An autonomous security system with AWS DeepLens

Csaba Kiss

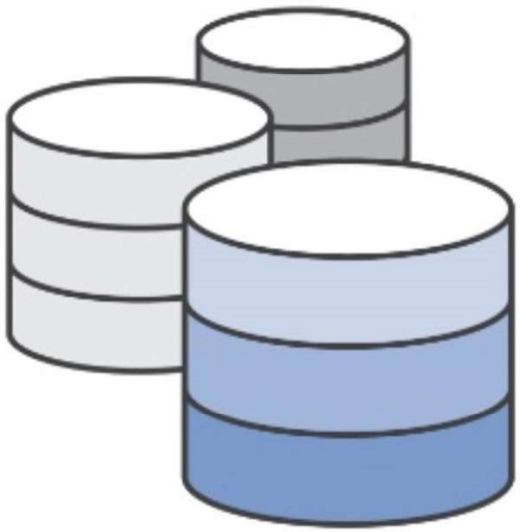
Objective: to introduce scientist and developers to machine learning in a playful manner by creating model that recognized bears and mountain lions with the help of various AWS services such as SageMaker, Lambda, Rekognition and DeepLens

DEEPLENS SPECIFICATIONS

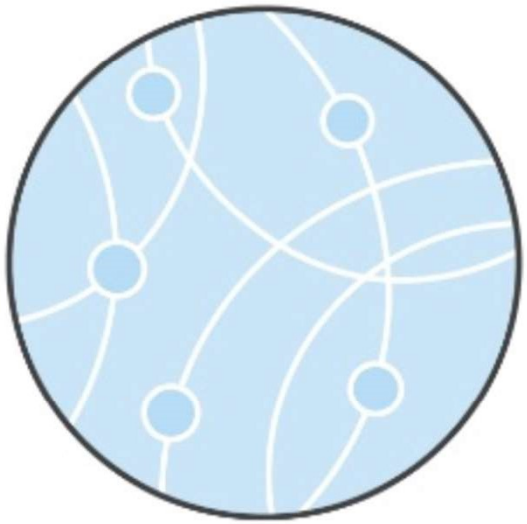
intel inside™

- Intel Atom Processor
- Gen9 graphics
- Ubuntu OS- 16.04 LTS
- 100 GFLOPS performance
- Dual band Wi-Fi
- 8 GB RAM
- 16 GB Storage (eMMC)
- 32 GB SD card
- 4 MP camera with MJPEG
- H.264 encoding at 1080p resolution
- 2 USB ports
- Micro HDMI
- Audio out
- AWS Greengrass preconfigured
- clDNN Optimized for MXNet

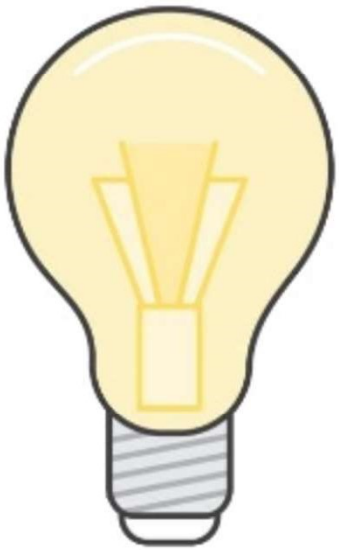
OVERVIEW OF DEEP LEARNING



Data



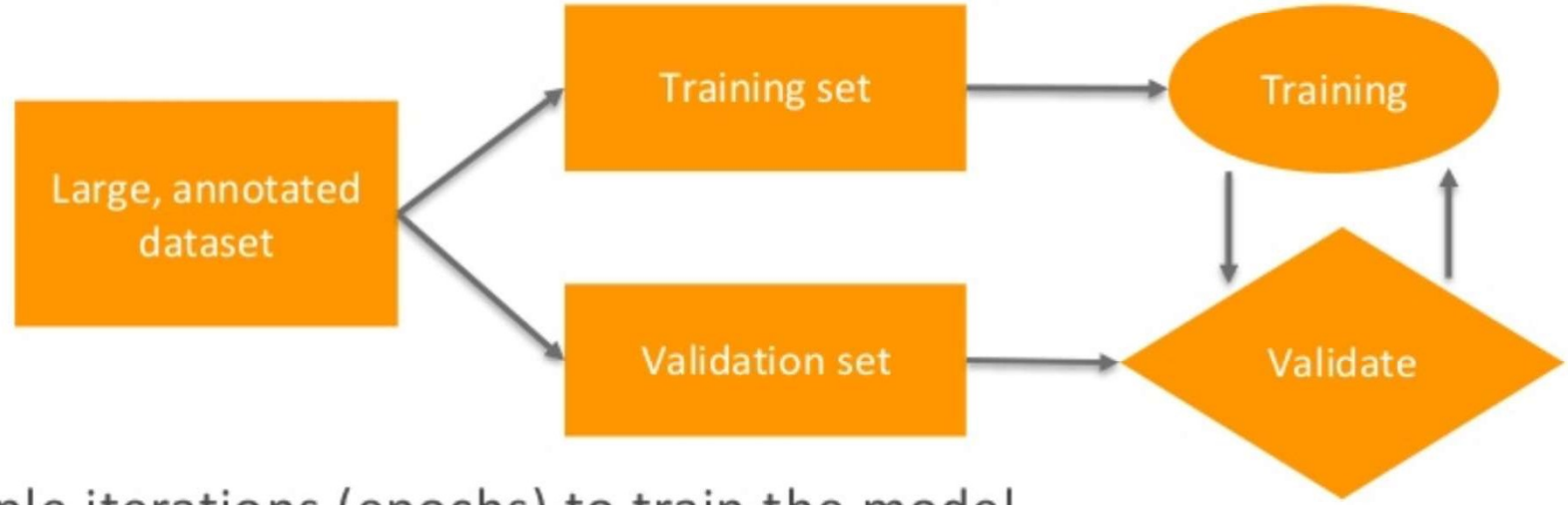
Model training



Inference

MODEL TRAINING

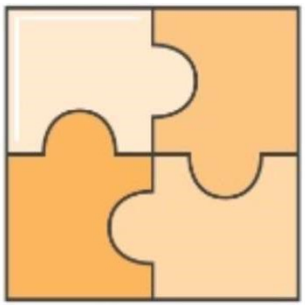
- Define model architecture
- Input the annotated and cleaned data into the model



