DATA SCIENCE UA

APOSTERA

Automotive perception for advanced driver assistance using deep neural networks

Sergii Bykov

Technical Lead R&D

2019-10-19

Agenda

- Company Introduction
- System Concept
- Data Preparation
- Object Detection DNN Showcase
- Future Work and Conclusions

Company Introduction

Company Introduction

- Unique augmented reality in the vehicle
- Ultimately easy and safe driving
- Full visibility of autonomous driving decisions

APOSTERA

- Headquarters in Munich
- Development centers in Eastern Europe, presence in Asia
- 50+ experienced and talented engineers in 4 countries
- 10+ years of automotive experience
- Know-how in core automotive domains: Vehicle Infotainment, Vehicle Sensors and Networks, Telematics, Advanced Driver Assistance Systems, Navigation and Maps,
- Collaboration with scientific groups in fields of Computer Vision and Machine Learning, unique IP and mathematical talents

Technology

Recognition and Tracking

- Road boundaries and lane detection
- Slopes estimation
- Vehicle recognition and tracking
- Distance & time to collision estimation
- Pedestrian detection and tracking
- Facade recognition and texture extraction
- Road signs recognition

Integration with HD Maps

- HD Maps utilization for Precise positioning, Map matching and Path planning, Junction assistance
- Data generation for HD Maps

Computer Vision Approaches

- Real-time objects extraction from video sensors
- Road scene semantic segmentation
- Adaptability and c output data confidence
  ou estimation
- GPU optimization for different platforms

Augmented Reality

- LCD, HUD & further output devices
- Natural navigation hints & infographics
- Collison, Lane departure,
  Blind spots warnings, etc.
- POIs and supportive information (facades and parking slots highlighting, etc.)

Sensor Fusion

- Flexible fusion of data from internal and external sources
- LIDAR data merging
- 3D-environment model reconstruction based on different sensors
- Latency compensation & data extrapolation

Machine Learning Specifics

- CNN and DNN approaches
- Supervised MRF parameters adjustment
- CSP-based structure & parameters adjustment (both supervised and unsupervised)
- Weak classifiers boosting & others

Challenges of ADAS Embedded Platforms

- ***Power*** vs ***Performance***
  - Focus on performance while presuming the low power consumption
- ***Low latency*** and ***High response frequency***
  - Fast responses to environment changes are crucial for working in real-time
- ***Robustness*** and ***Quality***
  - Ensure robustness and presume quality in difficult operating conditions
  - Requires a lot of verification scenarios as well as adaptive heuristics
- System architecture specifics for embedded ***real-time***
  - Designed for real-time requirements and portability to fit to most effective hardware platforms
- Hardware and software ***sensor fusion***
  - Fuse available data sources (sensors, maps, etc.) for robustness and quality
- ***Big data analysis***
  - Huge amount of data should be stored and used for development and testing
- In- and Off-field ***automated testing***
  - Adaptive heuristics development
  - System validation
  - Collecting special cases

Challenges of ADAS Machine Learning

- Machine Learning needs large volumes of quality ***data***
  - Real need to ensure greater stability and accuracy in ML
  - High volumes of data might not be available for some tasks, limiting ML's adoption
- AI vs ***Expectations***
  - Understanding the limits of technology
  - Address expectations of replacing human jobs
- Becoming ***production-ready***
  - Transition from modeling to releasing production-grade AI solutions
- Current ML doesn't understand ***context*** well
  - Increased demand for real-time local data analysis
  - A need to quickly retrain ML models to understand new data
- Machine Learning ***security***
  - Addressing security concerns such as informational integrity

