

e volve

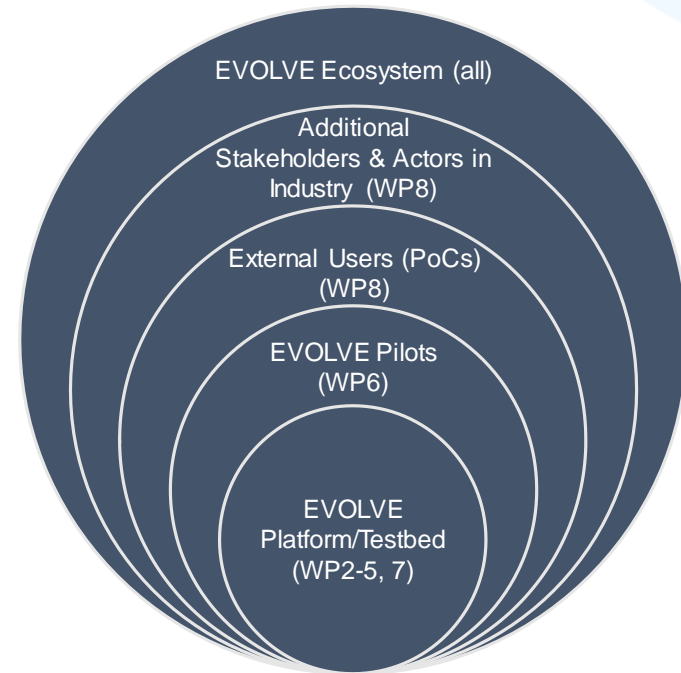
# EVOLVE

## Proof of Concepts and Ecosystem Development

Alexander Stocker (Virtual Vehicle)

# Evolve Ecosystem Development

- **Achieve and demonstrate the breadth of applicability of the EVOLVE testbed**
  - Identify and implement several Proof-of-Concepts (PoCs) between pilot partners and additional users (external to the project), showing the impact of the testbed to the market
  - Perform further activities that will result in developing a broader ecosystem around the testbed with the purpose of continuing work and ensuring impact beyond the end of the project



# Proof of Concepts

- Status detection of assistance systems from video streams of a vehicle dashboard (vehicle monitoring)
- Scalable pattern detection in parsimonious streams of vehicle sensor measurements
- Population genetics study on 1000 genomes project
- Distributed genotype imputation
- Improved generative machine-learning models for image processing (different datasets)
- Improvement of PT service performance/reliability
- Improvement of service operation and generation of user information



# Proof of Concepts

- Learning acceleration of retail AI
- Adapt vine varieties to soil properties
- Automatic cadastral map generation for tropical countries
- Network security analytics
- Accelerated bowtie2 for high throughput next generation sequencing (DNA/RNA sequencing)
- Scale-out beam longitudinal dynamics simulations (particle simulation)
- Telecom usage analytics: churn prediction & adoption



# Proof of Concepts

- Driver distraction detection (driver monitoring)
- Object detection using knowledge graph-based data integration for automated driving



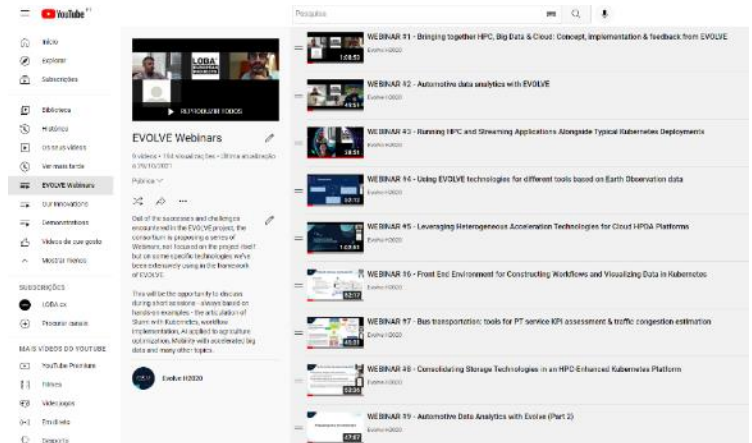
# Evolve Webinar Series



...with recorded talks

Playlist:

