



AWS - Confidential

Doc#: HDDZIDS-BD-101321-1

Human Detection and Danger Zone Intrusion Detection System User Manual Version 1.0 October 13, 2021

This documentation contains information of a proprietary nature. **All information contained herein shall be kept confidential.** None of this information shall be divulged to individuals other than AWS employees authorized by the nature of their duties to receive such information, or individuals or organizations authorized by AWS in accordance with existing policy regarding release of company information.



Revision History

Version	Modified By	Date	Changes
0.1	JMB	09-27-2021	Initial creation
1.0	JMB	10-13-2021	<ul style="list-style-type: none">• Add Shapely and Matplotlib installation guide• Update the path to run the main application

Table of Contents

I.	Overview	4
II.	Hardware Setup	4
III.	Setup Prerequisites	
	A. Software Requirement	4
	B. Prerequisite Installation in Jetson Xavier NX	4
	C. Edge Device Application Setup	6
	D. Mobile Application Setup	7
	E. Buzzer Device Setup	7
IV.	Running the Application	8
	A. Mobile Application	8
	B. Edge Device Application	12
	C. Buzzer Notification	13

I. Overview

This document describes how the Human Detection and Danger Zone Intrusion Detection System is used for end-users.

II. Hardware Requirements

Workstation
Jetson Xavier NX
8MP camera (or equivalent)
7" IPS capacitive touch display (or equivalent)
RPi-GPIO Buzzer
Android Device (OS: Android 8-Oreo or greater, Preferably Tablet Device)

III. Setup Prerequisites

A. Software Requirements

Software Requirements
DeepStream SDK 6.0 EA
Jetpack 4.5.1 GA
CUDA 11.1
TensorRT 7.2.2
NVIDIA driver 465.31
GStreamer 1.14.1
OpenCV 4
Jetson.GPIO
Paho Mosquitto 1.5.1
Python 3.6.9
Shapely 1.5.9
Matplotlib 2.1.1

B. Prerequisite Installation in Jetson Xavier NX

DeepStream SDK 6.0 EA

To download DeepStream SDK 6.0 EA from NVIDIA SDK, you should apply for early access. Get membership before you can download the installer including documentation.

1. Install Jetson SDK components (Jetpack SDK 4.5.1 GA)

Download NVIDIA SDK Manager from <https://developer.nvidia.com/jetpack-sdk-451-archive>. You will use this to install JetPack 4.5.1 GA (corresponding to L4T 32.5.1 release). This comes packaged with **CUDA**, **TensorRT** and **cuDNN**.

Follow this reference for the Developer Kit Guide for Jetson Xavier NX:

<https://developer.nvidia.com/embedded/learn/get-started-jetson-xavier-nx-devkit>

2. Install Dependencies

Enter the following commands to install the prerequisite packages:

```
$ sudo apt install \
libssl1.0.0 \
libgstreamer1.0-0 \
gstreamer1.0-tools \
gstreamer1.0-plugins-good \
gstreamer1.0-plugins-bad \
gstreamer1.0-plugins-ugly \
gstreamer1.0-libav \
libgstrtspserver-1.0-0 \
libjansson4=2.11-1
```

3. To install latest NVIDIA BSP packages

1. Open the apt source configuration file in a text editor, for example:

```
$ sudo vi /etc/apt/sources.list.d/nvidia-l4t-apt-source.list
```

2. Change the repository name and download URL in the deb commands shown below:

```
deb https://repo.download.nvidia.com/jetson/common r32.5 main
deb https://repo.download.nvidia.com/jetson/<platform> r32.5
main
```

Where <platform> identifies the platform's processor:

- t186 for Jetson TX2 series
- t194 for Jetson AGX Xavier series or Jetson Xavier NX
- t210 for Jetson Nano or Jetson TX1

For example, if your platform is Jetson Xavier NX:

- deb https://repo.download.nvidia.com/jetson/common r32.5 main
- deb https://repo.download.nvidia.com/jetson/t194 r32.5 main

3. Save and close the source configuration file.

4. Enter the commands:

```
$ sudo apt update
```

5. Install latest NVIDIA V4L2 Gstreamer Plugin using the following command:

```
$ sudo apt install --reinstall nvidia-l4t-gstreamer
```

If apt prompts you to choose a configuration file, reply Y for yes (to use the NVIDIA updated version of the file)

6. Install latest L4T MM and L4T Core packages using following commands:

```
$ sudo apt install --reinstall nvidia-l4t-multimedia
$ sudo apt install --reinstall nvidia-l4t-core
```

Note: Updating NVIDIA V4L2 GStreamer plugin should be performed after flashing Jetson OS from SDK Manager.

