Workplace Health Safety Monitoring System With Integrated COVID-19 Guidelines

1. Introduction

1.1 **Overview**

Project Build-A-Thon powered by AWS Educate - is a tech building initiative by AWS Educate & SmartBridge to enable students with practical skills of emerging technologies. This program will provide project based learning & development experience through SmartInternz Platform.

Project Build-A-Thon had projects based around the current times - the Covid-19 situation and my project was based around the problem statement - **Workplace Health Safety Monitoring System with Integrated COVID-19 Guidelines!**

1.2 **Purpose**

COVID-19 spread is emotionally challenging for many people, changing day-to-day life in unprecedented ways. All sections of society – including employers and employees, daily wage workers – should play a role to protect themselves and each other and help prevent further spread of the disease. WHO is providing advice and updated information on COVID-19, and on how employers can protect their employees, what measures they should take in the workplace, and other related factors. Even though the guidelines are issued it would become really a mess if any of the workers do not follow the guidelines.

The purpose of this project is to build an Artificial Intelligence model that can analyze video stream to check whether the environement of the workplace is safe or not by checking whether people are wearing masks and maintaining social distancing. The model will be built in AWS cloud using various services like S3 bucket and Amazon Rekognition. The code will be written in Python and will connect with AWS cloud using the Boto3 package. The model will be accessed through the AWS CLI and SDK.

2. Literature Survey

2.1 Existing Problem

At present, All organisations have a duty to ensure so far as is reasonably practicable, the health and safety of their workers and ensure that the health and safety of other persons is not put at risk from work carried out as part of the undertaking

Before 2020, most organisations would not have considered global pandemic at the top of their business continuity risks or priorities. Indeed in February 2020, Mercer conducted a survey to determine how many organisations had business continuity plans or protocols in place to combat a global pandemic such as COVID-19 and found that 51% of organisations surveyed did not have such a business continuity plan.

What the February 2020 Mercer study did show is that most organisations are monitoring the situation and adopting the following practical steps:

- curtailing or postponing nonessential travel where there are confirmed cases of COVID-19 (72%)
- requesting staff self-quarantine for 14 days where they have recently travelled to Mainland China (58%)
- monitoring the situation to determine the global need to work remotely (33%)
- arranging greater flexibility in terms of working from home arrangements (58%)
- providing hand sanitizer in the workplace (68%), and
- providing masks in the workplace (48%)

The existing limitation is that all of this is done manually. With an AI model to monitor the enviroment of workplace, the health and safety of the workers will be ensured in a better way.

2.2 Proposed solution

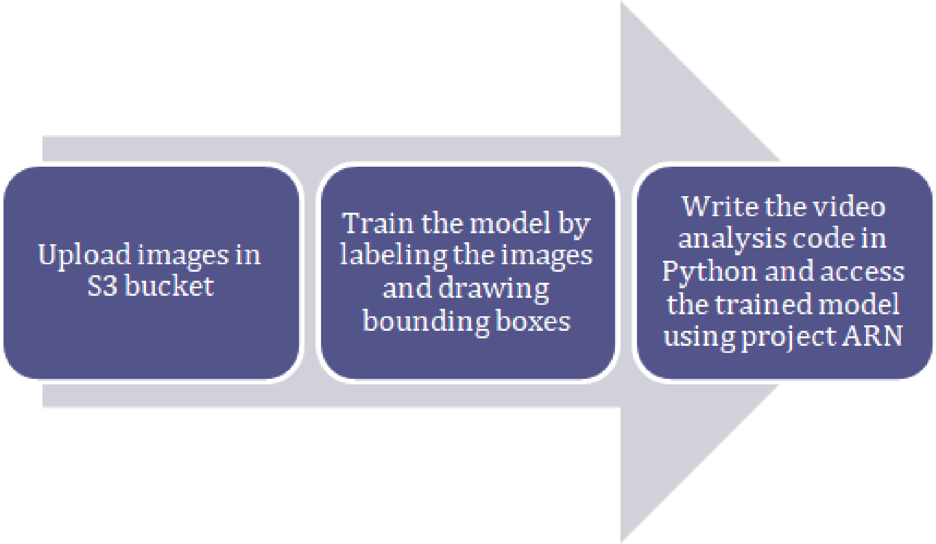
The proposed solution is to build an AI model that can be deployed in workplace architecture to analyze video streams from the CCTV cameras and tell the admin when the situation is safe i.e. when people are adhering to Covid guidelines like social distancing, wearing a mask etc and when the situation is unsafe i.e when no proper distancing is being followed or when people are not wearing a mask.