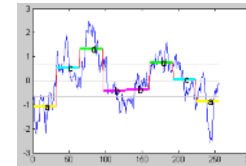
[https://www.youtube.com/watch?v=t6zsKaGoadI&list=PLh2iZalApB8g83frVlzw\\_Kq700pwYZh-P](https://www.youtube.com/watch?v=t6zsKaGoadI&list=PLh2iZalApB8g83frVlzw_Kq700pwYZh-P)



# Examples of Proof-of-Concepts

# Examples of Proof-of-Concepts

1. Status detection of vehicle assistance systems from video streams of a vehicle dashboard
2. Scalable pattern detection in parsimonious streams of vehicle sensor measurements
3. Driver distraction detection using camera-recorded driver videos
4. Object detection using knowledge graph-based data integration for automated driving





1. Status detection of vehicle assistance systems from video streams of a vehicle dashboard

## PoC: Dashboard Recognition

- Car dashboards were filmed during driving
- Our goal: Detection of various symbols to infer the state of driving assistance systems



## PoC: Detection of symbol state

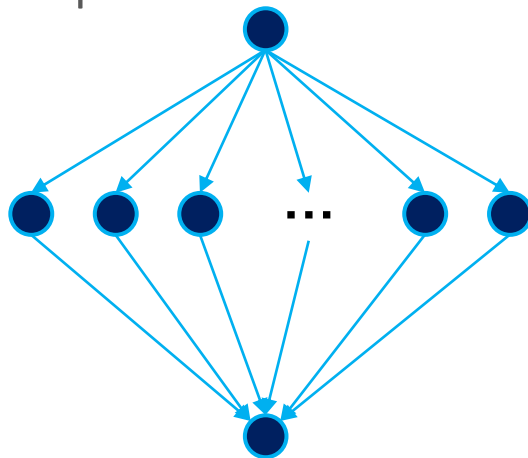
- Apply CLAHE (contrast limited adaptive histogram equalization) to each frame.
- Detect matches using cross correlation with normed coefficients
- Operations use GPU-ready data-structures
- Implemented in dockerized OpenCV



$$R(x, y) = \frac{\sum_{x', y'} (T'(x', y') \cdot I'(x + x', y + y'))}{\sqrt{\sum_{x', y'} T'(x', y')^2 \cdot \sum_{x', y'} I'(x + x', y + y')^2}}$$

# PoC: Workflow management

- Computational chain implemented in Kubernetes via CRDs using Argo
- Utilizes DAGs (directed acyclic graphs) to describe workflow steps



```
- name: video-processor
inputs:
  parameters:
    - name: file
  container:
    image: 172.9.0.240:5000/vif/video_processor_gpu:latest
  env:
    - name: VIDEO_FILE
      value: "{{inputs.parameters.file}}"

- name: video-statistics
inputs:
  container:
    image: 172.9.0.240:5000/vif/video_statistics:latest
  env:
    - name: VideoDataPath
      value: "/private/shortvideodata/"

- name: process-all
  parallelism: 50
  dag:
    tasks:
      - name: list-files
        template: file-finder

      - name: process-videos
        dependencies: [list-files]
        template: video-processor
        arguments:
          parameters:
            - name: file
              value: "{{item}}"
        withParam: "{{tasks.list-files.outputs.result}}"
```

# Video processing times

Processing on	Processing time for one video (averaged from 3 runs)	Estimated processing time for all 300 videos
Local laptop	0h 41min 14sec	8d 14h 10min (no parallelization)
EVOLVE platform w/o GPU acceleration	1h 07min 00sec	0d 06h 42min (50 CPU instances in parallel)
EVOLVE platform with GPU acceleration	0h 24min 19sec	0d 15h 11min (8 GPU instances in parallel)
EVOLVE platform with and w/o GPU	-	0d 04h 39min (8 GPU instances + 50 CPU instances in parallel)

