

MISSING SUBSTANCE DETECTION

COOKBOOK 2023-24

Table of Contents:

- [Introduction](#)
- [Requirements](#)
 1. [Python Libraries Required](#)
- [Steps to get started](#)
 1. [Python](#)
 2. [App Installation](#)
- [Using TIFR Machine](#)

Introduction:

In recent years, the escalating rate of vehicle thefts has emerged as a significant societal concern, posing economic losses and public safety threats on multiple fronts. This problem is particularly pronounced in urban areas, where dense populations and extensive surveillance systems coexist. Law enforcement agencies grapple with managing the overwhelming surveillance data generated by Closed-Circuit Television (CCTV) systems. As vehicles go missing or are stolen, investigators face the daunting challenge of swiftly and accurately identifying, locating, and recovering these missing assets.

Requirements:

- Python 3.8

Python Libraries Required:

The major libraries required are:

- PyQt5
- lxml
- bcrypt
- psutil
- python-dateutil
- Qt-Material
- QtWidgets
- opencv-python
- numpy
- enchant
- ultralytics
- torchvision
- torch
- tensorflow
- scipy
- scikit-image
- pillow
- pandas
- numpy
- easyocr

Steps to get started:

1. Python

Python (version 3.8.10) can be downloaded and installed from [here](#) for Windows-based systems. In most cases, it is already installed on Linux distros. If not:

Python installation steps for Ubuntu:

1. Open a new terminal.
2. Update the packages list and install the packages necessary to build Python:

```
$ sudo apt update
```

```
$ sudo apt install build-essential zlib1g-dev  
libncurses5-dev libgdbm-dev libnss3-dev  
libssl-dev libreadline-dev libffi-dev  
libsqlite3-dev wget libbz2-dev
```

3. Download the latest release's source code from the [Python download page](#) using [wget](#) :

```
$ wget  
https://www.python.org/ftp/python/3.8.12/Python-3.8.12.tgz
```

4. When the download finishes, [extract the gzipped archive](#):

```
$ tar -xf Python-3.8.12.tgz
```

5. [Switch](#) to the Python source directory and execute the configure script which performs several checks to make sure all of the dependencies on your system are present:

```
$ cd Python-3.8.12  
  
$ ./configure --enable-optimizations
```

6. Start the Python 3.8 build process:

```
$ make
```

7. When the build process is complete, install the Python binaries by typing:

```
$ sudo make altinstall
```

8. That's it. Python 3.8 has been installed and ready to be used. Verify it by typing:

```
$ python3.8 --version
```

9. The output should show the Python version installed.

Python installation steps for CentOS:

1. Open a terminal.
2. Install Python Dependencies

```
sudo yum -y install epel-release  
sudo yum -y update
```

3. Reboot after the upgrade

```
sudo reboot
```

4. After the reboot, install the build dependencies

```
sudo yum -y groupinstall "Development Tools"  
sudo yum -y install openssl-devel bzip2-devel  
libffi-devel xz-devel
```

5. Confirm GCC is available:

```
gcc --version
```

6. Download the latest Python 3.8 Archive

```
sudo yum -y install wget  
wget  
https://www.python.org/ftp/python/3.8.12/Python-3.8.12.tgz
```

7. Extract the package.

```
tar xvf Python-3.8.12.tgz
```

8. Change the created directory:

```
cd Python-3.8*/
```

9. Set up installation by running the configure script.

```
./configure --enable-optimizations
```

10. Initiate compilation of Python 3.8 on CentOS 7.

```
sudo make altinstall
```

11. If this was successful, you should get a message like below:

```
Installing collected packages: setuptools, pip
```

```
Successfully installed pip-19.2.3 setuptools-41.2.0
```

12. Confirm that the installation of Python 3.8 on CentOS 8 / CentOS 7 was successful.

```
python3.8 --version
```

13. The output should show the Python version: Python 3.8.12

2. App Installation

We recommend creating a virtual environment to install the application.

1. Download the ZIP file of code from this [link](#).
2. Extract the downloaded ZIP file.
3. Open a terminal.
4. Install Python Virtual Environments if not installed.

```
$ sudo apt install -y python3-venv
```

5. Navigate to the directory where the application has been cloned.
6. Create a virtual environment and activate it.

```
$ python3 -m venv <venv-name>
$ <venv-name>\Scripts\activate (for Windows)
$ source <venv-name>/bin/activate (for Linux)
```

- Add the name of your virtual environment in place of <venv-name>

7. Run the following command to install all requirements:

```
$ pip3 install -r requirements.txt
```

8. Launch the application by executing the following command in the application's root directory:

```
$ python3 integrate.py
```

9. The application should start executing.

Using TIFR Server:

Steps for accessing the **TIFR GPU** machine:

1. Open a terminal and enter the following commands -
 - ssh -p 4748 -tt sipm@gwssh.tifr.res.in
 - Enter password 1
 - ssh -p 22 -tt msd_group@<ip-address>
 - Enter password 2
2. Below is the link to the document containing commands for using the GPU - [TIFR GPU Commands](#)