The Building Blocks of Autonomous Control

Phil Magney, Founder & Principal Advisor
July 2016
• VSI Remarks
• The Building Blocks of Autonomy
• Elements of Autonomous Control
• Motion Control (path, maneuver, trajectory)
• Sensor Fusion
• Processing for Autonomous Control
• ECU Consolidation
• Software Defined Car
• Functional Safety Requirements
AV solutions are very complex

Nobody owns the whole value chain – various domains of expertise

Big Tech cannot go it alone – they need OEM and tier-one expertise

Traditional auto cannot go it alone – they need robotics, AI, machine learning, deep learning and massive IT support!
  - Active safety is largely rules based -- “deterministic” approach
  - Full autonomy may not be rule based – there are too many corner cases

Traditional suppliers can stay relevant with focus on the mechatronics – where the digital world meets the mechanical world!

The balance of power is definitely changing, particularly in the context of “new mobility”
The Challenges of Autonomy

- Level 2 autonomy well under way, but L3 challenging
- L4/L5 imminent over time – heavy tech push
- Take lots of software expertise: Perception, Motion Control, Predictive Learning, Safety, Security, etc. Lots of M&A activity
- Machine learning will be vital for L4/L5:
  - For training the classifiers to handle nearly infinite scenes and images
  - For training predictive control (maneuvers and trajectories) as there are infinite possible situations in which an AV will need to react.
- The sensing problem is largely understood – motion planning, decision and arbitration is the big challenge now!
- The processor companies are in a race to deliver the best instruction sets that can consolidate perception and control
- The AV Cloud = Data Aggregation + Deep Learning + OTA + Supervision + Security + Enterprise Management
What is Driving Auto Tech?
From ADAS to Autonomous Control
Two Approaches to Autonomy – Not a Zero Sum Game!

Automated Driving

**Incremental Approach**
- Evolution of ADAS
- SAE Levels 1, 2, 3
- Conditional Autonomy
- Step by Step approach
- Convenience and safety
- Existing automotive strategy

Highly/Fully Automated Driving

**Driverless Car Approach**
- Mobility as a Service (MaaS)
- SAE Levels 4 and above
- Full or highly automated
- Pursued by tech industry
- Requires machine learning
- Requires AV Cloud

Copyright 2016 - Vision Systems Intelligence, LLC.
The Evolution of Autonomy

- **Level 1**: Active (control-based) ADAS Solutions
  - Active lane keeping, adaptive cruise control, automatic emergency braking, etc.
- **Level 2**: Partial Autonomy
  - GM SuperCruise, Audi Traffic Jam Assist, Tesla Autopilot
- **Level 3**: Conditional Autonomy (i.e. Piloted driving) but driver required
- **Level 4**: Highly Autonomous Vehicle
  - Driverless in certain areas
- **Level 5**: Fully Autonomous Vehicle
  - Driverless door-to-door

- **Driverless Cars**

- **SAE Level of Autonomy**

- **2010**
- **2015**
- **2020**
- **2025**
- **2030**
- **2035**
- **2040**

Copyright 2016 - Vision Systems Intelligence, LLC.
Elements of Autonomous Control Systems
## Technologies for Autonomous Control

