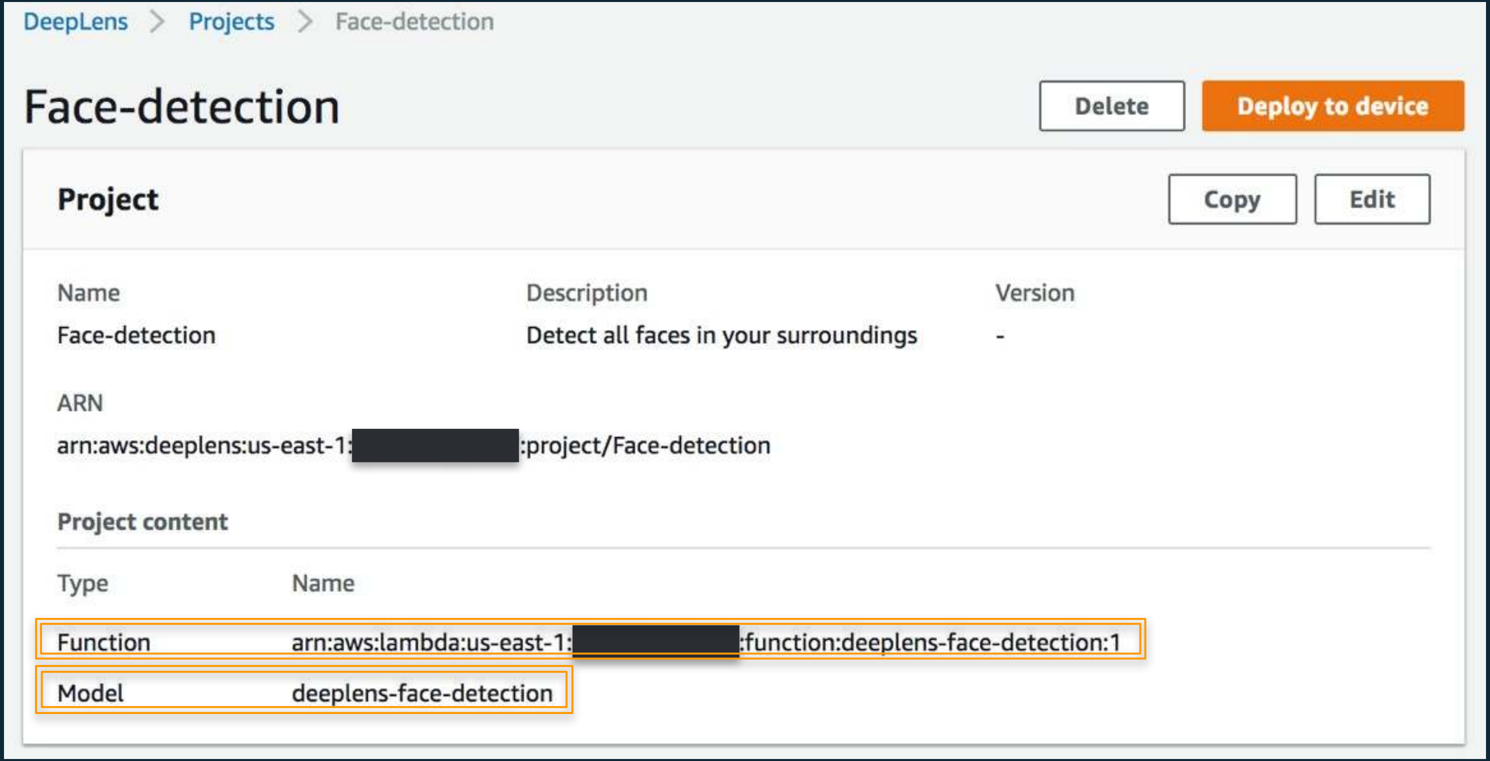
AWS DeepLens

AWS DeepLens

World's first Deep Learning enabled video camera for developers

- A new way to learn
- Custom built for Deep Learning
- Broad Framework Support
- Deploy models from Amazon SageMaker
- Integrated with AWS
- Fully programmable with AWS Lambda

