CV PRO COMPETITION KIT

Figure 1 Fully Assembled Kit

SOFTWARE SETUP

Downloading files from GitHub:

Download Resource Package from the link provided below: To obtain and utilize Python packages, training code and training data.

https://github.com/robotixdevteam/CVPro-Competition-Kit

- Extract or clone the downloaded "*CVPRO-Competition-Kit*" file and save them in a dedicated folder for working with CVPRO-Competition files.
- In the extracted folder, you will be having **'.gitkeep'** files. Go to the extracted folder and search for '.gitkeep' files. *Delete those files permanently*.

Software Required:

The software applications required for working with 'Competition kit' are as follows:

1. Arduino IDE. Since we are using ESP32 board manager has to be installed. Refer to the link, for the installation steps.

2. Miniconda environment - Create a conda environment for CVPRO. Instructions on installing conda can be found here. The easiest way to create a new environment with all dependencies is to use one of the provided environment files. On Windows, you will also need to install Microsoft C++ Build Tools.

3. Android application - To operate the CV Pro competition kit, you need the corresponding Android application. Build the APK on your computer using the Build option in Android Studio; the code is already provided.

    a. Android Studio Giraffe | 2022.3.1 or later, click here for building and installing the Android Studio.

    b. Android device and Android development environment **with minimum API 29**.

    c. Currently, we use **API 33** as **compile SDK** and **API 33** as **target SDK**. It should get installed automatically, but if not, you can install the SDK manually. Go to **Android Studio --> Preferences --> Appearance & Behaviour --> System Settings --> Android SDK**. Make sure API 33 is checked and click apply.

Building the Application process

1. Open *Android Studio* and select Open an existing Android Studio project.

2. Select the ***CVPRO-Competition-kit/Application*** directory and click **OK**.

3. Confirm Gradle Sync if necessary. To perform a Gradle Sync manually, click on the Gradle icon.

4. Connect your Android device to computer using *USB Cable* and make sure USB Debugging in the developer options is enabled. Depending on your development environment further steps might be necessary. You should see your device in the navigation bar at the top now.

5. Click the Run button (**the green arrow**) or select ***Run > Run 'android'*** from the top menu. You may need to rebuild the project using **Build > Rebuild Project**.

6. If it asks you to use Instant Run, click **Proceed Without Instant Run**.

7. The app has three different options in home screen, **SAMPLE CHALLENGE**, **DATA COLLECTION** and **AUTONOMOUS**, with different functionalities. Refer to the Figure (2) given aside.
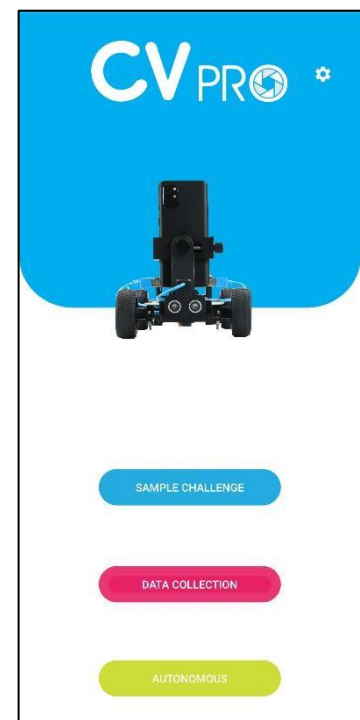

Figure 2 Home Screen

Upload Firmware to the Kit

To start, upload the **CVPRO-Competition-Firmware** into the competition kit, from the Firmware folder in your `*CVPRO-Competition-Kit*` repository.

How to connect the Phone and CV Pro Competition kit using connecting cable?

1. Connect the female-end of Type-C-OTG cable with male-end of Type-C-USB cable to establish a link.

2. Connect the Type-C end of Type-C-OTG cable with phone.

3. Connect the Type-C end of Type-C-USB cable with Type-C port on the PCB board.

4. If the connection process is correct, it will seek for permissions on your phone.

Sample Challenge

- Establish a connection between the phone and the CV Pro Competition kit using the connecting cable.

- Access the app's '*Home*' screen and select '**Sample challenge'**. Initiate the challenge by tapping on '**START'**. A countdown timer will then appear.