4. Install the DeepStream SDK (Using the DeepStream Debian package)

Download the DeepStream 6.0 Jetson Debian package

deepstream-6.0_6.0.0-1_arm64.deb:

<https://developer.nvidia.com/deepstream-sdk-6.0-early-access>, to the Jetson device. Then enter the command:

```
$ sudo apt-get install ./deepstream-6.0_6.0.0-1_arm64.deb
```

OpenCV 4

Install Opencv using the below command:

```
$ sudo apt install python3-opencv
```

Jetson.GPIO

Install using the below command to control Jetson GPIO channels:

```
$ sudo pip install Jetson.GPIO
```

Paho Mosquitto 1.5.1

Install MQTT Client and Broker using the below command:

```
$ pip install paho-mqtt
```

```
$ sudo apt-add-repository ppa:mosquitto-dev/mosquitto-ppa
```

```
$ sudo apt-get update
```

```
$ sudo apt-get install mosquitto
```

Shapely 1.5.9

Install Shapely using the below command:

```
$ pip install Shapely
```

If errors are encountered related to geos_c, install libgeos-dev.

```
$ sudo apt-get install libgeos-dev
```

Matplotlib 2.1.1

Install Matplotlib using the below command:

```
$ sudo apt-get update
```

```
$ sudo apt-get install python3-matplotlib
```

C. Edge Device Application Setup

Human Detection and Danger Zone Intrusion Detection application is configured to work with DeepStream SDK 6.0 EA.

1. Extract the dangerzone_beta_rel.zip in the deepstream root directory

Deepstream directory:

```
/opt/nvidia/deepstream/deepstream-6.0/sources/deepstream_python_apps/apps
```

Note: Make sure that the extracted folder is hddzids, this is the main folder.

2. Add user permission to the extracted folder using this command:

```
$ sudo chmod -R 777 ./hddzids
```

3. Edit the MOBILE_HOST constant in `/hddzids/dist/common/constant.py` with the IP Address of the Jetson device. *(This is necessary for the jetson-mobile app connection via mqtt)*

```
Ex. MOBILE_HOST = "192.168.1.17"
```

D. Mobile Application Setup

Please download the mobile application APK in your Android phone and allow it to be installed. (Included in the release package: `Mobile/danger-zone-final.apk`)

APK installation:

1. Download the `danger-zone-final.apk` file on a mobile device.
2. Open the APK file and click Install
3. Click Install anyway
4. Click Don't send.
5. Mobile App is successfully installed and is now ready to use. Click Open to open the application.

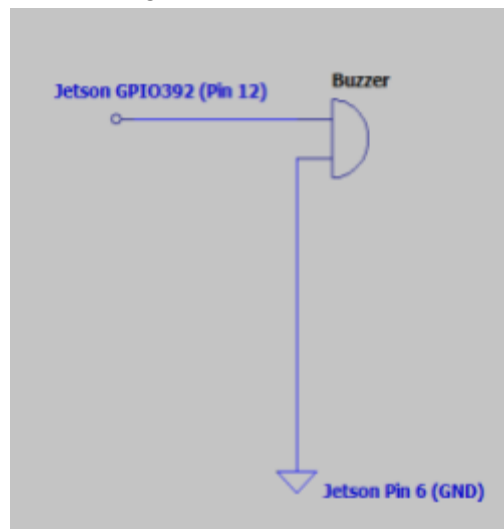
Note: APK installation may vary depending on the mobile device in use. The main goal is to install the apk on your mobile device and be able to run the app.

E. Buzzer Device Setup

Hardware Requirements:

- Jetson Device
- Active Buzzer
- Breadboard
- Connecting Wires

1. Active Buzzer will be directly connected to the Jetson Pins. Follow the Schematic Diagram:



IV. Running the Application

A. Mobile Application

Receiving Alert Message:

Precondition:

- Start the mqtt message broker service in jetson device, run command:
`$ sudo service mosquitto start`
- Mobile and Jetson Xavier must be connected in the same area network.
- Note: Stopping the mosquitto message broker service will disable mobile connections to all client applications. To stop run command on jetson device:
`$ sudo service mosquitto stop`

1. Open the application. (*App Name: Human Detection and Danger Zone Intrusion*)
2. Enter the IP address of Jetson Xavier NX and toggle on the switch button to connect to the server. "Connected" status should be displayed.