AWS DeepLens



DeepLens > Projects > Face-detection

Name	Description	Version
Face-detection	Detect all faces in your surroundings	-

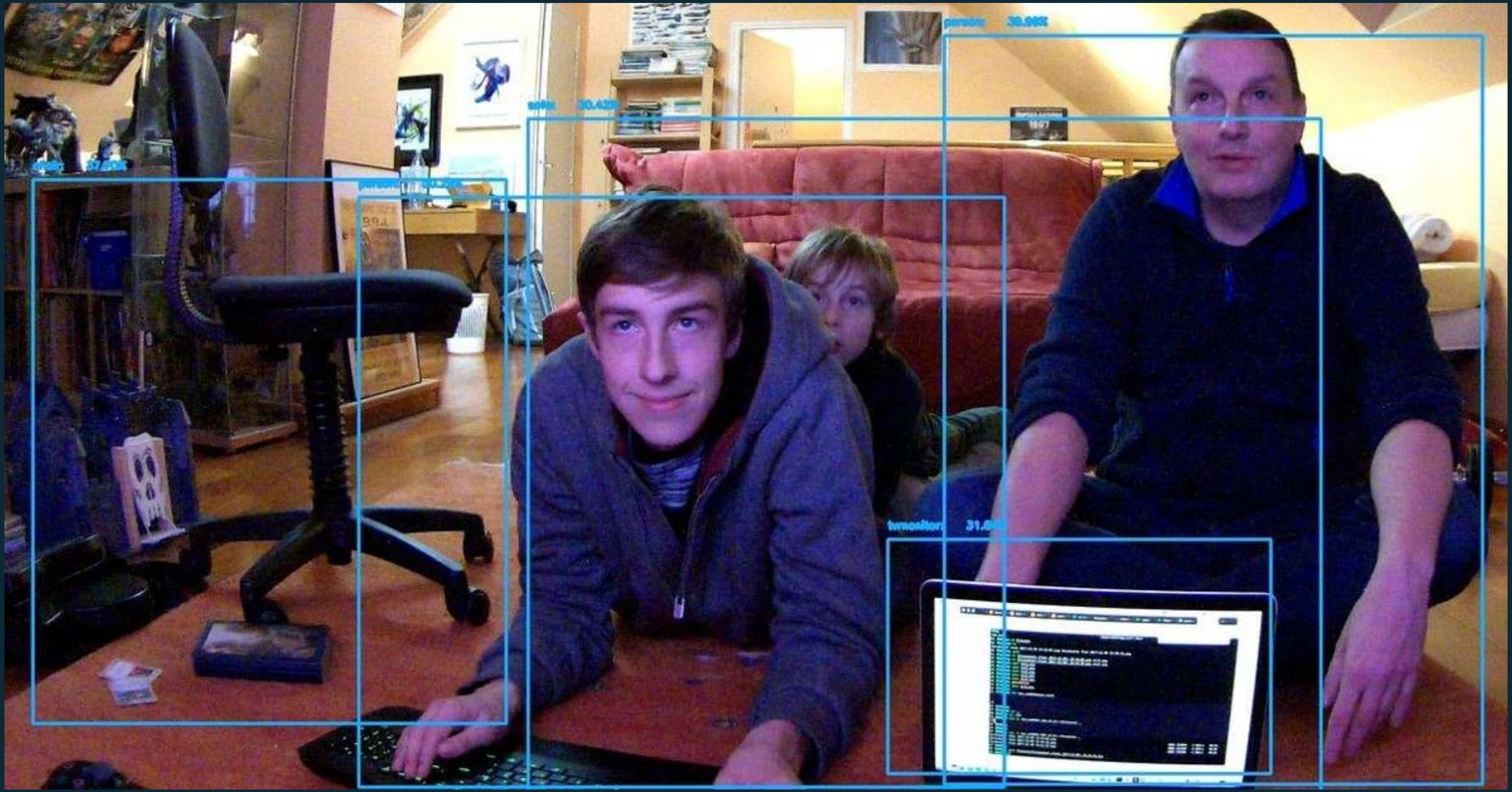
ARN

arn:aws:deeplens:us-east-1:[redacted]:project/Face-detection

Project content

Type	Name
Function	arn:aws:lambda:us-east-1:[redacted]:function:deeplens-face-detection:1
Model	deeplens-face-detection