- The vehicle will autonomously navigate around the track within the specified 90second timeframe. Refer to the images provided.
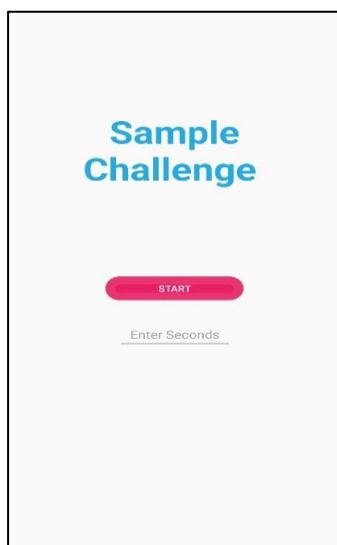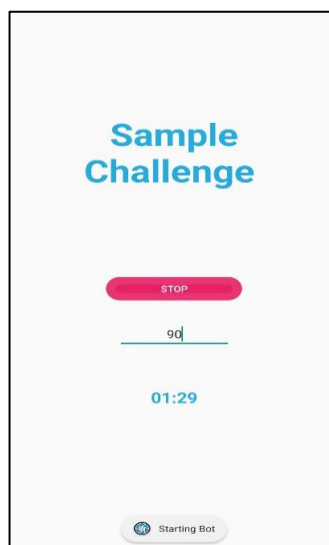

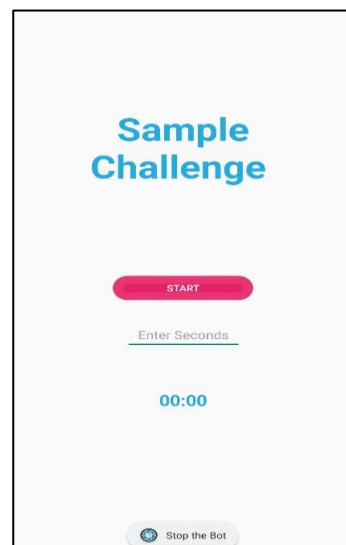*Figure 3(a) Sample challenge* *Figure 3(b) Countdown Timer in Progress* *Figure 3(c) Countdown Finished*

Environment Setup

First create a new conda environment with the following command:

conda create -n cvpro python=3.9 -y

Next, you need to activate your conda environment:

conda activate cvpro

Once your environment is active, install *TensorFlow*. For optimal training speed, especially on dedicated workstations or computers with dedicated GPUs, we recommend installing the necessary libraries and keeping your GPU drivers up to date. See below for TensorFlow installation commands on different Operating Systems.

Windows

pip install tensorflow~=2.9.0

GPU support for windows

conda install cudatoolkit=11.3 cudnn=8.2 -y

Linux

pip install tensorflow~=2.9.0

GPU support for Linux

sudo apt-get install nvidia-driver-510

conda install -c conda-forge cudatoolkit=11.2 cudnn=8.1 -y

echo 'export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$CONDA_PREFIX/lib/' >> ~/.bashrc

source ~/.bashrc

*Troubleshooting

Mac

conda install -c apple tensorflow-deps -y pip

install tensorflow-macos~=2.9.0

GPU support for Mac

pip install tensorflow-metal~=0.5.0

*[Troubleshooting](#)

Additional Requirements

Make sure you are in the folder within your local CVPRO-Competition-Kit repository.

Now, you can install all the remaining dependencies using the following command:

pip install -r requirements.txt

Notes

- Remember to activate the environment, before running commands in the terminal:

  `*conda activate cvpro*`

- If your tensorflow import does not work, try installing via

  `*pip install tensorflow --user*`.

Establishing Connections and Data Collection:

How to Set up the Local Connection:

For data collection follow the steps below:

1. Securely place your Android phone on the designated phone-mount within the competition kit and establish a connection with the kit *using the connecting cable.*

2. Power on the kit and grant the necessary permissions as prompted.

3. Activate the `**Mobile Hotspot**` option on your computer. Once your computer's hotspot name appears on your phone, initiate the connection process by entering the login credentials.

   Note: If you are unable to establish connection, repeat the steps 2-to-3.