System Concept

Apostera Approach. High Level and Highlights

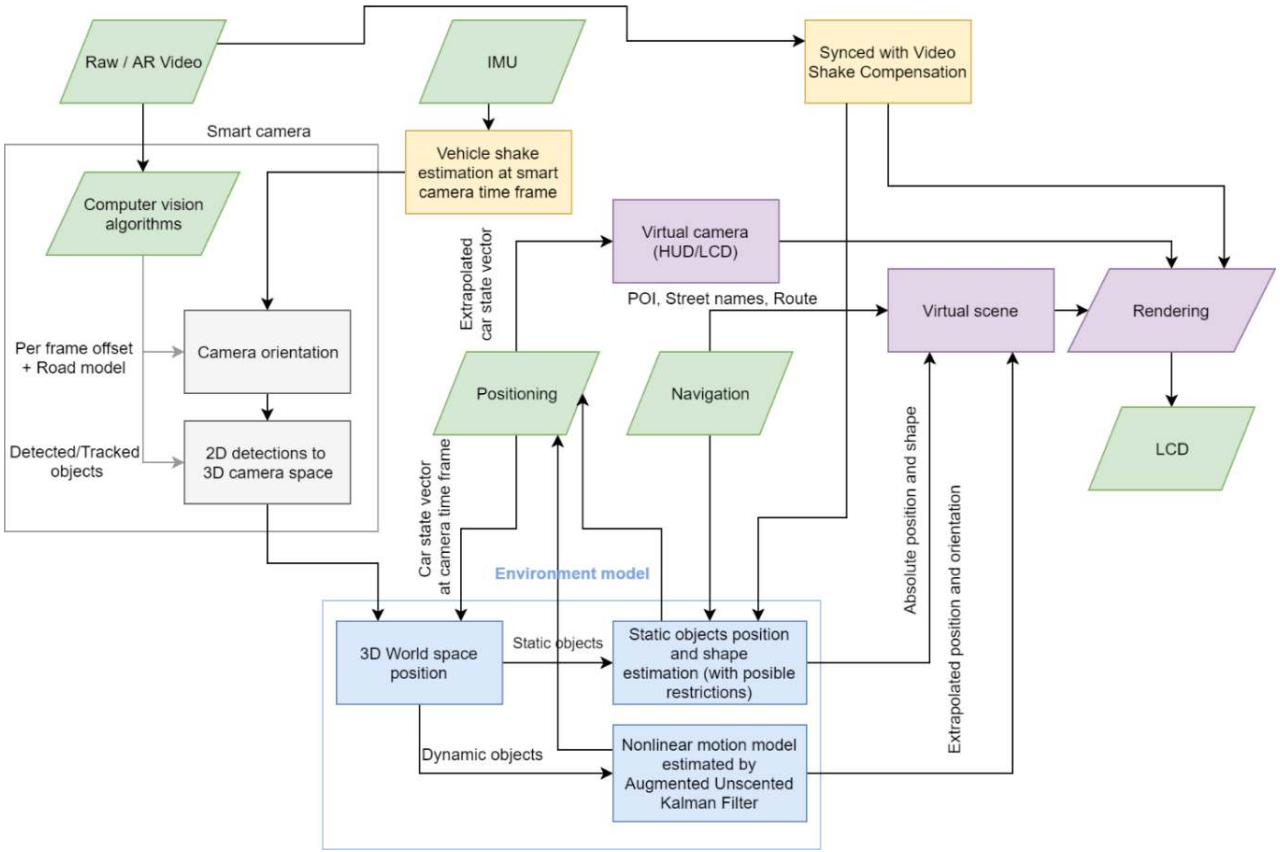


Figure – System architecture overview

Apostera Approach. High Level and Highlights

- Hardware agnostic
- Vehicle sensors agnostic
- Confidence estimation of fusion/visualization
- Real-time with low resource consumption
- Latency compensation and prediction model
  - Pitch, roll, low- and high-frequency
- Configurable design for different OEMs

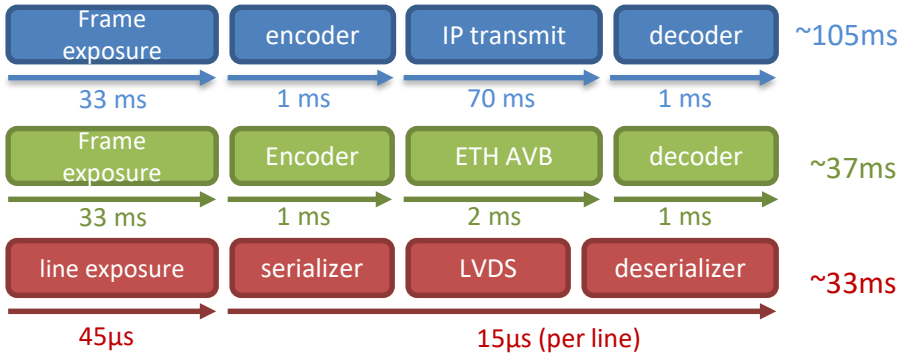
- Configurable logic requirements (including models and regions)
  - User interface logic considers confidence or probability of input data
  - Considers the dynamic environment and objects occlusion logic
- Integration with different navigation systems and map formats
  - Compensation of map data inaccuracy
  - Precise relative and absolute positioning

Cameras. Transport and Sensors

ADAS camera challenges

Low latency	Demand for algorithms reaction time Resolving data source synchronization issue
Small footprint	Demand for increasing number of ADAS sensors Increasingly space constrained
Low power	Reduced heat improves image quality & reliability Battery applications
High Reliability	Harsh environment Passenger and industrial vehicles

IP / ETH AVB / GMSL transport comparison



Supplier Type		Aptina AR0130	Aptina AR0231	Omnivision OV 10635
Resolution	pixel	1280x960	1928x1208	1280x800
Dynamic	dB	115 (HDR)	120(HDR)	115(HDR)
Response	V/L-sec	5.48	-	3.65
Frames	fps	60	40	30
Shutter Type	GS/ERS	ERS	ERS	ERS
Sensor optical format	Inch (“)	1/3”	1/2.7”	1/2.7”
Pixel size	µm	3.75	3	4.2
Interface		Parallel RGB	MIPI CSI2	Parallel DVP
Application		ADAS	ADAS	ADAS
Operation temp.	°C	-40...+105	-40...+105	-40...+105

Table – camera sensors comparison

Data Preparation

Data Preparation. Public Datasets

Desired road object classes:

- Vehicle
- Large Vehicle (Truck/Bus)
- Pedestrian
- Cyclist / motorcyclist
- Traffic sign
- Traffic light

General road scene annotations:

- Time of day
- Weather type
- Street type
- Country

Public datasets for object detection:

- Different annotation formats and classes
- Noncommercial use or after agreement
- Good for quick prototyping

