SHRIHARI MURALI

shrihari@usc.edu
Master of Science, Computer Science
University of Southern California

Projecting the Project

- An end-to-end Sign Language Detection Application
- Categorizing the 5 following signs:
  - Hello
  - Yes
  - No
  - I Love You
  - Thank You
- Trained on Custom Dataset
- Everything on AWS!

Table of contents

1 Introduction

2 Problem & Solution

3 Application Architecture

4 Project Demo

5 Code Review & SQL Queries

6 Discussion

1

Introduction

Introduction

Title

Real-Time Sign Language Detection Using Deep Learning

Overview

- A web application that detects signs in real-time using deep learning
- Client streams the real-time data to the cloud for analytics
- Admin dashboard to show application analytics

Extro-duction

Objective

To help the hard-of-hearing community bridge the gaps in communication using technologies that are simple to use and for FREE!

Team or Individual?

At this point in time, the whole project has been an individual journey.

Why it Matters?

My Cousin

...

- There are no viable or reliable options
- Efficient communication
- Works both ways!

2

Problem & Solution

Problems

Technologies?

The trade-off between complex technologies and usability

Real-Time

How do we build a tool that can make decisions on the fly?

Availability

The application must be highly available and fault tolerant

Features

Pipelining different features into a single application

Users

Ease-of-use and accessibility for our users base

Price

All at what cost?

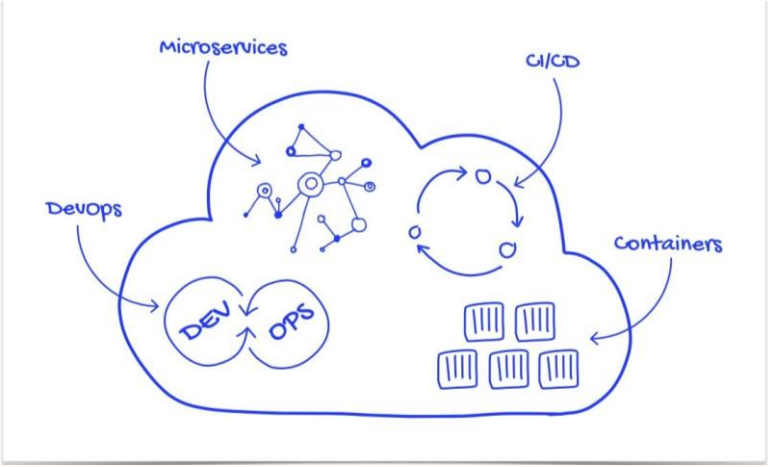
Solutions!

	Technologies
Sign App	React.js
Admin App	React.js
Sign Streaming	Amazon Kinesis
Analytics	Managed Apache Flink
Containerizing	Docker
Querying	SQL for Flink
Docker Repo	Docker Hub
Cloud Provider	AWS