4. To activate your environment,

   - For ***Windows***, Open the Anaconda Prompt,

     conda activate cvpro

   - For ***Linux/Ubuntu***, Open the Terminal,

     conda activate cvpro

   - For **Mac**, Open the Terminal,
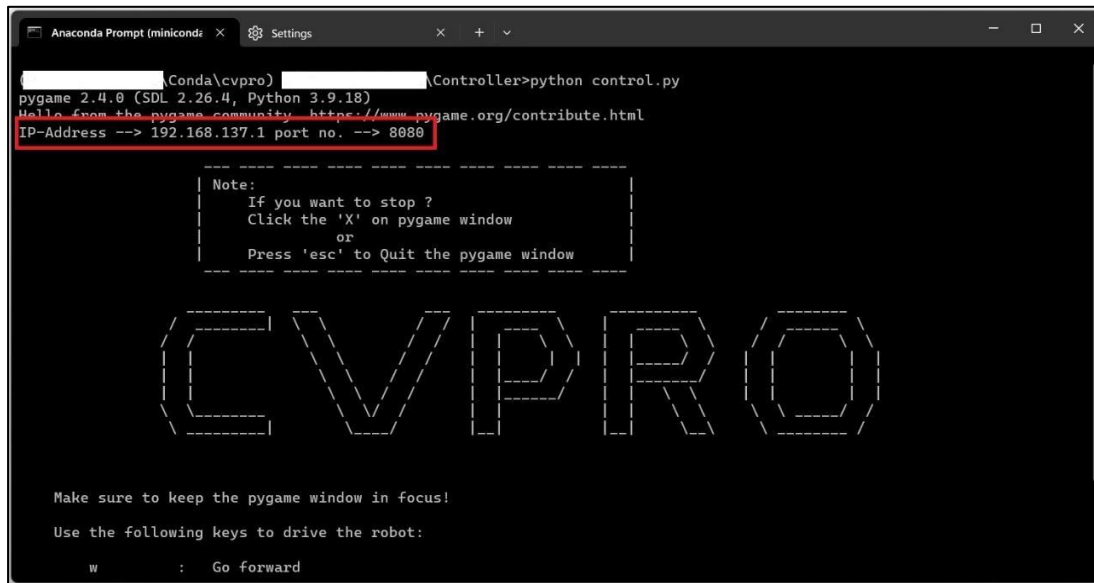
     conda activate cvpro

   Note: If you receive message `*EnvironmentNameNotFound*` after this step, create and activate the environment, by following steps in ′Environment Setup′.

5. Change to the `Controller` folder within the `CVPRO-Competition-kit` repository. Then, open the Anaconda prompt. Alternatively, you can use the command provided below to navigate to this path.

   - command >> cd *'path to the python file which is present in downloaded local `CVPRO-Competition-kit` repository in folder `Controller`'.*

   - Type or paste the following command in Anaconda Prompt, for *only Data Collection and bot control* use this command.

     python control.py

6. When the command `*python control.py*` is executed, a window will appear, displaying the IP address and port number. Refer to Figure (4) given below.
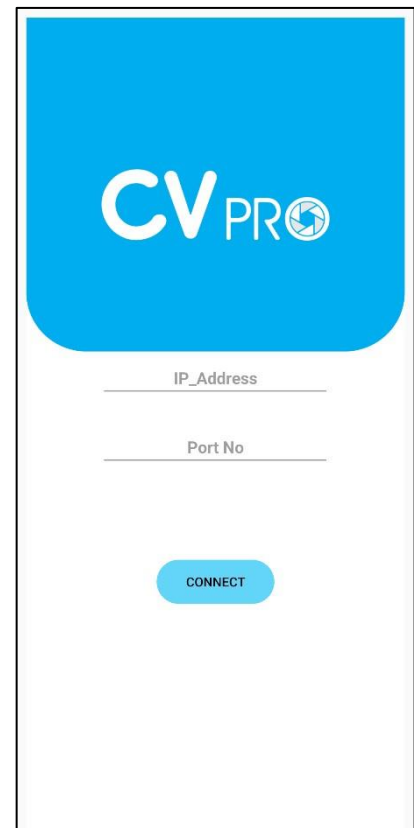
Figure 4 Control Window

7. Tap-on the `***Data collection***` option on the home screen of the `**CVPRO Competition**` application on your Android phone. *Allow the necessary permissions to take pictures and record videos*.

8. A screen will be presented for entering the IP address and port number. Input the `**IP_address**` and `**Port No**` shown on your computer into the respective fields on the IP address and port number screen of the `**CVPRO Competition**` application on your phone. Refer to Figure (5) given aside.