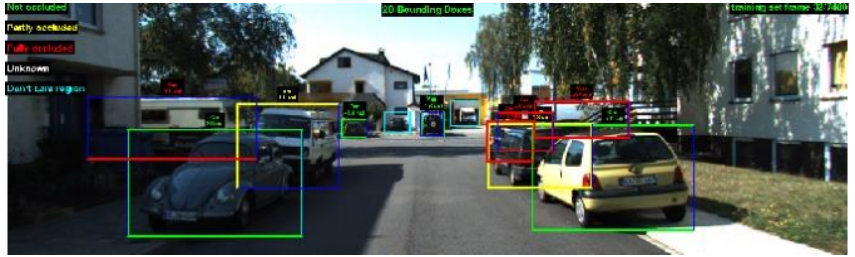
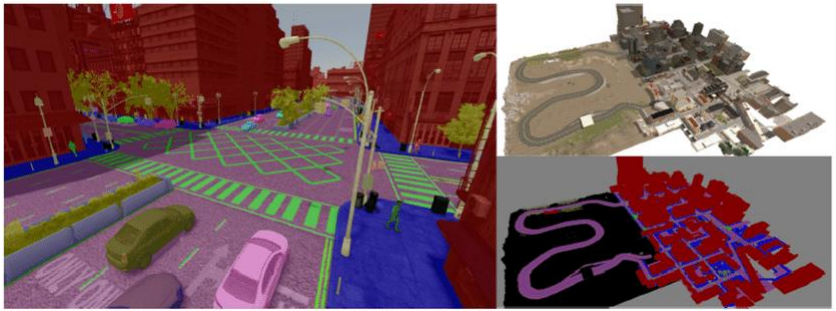


Figure – Public dataset samples

Data Preparation. Simulation

Pros	Cons
Faster development cycles	Limited physical, perceptual fidelity
Easy scalability	Limited behavioral fidelity
Controllability, reproducibility	Flexible, but not standardized
Easy data collection (inc. corner cases)	



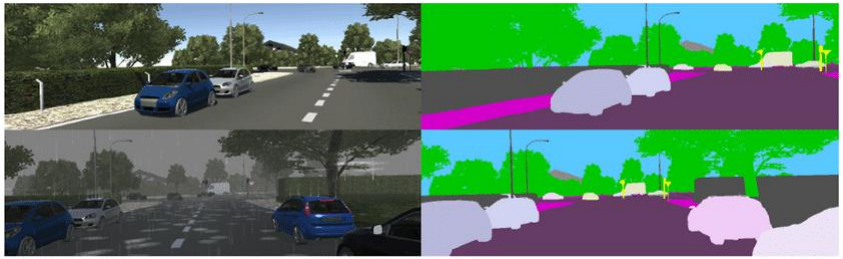


Figure – SYNTHIA and VirtualKITTI datasets

Data Preparation. Real and Simulated Data

Ground truth data sources:

- Raw driving data
  - 100x hours of in-field testing
  - Challenging manual or semi-automatic labeling
  - Hard to reuse after setup changes
- Simulated data
  - Fast automatic labeling
  - Special cases are easier to collect
  - Possible issues with real world deployment
- Data augmentation
  - Flip, crop, color changes
  - Quick dataset extension
  - Prevents model from irrelevant patterns, improves robustness

