



Vehicle Event Recording Reference Implementation

Version: 2021.3 Published: 04/12/2021

Last Updated: 12/17/2021

Overview

The Vehicle Event Recording Reference Implementation (RI) applies Artificial Intelligence (AI) to monitor the exterior of a vehicle and send events to the cloud dashboard. This includes event-based recording, remote view, driver coaching, and traffic violation detection features. The information is used for historical analysis, evidence support, and driver coaching by fleet management.

The RI includes an interface for the driver to be able to review events and check camera status on demand when the vehicle is not moving. The RI also sends event notifications and video clips to Amazon Web Services* (AWS*), which can be displayed on the cloud dashboard.

Select **Configure & Download**

implementation and the software listed below.

[Configure & Download](#) to download the reference

[Configure & Download](#)



Time to Complete

Approximately 60 minutes

Programming Language

Python*

Available Software

Intel® Distribution of OpenVINO™ toolkit 2020 Release

Recommended Hardware

The below hardware is recommended for use with this reference implementation. See the Recommended Hardware page for other suggestions.

- ADLINK MXE-5500 Series
- NEXCOM VTC 7252-7C4IP

Target System Requirements

- Ubuntu* 18.04.02
- 6th to 10th Generation Intel® Core™ processors with Intel® Iris® Plus graphics or Intel® HD Graphics

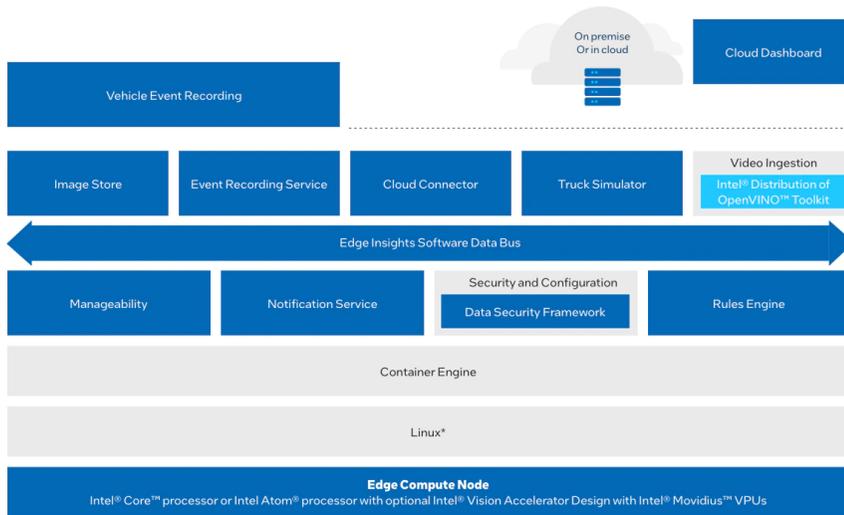
How It Works

Vehicle Event Recording utilizes external facing cameras to detect objects and provide event-based video recording, remote view and driver coaching, and traffic violation detection features.

The RI is used for historical analysis, evidence support and driver coaching by fleet management.

Vehicle Event Recording has an interface for the driver to be able to review events and check camera status on demand when the vehicle is not moving.

The RI sends event notifications and video clips to AWS, which can be displayed on a dashboard.



Get Started

Step 1: Install the Reference Implementation

Select **Configure & Download** implementation and then follow the steps below to install it.

[to download the reference](#)

[Configure & Download](#)

NOTE: If the host system already has Docker images and containers, you might encounter errors while building the reference implementation packages. If you do encounter errors, refer to the Troubleshooting section at the end of this document before starting the reference implementation installation.

1. Open a new terminal, go to the downloaded folder and unzip the downloaded RI package.

```
1 | unzip vehicle_event_recording.zip
```

2. Go to the `vehicle_event_recording/` directory.

```
1 | cd vehicle_event_recording/
```

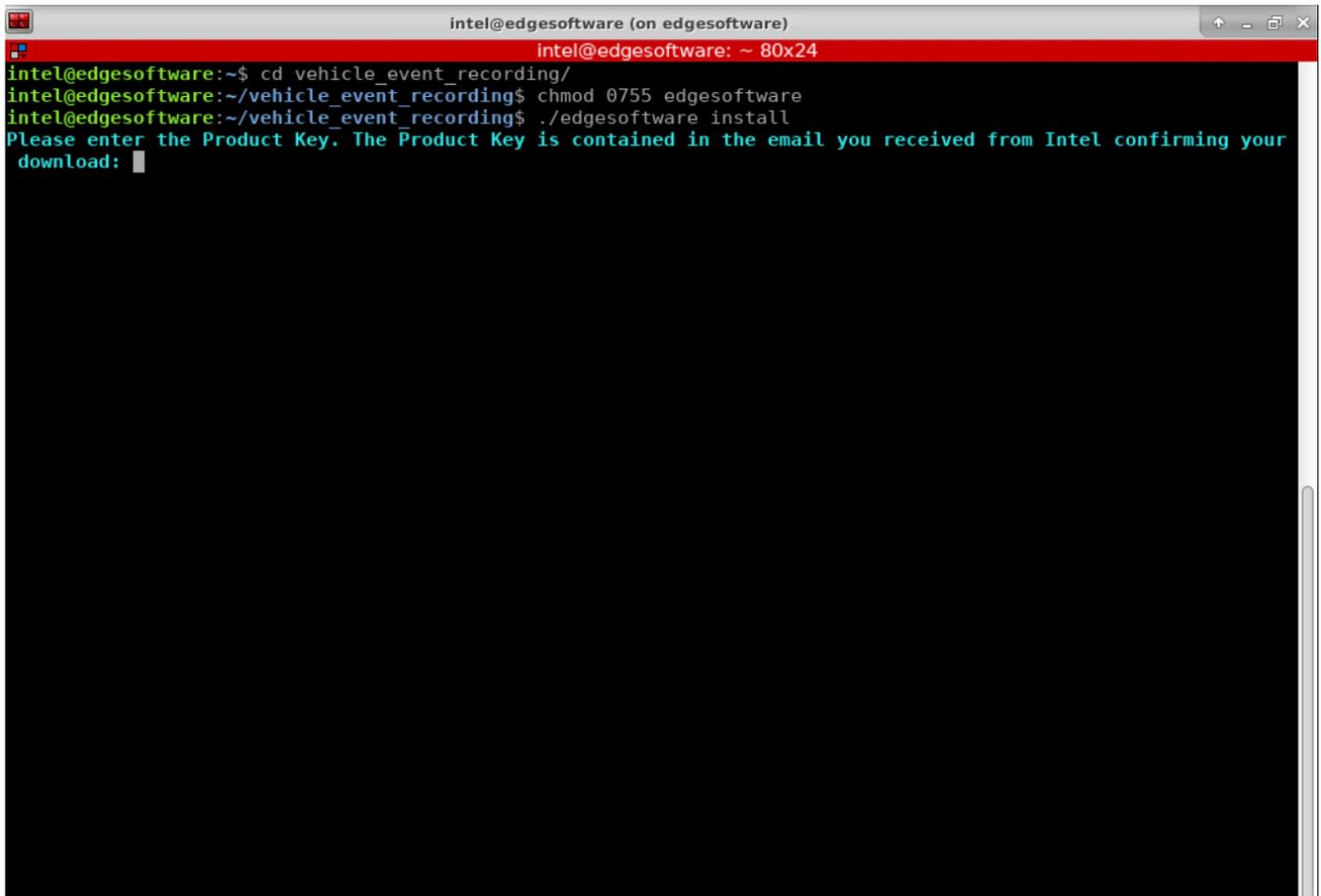
3. Change permission of the executable `edgesoftware` file.

```
1 | chmod 755 edgesoftware
```