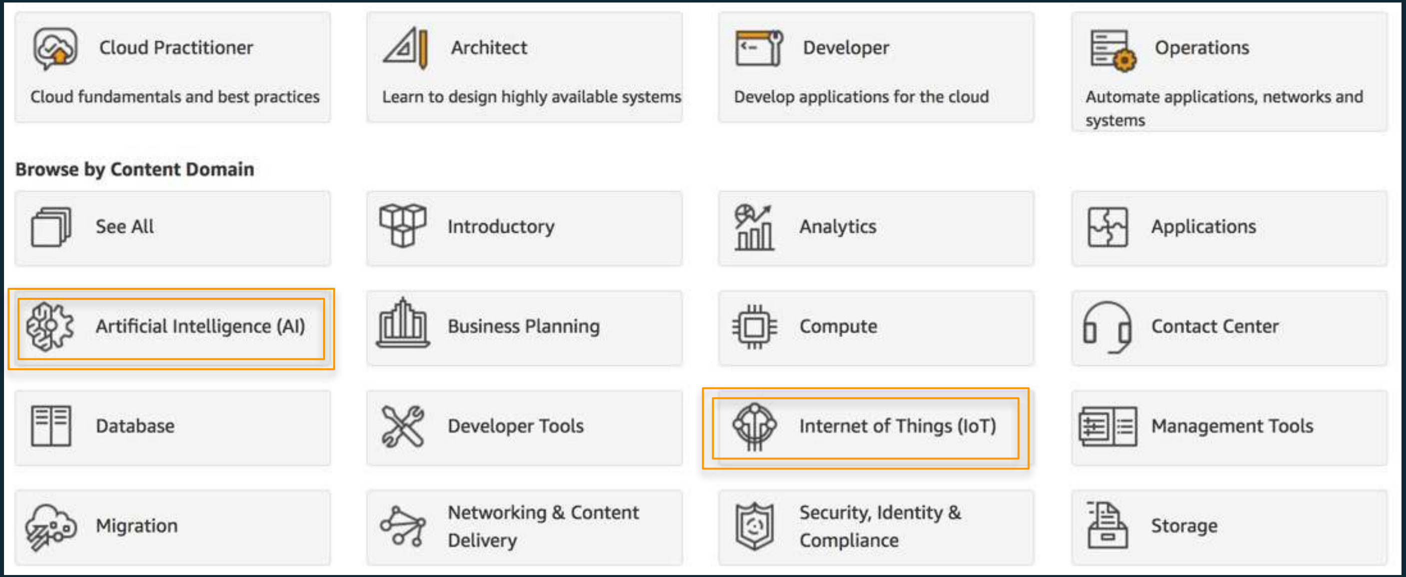
Object detection with AWS DeepLens



Getting started

Digital Training

AWS Training and Certification released free digital training courses that will make it easier for you to build your cloud skills.



Cloud Practitioner
Cloud fundamentals and best practices

Architect
Learn to design highly available systems

Developer
Develop applications for the cloud

Operations
Automate applications, networks and systems

Browse by Content Domain

See All

Introductory

Analytics

Applications

Artificial Intelligence (AI)

Business Planning

Compute

Contact Center

Database

Developer Tools

Internet of Things (IoT)

Management Tools

Migration

Networking & Content Delivery

Security, Identity & Compliance

Storage

https://aws.training

Amazon Machine Learning Lab

Lots of companies doing Machine Learning

Lack ML expertise

Unable to unlock business potential

Amazon ML Lab provides the missing ML expertise

Leverage Amazon experts with decades of ML experience with technologies like Amazon Echo, Amazon Alexa, Prime Air and Amazon Go

Brainstorming

Modeling

Teaching

https://aws.amazon.com/ml-solutions-lab/

Resources

https://aws.amazon.com/machine-learning

https://aws.amazon.com/sagemaker

https://aws.amazon.com/greengrass

https://aws.amazon.com/greengrass/ml

https://aws.amazon.com/deeplens

https://aws.amazon.com/machine-learning/amis/

https://mxnet.incubator.apache.org

http://gluon.mxnet.io/

https://medium.com/@julsimon

Thank you!

Julien Simon
Principal AI/ML Evangelist, EMEA
@julsimon