## Performance gain:

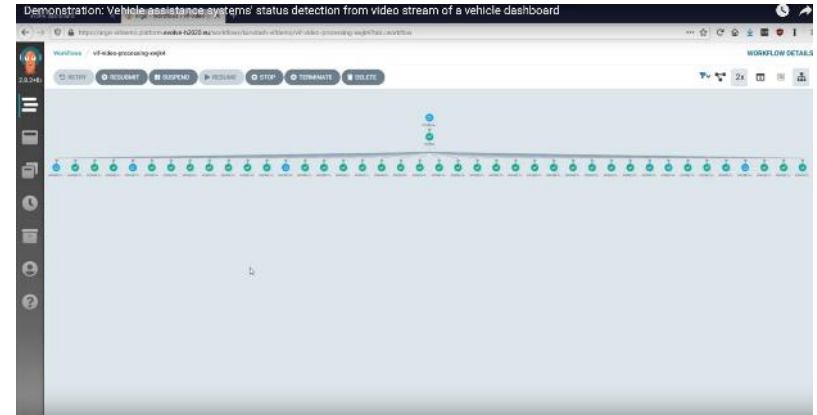
- From 8d14h to 4h39
- That's 41 times faster

# PoC: Video Demo

## Video with detection result



## Screencast of live PoC1-Demo



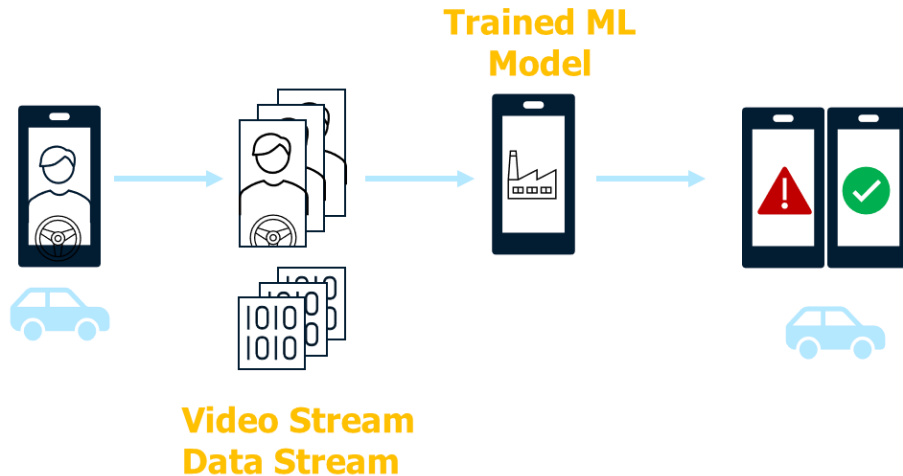
<https://www.youtube.com/watch?v=qV28LshCQK4>

# **Driver distraction detection using camera-recorded driver videos**

# PoC: Driver Distraction Detection

- **Distracted driving** is one of the most common cause for serious road accidents
- **Distraction** remains a problem in automated driving

**Overall goal:**

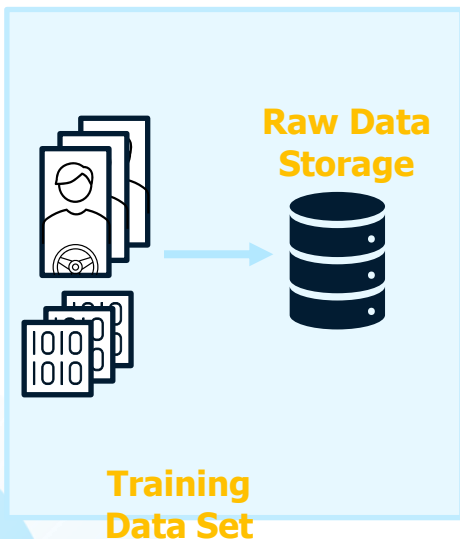






# Driver Distraction Detection

## EVOLVE PLATFORM



Texting Right



Reaching Behind



Talking Right



## Example images

Details
<b>10 classes:</b> Safe Driving Texting Right Talking Right Texting Left Talking Left Adjust Radio\\Music Player Drinking Reaching Behind Hair and Makeup Talking to Passenger
> 22,000 labeled training images (640x480); 950 MB
almost 80,000 unlabeled test images; 3,270 MB