Figure – Raw vs simulated data

Object Detection DNN Showcase

Object Detection DNNs. Speed vs Accuracy

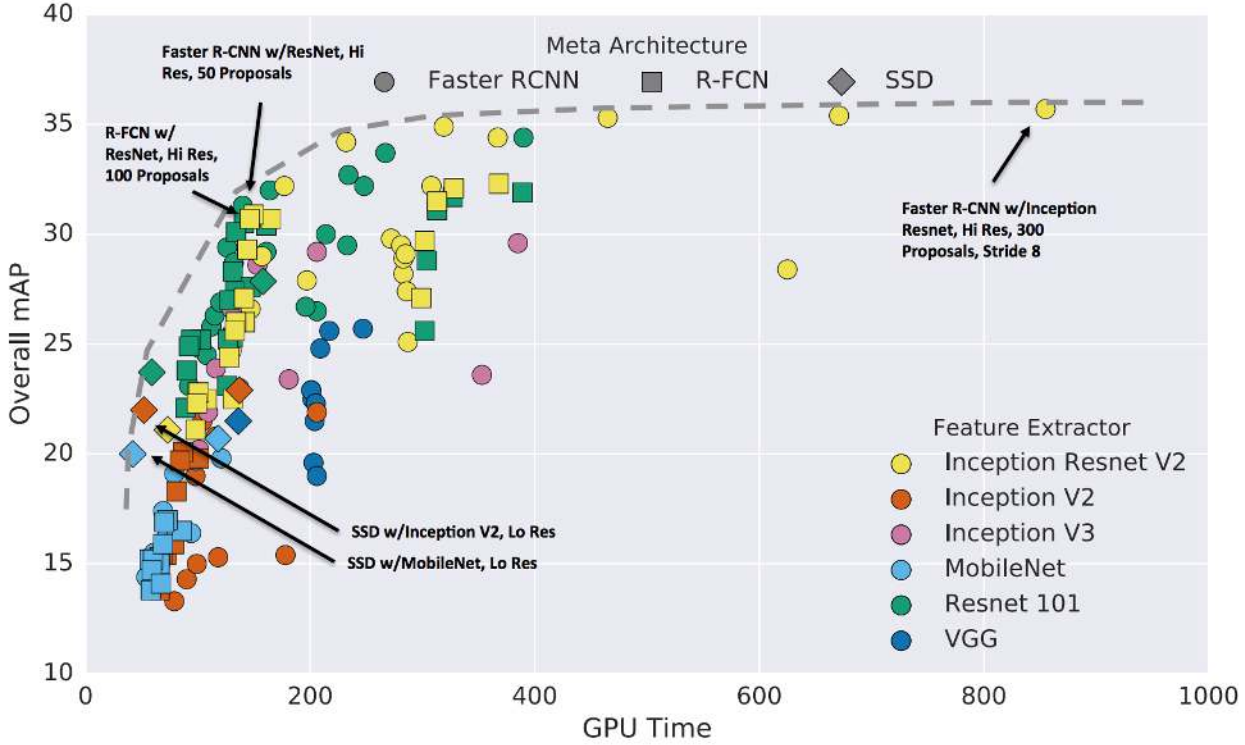


Figure – Accuracy (mAP) vs inference time of different meta architecture / feature extractor combinations for MS COCO dataset

Single Shot Multibox Detector

- Discretizes the output space of bounding boxes into a set of default boxes over different aspect ratios and scales per feature map location
- Generates scores for the presence of each object category in each default box and produces adjustments to the box to better match the object shape
- Combines predictions from multiple feature maps with different resolutions to handle various sizes
- Simple relative to methods that require object proposals, eliminates proposal generation and subsequent pixel or feature resampling stages, encapsulates all computation in a single network

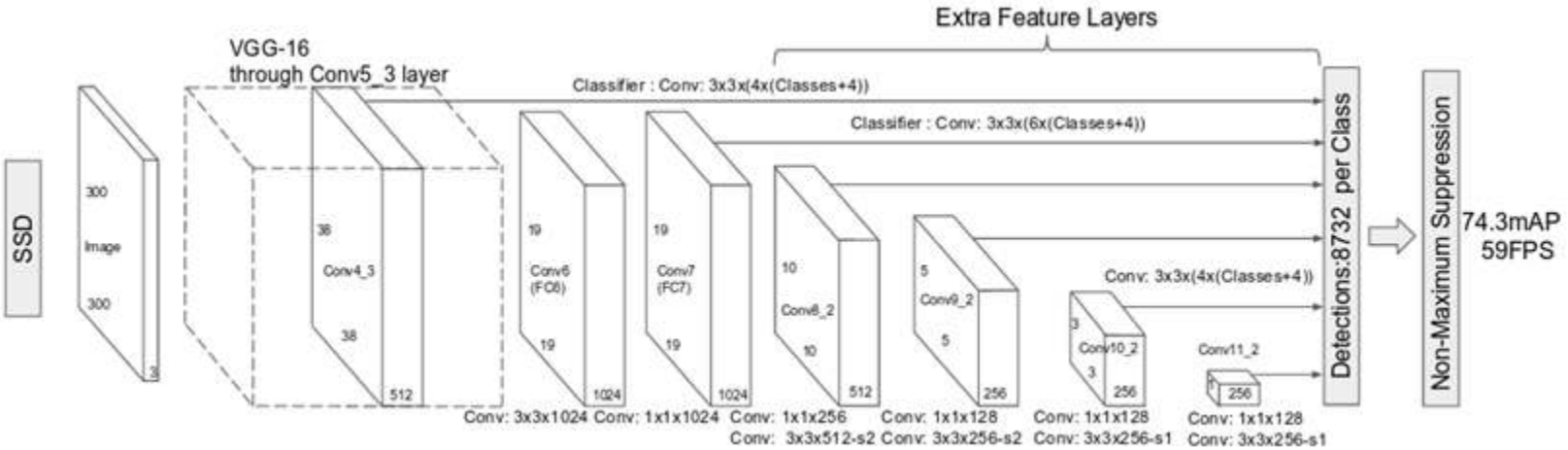


Figure – SSD model architecture

MobileNet as a Feature Extractor

- Streamlined architecture that uses depthwise separable convolutions to build light weight deep neural networks
- Uses two global hyper parameters to adjust between latency and accuracy
- Strong performance compared to other popular models on ImageNet classification
- Effective across a wide range of applications and use cases
  - object detection
  - fine grain classification
  - face attributes
  - large scale geo-localization

MobileNet Architecture. Convolution Block

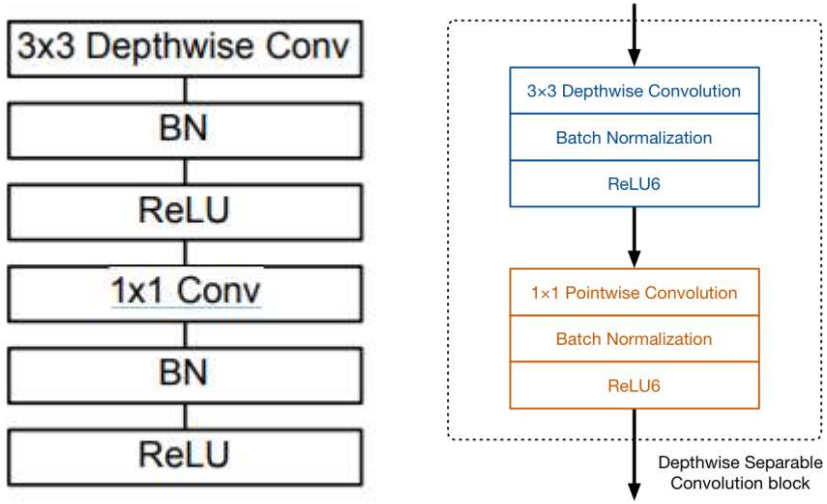


Figure - Depthwise separable convolution block

MobileNet Architecture

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
$5\times$ Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
$5\times$ Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

