Training and Deploying Your Custom Magic Wand Project

In this document we are going to train and then deploy a custom magic wand model based on the custom gestures we just collected.

Screencast of Brian walking through this section goes here on the edX course

Training the Magic Wand Model

The first thing you'll need to do is to upload your gesture dataset into Colab and train a new magic wand model. Then in that Colab we'll need to convert that model first into a quantized `.tflite` file and then into a `.cc` file for use with the Arduino IDE.

As with the KWS examples, we will be using the resulting `.cc` file, so make sure to download it or leave the tab open with the printout!

https://colab.research.google.com/github/tinyMLx/colabs/blob/master/4-8-11-CustomMagicWand.ipynb

Deploying the Trained Model

1. Use a USB cable to connect the Arduino Nano 33 BLE Sense to your machine. You should see the green LED power indicator come on when the board first receives power.

2. Open the magic_wand.ino sketch, which you can find via the File drop-down menu. Navigate, as follows: `File → Examples → Harvard_TinyMLx → magic_wand.`

3. You'll then need to make two changes to the `magic_wand.ino` file to alert it of your number of gestures and gesture labels. These changes occur on lines 93-97 which currently read as:

```
// -------------------------------------------------------------------------------- //
// UPDATE THESE VARIABLES TO MATCH THE NUMBER AND LIST OF GESTURES IN YOUR DATASET  //
// -------------------------------------------------------------------------------- //
constexpr int label_count = 10;
const char* labels[label_count] = {“0”, "1", "2", "3", "4", "5", "6", "7", "8", "9"};
```

   a. Update the `label_count` to reflect the number of gestures in your dataset.

   b. Update the list of `labels` to reflect the gestures in your dataset. **Note: the order matters!** Make sure it matches the alphanumeric order as printed out in the training script!

4. When you save, just like with the KWS examples you will be asked to save a copy. Again, we suggest that you make a folder called e.g., `TinyML` inside of your `Arduino` folder. You can find your main `Arduino` folder either inside of your `Documents` folder or in your `Home` folder, and save it in that folder with a descriptive name like `magic_wand_custom`. That said, you can save it wherever you like with whatever name you want!

5. As always, use the Tools drop down menu to select appropriate Port and Board.

   a. Select the Arduino Nano 33 BLE as the board by going to `Tools → Board: <Current Board Name> → Arduino Mbed OS Boards (nRF52840) → Arduino Nano 33 BLE.` Note that on different operating systems the exact name of the board may vary but/and it should include the word Nano at a minimum. If you do not see that as an option then please go back to Setting up the Software and make sure you have installed the necessary board files.

   b. Then select the USB Port associated with your board. This will appear differently on Windows, macOS, Linux but will likely indicate ‘Arduino Nano 33 BLE” in parenthesis. You can select this by going to `Tools → Port: <Current Port (Board on Port)> → <TBD Based on OS> (Arduino Nano 33 BLE).` Where <TBD Based on OS> is most likely to come from the list below where <#> indicates some integer number

      i. Windows → `COM<#>`

      ii. macOS → `/dev/cu.usbmodem<#>`

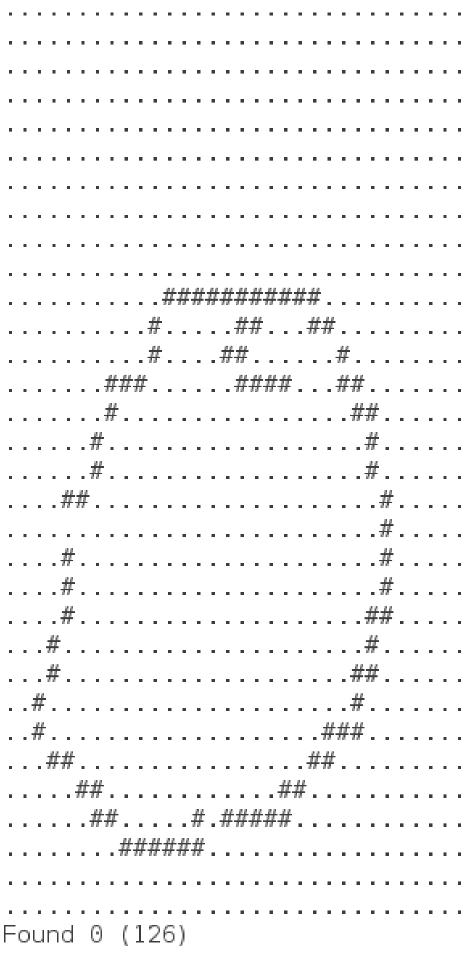
      iii. Linux → `ttyUSB<#>` or `ttyACM<#>`

6. Use the rightward arrow to upload / flash the code. Do not be alarmed if you see a series of orange warnings appear in the console. This is expected as we are working with bleeding edge code. You'll know the upload is complete when you red text in the console at the bottom of the IDE that shows 100% upload of the code and a statement that says something like "Done in <#.#> seconds."

   If you receive an error you will see an orange error bar appear and a red error message in the console. Don't worry -- there are many common reasons this may have occurred.

   To help you debug other issues please check out our FAQ appendix with answers to the most common errors!

7. Now open the serial monitor and test out your custom model. As a reminder, **the serial monitor will output first ASCII art showing the gesture you just performed and below it will be the best match label as well as a confidence score between -128 and 127**. The confidence score indicates how strongly the model believes you performed the gesture. Do note that every time you move the board and then stop a new gesture will be processed so don't be surprised to get some odd results as you move the board to prepare for a gesture.



```
................................
................................
................................
................................
................................
................................
................................
................................
................................
................................
...........###########..........
..........#.....##...##.........
..........#....##......#........
.......###......####...##.......
.......#................##......
......#..................#......
......#..................#......
....##....................#.....
..........................#.....
....#.....................#.....
....#.....................#.....
....#....................##.....
...#.....................#......
...#....................##......
..#.....................#.......
..#...................###.......
...##................##.........
.....##............##...........
......##.....#.#####............
........######..................
................................
................................
Found 0 (126)
```