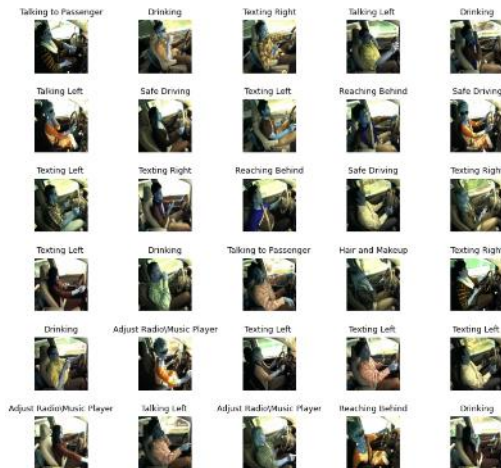


# PoC: Driver Distraction Detection

## EVOLVE PLATFORM

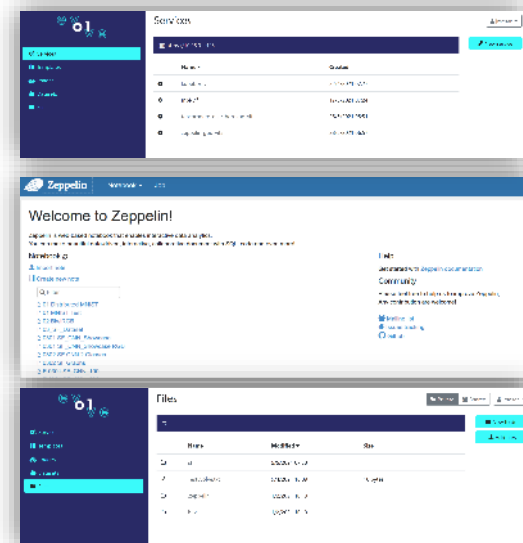


**Pre-Process  
and Store  
Data**



**Evolve-Service:  
Zeppelin-GPU+OpenCV**

**EVOLVE-Filestorage**

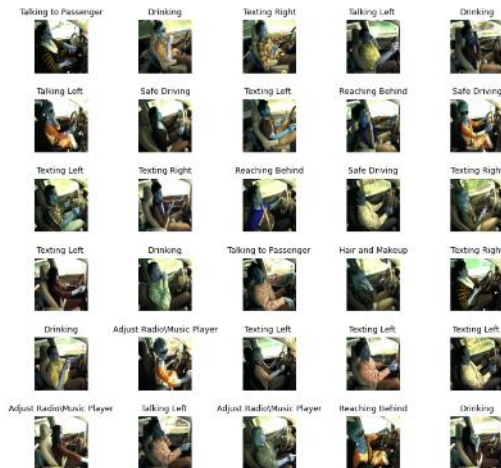


# PoC: Driver Distraction Detection

## EVOLVE PLATFORM

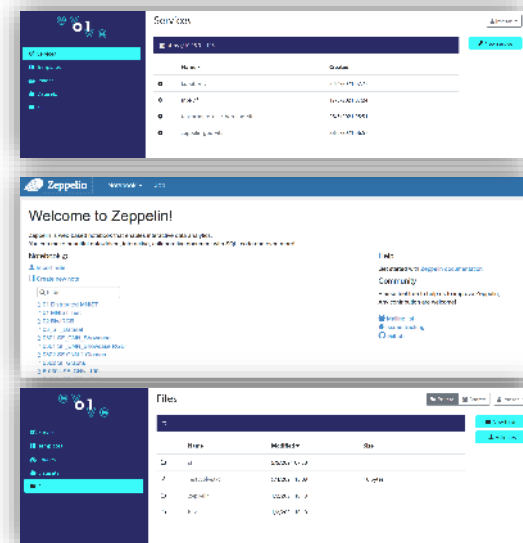


**Pre-Process  
and Store  
Data**

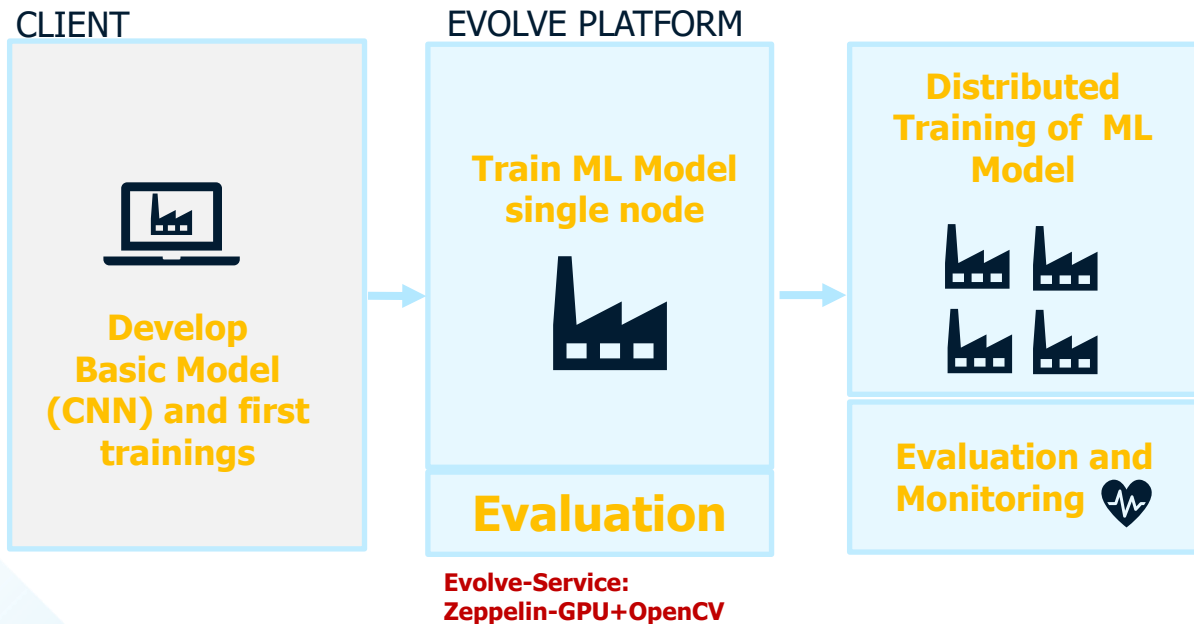


**Evolve-Service:  
Zeppelin-GPU+OpenCV**