4. Run the command below to install the Reference Implementation.

```
1 | ./edgesoftware install
```

5. During the installation, you will be prompted for the **Product Key**. The **Product Key** is contained in the email you received from Intel confirming your download.



```
intel@edgesoftware (on edgesoftware)
intel@edgesoftware: ~ 80x24
intel@edgesoftware:~$ cd vehicle_event_recording/
intel@edgesoftware:~/vehicle_event_recording$ chmod 0755 edgesoftware
intel@edgesoftware:~/vehicle_event_recording$ ./edgesoftware install
Please enter the Product Key. The Product Key is contained in the email you received from Intel confirming your
download: █
```



```
1 | make webui v=4 EII_BASE=<INSTALL_PATH>/vehicle_event_recording/Vehicle_Event_Recording_2021.3/IEdgeInsights
  | REPO_FOLDER=<INSTALL_PATH>/vehicle_event_recording/Vehicle_Event_Recording_2021.3/Vehicle_Event_Recording/EII-EVMSC-
  | UseCase
```

For example:

```
1 | make webui v=4 EII_BASE=/home/intel/vehicle_event_recording/Vehicle_Event_Recording_2021.3/IEdgeInsights
  | REPO_FOLDER=/home/intel/vehicle_event_recording/Vehicle_Event_Recording_2021.3/Vehicle_Event_Recording/EII-EVMSC-
  | UseCase
```

2. Open the Web UI and go to **127.0.0.1:9091** on your web browser.

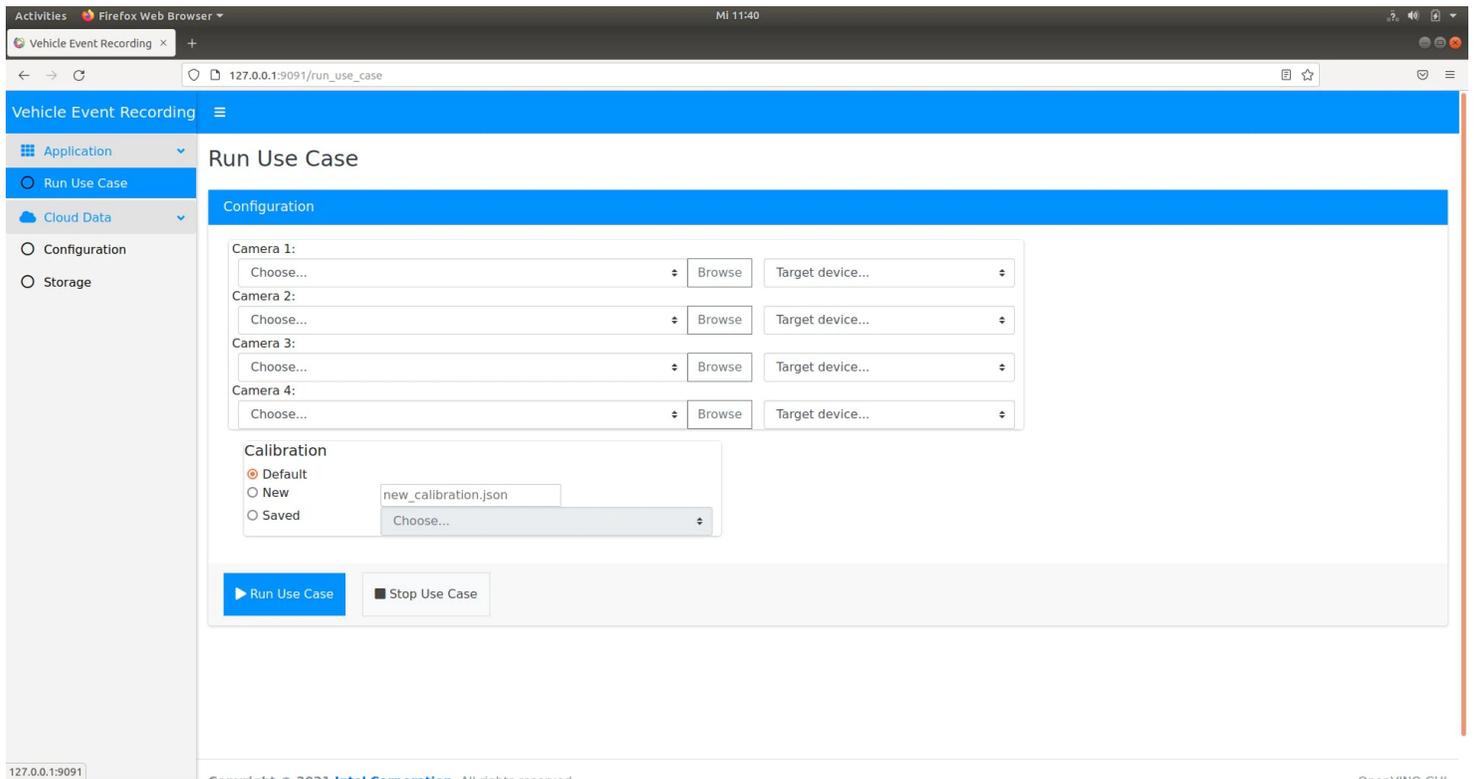


Figure 4: Web UI

3. If you installed your ThingsBoard Local Cloud Server and you have enabled S3 Bucket Server on your AWS account, you can provide your configured **AWS Access Key ID**, **AWS Secret Access Key**, **Thingsboard IP**, **Thingsboard Port** and **Thingsboard Device token** on the **Cloud Data Configuration** tab. After you complete the Cloud configuration, make sure you click on the **Save Credentials** and **Save Token** buttons. Now you can import the ThingsBoard dashboard as described at the end of the Set Up ThingsBoard* Local Cloud Data

to enable all dashboard features, including the cloud storage.

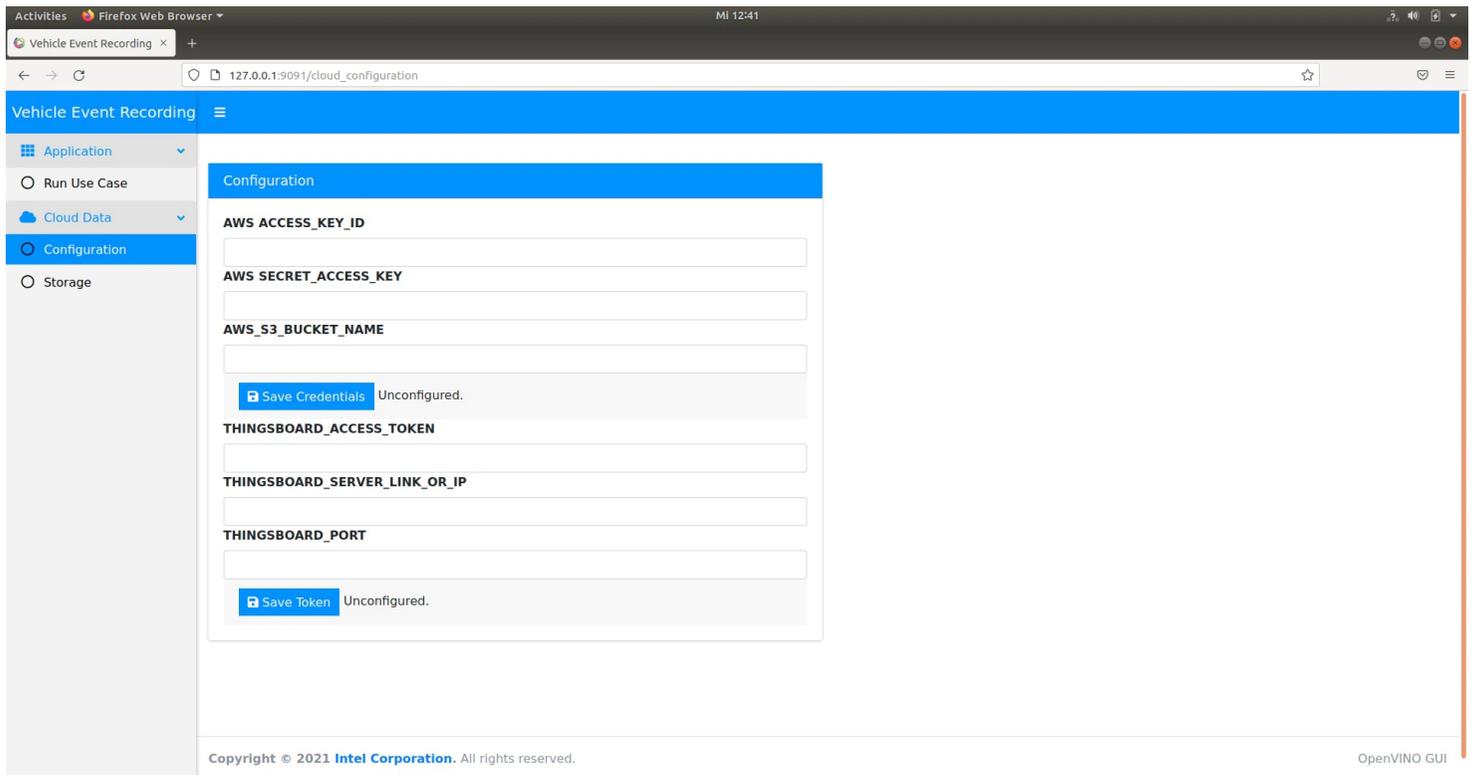


Figure 5: Configure AWS Login Information

NOTE: If you don't have an AWS account, you can still enable the ThingsBoard Cloud Data.