The propsed model will be built using Custom Labels service of the Artificial Intelligence service of AWS known as Amazon Rekognition. The model will be trained with a few images to make it understand what is safe and what is not. Then, with the help of open-CV, video analysis will be done by capturing frames of the video and passing them as images to the model.

3. Theoretical Analysis

3.1 **Block Diagram**

Workflow of the project can be divided into 3 sub tasks. These include aquiring the images and uploading them in an S3 bucket in AWS cloud, labeling the images, drawing bounding boxes and training the model and then acessing the model in the Python code of video analysis thorugh project ARN.


Upload images in S3 bucket
Train the model by labeling the images and drawing bounding boxes
Write the video analysis code in Python and access the trained model using project ARN

3.2 **Software Designing**

An AWS educate account was set up to build and train the model.

Following services have been used in this project:

1. **Amazon S3 bucket** - S3 bucket was used to store images that were trained to build the model
2. **Custom Labels** - This Amazon Rekognition service was used to train the model using the images stored in S3 bucket.
3. **AWS CLI and SDK** - This Amazon command line interface was used to run the model and detect labels in the video.

4. Experimental Investigation

4.1 Data acquisition

Here, data refers to the set of images that were downloaded from Google images according to the project - workplace images in which people are adhering to Covid guidelines and images where people are not.

4.2 Model Requirements

4.2.1 Python

Python is a multi-paradigm, general purpose, high level programming language, which focuses on code readability.
It has a large library, which provides tools for many tasks and has a wide support base.

4.2.2 Python Libraries

1. **json**

The json library can parse JSON from strings or files. The library parses JSON into a Python dictionary or list. It can also convert Python dictionaries or lists into JSON strings.

2. boto3

Boto is the Amazon Web Services (AWS) SDK for Python. It enables Python developers to create, configure, and manage AWS services, such as EC2 and S3. Boto provides an easy to use, object-oriented API, as well as low-level access to AWS services.

3. cv2

OpenCV-Python is a library of Python bindings designed to solve computer vision problems.

4. math

The Python Math Library provides us access to some common math functions and constants in Python, which we can use throughout our code for more complex mathematical computations. The library is a built-in Python module, therefore you don't have to do any installation to use it.

5. io

The io module provides the Python interfaces to stream handling. Under Python 2. x, this is proposed as an alternative to the built-in file object, but in Python 3. x it is the default interface to access files and streams.It defines the basic interface to a stream.

4.2.3 Training the model

This step involves uploading images in the S3 bucket, adding labels, assigning labels tothe images and drawing bounding boxes to train the model according to the labels.