- Multiple iterations (epochs) to train the model
- Validate with held back dataset

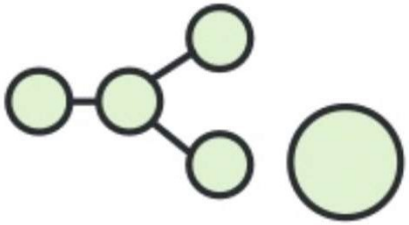
AMAZON SAGEMAKER

The quickest and easiest way to get ML models from idea to production



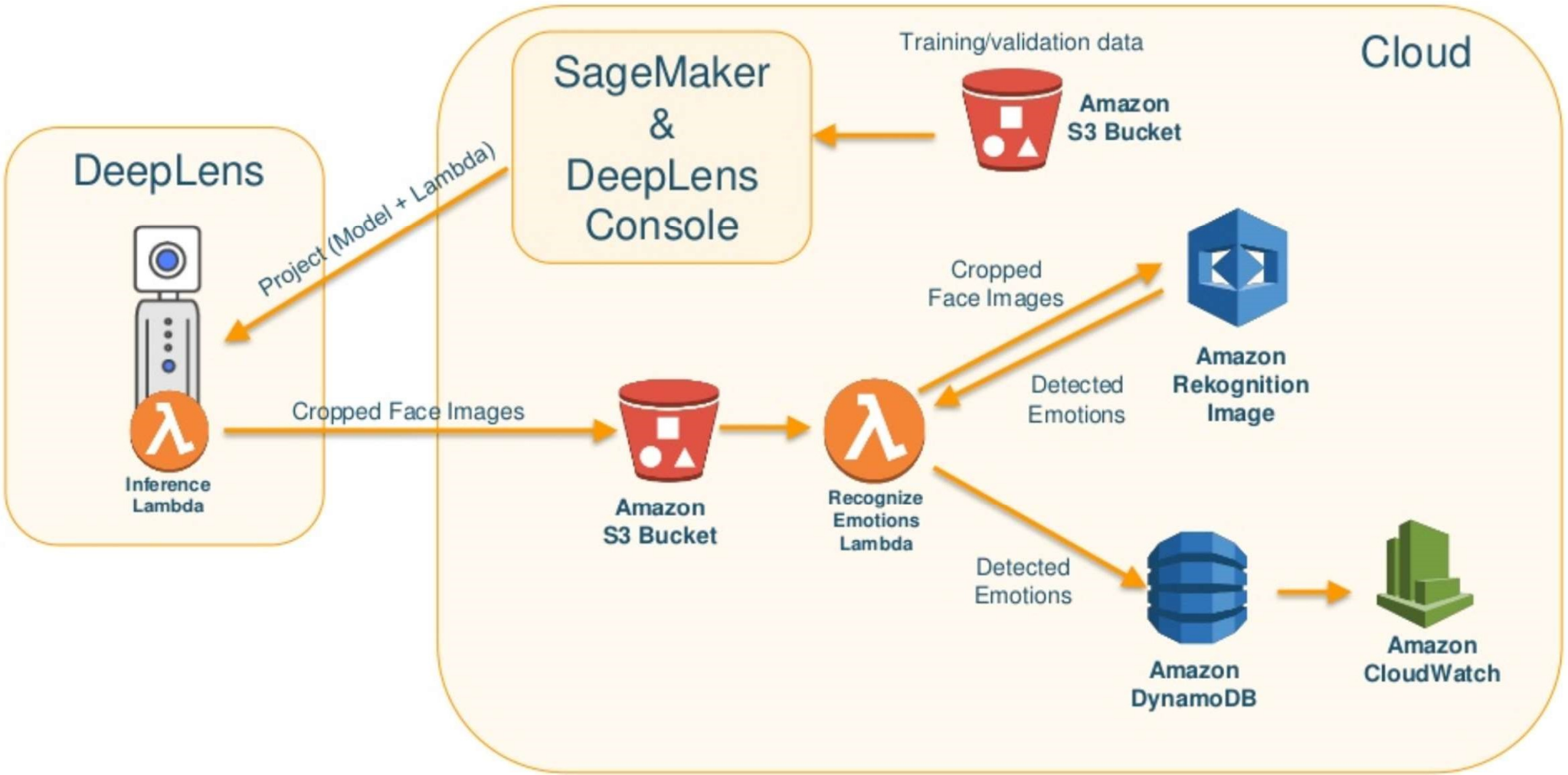
End-to-End Machine Learning Platform

Zero Setup



Flexible Model Training

Pay By The Second


DeepLens
Inference Lambda
Project (Model + Lambda)
SageMaker & DeepLens Console
Training/validation data
Amazon S3 Bucket
Cloud
Cropped Face Images
Amazon S3 Bucket
Recognize Emotions Lambda
Cropped Face Images
Detected Emotions
Amazon Rekognition Image
Detected Emotions
Amazon DynamoDB
Amazon CloudWatch

INFERENCE

It’s where the magic happens!

1. Preprocess new data/image just like training set.
2. Feed image back to the trained model to get a predicted output.