4. Access the Video Event Recording Dashboard with the following steps:

- Go to sidebar and select **Run Use Case**.

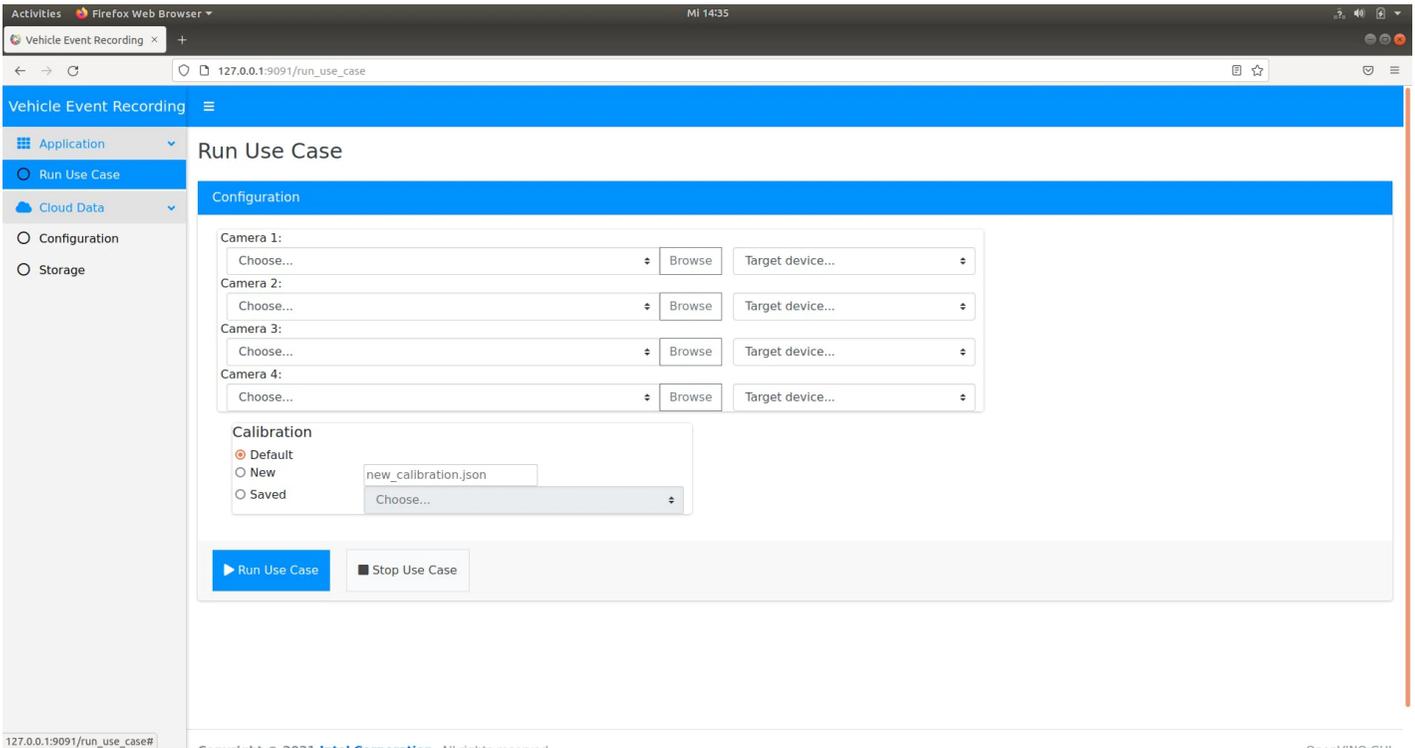


Figure 6: Select Use Case

- Configure the use case. Select video sample or multiple video samples and check CPU or GPU device to run the use case on it.