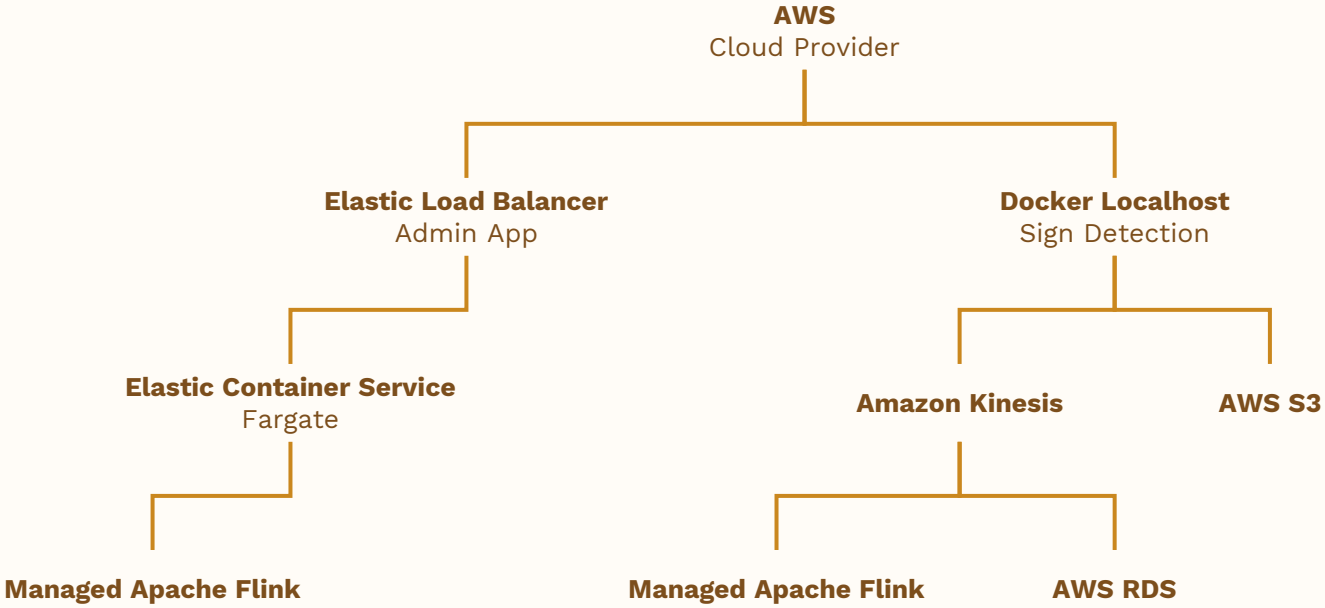
	Technologies
DL Framework	Tensorflow
Application ML Framework	Tensorflow JS
Computer Vision	OpenCV
DL Optimization 1	TensorflowJS_Converter
DL Optimization 2	TFRecord
Transfer Learning	my_ssd_mobnet
Labeling	LabelImg