**EVOLVE-Filestorage**

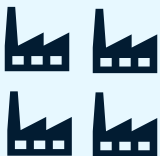


# PoC: Driver Distraction Detection

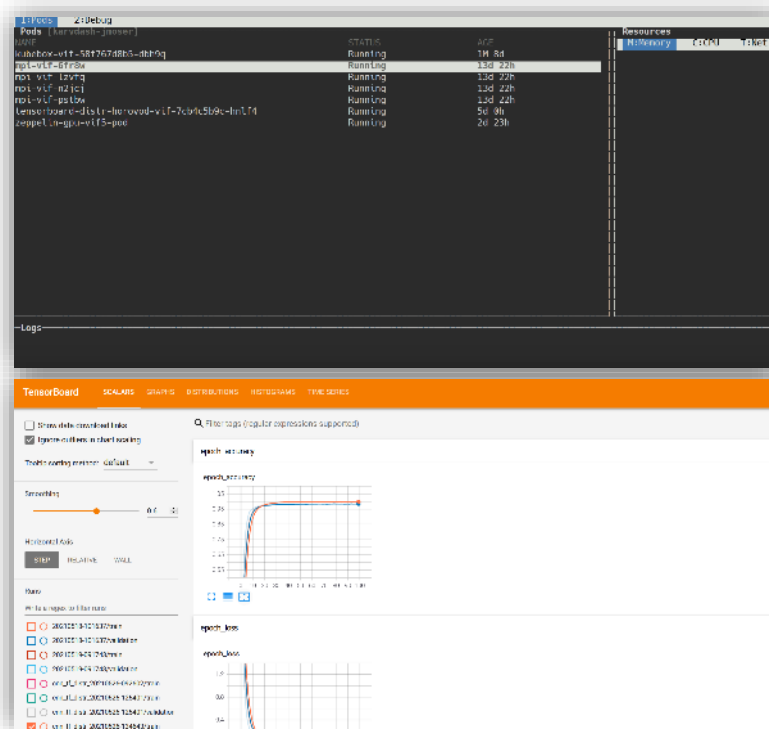


# PoC: Driver Distraction Detection

**Distributed  
Training of ML  
Model**



**Evaluation and  
Monitoring**



**Evolve-Services:**

**Kubebbox: Terminal and Webconsole**

**MPI TF/GPU: Cluster of pods  
supporting MPI running on GPU  
nodes (Horovod framework)**

**Tensorboard: Visualization Toolkit  
for Tensorflow**