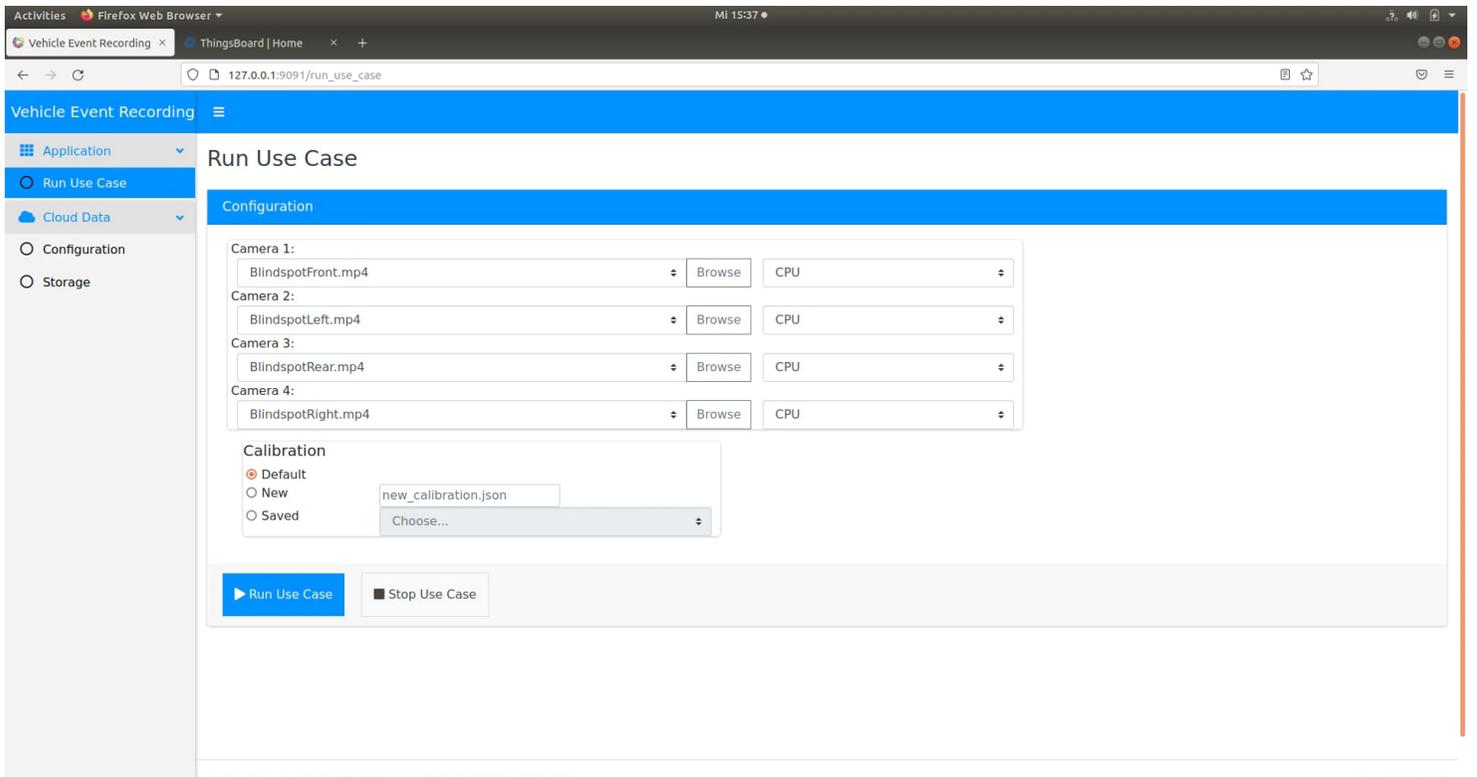
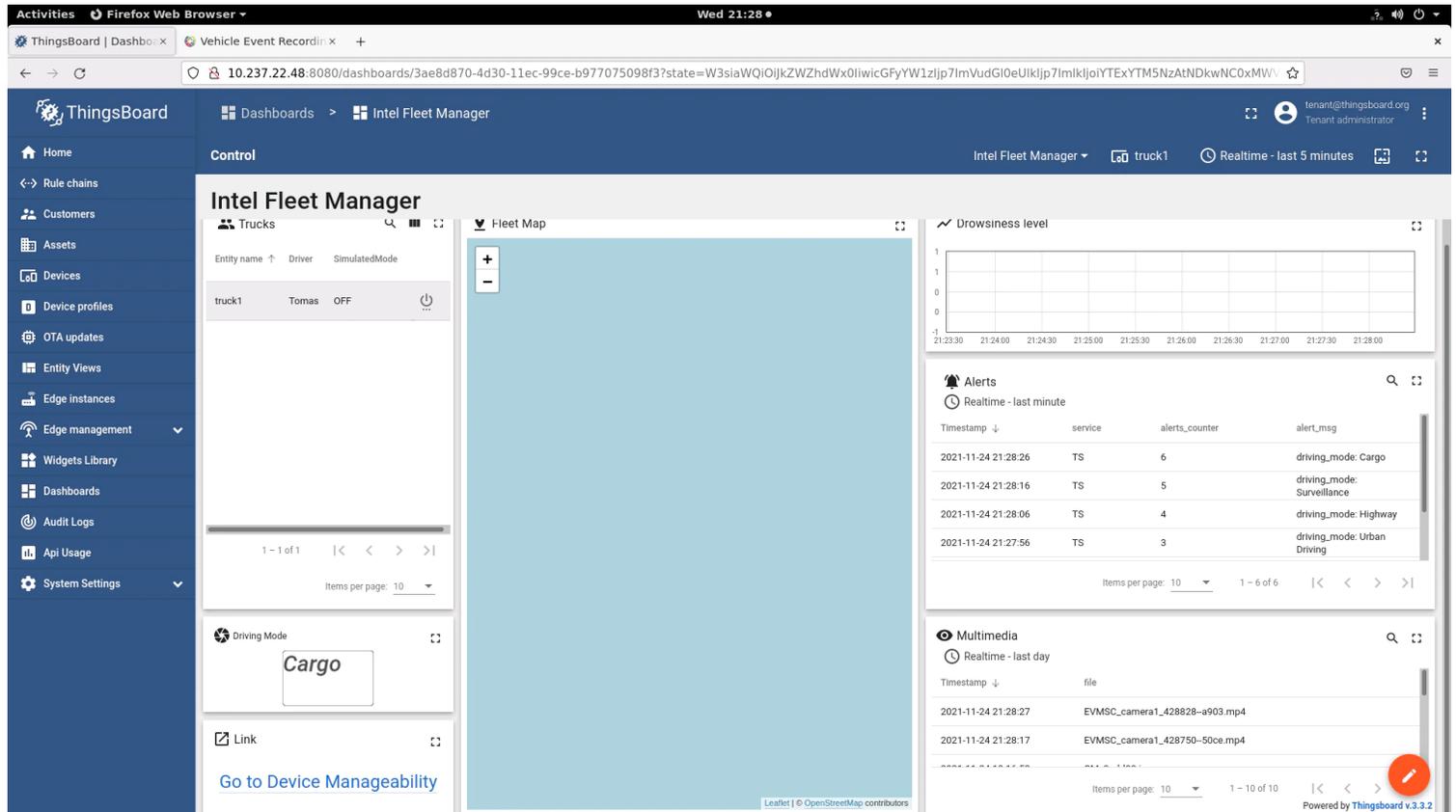


Figure 7: Configure Use Case

- Click on the **Browse** button and search for one of the sample videos delivered with the application at the following path: <INSTALL_PATH>/vehicle_event_recording/Vehicle_Event_Recording_2021.3/Vehicle_Event_Recording/EII-EVMSC-UseCase/config/VideoIngestion/test_videos/
- Select one of the available videos.
- After you configure all four videos, select the target CPU or GPU for all available options. Click on **Run Use Case**.

5. After the visualiser starts, you can go to the ThingsBoard link and check the alerts sent by the reference implementation. If you configured the AWS credentials, you will also have access to video snapshots taken by the application on the video stream.



6. You can also check the cloud storage from the Reference Implementation **Storage** tab.

Activities Firefox Web Browser Wed 21:11

Vehicle Event Recording ThingsBoard | Dashboard

127.0.0.1:9091/cloud_storage

Vehicle Event Recording

- Application
 - Run Use Case
 - Cloud Data
 - Configuration
 - Storage**

Cloud Video Clips

Filename	Size	Last Modified
EVMSC_camera1_21025--ca68.mp4	1.1 MB	Wed, 24 Nov 2021 19:06:38 GMT
EVMSC_camera1_20962--0427.mp4	1.0 MB	Wed, 24 Nov 2021 19:06:29 GMT
EVMSC_camera1_21104--c6c7.mp4	0.9 MB	Wed, 24 Nov 2021 18:35:38 GMT
EVMSC_camera1_21062--ab0a.mp4	1.1 MB	Wed, 24 Nov 2021 18:35:30 GMT
EVMSC_camera1_15205--6eb1.mp4	0.9 MB	Wed, 24 Nov 2021 17:37:40 GMT
EVMSC_camera1_15134--64a5.mp4	1.2 MB	Wed, 24 Nov 2021 17:37:31 GMT
EVMSC_camera2_15132--64a5.mp4	0.8 MB	Wed, 24 Nov 2021 17:37:30 GMT
EVMSC_camera1_12181--9844.mp4	1.1 MB	Wed, 24 Nov 2021 17:26:38 GMT
EVMSC_camera1_11399--4236.mp4	1.0 MB	Wed, 24 Nov 2021 17:26:00 GMT
EVMSC_camera1_10641--403e.mp4	1.2 MB	Wed, 24 Nov 2021 17:25:59 GMT
EVMSC_camera1_11329--c76a.mp4	1.3 MB	Wed, 24 Nov 2021 17:25:59 GMT
EVMSC_camera1_10576--e69e.mp4	1.3 MB	Wed, 24 Nov 2021 17:25:58 GMT
EVMSC_camera1_9877--739e.mp4	1.1 MB	Wed, 24 Nov 2021 17:25:57 GMT
EVMSC_camera1_9811--b2dd.mp4	1.4 MB	Wed, 24 Nov 2021 17:25:56 GMT
EVMSC_camera1_8073--a5a6.mp4	1.0 MB	Wed, 24 Nov 2021 14:41:17 GMT
EVMSC_camera1_8004--4d17.mp4	0.9 MB	Wed, 24 Nov 2021 14:41:09 GMT
EVMSC_camera1_7217--5fab.mp4	0.9 MB	Wed, 24 Nov 2021 14:40:19 GMT
EVMSC_camera1_7161--1774.mp4	1.2 MB	Wed, 24 Nov 2021 14:40:11 GMT
EVMSC_camera1_6259--3d8c.mp4	1.1 MB	Wed, 24 Nov 2021 14:39:17 GMT
EVMSC_camera1_6161--0392.mp4	1.5 MB	Wed, 24 Nov 2021 14:39:08 GMT
EVMSC_camera1_5312--1cf7.mp4	1.2 MB	Wed, 24 Nov 2021 14:38:24 GMT
EVMSC_camera1_5230--09a0.mp4	1.2 MB	Wed, 24 Nov 2021 14:38:10 GMT
EVMSC_camera1_4688--a23a.mp4	1.0 MB	Wed, 24 Nov 2021 13:38:50 GMT
EVMSC_camera1_2594--6404.mp4	1.4 MB	Wed, 24 Nov 2021 13:36:59 GMT
EVMSC_camera1_2496--3884.mp4	1.3 MB	Wed, 24 Nov 2021 13:36:47 GMT
EVMSC_camera1_1583--7bbc.mp4	1.2 MB	Wed, 24 Nov 2021 13:35:58 GMT
EVMSC_camera1_1493--4172.mp4	1.2 MB	Wed, 24 Nov 2021 13:35:48 GMT
EVMSC_camera1_555--688a.mp4	1.1 MB	Wed, 24 Nov 2021 13:34:59 GMT
EVMSC_camera1_453--68c2.mp4	1.2 MB	Wed, 24 Nov 2021 13:34:47 GMT

Use Case Running

Depending on the number of cameras configured, the application will start the Visualizer App that will analyze the video sample or video samples selected.