Figure 5 Login Credentials

9. Activate data collection on your competition kit app by toggling the `**Log**` button to the **on** position. Refer to Figure (6(a)) given below.

10. Gather data by maneuvering the kit along the track. Conclude the data collection process by toggling the `**Log**` button to the *off* position. You can repeat the data collection process as many times as you prefer. Refer to Figure (6(b)) given below.
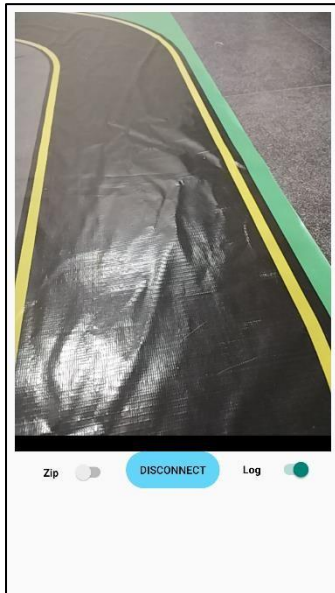

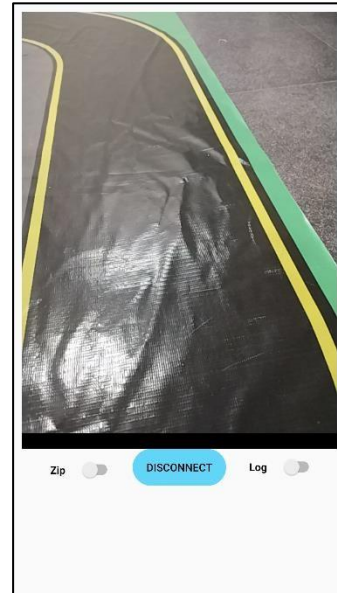Figure 6(a) Log ON Enabled

Figure 6(b) Log ON Disabled

11. Place the bot on the track and now you can control the bot as well as collect the data based on the control keys which will be displayed after running the above python command. Refer to Figure (7) given below.
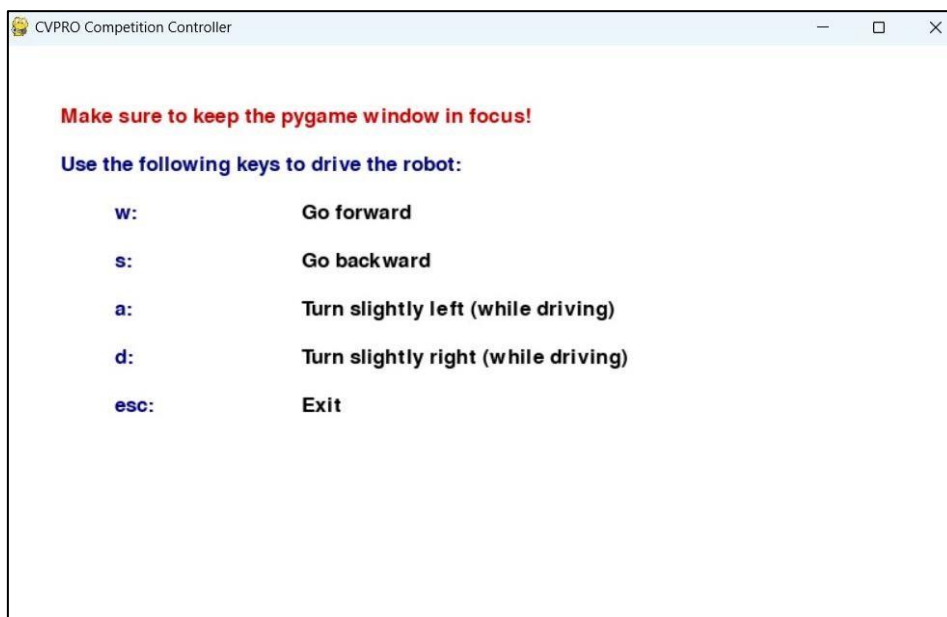