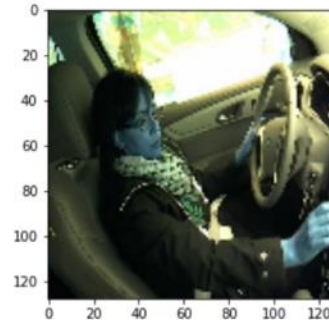
# PoC: Driver Distraction Detection

EVOLVE PLATTFORM: Train, test and validate Model

Reading Image with OpenCV - changed color-channels



Resize and normalize image



Predict Action



Statefarm Distracted Driver Dataset, [State Farm Distracted Driver Detection | Kaggle](#)

# PoC: Driver Distraction Detection

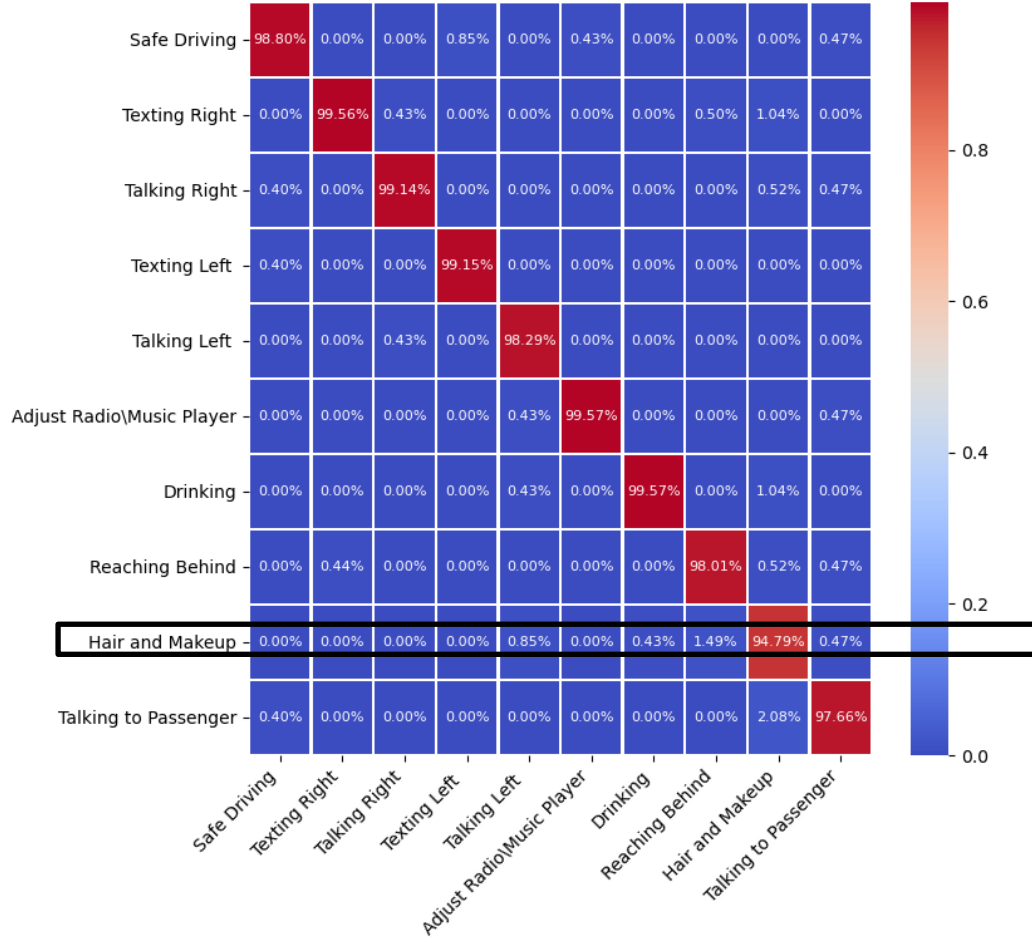
	in s	Client	EVOLVE distributed	
Train Model	657.33	42.99	15 x	faster
Predict Actions	22.29	3.01	7.3 x	faster