4.2.4 Code Development

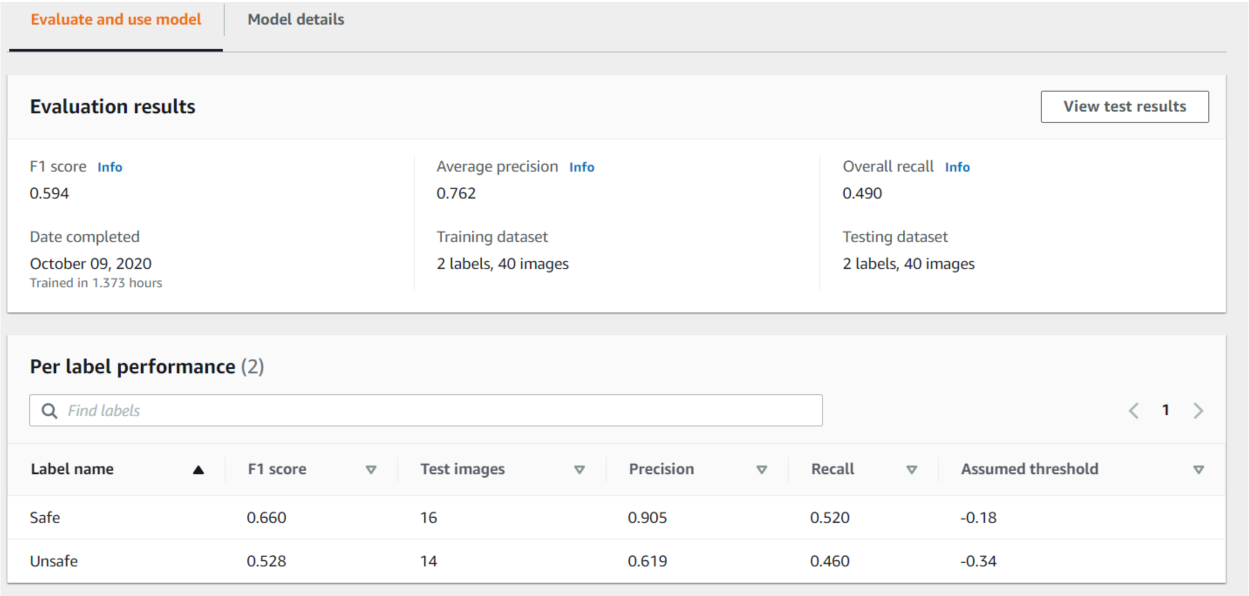
This step involves designing a code to access the trained model in Python.

4.2.5 Running the model

This step involves running the model in the command prompt and evaluating the results.