3

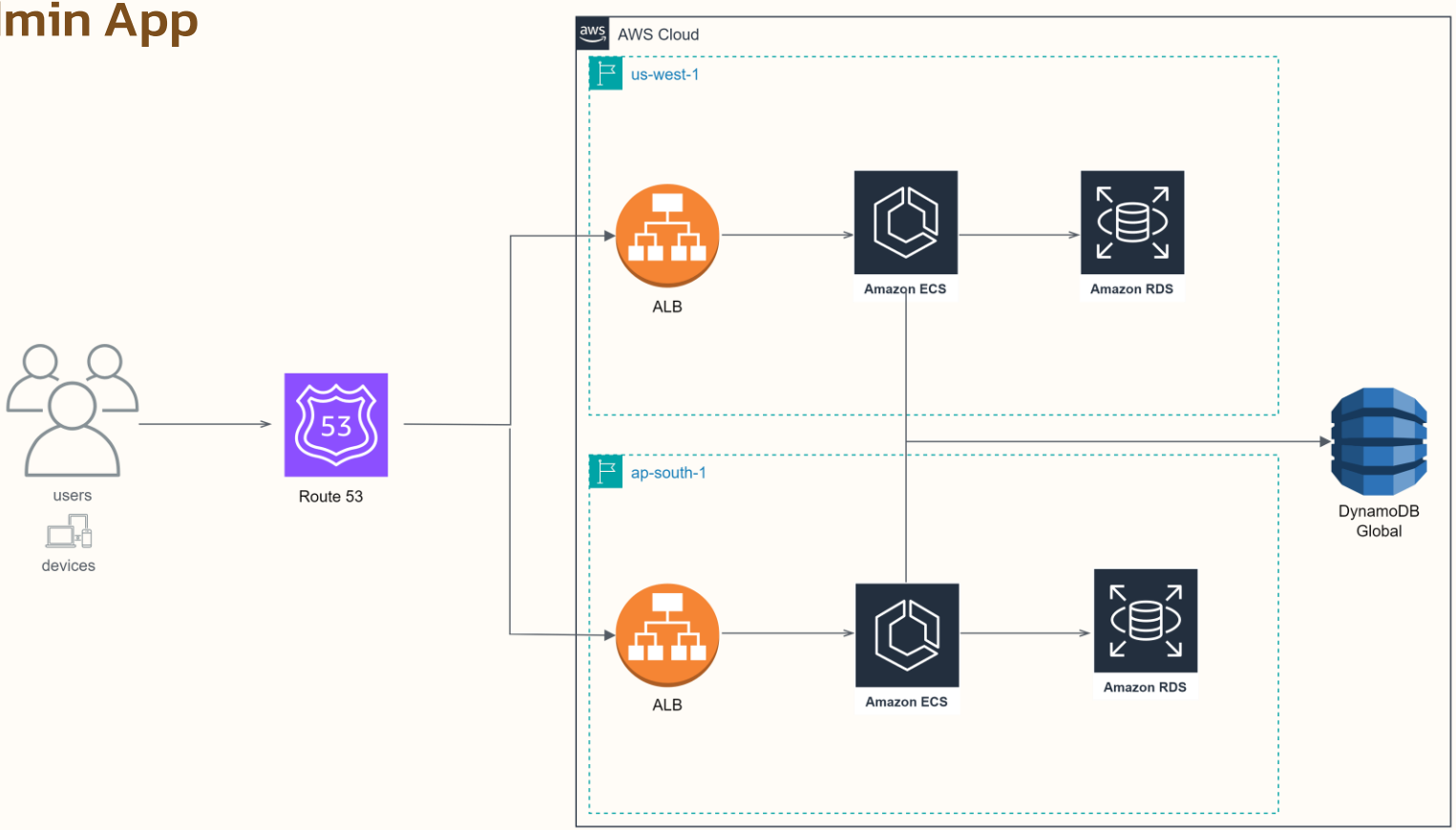
Application Architecture



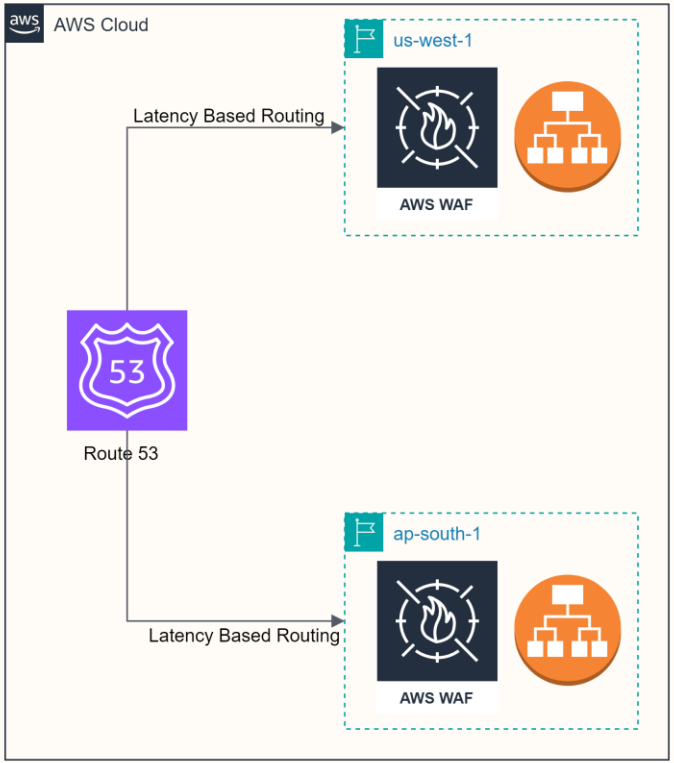
Application Architecture – Overview