Model predictions (green: correct, red: incorrect)



**Accuracy** on a Kaggle-competition-dataset (~22000 labeled Images): **98.66%** - using a base model from literature that we adapted to our needs

# PoC: Driver Distraction Detection



Hair and Makeup is the hardest type to classify (94.79%) as the model confuses this distraction type with Talking to Passenger (2.08%) and Texting Right and Drinking (both 1.04%).

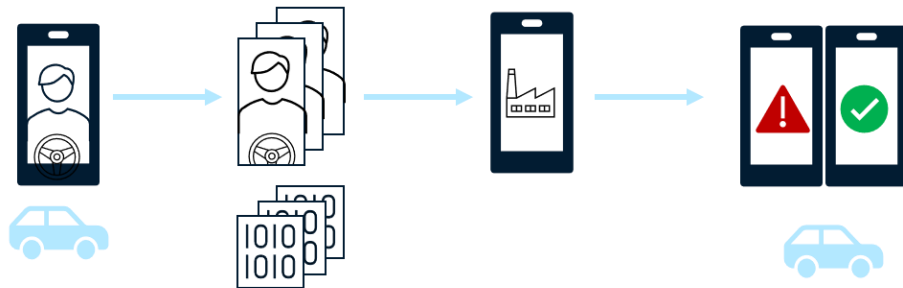


# PoC: Driver Distraction Detection

Live-Test:



Overall goal:



Video Stream  
Data Stream

Trained ML  
Model

Lower reliability due to different camera angles and light positions

Reaching Behind	100.00%
Talking Right	0.00%
Safe Driving	0.00%

Frame	640x480
Crop	128x128
View	480x480
Rotation	90
Inference Time	59ms

Threads

Model: FLOAT\_MOBILENET

Device: CPU

**Thank you!**

## Consortium



**DDN STORAGE**  
www.ddn.com



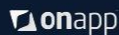
**BULL**  
www.atos.net



**IBM**  
www.ibm.com



**FORTH**  
www.ics.forth.gr



**OnApp**  
www.onapp.com



**Institute of communications and computer systems**  
www.microlab.ntua.gr



**MemoScale**  
www.memoscale.com



**webLyzard technology**  
www.weblyzard.com



**LOBA**  
www.loba.pt



**Thales Alenia Space**  
www.thalesgroup.com



**Space Hellas**  
www.space.gr



**CybeleTech**  
www.cybeletech.com



**Neurocom Luxembourg**  
www.neurocom.eu



**MemEX**  
www.memexitaly.it



**Tiemme SPA**  
www.tiemmespa.it



**Virtual Vehicle**  
www.v2c2.at



**AVL List GmbH**  
www.avl.com



**BMW AG**  
www.bmw.com



**KOOLA**  
www.koola.io

[www.evolve-h2020.eu](http://www.evolve-h2020.eu)  
[info@evolve-h2020.eu](mailto:info@evolve-h2020.eu)



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825061