Engineering Requirements

Table of Content

[Summary 1](#_Toc16502531)

[Current technical considerations for “USB Device”: 1](#_Toc16502532)

[Current technical considerations for “Main Device” 2](#_Toc16502533)

[Known challenges: 3](#_Toc16502534)

[Alternative “Main Device” options: 3](#_Toc16502535)

[Option A1 “Use Entire Coral Dev Board” 3](#_Toc16502536)

[Option A2 “Use Nvidia Jetson TX2” 3](#_Toc16502537)

[Option A3 “Integrate Coral SoM chip” 4](#_Toc16502538)

[Optional but important considerations 4](#_Toc16502539)

## Summary

Goal is to develop 2 connected devices that can jointly perform ML classification tasks. There are two devices, which we will call “USB Device” and “Main Device”. Overall picture assumes the following:

1. “USB Device” and “Main Device” are connected to the same WiFi network
2. “Main Device” plays role of “WiFi Router”
3. “USB Device” is connected to external system as a storage device.
4. “Main Device” runs a web-server that servers as main communication channel between “Main Device” and “USB Device”. This web-server will be used to present api endpoints for another external device (tablet) to display results of machine learning
5. “Main Device” is capable of running ML tasks in timely manner

Overall, this is a proof of concept project

## Current technical considerations for “USB Device”:

Current idea is to use Raspberri Pi Zero W with USB expander. Similar to TeslaUsb project, this device will present itself as “storage device” once connected to “external device”.

Current scenario:

Prerequisites:

* “USB Device” is configured to connect to WiFi network provided by “Main Device”
* Once connected “USB Device” can access “Main Device” web-server (HTTP)

Main UX flow:

1. User connects “USB Device” to external device (PC). “USB Device” presents itself as storage device.
2. User copies files (200-500Mb) to the “USB Device”
3. Once “USB Device” receives a signal from “Main Device” (HTTP), “USB Device” disconnects “storage” and starts sending files to “Main Device” (using CIFS)

Technical notes:

1. Inspired by TeslaUSB: <https://github.com/cimryan/teslausb/>
2. Raspberry Pi Zero W: https://www.raspberrypi.org/products/raspberry-pi-zero-w/
3. Zero Stem: <https://www.adafruit.com/product/3945>

## Current technical considerations for “Main Device”

Current idea is to use Raspberry Pi 4 B as a main module with one or two Coral USB Accelerators for machine learning processing.

Current scenario:

Prerequisites:

* “Main Device” is powered by USB-C power cable (120V)
* “Main Device” is hidden WiFi Router
* “Main Device” exposes a shared folder (SMB/CIFS) for “USB Device”
* “Main Device” runs web server (assuming Python Flask)
* “Main Device” has 2 Coral USB Accelerators connected using USB3 ports
* “Main Device” has TF Lite model uploaded already (machine learning is limited to inference)

Main logical flow:

1. External factor (button click) initiates new “session”.
2. “USB Device” pings “Main Device” web server and upon “new session”, “USB Device” starts sending files to “Main Device” shared folder. Once all files transfer is completed, “USB Device” notifies “Main Device” web server.
3. “Main Device” create a queue task (Python rq) to do the following:
   1. Copy files from “share” to a temp processing directory (local to “Main Device)
   2. Open files using Python libraries (openCV, pydicom, gdcm, pillow)
   3. Send data to Machine Learning (TensorFlow Lite) (assuming here that processing will happen in Coral USB Accelerator)
   4. Save results of Machine Learning into SQLite database
4. “Main Device” web server exposes “results” as JSON data using web api endpoint

Technical notes:

* Raspberry Pi 4 B: <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/>
* Coral USB Accelerator: <https://coral.withgoogle.com/products/accelerator>
* TensorFlow Lite: <https://www.tensorflow.org/lite>

### Known challenges:

* Raspberry PI 4 B is still relatively new board and has limited documentation
* Making GDCM, OpenCV, and Pillow to work (build) is not trivial
* Not tested if it can work as WiFi router
* USB cables hanging out of Raspberry Pi 4 B board will affect closure design
* Need to address active cooling that will be necessary for 2 Coral USB Accelerators
* Need to figure out if we can run processing of files on two different USB Coral USB Accelerators (the same model) to speed up processing

## Alternative “Main Device” options

Understanding that primary “Main Device” may have to be adjusted due to various technical reasons, we understand and accept hardware design alternatives.

### Option A1 “Use Entire Coral Dev Board”

It is possible to use Coral Dev Board as alternative to Raspberry Pi 4 B. This approach will limit Coral AI Chip to SoM installed on Coral Dev Board.

This approach may provide less challenges with Linux distro as everything will be limited to one system.

Technical notes:

* Dev Board: <https://coral.withgoogle.com/products/dev-board>

Possible challenges may include:

* Only one Coral AI chip does not have enough of the ML processing power
* Very limited memory may affect ability to process files (OpenCV cannot be even build on it without SD card: https://medium.com/@balaji\_85683/installing-opencv-4-0-on-google-coral-dev-board-5c3a69d7f52f
* Linux Mendel distro has limited documented support
* May need external WiFi antenna to support “USB Device”

### Option A2 “Use Nvidia Jetson TX2”

It is possible to use Nvidia Jetson TX2 as alternative to Raspberry Pi 4 B and Coral USB Accelerators. Benefit can include high processing power and more resources.

Technical notes:

* TX2: https://developer.nvidia.com/embedded/jetson-tx2

Possible challenges may include:

* ML model will have to transferred from TF Lite to TensorRT
* No experience with Nvidia cards

### Option A3 “Integrate Coral SoM chip”

It is possible to detach “SoM” from Coral Dev Board as alternative of using Coral USB Accelerator. This approach will reduce size and provide already attached active cooling.

Technical notes:

* Dev Board: <https://coral.withgoogle.com/products/dev-board>
* Dev Board FAQ (2nd from the last): https://coral.withgoogle.com/docs/edgetpu/faq/

Possible challenges may include:

* Now we need to figure out how to integrate 100pin of SoM to Raspberry Pi 4 B (or another board)

## Optional but important considerations

* It is assumed that device will have LED indicators to show progress of ML processing.
* It is optional but will be important factor if we can add LiPo power support (and charging) to enable processing when power outlet is not available
* “Main Device” may contain touch button to initiate “new session”. To be discussed
* Active cooling is important as ML make Raspberry Pi 4 B and Coral USB Accelerator very hot.