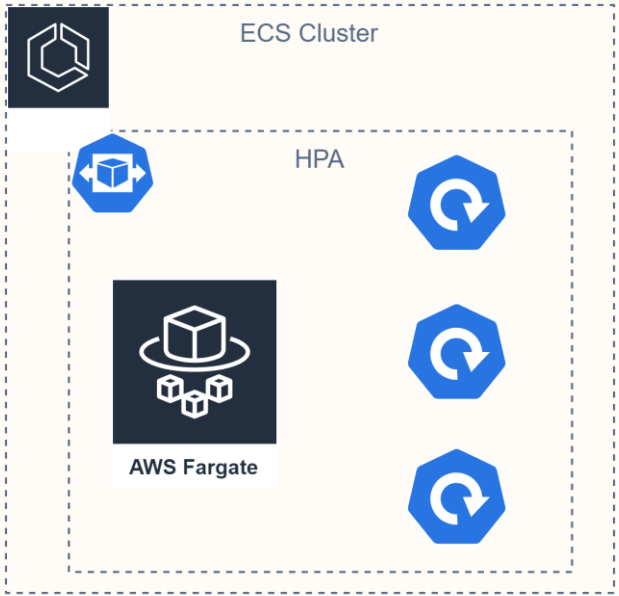
Admin App


AWS Cloud
us-west-1
ALB
Amazon ECS
Amazon RDS
users
devices
Route 53
ap-south-1
DynamoDB
Global
ALB
Amazon ECS
Amazon RDS

