

== REPORT ==

< 중간 과제#3 >

과 목	IOT 플랫폼과 IOT 장치 설계
담당 교수	최민 교수님
제출 일자	2022.05.25
전공	산업인공지능학과
학번	
이름	

1. AWS Lambda 기능을 활용하여 다음 기능을 수행하는 서비스를 생성하시오.

- 1) AWS Lambda 서비스와 AWS Rekognition 서비스를 사용,
- 2) runtime 은 python
- 3) trigger 는 S3 의 객체 생성 event 활용,
- 4) S3 에 파일 업로드시 AWS Lambda 함수가 실행되면서 S3 에 업로드된 파일에 대하여 AWS Rekognition 의 detect label 기능을 적용하여 이미지 내 객체인식한 결과를 JSON 형식으로 로그에 출력함.

제출 : AWS Lambda 에 작성한 코드와 화면 스크린샷을 제출합니다.

기한 : 2022 년 5 월 26 일

1) AWS Lambda 에서 작성한 코드

```
import boto3
from decimal import Decimal
import json
import urllib.request
import urllib.parse
import urllib.error

print('Loading function')

rekognition = boto3.client('rekognition')

# ----- Helper Functions to call Rekognition APIs -----

def detect_faces(bucket, key):
    response = rekognition.detect_faces(Image={"S3Object": {"Bucket": bucket, "Name": key}})
    return response

def detect_labels(bucket, key):
    response = rekognition.detect_labels(Image={"S3Object": {"Bucket": bucket, "Name": key}})

    # Sample code to write response to DynamoDB table 'MyTable' with 'PK' as Primary Key.
    # Note: role used for executing this Lambda function should have write access to the table.
    #table = boto3.resource('dynamodb').Table('MyTable')
    #labels = [{'Confidence': Decimal(str(label_prediction['Confidence'])), 'Name': label_prediction['Name']} for label_prediction in response['Labels']]
    #table.put_item(Item={'PK': key, 'Labels': labels})
    return response

def index_faces(bucket, key):
    # Note: Collection has to be created upfront. Use CreateCollection API to create a collection.
    #rekognition.create_collection(CollectionId='BLUEPRINT_COLLECTION')
    response = rekognition.index_faces(Image={"S3Object": {"Bucket": bucket, "Name": key}}, CollectionId='BLUEPRINT_COLLECTION')
    return response

# ----- Main handler -----

def lambda_handler(event, context):
    '''Demonstrates S3 trigger that uses
    Rekognition APIs to detect faces, labels and index faces in S3 Object.
    '''
    #print("Received event: " + json.dumps(event, indent=2))

    # Get the object from the event
    bucket = event['Records'][0]['s3']['bucket']['name']
    key = urllib.parse.unquote_plus(event['Records'][0]['s3']['object']['key'])

    try:
        # Calls rekognition DetectFaces API to detect faces in S3 object
        response = detect_faces(bucket, key)

        # Calls rekognition DetectLabels API to detect labels in S3 object
        #response = detect_labels(bucket, key)

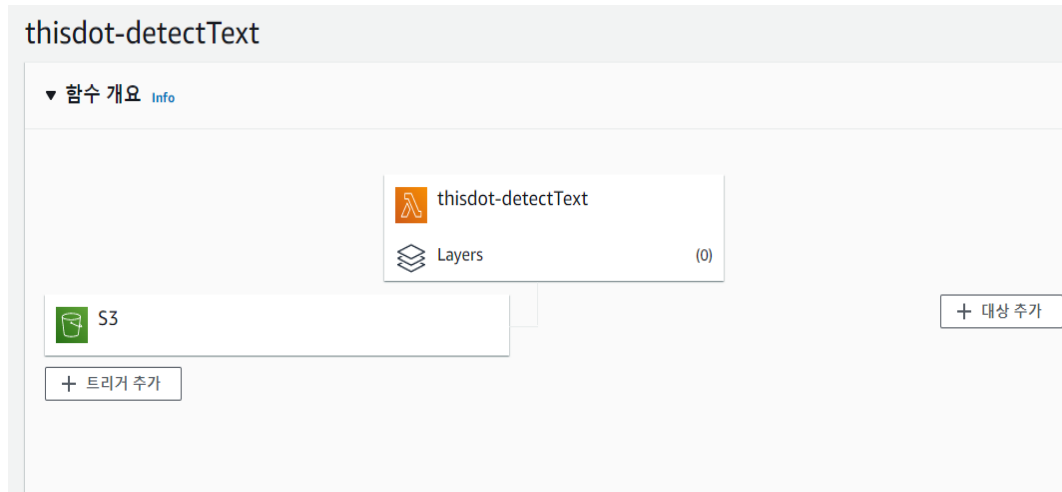
        # Calls rekognition IndexFaces API to detect faces in S3 object and index faces into specified collection
        #response = index_faces(bucket, key)

        # Print response to console.
        print(response)

        return response
    except Exception as e:
        print(e)
        print("Error processing object {} from bucket {}. ".format(key, bucket) +
              "Make sure your object and bucket exist and your bucket is in the same region as this function.")
        raise e
```

2) 결과 스크린 샷

(1) Lambda 구성



(2) 결과 스크린샷



(3) 인식 결과 로그

```
{'FaceDetails': [{ 'BoundingBox': { 'Width': 0.28637221455574036, 'Height': 0.7875415086746216, 'Left': 0.12115167826414108, 'Top': 0.11378925293684006, 'Landmarks': [{ 'Type': 'eyeLeft', 'X': 0.1819247603416443, 'Y': 0.4271235466003418, 'Type': 'eyeRight', 'X': 0.30765119194984436, 'Y': 0.3881513178348541, 'Type': 'mouthLeft', 'X': 0.20938913524150848, 'Y': 0.6735824942588806, 'Type': 'mouthRight', 'X': 0.3139011859893799, 'Y': 0.6415072083473206, 'Type': 'nose', 'X': 0.2403259575366974, 'Y': 0.5132160186767578}, 'Pose': { 'Roll': -9.969779968261719, 'Yaw': -4.963196754455566, 'Pitch': 16.848907470703125, 'Quality': { 'Brightness': 71.18637084960938, 'Sharpness': 98.08562469482422, 'Confidence': 99.99091339111328}, 'ResponseMetadata': { 'RequestId': 'e685ff8f-ec2b-49b3-9993-01622d3a1bf8', 'HTTPStatusCode': 200, 'HTTPHeaders': { 'x-amzn-requestid': 'e685ff8f-ec2b-49b3-9993-01622d3a1bf8', 'content-type': 'application/x-amz-json-1.1', 'content-length': '679', 'date': 'Sun, 01 May 2022 12:34:34 GMT', 'RetryAttempts': 0}}
```

(4) S3 스크린샷

photo-rek-test
Info

객체
속성
권한
지표
관리
엑세스 지정

객체 (1)

객체는 Amazon S3에 저장되어 있는 기본 엔티티입니다. [Amazon S3 인벤토리](#)를 사용하여 버킷에 있는 모든 객체의 목록을 얻을 수 있습니다. 다른 사용자가 객체에 액세스할 수 있게 하려면 명시적으로 권한을 부여해야 합니다. 자세히

S3 URI 복사

URL 복사

다운로드

열기

삭제

작업

폴더 만들기

업로드

☒
버전 표시

<input type="checkbox"/>	이름	▲	유형	▼	마지막 수정	▼	크기	▼	스토리지 클래스
<input type="checkbox"/>	people.jpg		jpg		2022. 5. 1. pm 9:35:47 PM KST		134.7KB		Standard