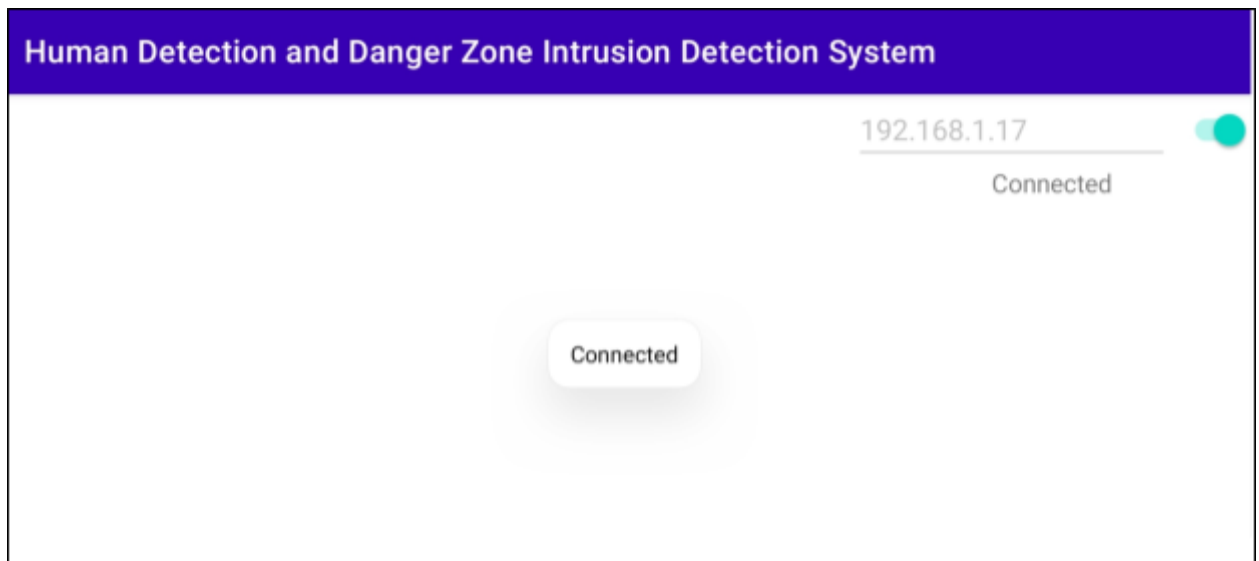


Figure 1: Connecting to server

If you wish to disconnect, toggle off the switch button to disconnect to the server. However, you will not be able to receive any alert from the Jetson device. "Disconnected" status should be displayed.