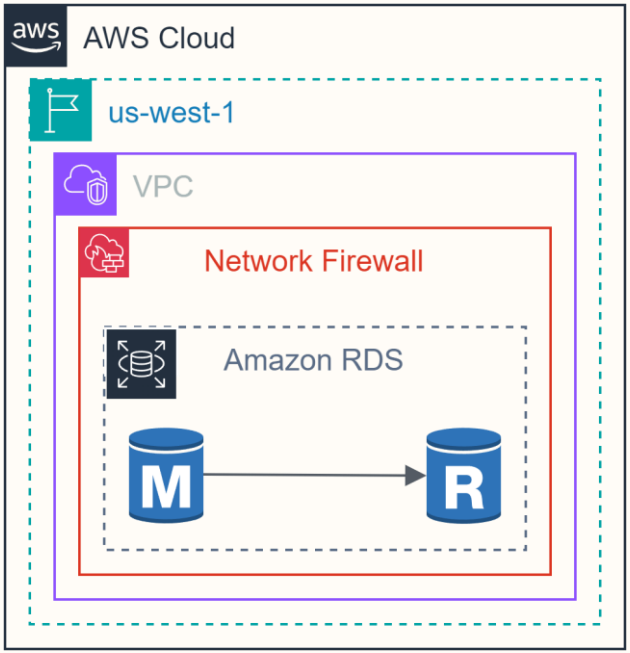
Admin App – Security



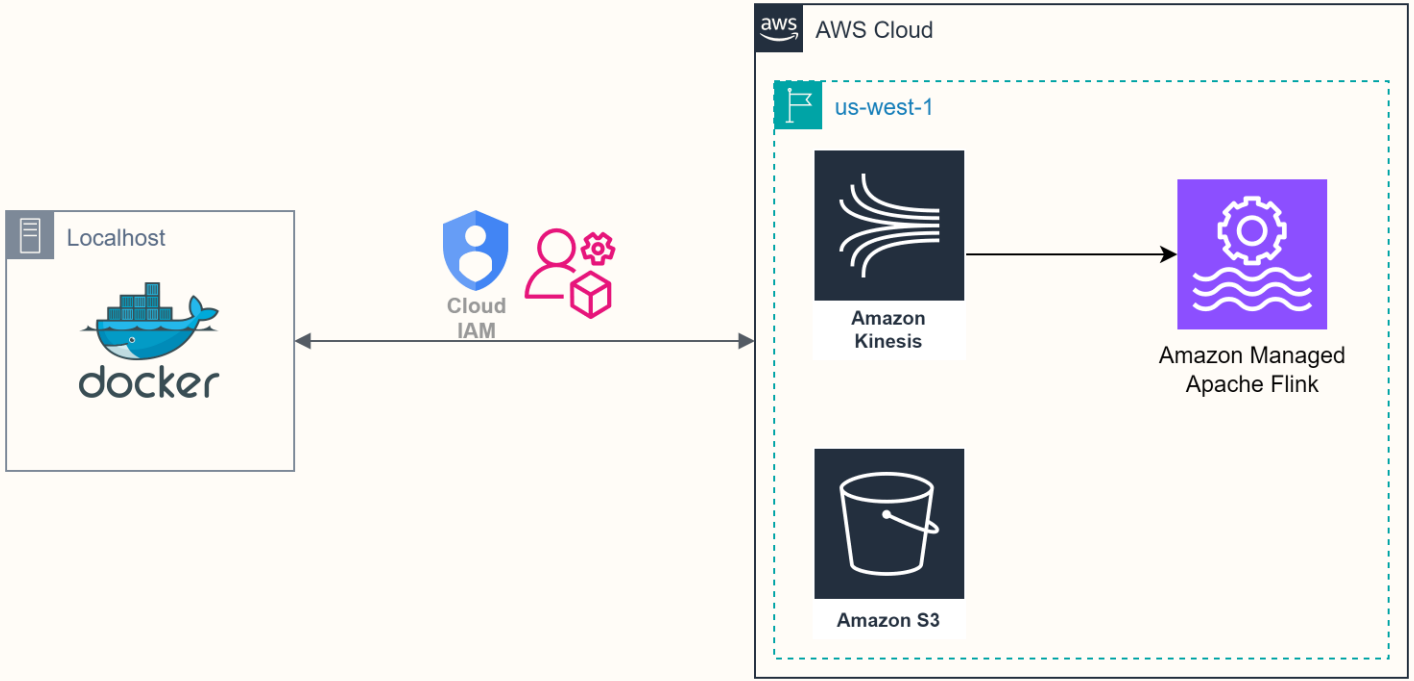
Admin App – ECS



Admin App – Database



Sign App


AWS Cloud
us-west-1
Localhost
docker
Cloud
IAM
Amazon
Kinesis
Amazon Managed
Apache Flink
Amazon S3