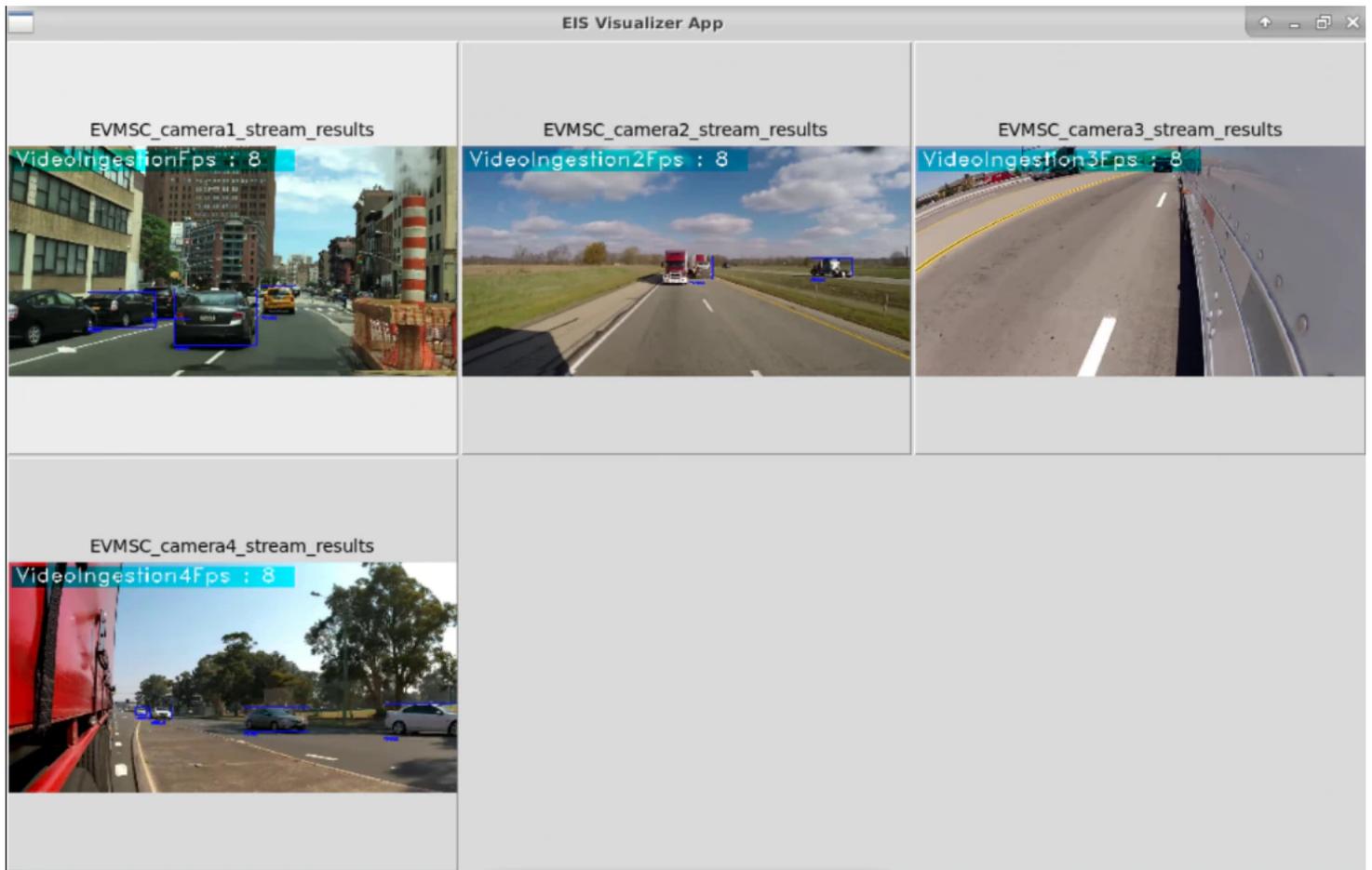


Figure 8: Use Case Analysis Results

At this point, you can see that the algorithm is analyzing the traffic from the video streams.

Run in Parallel with Driver Behavior Analytics Reference Implementation

To run this task, you will need to download and install the Driver Behavior Analytics Reference Implementation.

Prerequisites

- Two terminals
- Follow the steps to install Driver Behavior Analytics Reference Implementation after
installing Vehicle Event Recording

Steps to Run the Application

1. Change directory to **Vehicle Event Recording Use Case** path on terminal 1:

```
1 | cd <INSTALL_PATH>/vehicle_event_recording/Vehicle_Event_Recording_2021.3/Vehicle_Event_Recording/EII-EVMS-UseCase
```

```

intel@edgesoftware: ~/driver_behavior_analytics 104x55
Step 14/22 : COPY --from=common ${60_WORK_DIR}/common/Libs ${PY_WORK_DIR}/libs
Step 15/22 : COPY --from=common ${60_WORK_DIR}/common/util ${PY_WORK_DIR}/util
Step 16/22 : COPY --from=common ${60_WORK_DIR}/common/cmake ${PY_WORK_DIR}/common/cmake
Step 17/22 : COPY --from=common /usr/local/lib /usr/local/lib
Step 18/22 : COPY --from=common /usr/local/lib/python3.6/dist-packages/ /usr/local/lib/python3.6/dist-pa
Step 19/22 : COPY .
Step 20/22 : RUN apt-get remove -y wget && apt-get remove -y git && apt-get remove curl && a
Step 21/22 : HEALTHCHECK NONE
Step 22/22 : ENTRYPOINT ["python3.6", "visualize.py"]
[Warning] One or more build-args [EIS_UID] were not consumed
Successfully built 0c4837bc7d8a
Successfully tagged ia visualizer_db:2.4.2
Compiled successfully.
Successfully installed Driver Behavior Analytics.

In order to Launch server and open the WebUI:
# cd /home/intel/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/Driver_Behavior_Analytics/E
# make webui
Clean up function
Successfully installed Driver Behavior Analytics took 2 minutes 17.05 seconds
Installation of package complete
***Recommended to reboot system after installation***
-----
| Id | Module | Status |
-----
| 5f21392e9e63c9002a6fd88d | Docker Community Edition CE | SUCCESS |
| 5f213aae9e63c9002a6fd88e | Docker Compose | SUCCESS |
| 6116593b4d13002b0b4c3 | EIIcleaner | SUCCESS |
| 5face41098a1e002af292fd | eii installer | SUCCESS |
| 60dec2c3f4472c4002a91d311 | Driver Behavior Analytics | SUCCESS |
-----
intel@edgesoftware:~/driver_behavior_analytics$

intel@edgesoftware: ~/vehicle_event_recording 105x55
Step 12/20 : COPY --from=common /usr/local/lib/python3.6/dist-packages/ /usr/local/lib/python3.6/dist-pac
Step 13/20 : RUN apt-get install python-pip
Step 14/20 : RUN pip3.6 install ffmpeg-python==0.2.0
Step 15/20 : RUN pip3.6 install numpy==1.19.5
Step 16/20 : RUN pip3.6 install opencv-python==3.4.1.15
Step 17/20 : RUN apt-get update && apt-get install -y ffmpeg
Step 18/20 : COPY ./*.py ./
Step 19/20 : COPY ./common ${PY_WORK_DIR}/common
Step 20/20 : ENTRYPOINT ["python3.6", "-u", "main.py"]
[Warning] One or more build-args [MINIO_VERSION EIS_UID] were not consumed
Successfully built 547aa8d4972a
Successfully tagged ia image_store_evmsc:2.4.2
Compiled successfully.
Successfully installed Vehicle Event Recording.

In order to Launch server and open the WebUI:
# cd /home/intel/vehicle_event_recording/Vehicle_Event_Recording_2021.2/Vehicle_Event_Recording/EII-EVMS
# make webui
Clean up function
Successfully installed Vehicle Event Recording took 2 minutes 10.73 seconds
Installation of package complete
***Recommended to reboot system after installation***
-----
| Id | Module | Status |
-----
| 5f21392e9e63c9002a6fd88d | Docker Community Edition CE | SUCCESS |
| 5f213aae9e63c9002a6fd88e | Docker Compose | SUCCESS |
| 6116593b4d13002b0b4c3 | EIIcleaner | SUCCESS |
| 5face41098a1e002af292fd | eii installer | SUCCESS |
| 60dec2c3f4472c4002a91d311 | Driver Behavior Analytics | SUCCESS |
| 60dec2c3f4472c4002a6d6a0b | Vehicle Event Recording | SUCCESS |
-----
intel@edgesoftware:~/vehicle_event_recording$