Figure 7 Pygame Window

12. Use the `**Zip**` button, that acts as a toggle button to compress the collected data. After compression is complete, toggle '***off***' the '*zip*' button.
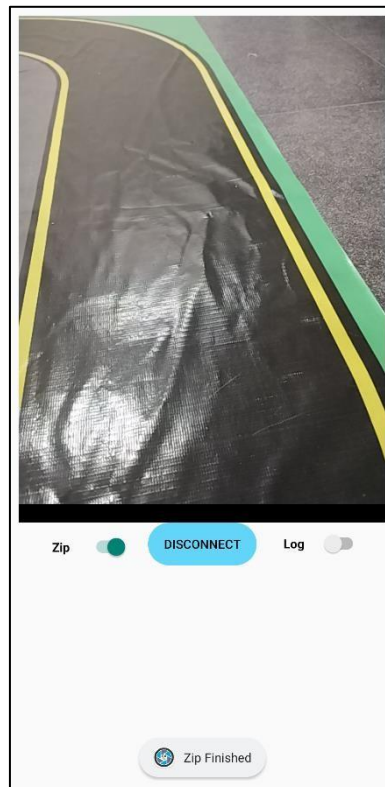

Figure 8 Zip enabled for compressing the data

13. To disconnect the connection between the computer and phone, tap on the `**Disconnect**` button and close the *pygame window* in your computer using '**Escape'** key or ***Close button in Window***.

14. Copy the collected data from the folder `**CVPRO_Competition**` in your smartphone to your computer. Open the local `***CVPRO-Competition-Kit***` repository and go to the location -> `**Training_Process\Training_Data\Dataset_CVPRO\**` and place it in `**Self_Driving**` folder to enable the movement of bot.

Note:

   Avoid pasting the compressed files directly. Make sure to extract the files first.

Training Process

> **Note:**
>
> You will be having **'.gitkeep'** files in CVPRO-Competition-Kit. Go to the extracted folder and search for '.gitkeep' files. Delete those files permanently.

1. Make sure your conda environment for CVPRO is activated by executing the following command:

   conda activate cvpro

2. The training process can be visualised either in *Python script or in jupyter notebook*:

   a. To start in the Python script, navigate to the folder, `*Training_Process*` within your local `*CVPRO-Competition-Kit*` repository. Type or paste the following command in Anaconda Prompt,

   python main.py

   b. To visualize the same using Jupyter notebook, you can use jupyter notebook file "*main.ipynb*". Type or paste the following command in Anaconda Prompt,

   jupyter notebook main.ipynb

3. After training process is complete, the script will generate two files (**best_model_labels.tflite and best_model_labels.txt**). The files are saved in the location `***Training_Process -> Training_Data -> Save_Model***`.

Model Management

1. Copy the files, `best_model_labels.tflite` and `best_model_labels.txt` and paste into the `**Assets**` folder in the Android studio application. Refer to Figure (9) given below.
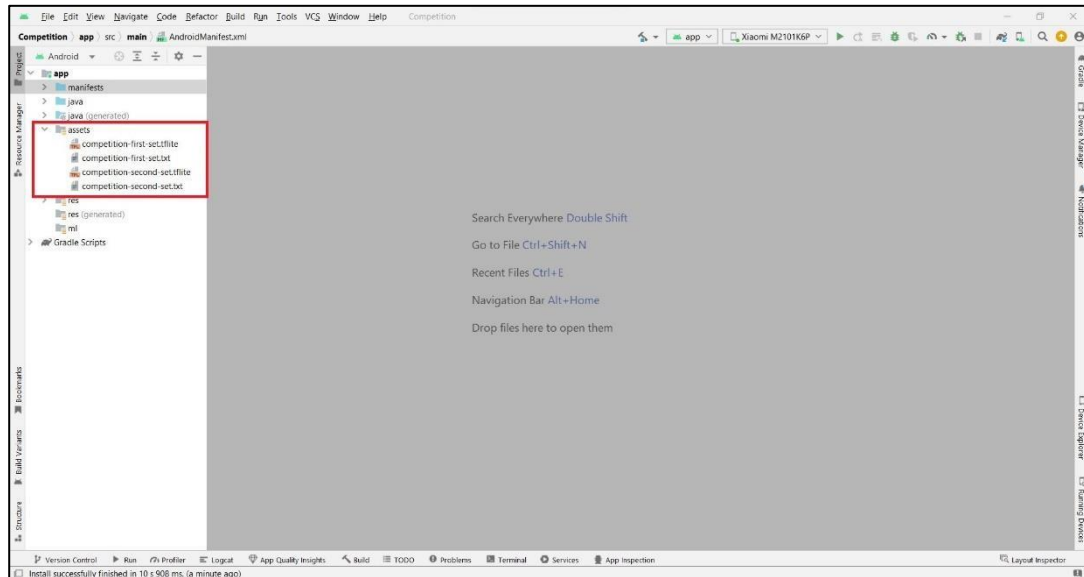

Figure 9 Android Studio Window to place zip file in Asset folder

2. Then, *connect your phone and computer using USB cable* and, for building the project, click on the `**Run**` button. The `**CVPRO Competition**` Application will be launched on your phone.