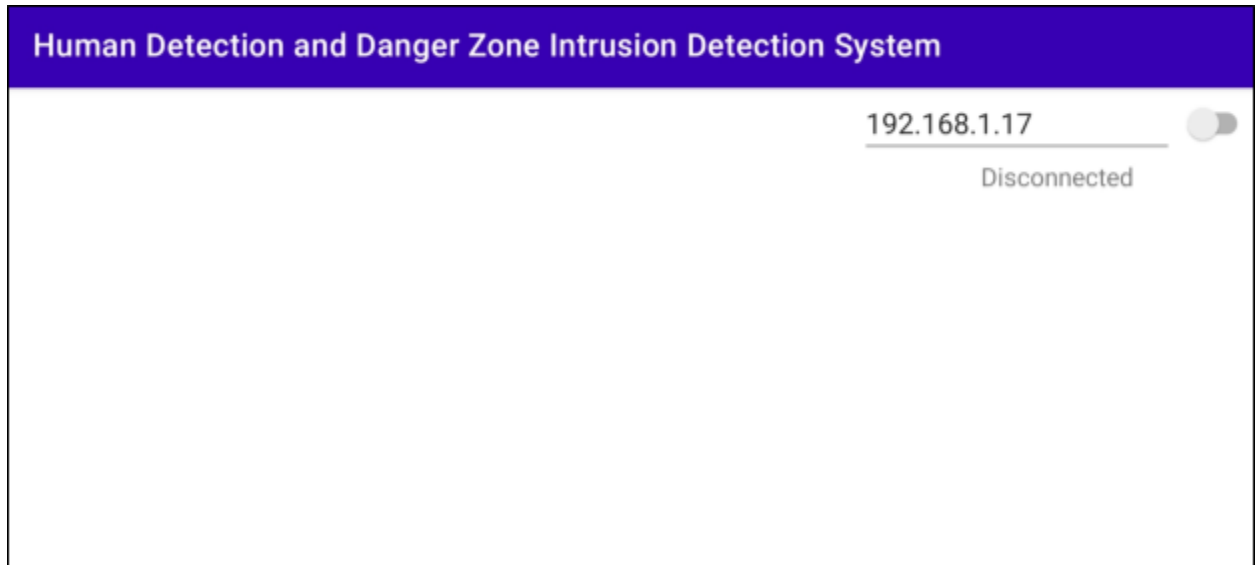


Figure 2: Disconnecting to server

3. Run the Edge Device Application (**See: B. Edge Device Application**)
4. Wait for grid display and alert notification to be received.

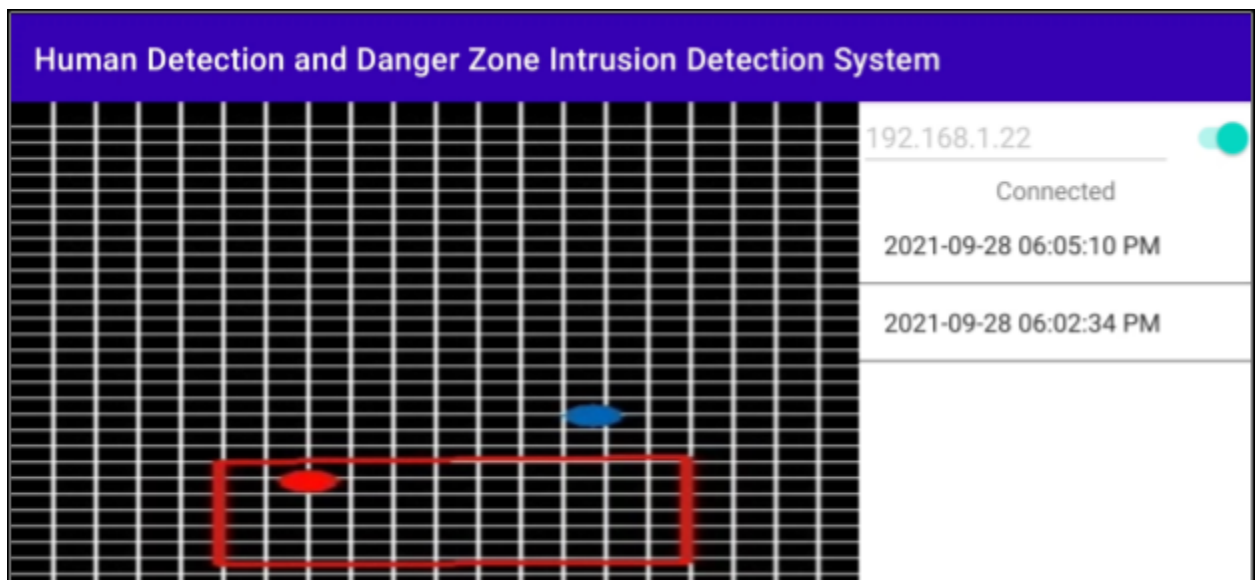


Figure 3: Receiving Grid Map and Alert Messages

5. Select (Press) item in the alert message list and it will transition to the next screen to display the captured image of detected object in the danger zone.