```

2. Change directory to Driver Behavior Analytics Use Case path on terminal 2:

```

1 | cd <INSTALL_PATH>/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/Driver_Behavior_Analytics/EII-
  | DriverBehavior-UseCase

```

```

intel@edgesoftware: ~/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/Driver_Behavior_Analytics/EII-Drive
Step 15/22 : COPY --from=common ${60_WORK_DIR}/common/util ${PY_WORK_DIR}/util
Step 16/22 : COPY --from=common ${60_WORK_DIR}/common/cmake ${PY_WORK_DIR}/common/cmake
Step 17/22 : COPY --from=common /usr/local/lib /usr/local/lib
Step 18/22 : COPY --from=common /usr/local/lib/python3.6/dist-packages/ /usr/local/lib/python3.6/dist-pa
Step 19/22 : COPY .
Step 20/22 : RUN apt-get remove -y wget && apt-get remove -y git && apt-get remove curl && a
Step 21/22 : HEALTHCHECK NONE
Step 22/22 : ENTRYPOINT ["python3.6", "visualize.py"]
[Warning] One or more build-args [EIS_UID] were not consumed
Successfully built 0c4837bc7d8a
Successfully tagged ia visualizer_db:2.4.2
Compiled successfully.
Successfully installed Driver Behavior Analytics.

In order to Launch server and open the WebUI:
# cd /home/intel/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/Driver_Behavior_Analytics/E
# make webui
Clean up function
Successfully installed Driver Behavior Analytics took 2 minutes 17.05 seconds
Installation of package complete
***Recommended to reboot system after installation***
-----
| Id | Module | Status |
-----
| 5f21392e9e63c9002a6fd88d | Docker Community Edition CE | SUCCESS |
| 5f213aae9e63c9002a6fd88e | Docker Compose | SUCCESS |
| 6116593b4d13002b0b4c3 | EIIcleaner | SUCCESS |
| 5face41098a1e002af292fd | eii installer | SUCCESS |
| 60dec2c3f4472c4002a91d311 | Driver Behavior Analytics | SUCCESS |
-----
intel@edgesoftware:~/driver_behavior_analytics$ cd /home/intel/driver_behavior_analytics/Driver_Behavior
intel@edgesoftware:~/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/Driver_Behavior_Analyt
s/EII-DriverBehavior-UseCases$

intel@edgesoftware: ~/vehicle_event_recording/Vehicle_Event_Recording_2021.2/Vehicle_Event_Recording/EII-EVMS-UseCase
Step 13/20 : RUN apt-get install python-pip
Step 14/20 : RUN pip3.6 install ffmpeg-python==0.2.0
Step 15/20 : RUN pip3.6 install numpy==1.19.5
Step 16/20 : RUN pip3.6 install opencv-python==3.4.1.15
Step 17/20 : RUN apt-get update && apt-get install -y ffmpeg
Step 18/20 : COPY ./*.py ./
Step 19/20 : COPY ./common ${PY_WORK_DIR}/common
Step 20/20 : ENTRYPOINT ["python3.6", "-u", "main.py"]
[Warning] One or more build-args [MINIO_VERSION EIS_UID] were not consumed
Successfully built 547aa8d4972a
Successfully tagged ia image_store_evmsc:2.4.2
Compiled successfully.
Successfully installed Vehicle Event Recording.

In order to Launch server and open the WebUI:
# cd /home/intel/vehicle_event_recording/Vehicle_Event_Recording_2021.2/Vehicle_Event_Recording/EII-EVMS
# make webui
Clean up function
Successfully installed Vehicle Event Recording took 2 minutes 10.73 seconds
Installation of package complete
***Recommended to reboot system after installation***
-----
| Id | Module | Status |
-----
| 5f21392e9e63c9002a6fd88d | Docker Community Edition CE | SUCCESS |
| 5f213aae9e63c9002a6fd88e | Docker Compose | SUCCESS |
| 6116593b4d13002b0b4c3 | EIIcleaner | SUCCESS |
| 5face41098a1e002af292fd | eii installer | SUCCESS |
| 60dec2c3f4472c4002a6d6a0b | Vehicle Event Recording | SUCCESS |
-----
intel@edgesoftware:~/vehicle_event_recording$ cd /home/intel/vehicle_event_recording/Vehicle_Event_Recor
ding_2021.2/Vehicle_Event_Recording/EII-EVMS-UseCase
intel@edgesoftware:~/vehicle_event_recording/Vehicle_Event_Recording_2021.2/Vehicle_Event_Recording/EII-E
VMS-UseCases$

```

- Run the following command on terminal 1 to start the webserver application. Copy and run the `make webui` command from the end of the installation log:

```
1 | make webui v=4 EII_BASE=<INSTALL_PATH>/vehicle_event_recording/Vehicle_Event_Recording_2021.3/IEdgeInsights
  REPO_FOLDER=
  <INSTALL_PATH>/vehicle_event_recording/Vehicle_Event_Recording_2021.3/Vehicle_Event_Recording/EII-EVMSC-
  UseCase
```