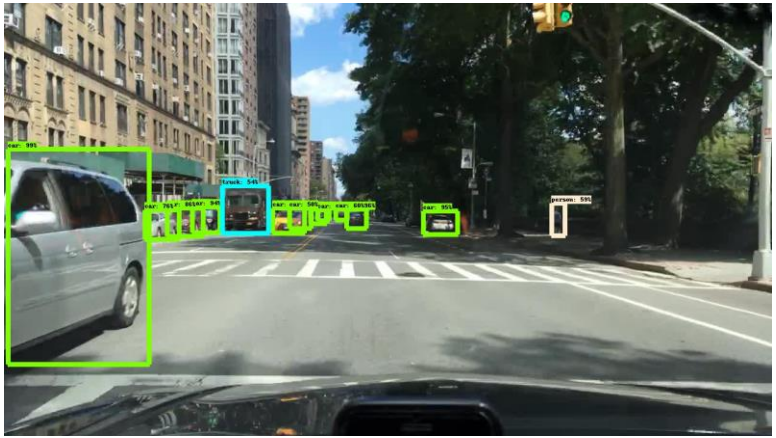
Figure – MobileNet architecture

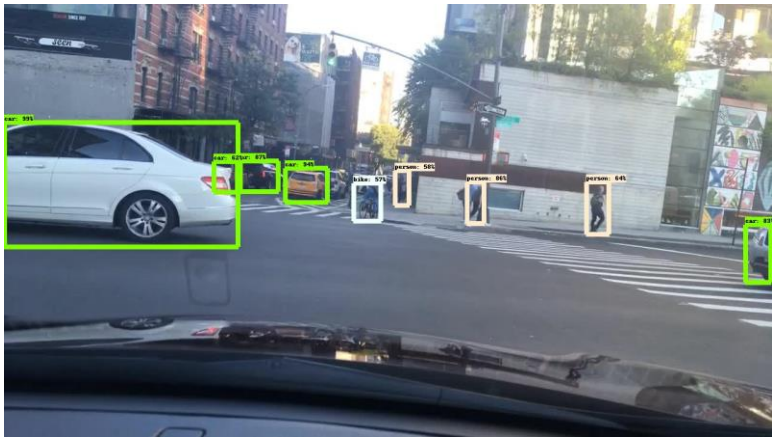
SSD-MobileNet Qualities

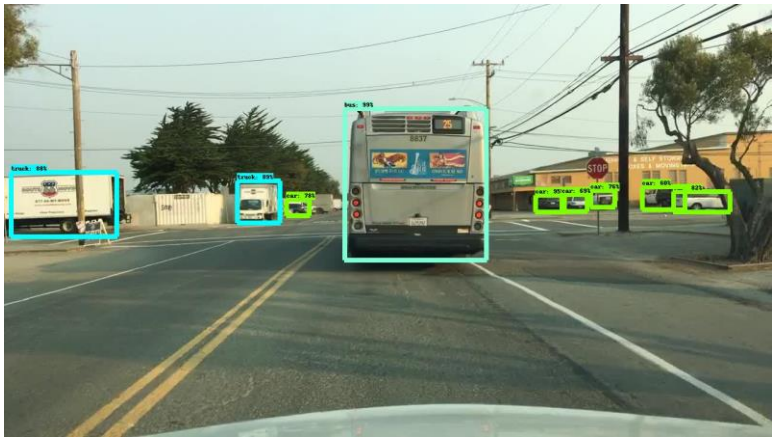
- Speed vs Accuracy:
  - SSD on MobileNet has the highest mAP among the models targeted for real-time processing
- Feature extractor:
  - The accuracy of the feature extractor impacts the detector accuracy, but it is less significant with SSD.
- Object size:
  - For large objects, SSD performs pretty well even with a simple extractor. SSD can even match other detectors' accuracies using better extractor. But SSD performs worse on small objects compared to other methods.
- Input image resolution
  - Higher resolution improves object detection for small objects significantly while also helping large objects. Decreasing resolution by 2x in both dimensions lowers accuracy, but the inference time is also reduced by 3x.
- Memory usage
  - MobileNet has the smallest RAM footprint. It requires less than 1Gb (total) memory.

SSD-MobileNet Detection Quality

- Input size: 640x360
- Detection quality for classes (AP@0.5IOU):
  - Light vehicle – 0.52
  - Truck/bus – 0.36
  - Cyclist/motorcyclist – 0.255
  - Pedestrian – 0.288







SSD-MobileNet Inference Performance

Desktop platform (PC)

- Quad-core Intel Core i5-7400
- 16 GB DDR4
- GeForce GTX 1060 (6 Gb)
- CUDA 8.0, CuDNN 6, TensorFlow v1.5