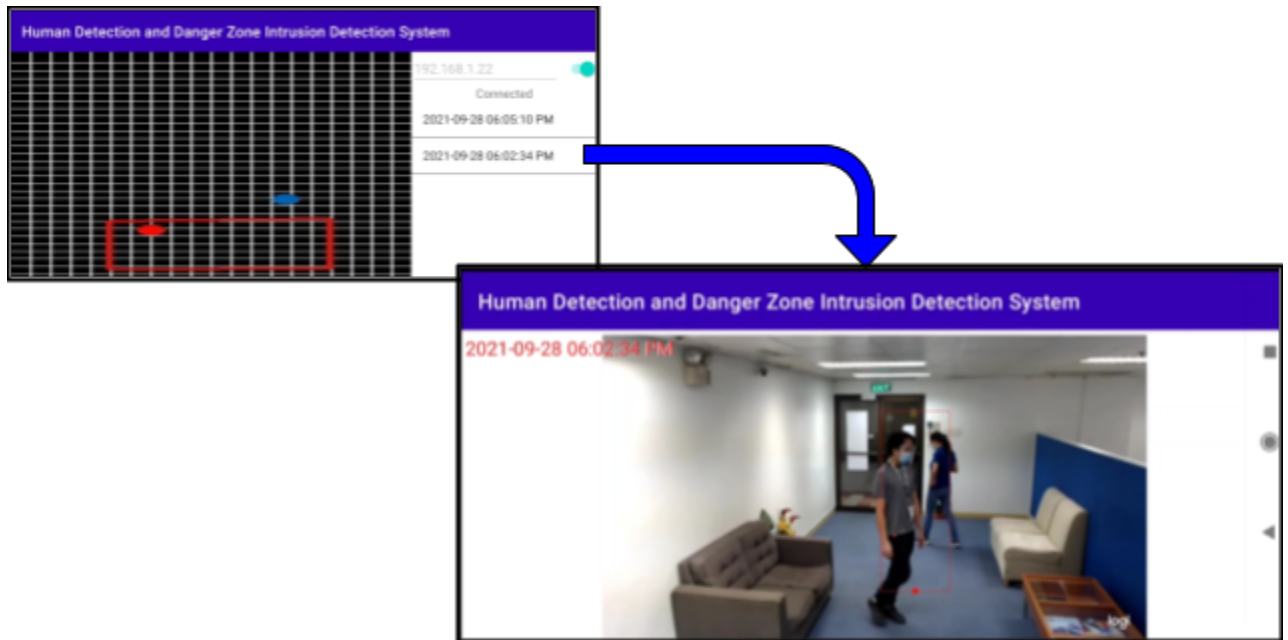


Figure 4: Alarm Message Image

6. Buzzer will alarm. (See: **C. Buzzer Notification**)

Receiving Push Notification:

1. In the mobile home screen, wait for the notification message to be received.
(* Mobile app is already connected to the edge device)

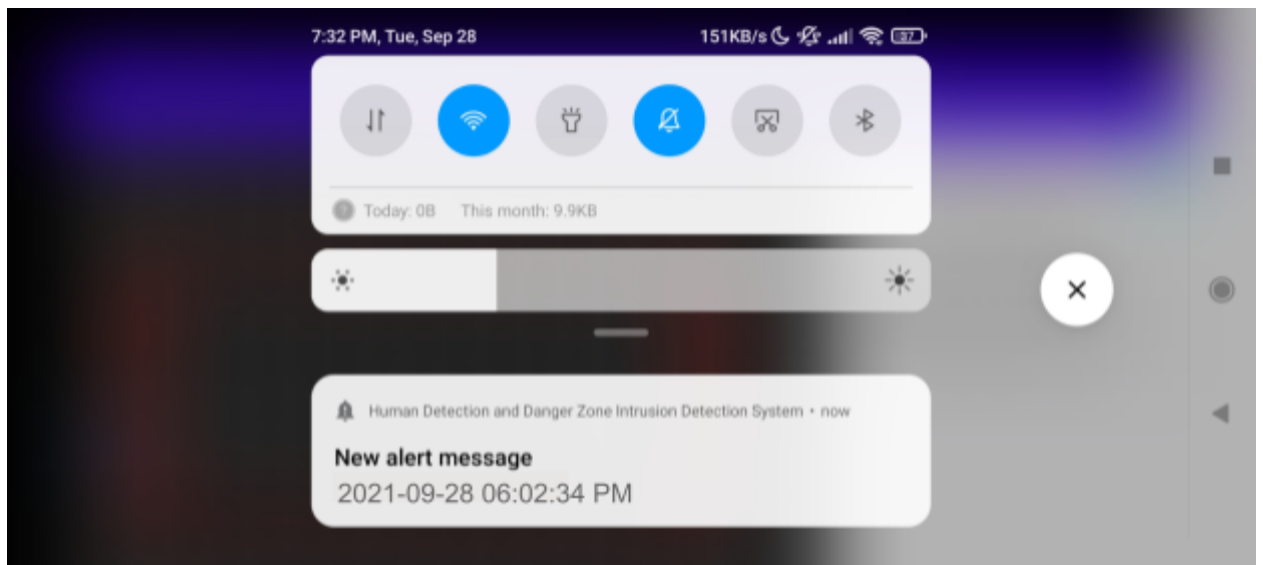


Figure 5: Alert Notification Message

2. When alert notification is received, press the notification message.
3. Mobile application will open and it will be redirected to the screen of captured image with detected object in the danger zone.