<table>
<thead>
<tr>
<th>Technologies</th>
<th>Reasoning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sensors</strong></td>
<td>Camera, Radar, LiDAR, IR, inertia, Ultrasonic, etc.</td>
</tr>
<tr>
<td><strong>Safety Domain Controllers</strong></td>
<td>These controllers can acquire, sense, interpret, classify and make decisions to actuate control</td>
</tr>
<tr>
<td><strong>Vision Processing</strong></td>
<td>Vision processing is very compute intensive. Requires massively parallel computing architectures</td>
</tr>
<tr>
<td><strong>Machine Learning</strong></td>
<td>Training the algorithms -- classifiers, pooling, semantic labeling, etc. Machine learning also includes CNN, DNN, SVM</td>
</tr>
<tr>
<td><strong>Sensor Fusion</strong></td>
<td>Requires a unique set of software tools for development of algorithms -- data association, clustering, segmentation, filtering, estimation, and motion prediction</td>
</tr>
<tr>
<td><strong>Control Systems</strong></td>
<td>Predictive control for optimal vehicle trajectory using information from environmental sensing systems</td>
</tr>
<tr>
<td><strong>Reference Maps</strong></td>
<td>Road features, road geometry, and very dynamic -- enables autonomous method called “sense and align” -- Also enables advanced safety</td>
</tr>
<tr>
<td><strong>Communications</strong></td>
<td>V2V, V2I, V2P, V2C (Vehicle, Infrastructure, Pedestrian, and Cloud)</td>
</tr>
<tr>
<td><strong>High Capacity Networks</strong></td>
<td>Necessary to handle the flow of information being generated in the vehicle. Transition from CAN to higher performance networks such as Ethernet</td>
</tr>
<tr>
<td>Sensor</td>
<td>Attributes</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Mono Camera</td>
<td>Single focal point – good for identification of objects, improving for 3D attributes</td>
</tr>
<tr>
<td>Stereo Camera</td>
<td>Dual stereoscopic focal points – very good for depth and motion within sweet spot of converged focal points</td>
</tr>
<tr>
<td>Tri-Focal Camera</td>
<td>Three different focal points – like 3 monocsms – increases range and field of view (FOW)</td>
</tr>
<tr>
<td>Radar</td>
<td>24-79GHz – many applications, very accurate ranging – new millimeter wave using CMOS – can sense pedestrians</td>
</tr>
<tr>
<td>Flash Lidar</td>
<td>Very low resolutions – ToF (time-of-flight), restricted for identification – good for sensor fusion applications</td>
</tr>
<tr>
<td>Solid State Lidar</td>
<td>Good resolutions via point cloud – better at identification – good for L3/L4/L5 perception</td>
</tr>
<tr>
<td>360 Degree Lidar</td>
<td>Good resolution via point cloud – best for data acquisition – required for L4/L5</td>
</tr>
<tr>
<td>Infrared (IR)</td>
<td>Thermal camera – useful for night vision and driver monitoring – sometime IR emitters used for ToF</td>
</tr>
<tr>
<td>HD Maps</td>
<td>Useful for localization – necessary for L4/L5 – very useful when combined with dynamic content on road situation</td>
</tr>
<tr>
<td>IMU</td>
<td>Inertia Measurement – necessary for localization and path planning – need high resolution dead reckoning</td>
</tr>
</tbody>
</table>
Sensors – RAW Sensors, Sensor Modules, Sensor Fusion
Perception Plus Control – Sense, Decision, Action...

Image Sensors
- Camera(s)
- Thermal
- Scanning LiDAR

Ranging Sensors
- Radar
- Flash LiDAR
- Ultrasonic

Map Data
- Path planning
- Localization
- Dynamic content

Sensor Fusion Algorithms
- Object
- Track-to-track fusion
- RAW data fusion

Control Algorithms
- Trajectory control
- PID Control
- State Machines
- MPC Control

Wheel speed sensor
Inertia measurement
Throttle position
Steering Angle

Longitudinal Control
Latitudinal Control

Copyright 2016 - Vision Systems Intelligence, LLC.
Who is Assembling Autonomous Solutions (perception + control)
• **Path Planning** – Dependent on the digital reference maps as well as the incremental data that supplement this. Incremental data includes dynamic content such as lane closures, detours, or traffic conditions.

• **Maneuverer Planning** – these are the decision algorithms that have to assess risk and evaluate dynamic elements. These algorithms are based on decision theory and best trained through massive data collection and rigorous training.

• **Trajectory planning** – this has a higher level role compared to a path or maneuvers. It contains the geometric concept of a path but it also encompasses vehicle states plus the kinematic properties.
Leading Issue → Petabytes of geospatial data generated from 3D scans are essential for the creation of 3D semantic maps. Data collection will become commoditized. Extracting features from the data remains bottlenecked. The culprit: a process dominated by manual labeling and vectorization of the input data.
Autonomous Control: The Use Case for Maps

- Maps enable **redundancy of sensors**.
- Maps better **optimize perception** systems — takes some load off of the environmental sensors.
- Maps aid in the **localization and path planning** for autonomous control — provides a guidance method.
- Autonomy requires **dynamic content** such as ice, potholes and construction-related course adjustments and/or obstacles.
- Autonomous Vehicle needs map to separate unlabeled objects from the static picture of the world with which to compare what it sees.
Mapping Assets That Will Enable Autonomy
Machine Learning:
Teaching Cars To Drive
Machine Learning vs. Deep Learning

**Deterministic**
This approach to self-driving cars comes with a measure of verifiability that gives confidence that the design works and why it doesn’t. The problem with this “engineering” approach is that the system can't cope with anything that it encounters that is new.