- Run the following command on terminal 2 to start the webserver application. Copy and run the `make webui` command from the end of the installation log:

```
1 | make webui EII_BASE=<INSTALL_PATH>/vehicle_event_recording/Vehicle_Event_Recording_2021.3/IEdgeInsights
  REPO_FOLDER=
  <INSTALL_PATH>/driver_behavior_analytics/Driver_Behavior_Analytics_2021.3/Driver_Behavior_Analytics/EII-
  DriverBehavior-UseCase
```

```
intel@edgesoftware: ~/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/Driver_Behavior_Analytics/EII-DriverBehavior-UseCase
git-get autoremove -y
--> Using cache
--> 4728330263e
Step 21/22 : HEALTHCHECK NONE
--> Using cache
--> 527f637af3f4
Step 22/22 : ENTRYPOINT ["python3.6", "visualize.py"]
--> Using cache
--> 0c4837bc7d8a

[Warning] One or more build-args [EIS_UID] were not consumed
Successfully built 0c4837bc7d8a
Successfully tagged ia_visualizer_db:2.4.2
✔ Compiled successfully.
Successfully installed Driver Behavior Analytics.

In order to Launch server and open the WebUI:
# cd /home/intel/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/Driver_Behavior_Analytics/EII-DriverBehavior-UseCase
# make webui
Clean up function
Successfully installed Driver Behavior Analytics took 2 minutes 17.05 seconds
Installation of package complete
***Recommended to reboot system after installation***
-----
| Id | Module | Status |
-----
| 5f21392e9e63c9002a6fd88d | Docker Community Edition CE | SUCCESS |
| 5f213aae9e63c9002a6fd88e | Docker Compose | SUCCESS |
| 61165593bd4d13002b6ba4c3 | EIIcleaner | SUCCESS |
| 5face41098a1ef002af292fd | eii installer | SUCCESS |
| 60db2c3f4472c4002a91d311 | Driver Behavior Analytics | SUCCESS |
-----
intel@edgesoftware:~/driver_behavior_analytics$ cd /home/intel/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/Driver_Behavior_Analytics/EII-DriverBehavior-UseCase
intel@edgesoftware:~/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/Driver_Behavior_Analytics/EII-DriverBehavior-UseCase$ make webui
Host +
access control disabled, clients can connect from any host
# Activate virtual env and Launch WebUI
source ".webui_env"/bin/activate && cd webui && python3 ./server.py
* Working directory /home/intel/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/Driver_Behavior_Analytics/./IEdgeInsights/build
* Videos directory /home/intel/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/Driver_Behavior_Analytics/./IEdgeInsights/VideoIngestion.DB/test_videos/
* Calibration configurations directory /home/intel/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/Driver_Behavior_Analytics/./IEdgeInsights/common/video/udfs/python/area_detection/saved_calibrations/
* Serving Flask app "server" (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:9092/ (Press CTRL+C to quit)

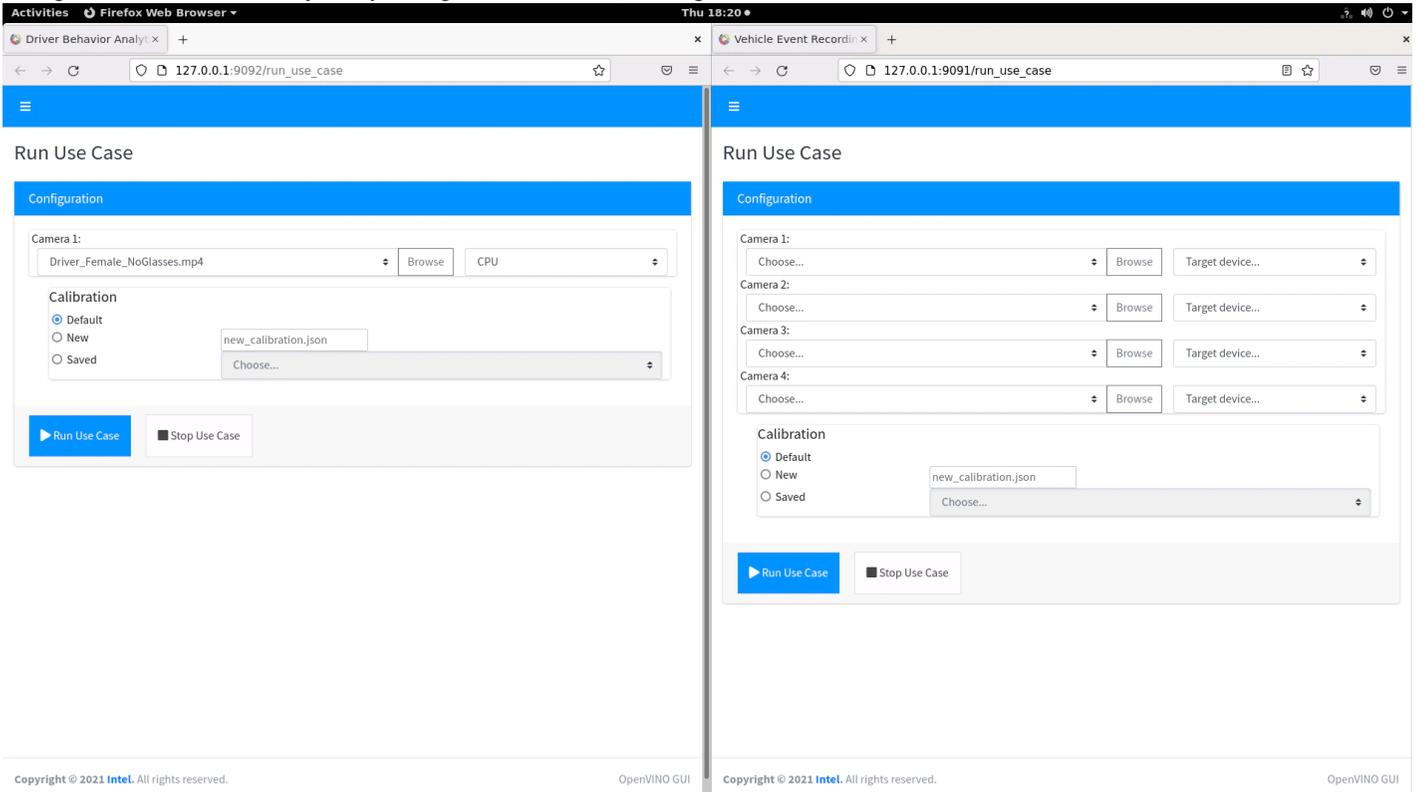
intel@edgesoftware:~/vehicle_event_recording/Vehicle_Event_Recording_2021.2/Vehicle_Event_Recording/EII-EVMSC-UseCase
--> 1a7f75938cd3
Step 18/20 : COPY ./*.py ./
--> Using cache
--> 3835128642f4
Step 19/20 : COPY ./common ${PY_WORK_DIR}/common
--> Using cache
--> de7493857be4
Step 20/20 : ENTRYPOINT ["python3.6", "-u", "main.py"]
--> Using cache
--> 547a88d4972a

[Warning] One or more build-args [MINIO_VERSION EIS_UID] were not consumed
Successfully built 547a88d4972a
Successfully tagged ia_image_store_evmsc:2.4.2
✔ Compiled successfully.
Successfully installed Vehicle Event Recording.

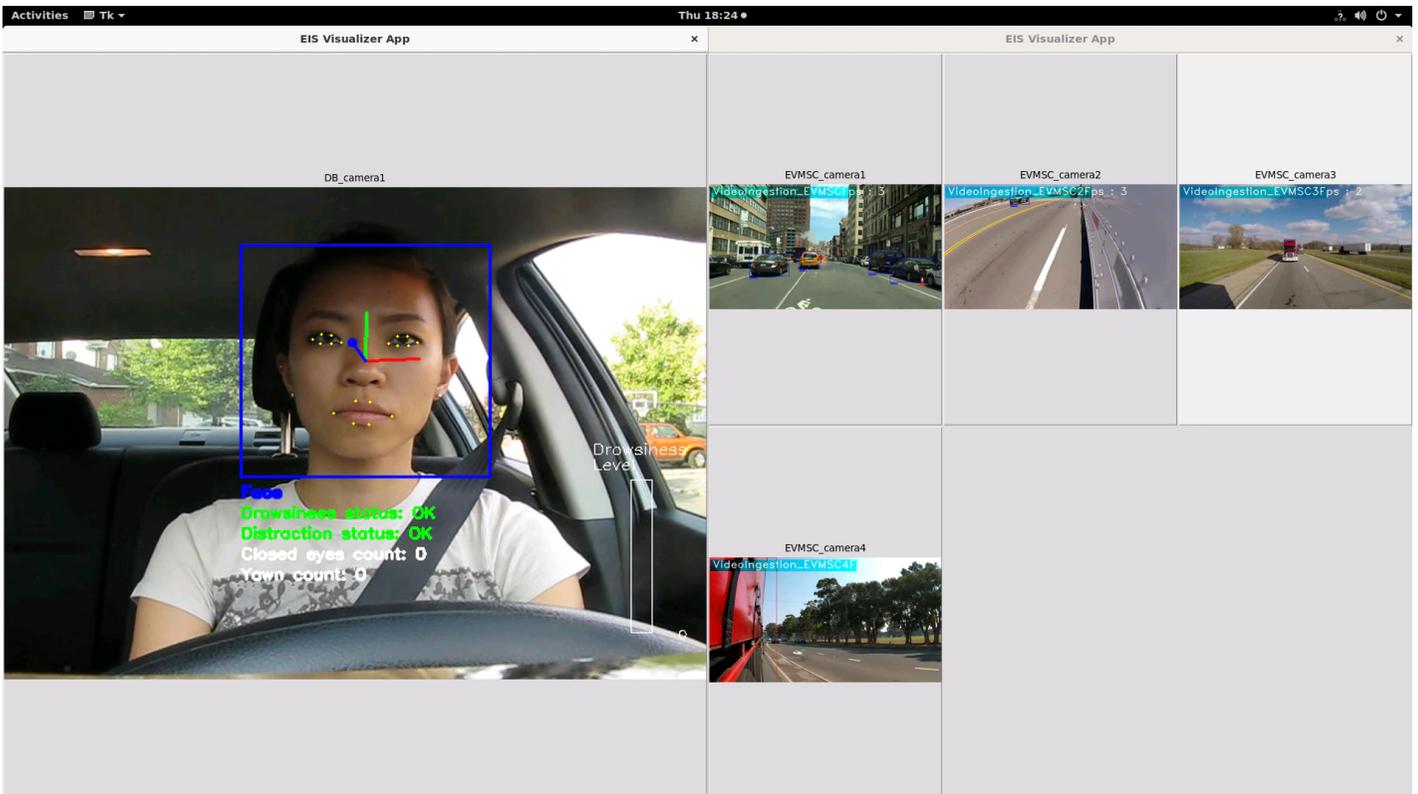
In order to Launch server and open the WebUI:
# cd /home/intel/vehicle_event_recording/Vehicle_Event_Recording_2021.2/Vehicle_Event_Recording/EII-EVMSC-UseCase
# make webui
Clean up function
Successfully installed Vehicle Event Recording took 2 minutes 10.73 seconds
Installation of package complete
***Recommended to reboot system after installation***
-----
| Id | Module | Status |
-----
| 5f21392e9e63c9002a6fd88d | Docker Community Edition CE | SUCCESS |
| 5f213aae9e63c9002a6fd88e | Docker Compose | SUCCESS |
| 61165593bd4d13002b6ba4c3 | EIIcleaner | SUCCESS |
| 5face41098a1ef002af292fd | eii installer | SUCCESS |
| 60db2c3f4472c4002a91d311 | Driver Behavior Analytics | SUCCESS |
| 608ec8df41e9002a6d6a0b | Vehicle Event Recording | SUCCESS |
-----
intel@edgesoftware:~/vehicle_event_recording/Vehicle_Event_Recording_2021.2/Vehicle_Event_Recording/EII-EVMSC-UseCase$ cd /home/intel/vehicle_event_recording/Vehicle_Event_Recording_2021.2/Vehicle_Event_Recording/EII-EVMSC-UseCase
intel@edgesoftware:~/vehicle_event_recording/Vehicle_Event_Recording_2021.2/Vehicle_Event_Recording/EII-EVMSC-UseCase$ make webui
Host +
access control disabled, clients can connect from any host
# Activate virtual env and Launch WebUI
source ".webui_env"/bin/activate && cd webui && python3 ./server.py
* Working directory /home/intel/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/IEdgeInsights/build
* Videos directory /home/intel/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/IEdgeInsights/VideoIngestion.EVMS/test_videos/
* Calibration configurations directory /home/intel/driver_behavior_analytics/Driver_Behavior_Analytics_2021.2/IEdgeInsights/common/video/udfs/python/area_detection/saved_calibrations/
* Serving Flask app "server" (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:9091/ (Press CTRL+C to quit)
```