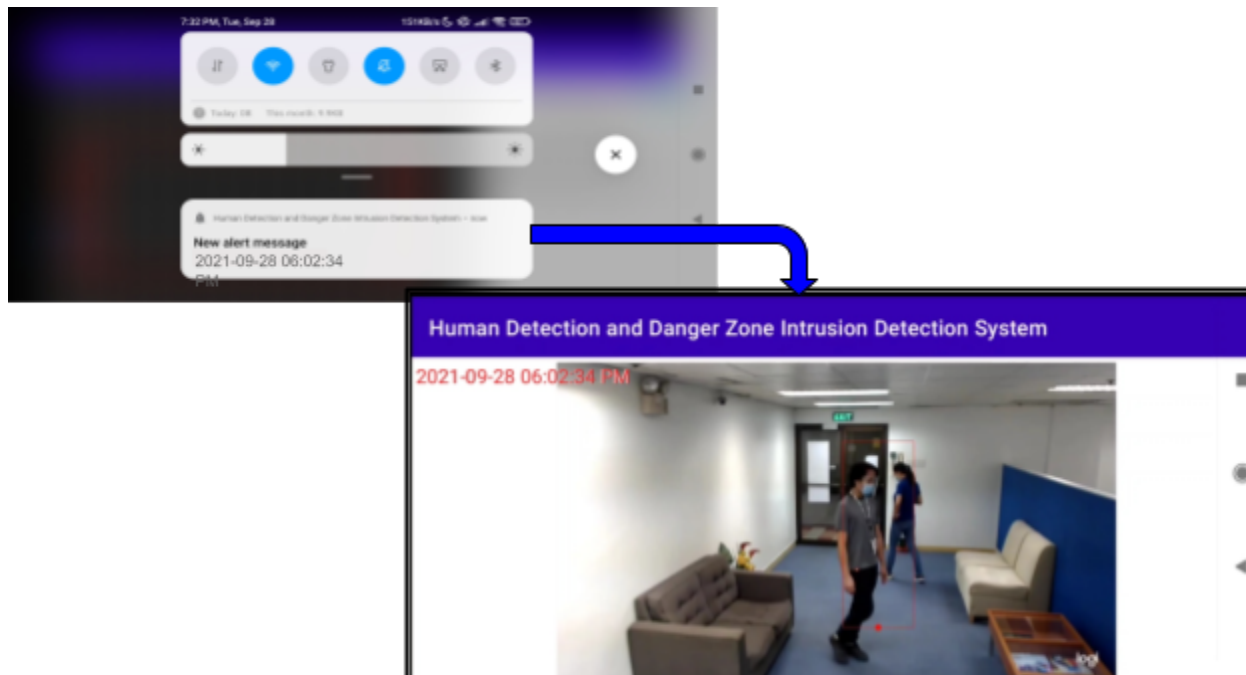


Figure 6: Alarm Message Image from Push Notification

Deletion of Alert Message:

1. Long press an item on the alert message list
2. Delete confirmation message box will display.

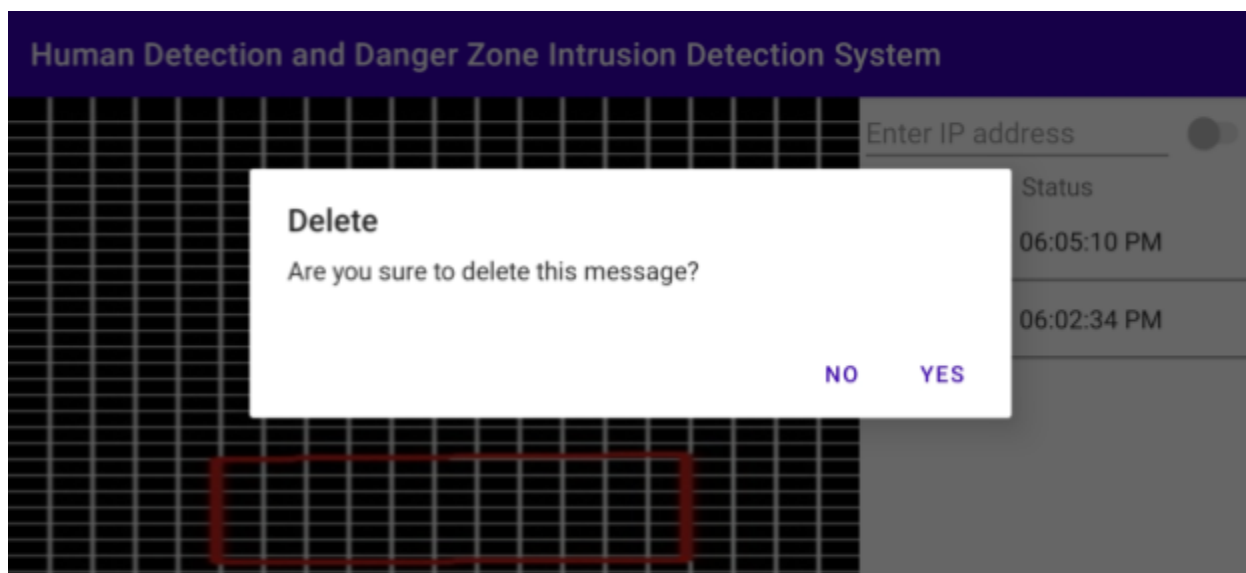


Figure 7: Delete Alert Message

3. Press 'No' to return to the main screen
4. Press 'Yes' to delete the message and update the list

B. Edge Device Application

1. Open a new terminal window and redirect to the application directory.
2. In the terminal window, enter the following command to run the application:

```
$ python3 main.py
```
3. Wait until the camera stream window displays for Bird's eye view and Danger Zone calibration.
4. In the camera stream window, do Bird's Eye View calibration by mouse drag and drop to select the initial area. (Corners are adjustable to align the bird's eye view area.)

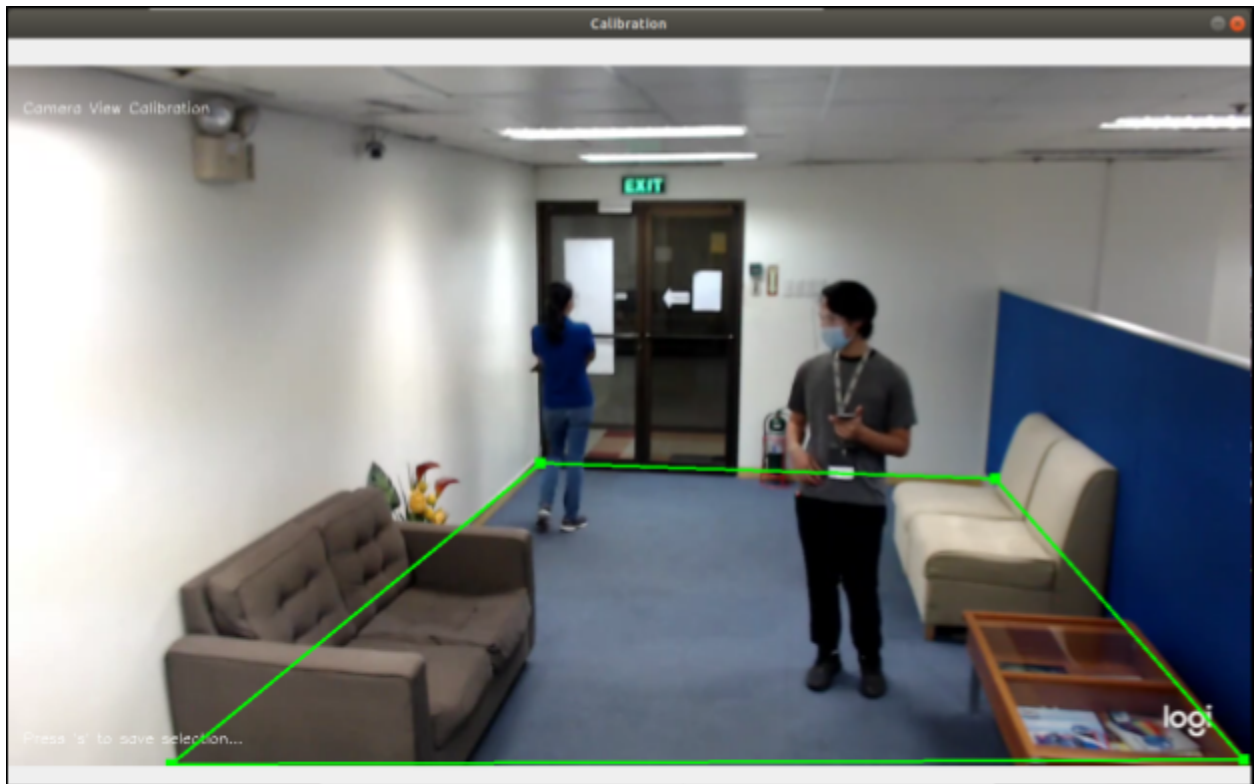


Figure 8: Bird's Eye View Calibration

5. Press 's' key to save the selected area for Bird's Eye view calibration
6. Next, do the Danger Zone calibration also by mouse drag and drop inside the initial area selected in Bird's Eye view calibration. (Corners are also adjustable to align the danger zone area.)