**Probabilistic**
This approach can generalize to situations it has never been trained on. This means that a self-driving car using an end-to-end approach could be more capable, but also perhaps less predictable.
Neural Networks

• Deep Neural Networks (DNN) is the latest hot topic for autonomous vehicle development.
• The goal is to “train” algorithms to handle limitless possibilities and outcomes – this is also known as “machine learning”
• Is applied to both perception (sense and understand) and control (maneuvers and trajectories)
• Algorithms need to be trained with real or simulated data
  • Supervised learning: Learn by examples as you try to predict output a target vector $y$, from a matrix of inputs, $x$. Training is done with known inputs and associated known outputs.
  • Unsupervised learning: Attempt to predict the target vector $y$ (unknown) using the very same matrix $x$ as the inputs. The network learns something intrinsic about the data without the help of a target or label vector that is often created by humans. Uses weights.
  • Semi-supervised learning: Training data includes a few desired outputs.
  • Reinforcement learning: Rewards from a sequence of actions.
• NVIDIA approach → purports “end-to-end” training for self driving car project, called Dave-2. The neural network learns to steer by being shown videos of a human driving and what the human driver did to the steering wheel as a result.
• Mobileye approach → favors semantic abstraction which breaks up the learning into blocks. Mobileye says there are too many corner cases with NVIDIA approach.
• Conventional approach → current self-driving car approach breaks the task into different components – sense & identify; localize against surroundings; then you plan your motions.
Sensor Fusion For Active Safety & Autonomous Control
Developing Sensor Fusion – Step-by-Step

1. **Modeling tools** are necessary to examine functional requirements as well as to determine hardware and software requirements.

2. Sensor selection considers all **environment sensors** such as camera, radar, LiDAR, ultrasonic, as well as other sensors such as inertia, positioning, throttle position, steering angle etc.

3. Safety goals and hazard ratings must be established formally according to **ISO 26262** – most active safety and autonomous control will be ASIL D resulting very stringent requirements.

4. **Fusion of sensor data** can be done with objects (complete sensors modules such as Mobileye) or can be done with RAW data or a combination.

5. Special tools are necessary for sensor signal synchronization and timing such as **ADTF, ROS, RT Maps, Polysync**, etc. They record, playback and analyze the “fused” sensor data.
6. Development will require a hardware development platform to test the feasibility of your applications – typically FPGA-based. This may be in advance of a chosen “target” processor.

7. Lots of simulation software will be required for Model-in-loop (MiL) for initial examination, Hardware-in-loop (HiL) for sensors and ECUs, Software-in-Loop (SiL) for application testing.

8. Environmental models (simulated roads, objects, signs etc.) will be necessary unless you have collected your own data sets.

9. Selection of a target processor for perception (typically SoC) that can handle the requirements of your applications (i.e. vision processor). You will need to select a target processor for control (typically dual-core lockstep)

10. You will need ISO 26262 compliant production code generation tools in accordance with MISRA requirements (Motor Industry Software Reliability Association).
Processing Requirements
CPU Load for Object Recognition

Figure 2: Computational Load and Power Consumption for Situation-Analysis Functions (Concept)

Chart Source: Renesas
Processing for Autonomous Functions

- **ASICs** – have the advantage of high performance and low power consumption. They are not suitable for rapid prototyping and they are not reconfigurable.

- **FPGA** – often used for prototyping to implement the architecture of a microprocessor “soft processor core” and the required additional custom hardware accelerators. They are not so good for the serial processing necessary in mid and high levels.

- **GPU** – Current GPUs are power hungry and are not ideal for embedded applications. However, GPUs are very good with vector processing and as well as deep learning routines.

- **DSPs** – Attractive for embedded automotive applications since they offer a good price to performance ratio. DSPs are essentially for signal processing which is a core element of sensor based solutions. DSP logic in now commonly found in SoCs.

- **MPUs** – a good option for high-level vision processing. Additionally, they are easy to program, since it is possible to use the same tools and libraries used for standard PC applications. ARM architectures are leading MPU instruction sets.

- **SoCs** – Many new products are entering the market which include the elements of the above approaches combining standard microprocessors with added functional such as FPGA fabric, DSP functionality and unique memory cores for pipelined processing.
Processor Activities Related to Active Safety & Autonomy
Key Enablement Platforms

• **NVIDIA DriveWorks** -- NVIDIA DriveWorks software development kit (SDK) gives developers a foundation upon which to build applications for object detection, map localization, and path planning.

• **NXP’s BlueBox** -- is a central computing engine that enables all aspects of perception, decision and motion planning tasks. The NXP BlueBox platform is a Linux-based open-platform that is programmable in C allowing automotive manufacturers to customize their solutions.

• **Mobileye** – now engaged in AV development activities on two OEM programs with a pending third OEM. Most AV activities centered around EyeQ5 SoC.

• **Elektrobit** – robitos is an interface specification for autonomous development. Includes many EB blocksets and tools.

• **Polysync** – comes from start up (Harbrick) that has created an abstraction layer with APIs so that developers can add their applications for AV control. (pure tech vision as “software define car”)

• **Renesas Skyline Concept** – aggregates all nodes into a vehicle, along with various partners, for test and development of AV solutions.