Reference platform – ***Nvidia Jetson TX2***

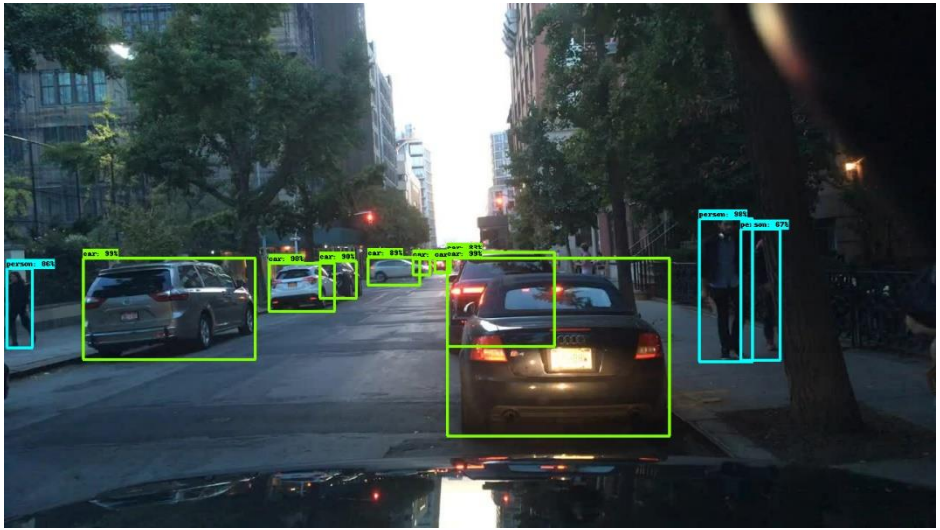
- Dual-core NVIDIA Denver2
- Quad-core ARM Cortex-A57
- 8GB 128-bit LPDDR4
- 256-core Pascal GPU (max freq)
- CUDA 8.0, CuDNN 6, TensorFlow v1.5

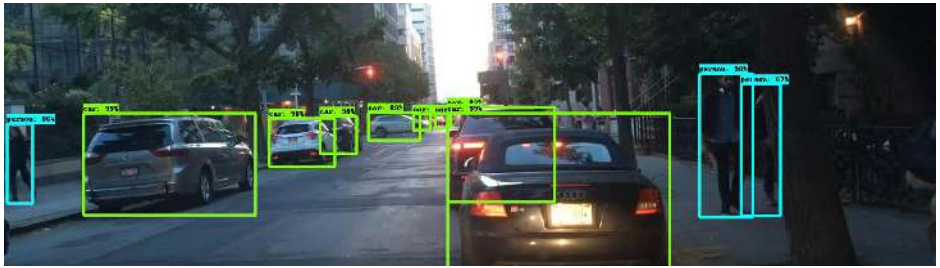
Input image resolution	PC GPU inference (ms/frame)	TX2 GPU inference (ms/frame)
1280x720	*49.55*	*185.0*
853x480	*26.3*	*84.87*
640x360	*15.7*	*56.21*
427x240	*8.25*	*32.51*

Table – Inference performance

DNN Inference Speedup. ROI

- **Challenge**: reducing input horizontal resolution under 640p resulted in serious decrease of narrow object accuracy (e.g. pedestrians)
- **Solution**: reduce ROI further only by height, remove small objects from training
  - Most road objects occupy center half of the frame
  - Use dynamic frame crop by horizon level
  - SSD can deal with truncated/occluded closer large objects





DNN Inference Speedup. Model Depth

- MobileNet provides two hyper parameters
  - width multiplier, resolution multiplier
- The role of the width multiplier α is to thin a network uniformly at each layer
- **Solution**: decrease the width multiplier to thin the network and remove redundant convolutions
  - width multiplier **0.75** was chosen for current road objects dataset

Width Multiplier (alpha)	ImageNet Acc (%)	Multiply-Adds (M)	Params (M)
1.0 MobileNet-224	*70.6*	*529*	*4.2*
0.75 MobileNet-224	*68.4*	*325*	*2.6*
0.50 MobileNet-224	*63.7*	*149*	*1.3*
0.25 MobileNet-224	*50.6*	*41*	*0.5*

Table – MobileNet accuracy vs width multiplier on ImageNet dataset

DNN Inference Speedup. Runtime

- Runtime and driver update
  - From: CUDA 8.0 + cuDNN 6
  - To: CUDA 9.0 + cuDNN 7
- Utilizing low level optimization efforts from specialized libraries
- Performance upgrade at low development cost

Input image resolution	TX2 CUDA 8 (ms/frame)	TX2 CUDA 9 (ms/frame)	Speedup
640x360	*56.2*	*54.5*	*+3.1%*

Table – Runtime performance comparison

SSD-MobileNet Optimized Performance

- Input size: 640x360
- Detection quality for classes (AP@0.5IOU):
  - Vehicle – 0.52
  - Pedestrian – 0.288

- New input size: 640x180
- Width multiplier: 0.75
- Detection quality for classes (AP@0.5IOU):
  - Vehicle – 0.6891 (***small obj removed***)
  - Pedestrian – 0.2902

Input image resolution	Width Multiplier (alpha)	TX2 CPU inference (ms/frame)	TX2 GPU inference (ms/frame)	CPU/GPU speedup
640x360	*1.0*	*262*	*56.2*	*4.66x*
640x180	*0.75*	*115.5*	***30.3***	*3.81x*

Table – Final performance comparison

Inference Acceleration. Hardware

GPU

- high throughput(up to 33x CPU)
- lower latency (up to 31x CPU)
- cuDNN
- FP16

FPGA

- performance per watt
- low precision types
- sparsity

TPU

- Int8 quantization
- DNN-inference-specific CISC instruction set
- massively parallel matrix processor
- minimal deterministic design