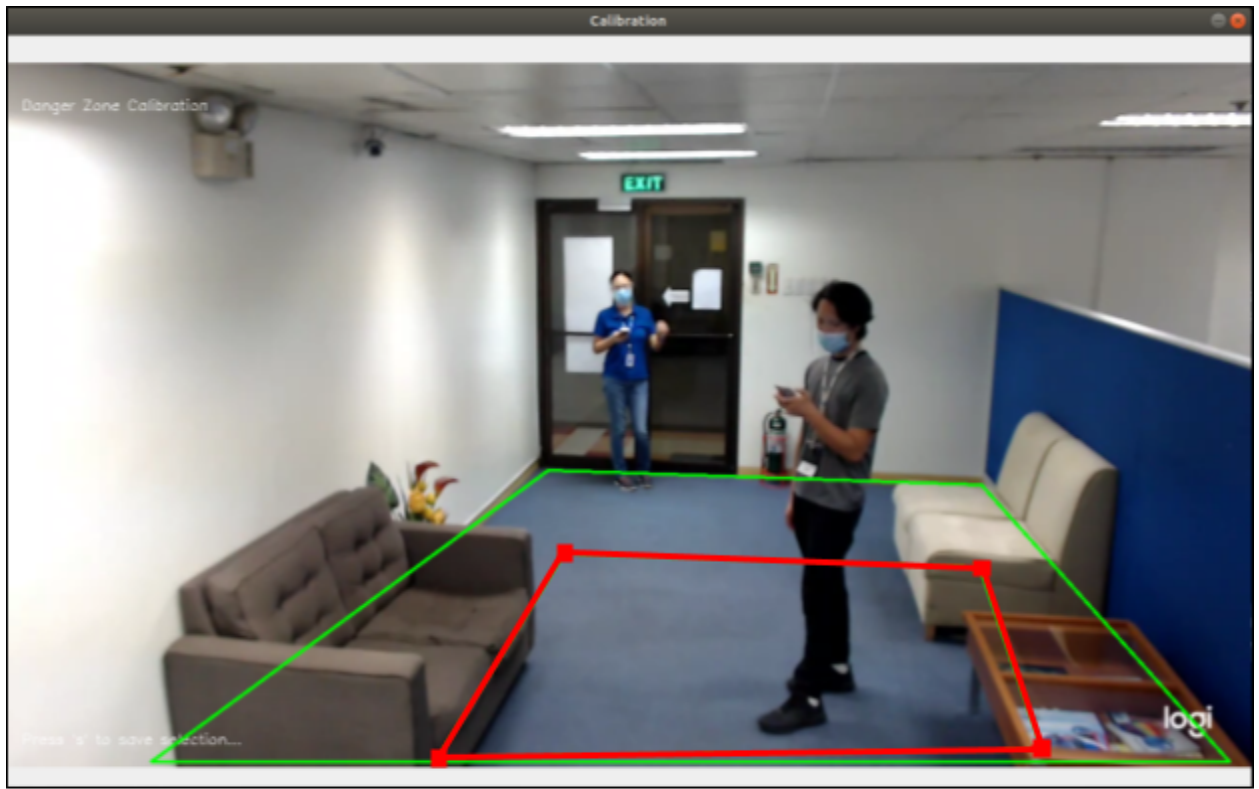


Figure 9: Danger Zone Calibration

7. Press 's' key to save the selected area for Danger Zone calibration
8. Press 'q' key to continue detection and tracking.
9. Windows will close and the app will continue running for object detection and tracking. (Display of the result will be seen on the Mobile Application)

Note:

- a. Calibration window will only display when there's no existing calibration beforehand.
- b. You can retry to do the calibration by deleting the calibration file save in `/hddzids/data/calibration.txt`

C. Buzzer Notification

1. When the object detected was in the danger zone and reached the maximum dwell time limit (as of 5 seconds), the alarm will turn-on and will make a sound.
2. When the object went outside the danger zone, the alarm would turn-off.