3. Upon launching the application, click the `**Settings**` button.

4. The `***Select Models***` screen will appear. Select the newly created model and click `**Save**` to store the model. Refer the Figure (10) to select model.
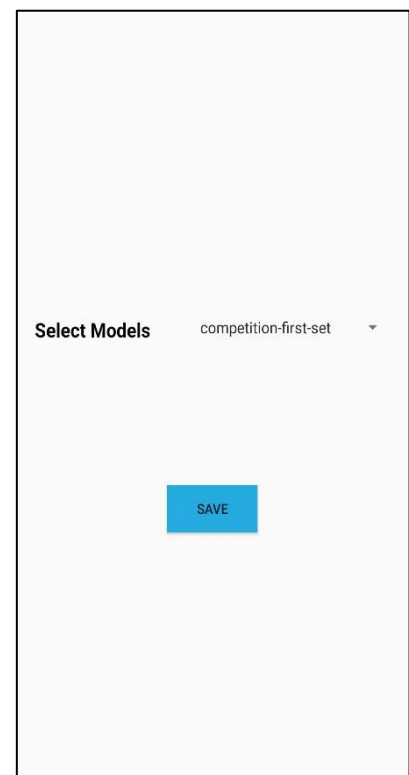

Figure 10 Select Model Screen

Autonomous

1. Access the '*Home*' screen of the app. Choose '**Autonomous**', and on the subsequent screen, pick the most recent model from the list of displayed models by tapping on '*Model Names*' within the '**Model**' option.

2. Enable autonomous navigation of the competition kit by tapping the `**USB**`*ON* button. The kit will navigate around the area where the data has already been collected when it is positioned there. Refer to Figure(11(a) & 11(b)) to select the model and tap the USB -ON Button.
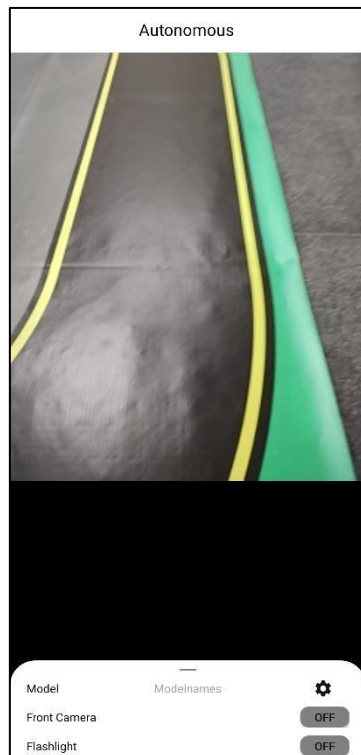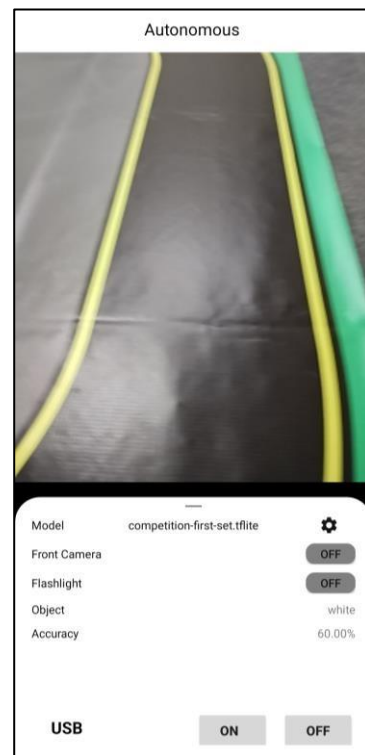

Figure 11(a) Autonomous Screen   *Figure 11(b) Tap ON Button*