• **Texas Instruments RT-RK Platform** – a development kit for ADAS and autonomous solutions.
ECU Consolidation
Vehicles are based on a distributed architecture where data is shared among different ECUs using high-speed deterministic networks!
The amount of bandwidth necessary to handle raw data from all sensors is a big challenge for designers of active safety of autonomous control systems!
Consolidation of Sensing ECUs

Singular Approach

- **Sensors / Acquisition**
  - Camera
  - Radar
  - Laser
  - Ultrasonic
  - V2X
- **Sensor Processing**
  - Vision Processor ECU
  - Radar Processor ECU
  - Laser Processor ECU
  - Ultrasonic Processor ECU
  - V2X Processor ECU
- **Data Processing**
- **Fusion**
  - Decision - Making MCU
- **Vehicle Systems**
  - Braking
  - Throttle
  - Steering
  - Driver Alert

Consolidated Approach

- **Sensors / Acquisition**
  - Camera
  - Radar
  - Laser
  - Ultrasonic
  - GPS
  - V2X
  - Vehicle Telemetry
- **Sensor Processing**
- **Data Processing and Fusion**
- **Vehicle Systems**
  - Braking
  - Throttle
  - Steering
  - Driver Alert
Functional Elements of Autonomous Control

Perception

- Sensor Data Fusion
  - HD Positioning
  - Object Fusion
  - Grid Fusion
  - Road and Lane Fusion
  - Vehicle Database

Function Specific Views

Decision

- Situative Behavior Arbitration
  - Situation Analysis
  - Path Planning

Control

- Motion Management
  - Trajectory Control
  - Longitudinal Control
  - Lateral Control

- Vehicle Abstraction - Actuators

HMI Management

Safety and Error Management

Fail Safe

Elektrobit robinos Architecture

Copyright 2016 - Vision Systems Intelligence, LLC.
Software Defined Car

Conventional Architecture

Software Centric Approach
Audi zFAS Autonomous Domain Controller

**Processors**
- nVidia Tegra K1 SoC
- Mobile EyeQ3 SoC
- TTTech NIC
- Infineon TC29xT (tri-core)
- Altera FPGA
- Delphi

**Sensors**
- Ultrasonic
- Front Radar 250m
- Front Laser Scanner 80m
- Rear Radar
- Surround Cameras
- Wide angle front camera
- GPS up to 2cm
- LTE wireless
- Cloud-based Deep Learning

*The zFAS controller represent the “brains” of autonomous control. Massive processing support is necessary to handle all sensor inputs, decipher objects and react in real-time!*
The NVIDIA DRIVE PX platform enables the development of systems that capture and process multiple HD camera and sensor inputs and supports advanced graphics, computer vision and machine learning.
Functional Safety
• **Functional Safety** means “absence of unreasonable risk due to hazards caused by malfunctioning behavior of E/E systems”

• ISO 26262 is still firmly grounded in the constraints of a traditional vehicle where the presence of a driver provides a failsafe mechanism. As such, under the current scope of ISO 26262, there are no current provisions for a driverless vehicle.

• It is applicable to safety-related automotive systems that include one or more E/E systems and that are installed in series production passenger cars with a max gross weight up to 3.5t”

• ISO 26262 addresses architectural & functional aspects and procedural aspects – to avoid systematic faults and to control random faults

• Automotive Safety Integrity Level (ASIL) are assigned values to systems from A (lowest) to D (highest)

• Key components that are part of a system can be certified as a Safety Element out of Context (SEooC)

• ASIL decomposition provides method for reducing ASIL levels (ASIL A+C = ASIL D)
ASIL Rating Examples

Functional Safety: a topic for all four automotive sub-segments

Source: Freescale: Expectations based on global customer feedback

Copyright 2016 - Vision Systems Intelligence, LLC.
Key Take Aways

- Active ADAS systems are the building blocks of autonomous control and lend to the traditional OEMs in terms of strategy
- Companies (OEMs, tier-1s) wanting to go deeper into the value chain to have more control over the functionality – especially for autonomous control features
- Big Tech Brings disruptive technology to traditional automotive school of thought – probabilistic vs. deterministic architecture.
- New methods in machine learning offer great promise for autonomous control
- Data acquisition including highly detailed mapping and crowd sourced dynamic content becomes vital to autonomy and to “learning.”
- HMI will see a lot of innovation from autonomous control features – lots of new research on this but current concepts appear fragmented
- Functional Safety practices do not currently align with driverless technologies. Should safety practices adopt “outcome-based” examination?