5. Result

Screenshot of the result of Custum Labels model



Evaluation results

F1 score
0.594

Average precision
0.762

Overall recall
0.490

Date completed
October 09, 2020
Trained in 1.373 hours

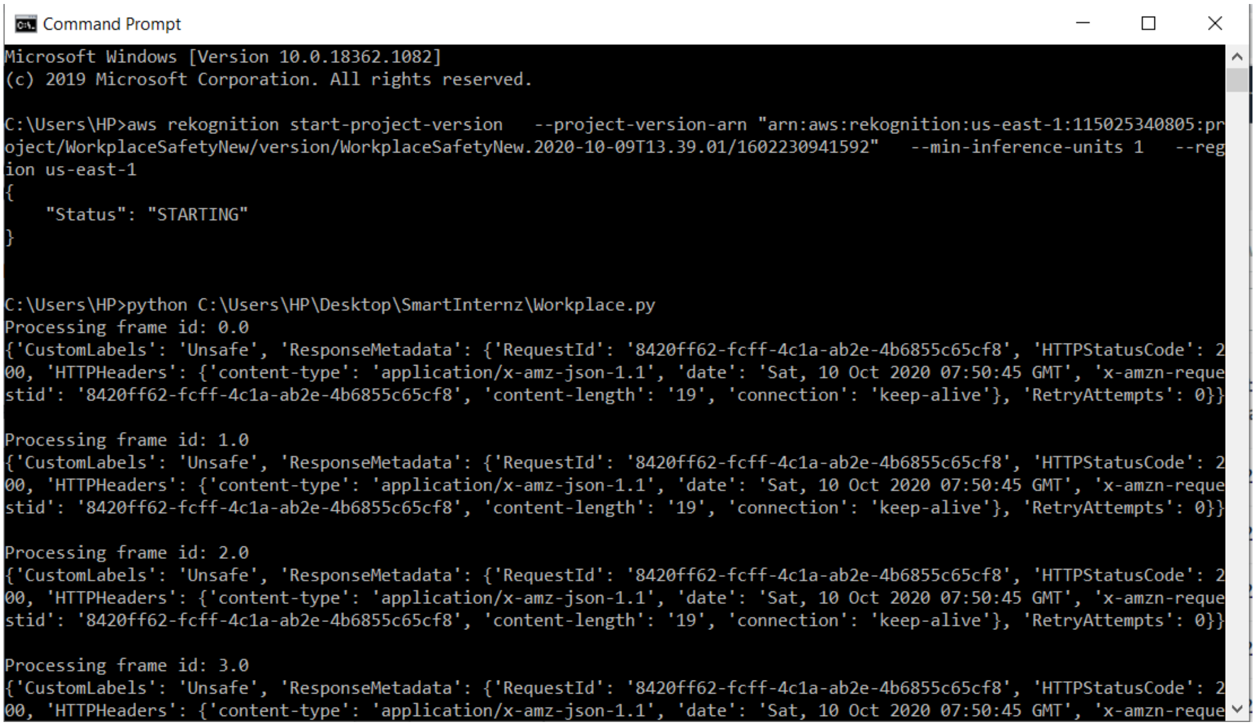
Training dataset
2 labels, 40 images

Testing dataset
2 labels, 40 images

Per label performance (2)

Find labels

Label name	F1 score	Test images	Precision	Recall	Assumed threshold
Safe	0.660	16	0.905	0.520	-0.18
Unsafe	0.528	14	0.619	0.460	-0.34



```
Microsoft Windows [Version 10.0.18362.1082]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\HP>aws rekognition start-project-version   --project-version-arn "arn:aws:rekognition:us-east-1:115025340805:pr
oject/WorkplaceSafetyNew/version/WorkplaceSafetyNew.2020-10-09T13.39.01/1602230941592"   --min-inference-units 1   --reg
ion us-east-1
{
    "Status": "STARTING"
}


C:\Users\HP>python C:\Users\HP\Desktop\SmartInternz\Workplace.py
Processing frame id: 0.0
{'CustomLabels': 'Unsafe', 'ResponseMetadata': {'RequestId': '8420ff62-fcff-4c1a-ab2e-4b6855c65cf8', 'HTTPStatusCode': 2
00, 'HTTPHeaders': {'content-type': 'application/x-amz-json-1.1', 'date': 'Sat, 10 Oct 2020 07:50:45 GMT', 'x-amzn-reque
stid': '8420ff62-fcff-4c1a-ab2e-4b6855c65cf8', 'content-length': '19', 'connection': 'keep-alive'}, 'RetryAttempts': 0}}

Processing frame id: 1.0
{'CustomLabels': 'Unsafe', 'ResponseMetadata': {'RequestId': '8420ff62-fcff-4c1a-ab2e-4b6855c65cf8', 'HTTPStatusCode': 2
00, 'HTTPHeaders': {'content-type': 'application/x-amz-json-1.1', 'date': 'Sat, 10 Oct 2020 07:50:45 GMT', 'x-amzn-reque
stid': '8420ff62-fcff-4c1a-ab2e-4b6855c65cf8', 'content-length': '19', 'connection': 'keep-alive'}, 'RetryAttempts': 0}}

Processing frame id: 2.0
{'CustomLabels': 'Unsafe', 'ResponseMetadata': {'RequestId': '8420ff62-fcff-4c1a-ab2e-4b6855c65cf8', 'HTTPStatusCode': 2
00, 'HTTPHeaders': {'content-type': 'application/x-amz-json-1.1', 'date': 'Sat, 10 Oct 2020 07:50:45 GMT', 'x-amzn-reque
stid': '8420ff62-fcff-4c1a-ab2e-4b6855c65cf8', 'content-length': '19', 'connection': 'keep-alive'}, 'RetryAttempts': 0}}

Processing frame id: 3.0
{'CustomLabels': 'Unsafe', 'ResponseMetadata': {'RequestId': '8420ff62-fcff-4c1a-ab2e-4b6855c65cf8', 'HTTPStatusCode': 2
00, 'HTTPHeaders': {'content-type': 'application/x-amz-json-1.1', 'date': 'Sat, 10 Oct 2020 07:50:45 GMT', 'x-amzn-reque
```