- Open your browser and go to **127.0.0.1:9091**.
- Configure all available cameras with the desired videos and set the target for each one (**CPU** or **GPU**) and click **Run Use Case**.
- Wait for Visualizer to run.
- Open the **Driver Behavior Analytics** page by going to address **127.0.0.1:9092**.

9. Configure **Driver Behavior Analytics** by setting the **video source**, the **target** and click on **Run Use Case**.



At this point, Vehicle Event Recording will close and then both use cases will start.



NOTE: If you reinstall the first reference implementation, you must also reinstall the second reference implementation.

Summary and Next Steps

This reference implementation successfully implements Intel® Distribution of OpenVINO™ toolkit plugins for detecting objects and provides event-based video recording. It uses Edge Insights for Fleet framework to cover historical analysis, evidence support, driver coaching, remote view, and traffic violation detection.

As a next step, try one of the following:

- Use deep learning models, Edge Insights for Fleet framework and a live external video camera stream to capture evidence support, remote view, traffic violations and coach the decisions that must be made by the algorithm.
- This reference implementation uses Intel® Distribution of OpenVINO™ toolkit Open Model Zoo pre-trained models and 3rd party models, but you can extend it to use your own models.

Learn More

To continue learning, see the following guides and software resources:

- Intel® Distribution of OpenVINO™ toolkit documentation

Known Issues

Uninstall Reference Implementation

If you uninstall one of the reference implementations, you need to reinstall the other reference implementations because the Docker images will be cleared.

Troubleshooting

Installation Failure

If host system already has Docker images and its containers running, you will have issues during the RI installation. You must stop/force stop existing containers and images.

- To remove all stopped containers, dangling images, and unused networks:

```
1 | sudo docker system prune --volumes
```

- To stop Docker containers:

```
1 | sudo docker stop $(sudo docker ps -aq)
```

- To remove Docker containers:

```
1 | sudo docker rm $(sudo docker ps -aq)
```

- To remove all Docker images:

```
1 | sudo docker rmi -f $(sudo docker images -aq)
```

Docker Image Build Failure

If Docker image build on corporate network fails, follow the steps below.

1. Get DNS server using the command:

```
1 | nmcli dev show | grep 'IP4.DNS'
```

2. Configure Docker to use the server. Paste the line below in the `/etc/docker/daemon.json` file:

```
1 | {  
2 |     "dns": ["<dns-server-from-above-command>"]  
3 | }
```

3. Restart Docker:

```
1 | sudo systemctl daemon-reload && sudo systemctl restart docker
```

Installation Failure Due to Ubuntu Timezone Setting

While building the reference implementation, if you see `/etc/timezone && apt-get install -y tzdata && ln -sf /usr/share/zoneinfo/${HOST_TIME_ZONE} /etc/localtime && dpkg-reconfigure -f noninteractive tzdata` returned a non-zero code: `1 make: *** [config] Error 1`

Run the following command in your terminal:

```
1 | sudo timedatectl set-local-rtc 0
```

Installation Encoding Issue

While building the reference implementation, if you see `ERROR: 'latin-1' codec can't encode character '\u2615' in position 3: ordinal not in range(256)`

Run the following command in your terminal:

```
1 | export LANG=en_US.UTF-8
```

Can't Connect to Docker Daemon

If you can't connect to Docker Daemon at `http+docker://localhost`, run the following command in your terminal:

```
1 | sudo usermod -aG docker $USER
```

Log out and log in to Ubuntu.

Check before retrying to install if group Docker is available for you by running the following command in a terminal:

```
1 | groups
```

The output should contain **Docker**.

Support Forum

If you're unable to resolve your issues, contact the [Support Forum](#)

Product and Performance Information

¹ Performance varies by use, configuration and other factors. Learn more at www.Intel.com/PerformanceIndex

Company Overview

[Contact Intel](#)

[Newsroom](#)

[Investors](#)

[Careers](#)

[Corporate Responsibility](#)

[Diversity & Inclusion](#)

[Public Policy](#)



© Intel Corporation

[Terms of Use](#)

[*Trademarks](#)

[Cookies](#)

[Privacy](#)

[Supply Chain Transparency](#)

[Site Map](#)

Intel technologies may require enabled hardware, software or service activation. // No product or component can be absolutely secure. // Your costs and results may vary. // Performance varies by use, configuration and other factors. // See our complete legal [Notices and Disclaimers](#)

. // Intel is committed to respecting human rights and avoiding complicity in human rights abuses. See Intel's [Global Human Rights Principles](#). Intel's products and software are intended only to be used in applications that do not cause or contribute to a violation of an internationally recognized human right.



