Machine Learning report:-

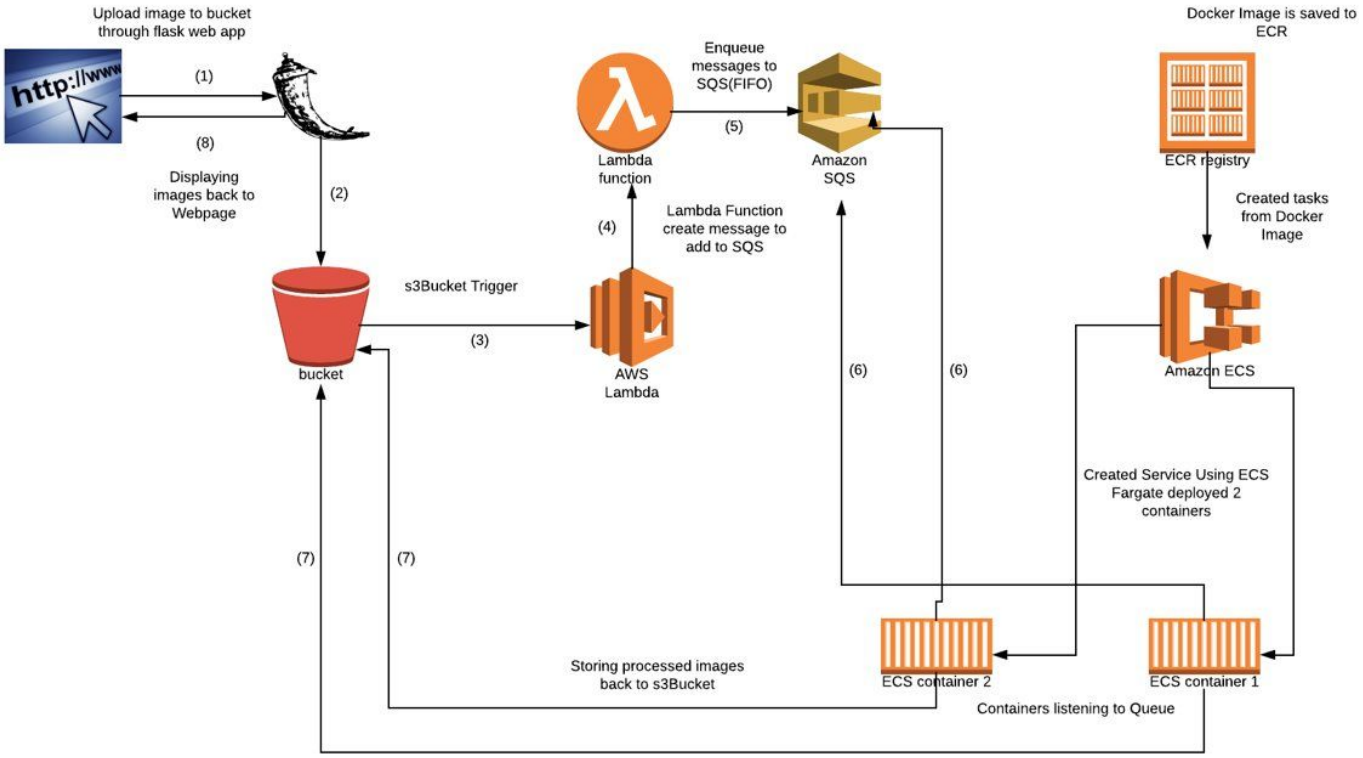
Introduction :-

For this project Deepak and me collaborated to work on hand gestures recognition data taken from kaggle.

Description of the individual work :

I mostly worked on creating the cloud architecture for deploying deep learning model trained on hand recognition data. Apart from cloud architecture I worked on developing scripts to predict label for a single image which is added to docker image. I helped deepak while he developed script to train model, we brainstormed ideas together for the whole project.

This is the architecture that I developed on cloud.



Flask app is deployed on EC2 t2 micro. It is attached to port 80. Through web page an image can be uploaded to s3 using the flask web app. Flask is a microframework for python, I used it to upload images to s3 and retrieve those images after model classifies them. Then a trigger is set on AWS lambda for a particular bucket where the files will be uploaded. After image is uploaded to s3, lamda enqueues a message to SQS. SQS is a queue(FIFO), this can have any