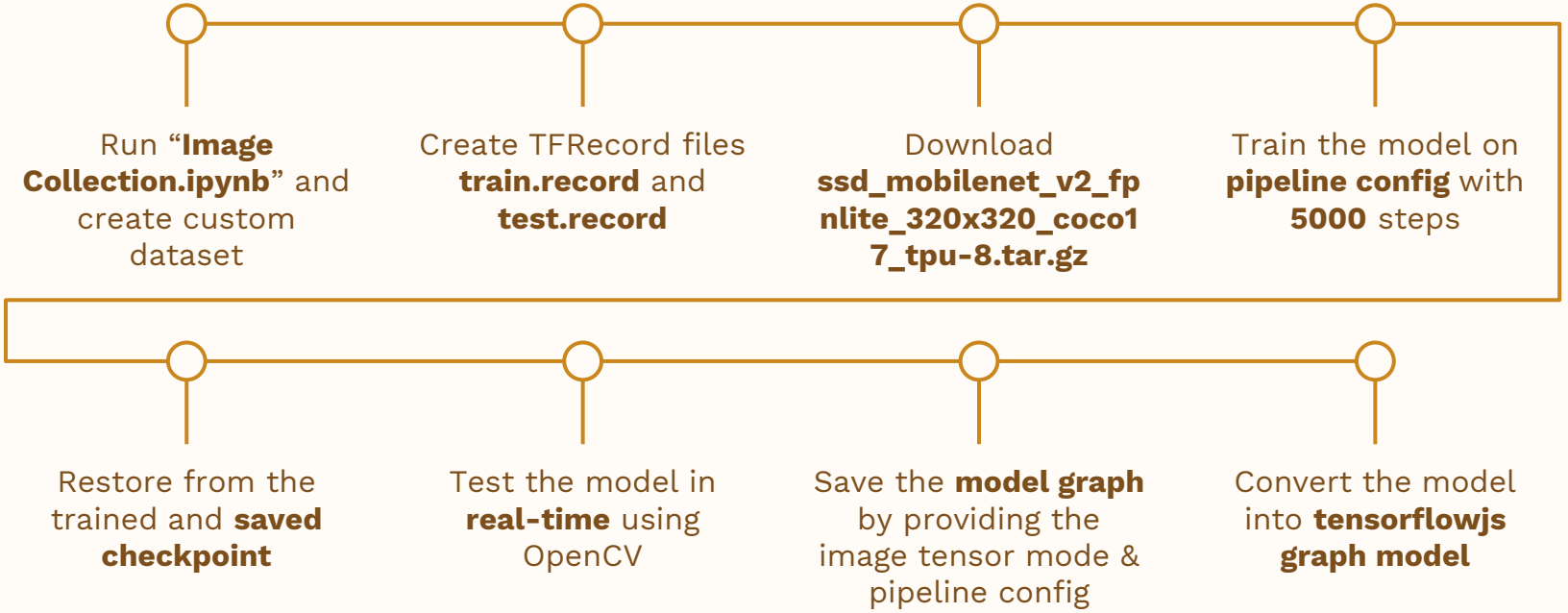
4

Project Demo

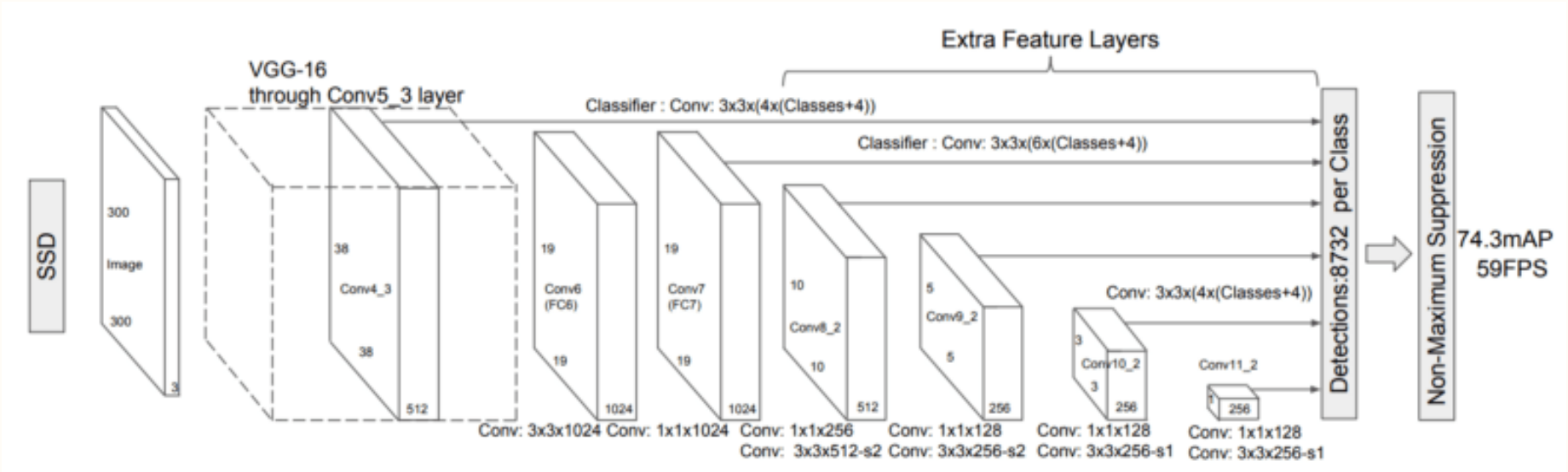
5

Code Review & SQL Queries

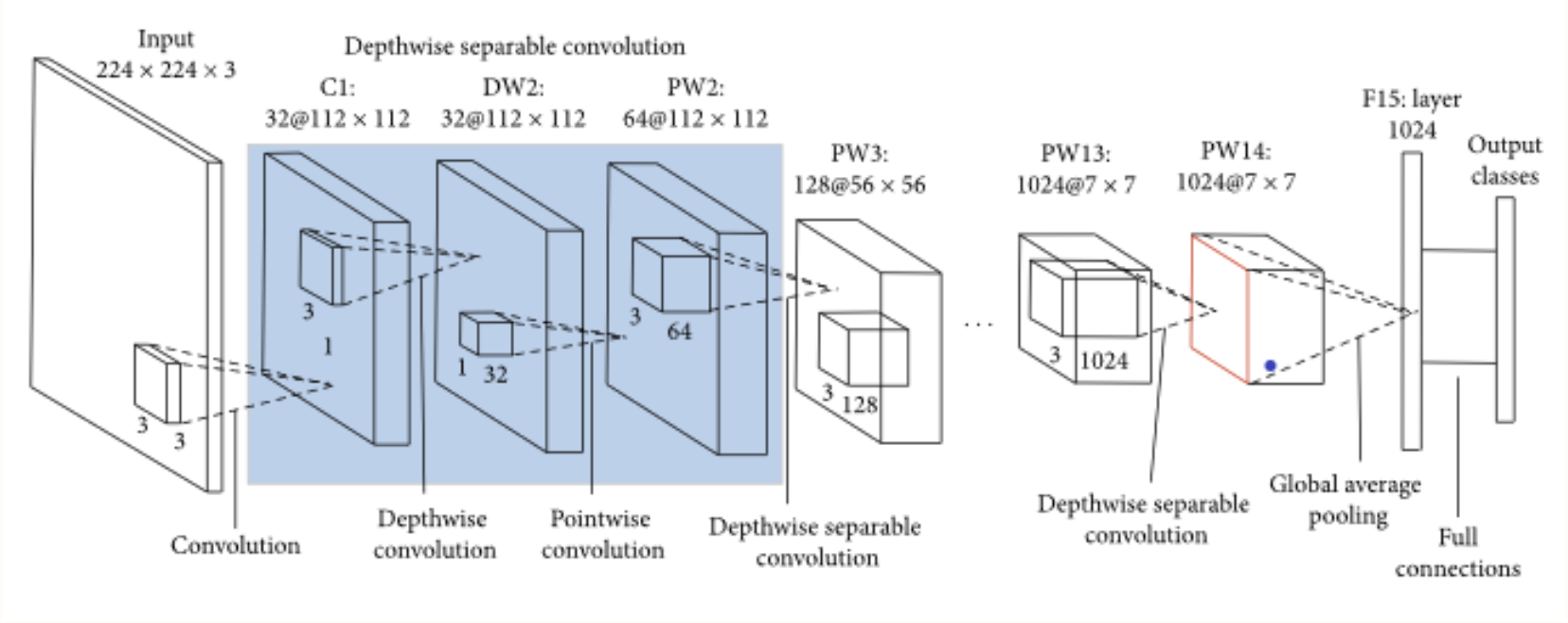
Deep Learning Training Life Cycle



Single Shot Detector [1]



MobileNet [2]


Input
224 × 224 × 3
Depthwise separable convolution
C1:
32@112 × 112
DW2:
32@112 × 112
PW2:
64@112 × 112
PW3:
128@56 × 56
PW13:
1024@7 × 7
PW14:
1024@7 × 7
F15: layer
1024
Output
classes
3
3
3
1
1
32
3
64
3
128
3
1024
Convolution
Depthwise
convolution
Pointwise
convolution
Depthwise separable
convolution
Depthwise separable
convolution
Global average
pooling
Full
connections

6

Discussion

Thanks!