Inference Acceleration. Model Compression

Network pruning

- remove weights below threshold
- retrain
- AlexNet: 10x less params

Quantization

- binning
- weight sharing
- AlexNet: 3x size compression

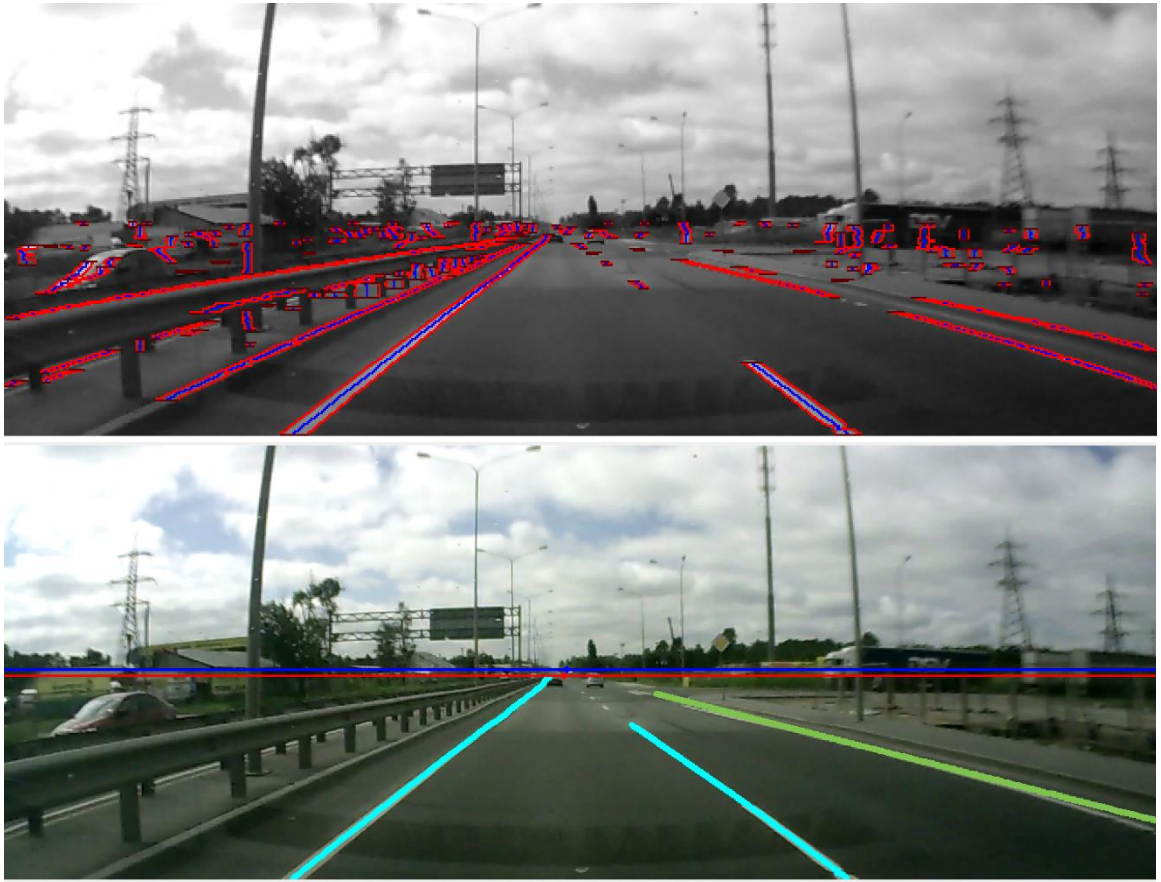
Huffman coding

- weight distributions are biased
- AlexNet: ~25% compression

Future Work and Conclusions

Ongoing Work. Lane Markings Detection

- Low level invariant features
  - Single camera
  - Stereo data
  - Point clouds
- Structural analysis
- Probabilistic models
  - Real-world features
  - Physical objects
  - 3D scene reconstruction
  - Road situation
- 3D space scene fusion (different sensors input)
- Backward knowledge propagation from high levels



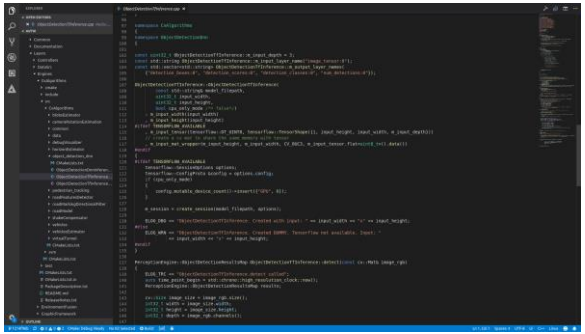
Ongoing Work. More Detection Classes

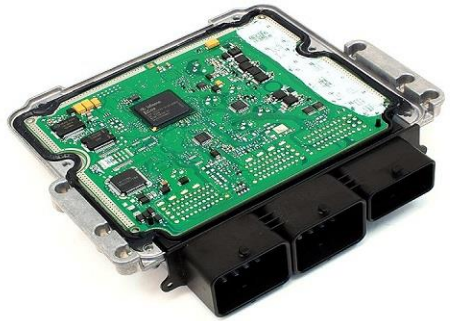
- Road object classes extension
  (without a loss of quality for existing classes)
  - Adding traffic signs recognition (detector + classifier)
  - Adding traffic lights recognition