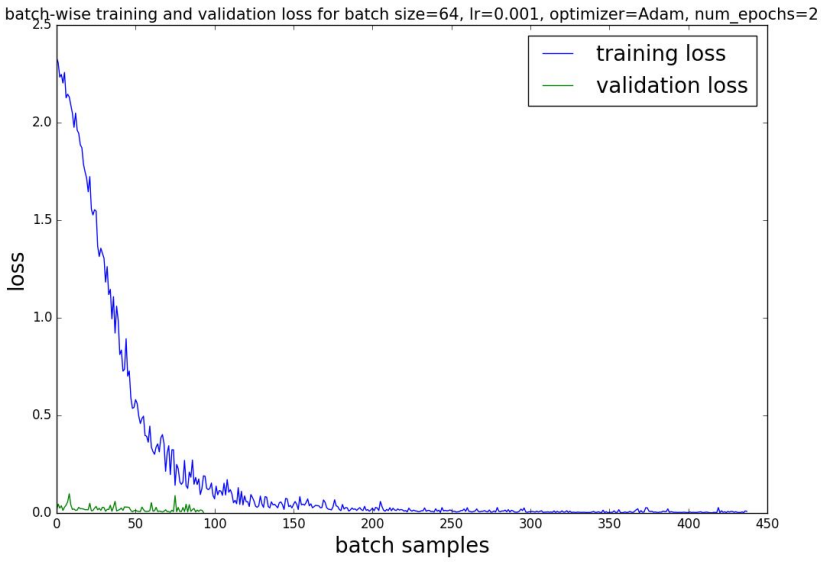
number of messages. Docker containers are running, python script polling messages are being executed, as soon as the message appears in the SQS. One container will take the message, message contains the path to image uploaded, it takes that image, executes the classifier the saves the image back to a different S3 bucket.

The flask webapp, reads all the images from the output S3 bucket and outputs labels and images on a web page. A

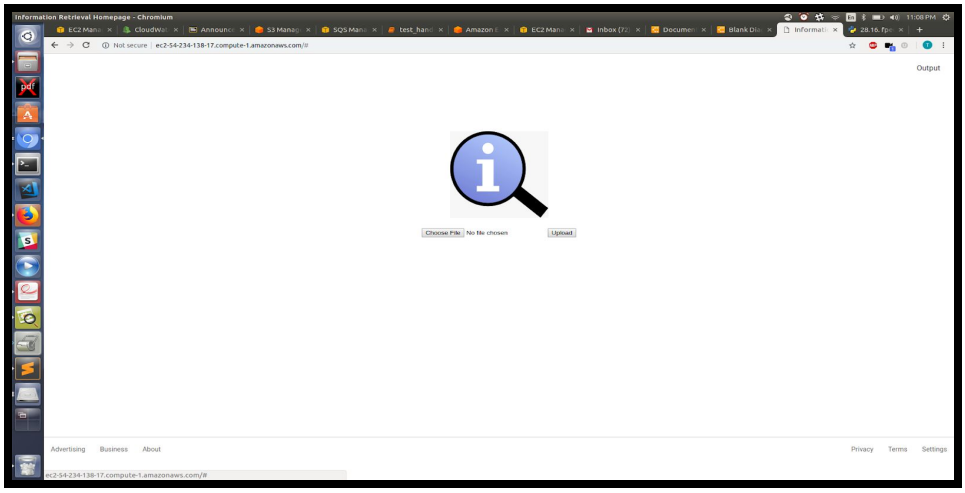
Automating the whole process in AWS was a challenge for me. I had to learn about different cloud services, how can they be integrated to make a autonomous system.

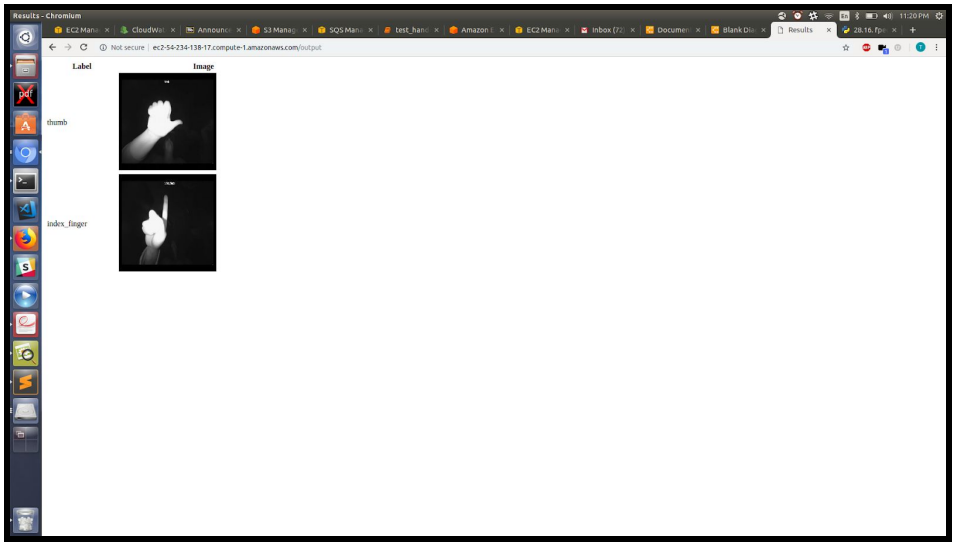
Results : -For our data set we have found best accuracy/ f-score for below parameters:

batch size	learning rate	optimizer method	num epochs	time((sec)	accuracy
64	0.001	Adam	2	7.26	99.47



Flask web app screenshots





Summary:-

This project has been really helpful. I learned a lot about Amazon web services, I feel that this will be really helpful for future endeavours. I also learned to experiment with different layers in pycharm and work with CNN networks. Working is CNN is complex as determine sizes between layers is very important, working with different strides and kernel sizes helped me understand CNN really well.

References:

https://pytorch.org/docs/stable/index.html

https://docs.aws.amazon.com/index.html#lang/en_us

https://boto3.amazonaws.com/v1/documentation/api/latest/index.html