Project Synopsis:

Building Face Recognition Service Using Amazon DeepLens

Siyuan Liu siyuanliu2022@u.northwestern.edu MSDS462 Computer Vision

Introduction & Problem Definition

This project aims at developing a computer vision application with capability of detecting and recognizing faces based on ML algorithms and AWS cloud services and deployed the application on Amazon DeepLens as edge device.

Architecture & Environment Overview

The programmatic components including primarily ML models and Lambda functions are developed with Python 3.7 runtime under AWS Linux ecosystem. The application workflow runs dependent on the collaboration between edge and cloud section. The graph below demonstrates the workflow diagram.

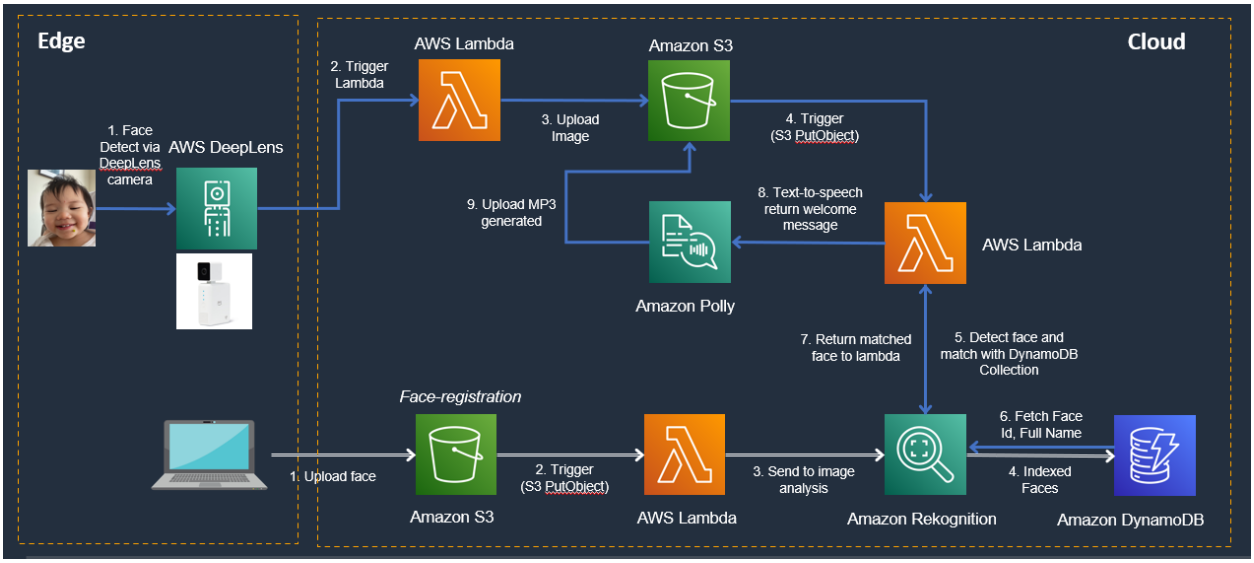


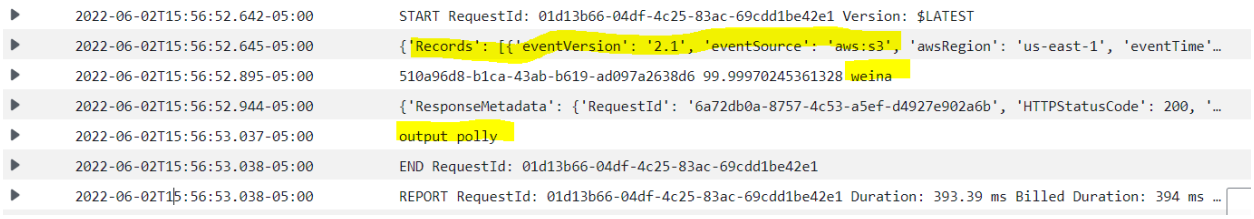
Figure 1. Project Workflow

The edge, DeepLens camera, takes care of face detection duty with a computer vision model and Lambda function deployed. The model is tasked to conduct image analysis job in consecutive basis and the Lambda function uploads the image if any face is detected. Once the image is uploaded to the assigned S3 bucket on the cloud. another Lambda function is triggered to call for Amazon Rekognition service to conduct image analysis and match the faced detected from the uploaded

image with ones registered in DynamoDB collection, The DynamoDB keeps the record of index faces with FaceID, pattern and label for each. If a match is returned, the Lambda function will call for Polly's text-to-speech service to generate welcome message as mp3 file and upload it to S3 bucket.

Experimental Evaluation

Multiple tests have been conducted. Since 16 images of 5 faces have been indexed and registered with DynamoDB to make sure the pattern of each face is fully learned by Rekognition, all tests work out (5/5 rate) given the small sample of indexed faces. Below are the logs documented by AWS CloudWatch service regarding the label identification and Polly output process.



Related Work

There are several trials which perform similar features but were built upon Python 2.7 runtime back in 2018 Amazon re:Invent. OneEye, developed by Dr. Yazdan Shivany's team, is capable of detecting customer face and pulling out the matched customer profile from backend database based on AWS ecosystem (link to more details). DeepLens Family Assistant is another similar face recognition application designed based on AWS computer vision algorithm but extend with additional front-end user interface using Amazon Elastic Beanstalk.

Future Work

A customized MXNet model could be trained with SageMaker using my own data and deployed so that the application can run on DeepLens independent of cloud and internet. Additionally, Alexa-enabled speaker could be brought in to automatically play the generated audio file.

Conclusion

AWS ML platform including DeepLens, SageMaker, and Rekognition along with its other cloud service allow for turning hypothetical machine learning into productivity with end-to-end solution

Github Repository & Jupyterbook

https://github.com/ShawnLiu119/aws_deeplens_face_recognition_project