Ongoing Work. Drivable Area Detection

- Drivable area detection using semantic segmentation
- Model is inspired by Squeeze-net and U-Net.
- Current performance (Jetson TX2):
  - Input size: 640x320 (lowres)
  - Inference speed: 75 ms/frame

Safety and Autonomous Vehicles





Algorithms

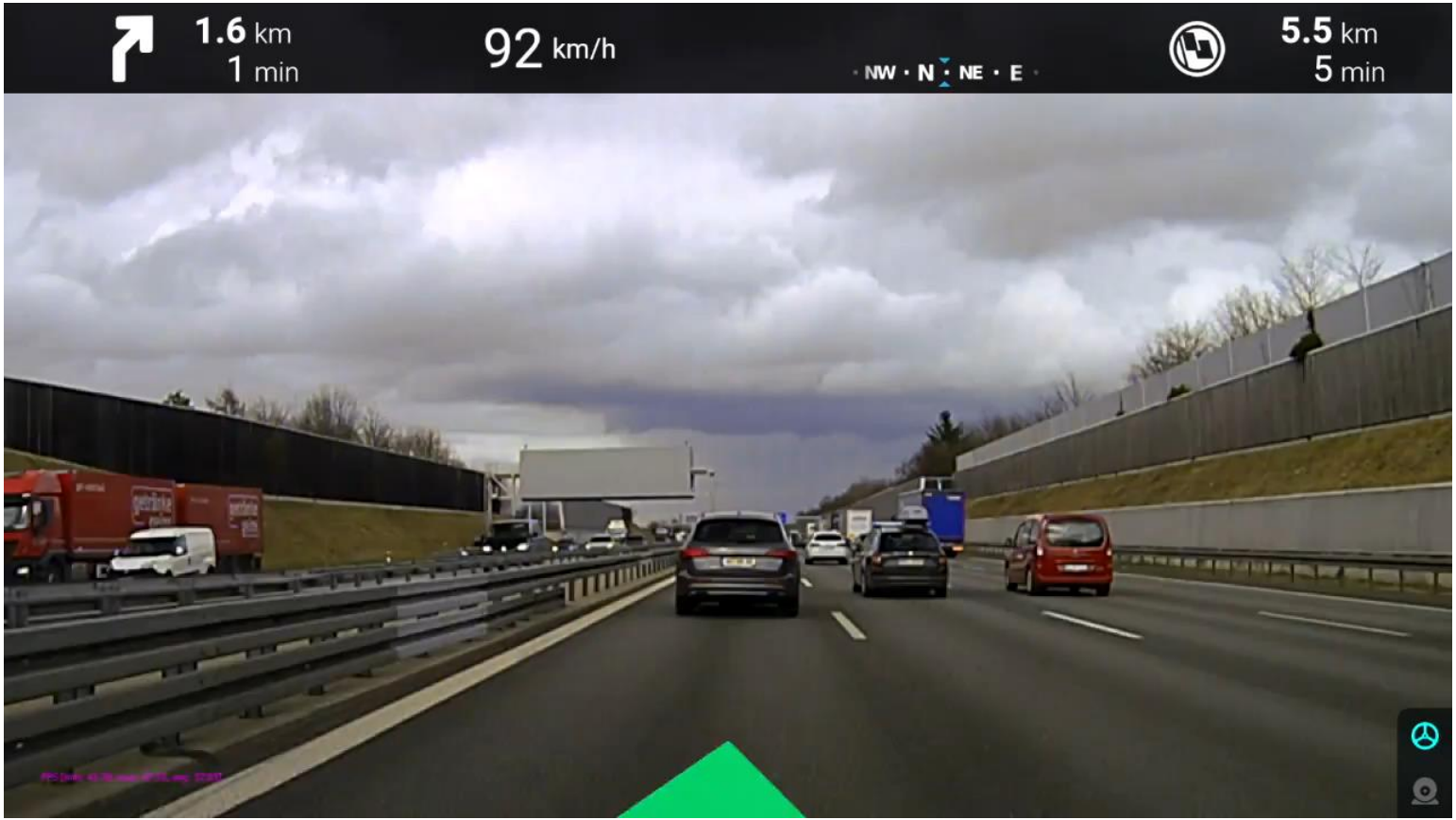
Software

Hardware

Safety of the intended functionality (SOTIF ISO/PAS 21448, published)

Road vehicles – Functional safety (ISO 26262, published)

Augmented Guidance Demo Application



Conclusions

- Besides the DNN architecture, many aspects impact the performance of object detectors
  - Model specific: feature extractor, input resolutions, matching strategy, IoU threshold
  - Data specific: training data, augmentation
- SSD with MobileNet provides the best accuracy tradeoff within the fastest detectors
  - SSD is fast but performs worse for small objects comparing with others
  - For large objects, SSD can outperform other meta architectures with lighter extractors
- Higher detection frame rates with lower accuracy (mAP) are better for consistent object tracking
  - Lower latency (better estimation of real time position)
  - High refresh (smooth tracking)
- GPU computing capabilities and solid library optimization enable real time perception for complex recognition models
- Full road object coverage will require more computing power from next generation embedded PCs

APOSTERA

DS DATA SCIENCE UA

THANK YOU

Sergii Bykov

Technical Lead R&D

sergii.bykov@apostera.com

+38 050 4191953