6. Application

This model can be deployed at any kind of workplace where there are CCTV cameras intsalled. These cameras can send the video to the model where it can analyze the situation and alert the system when Covid guidelines are not being followed.

7. Conclusion

This solution can help companies to ensure better if the workers are maintaing social distancing, wearing a msk or helmet etc which in turn will make the workplace environment more safe.This is an efficient method to check for the implementation of COvid guidelines.

8. Future Scope

Further, the analytics, and big data can be used to analyse the data collected at a very large level and get better insights, This data can be used for better learning. Real time data can be used to make model relearn and process data to give better results.

People who have had Covid symptoms can be tracked through people pathing systems and their contacts can be traced by the model in the future too.

9. Bibliography

Appendix

Source code

```
import json
```

```python
import boto3
import cv2
import math
import io
def analyzeVideo():
    videoFile = "C:\\Users\\HP\\Downloads\\Crowd in a mall.mp4"
    projectVersionArn = "arn:aws:rekognition:us-east-1:115025340805:project/WorkplaceSafetyNew/version/WorkplaceSafetyNew.2020-10-06T18.55.34/1601990734982"
    rekognition = boto3.client('rekognition',
                           aws_access_key_id="ASIARVSALNGCWUMELSZW",
                           aws_secret_access_key="pRUSwCT0Z0K6dkkq2HxpNwHGsGXQWpI0DFkssxsB",
                           aws_session_token="FwoGZXIvYXdzEP///////////wEaDJpezaC6jCpHK+LWxiLFAUHnA7w9gRLhf7gMfDyUhKCLXtAVm8+kncg5jrZSY4B2eD/o5MXQvElcZh7Od1eYu1sektTOAHzuziMWj1refll5wkmBXT55nmK2Pj4ZR3wtsRnvM6jg6mweEqc3ZXGxGleCrE9/JClbQmMuJW/r+ARAWkqw3z8wdTylmqN3Sum1USsXI4oKMRHZjd16SKwUO34RNvn4XxyxFy2BI2xXWbVLRn6+E0cVfrKRCDpL8+Kz6WYYmvnk5+GIj8saOwYh4U0bkeM+KNbG+vsFMi2bTPPDjVTXGL477HZ1GMyILXnQCV61v8hpyhE8ALvxoNQiBdMLCZcz4eQWqvo=",
                           region_name='us-east-1')
    customLabels = []
    cap = cv2.VideoCapture(videoFile)
    frameRate = cap.get(5) #frame rate
    while(cap.isOpened()):
        frameId = cap.get(1) #current frame number
        print("Processing frame id: {}".format(frameId))
        ret, frame = cap.read()
        if (ret != True):
            break
        if (frameId % math.floor(frameRate) == 0):
            hasFrame, imageBytes = cv2.imencode(".jpg",
frame)
            if(hasFrame):
                response = rekognition.detect_custom_labels(
                    Image={
                        'Bytes': imageBytes.tobytes(),
                    },
                    ProjectVersionArn = projectVersionArn
                )

        for elabel in response["CustomLabels"]:
            elabel["Timestamp"] = (frameId/frameRate)*1000
            customLabels.append(elabel)

    for i in customLabels:
        print(i)
        print('\n')
    with open(videoFile + ".json", "w") as f:
        f.write(json.dumps(customLabels))
    cap.release()
analyzeVideo()

```