

*Linear Algebra*

# Course Introduction

Automotive Intelligence Lab.



# Contents

## ■ Introduction

- ▶ Staff
- ▶ Prerequisites

## ■ Course background

## ■ Course objectives

## ■ Course schedule and evaluation

## ■ Assignment

# Course Staff

## ■ Professor

- ▶ Kichun Jo
- ▶ Email : [kichunjo@hanyang.ac.kr](mailto:kichunjo@hanyang.ac.kr)

## ■ Teaching assistant

- ▶ Seungwoo Heo
  - Email : [seungwooheo@hanyang.ac.kr](mailto:seungwooheo@hanyang.ac.kr)
- ▶ Jahyun Koo
  - Email : [jahyunkoo@hanyang.ac.kr](mailto:jahyunkoo@hanyang.ac.kr)
- ▶ Jeongwoo Park
  - Email : [jeongwoopark@hanyang.ac.kr](mailto:jeongwoopark@hanyang.ac.kr)



**허승우**  
Seungwoo Heo  
석사과정  
M.S. student

### Research Interests

Model Predictive Control



[seungwooheo@hanyang.ac.kr](mailto:seungwooheo@hanyang.ac.kr)



**구자현**  
Jahyun Koo  
석사과정  
M.S. student

### Research Interests

Motion Prediction, Local Planning, MPC



[jahyunkoo@hanyang.ac.kr](mailto:jahyunkoo@hanyang.ac.kr)



**박정우**  
Jeongwoo Park  
석박사통합과정  
M.S. student

### Research Interests

Object Tracking, Localization



[jeongwoopark@hanyang.ac.kr](mailto:jeongwoopark@hanyang.ac.kr)

# Course Information

## ■ Class

- ▶ Lecture & Exercise

## ■ Communication with LMS

- ▶ Announcements
- ▶ Lecture notes
- ▶ Assignments

## ■ Prerequisites

- ▶ Computer(laptop, tablet, ...) to write code on MATLAB.
- ▶ Basic algebra and geometry.
  - High-school level
- ▶ Basic programming skills.
  - Freshman level

## ■ Textbook

- ▶ We will provide lecture slides.

## ■ Reference

- ▶ “Introduction to Linear Algebra” – Gilbert Strang
- ▶ 3Blue1Brown (Youtube)
- ▶ Angelo’s math notes (공돌이의 수학노트)
- ▶ Matlab

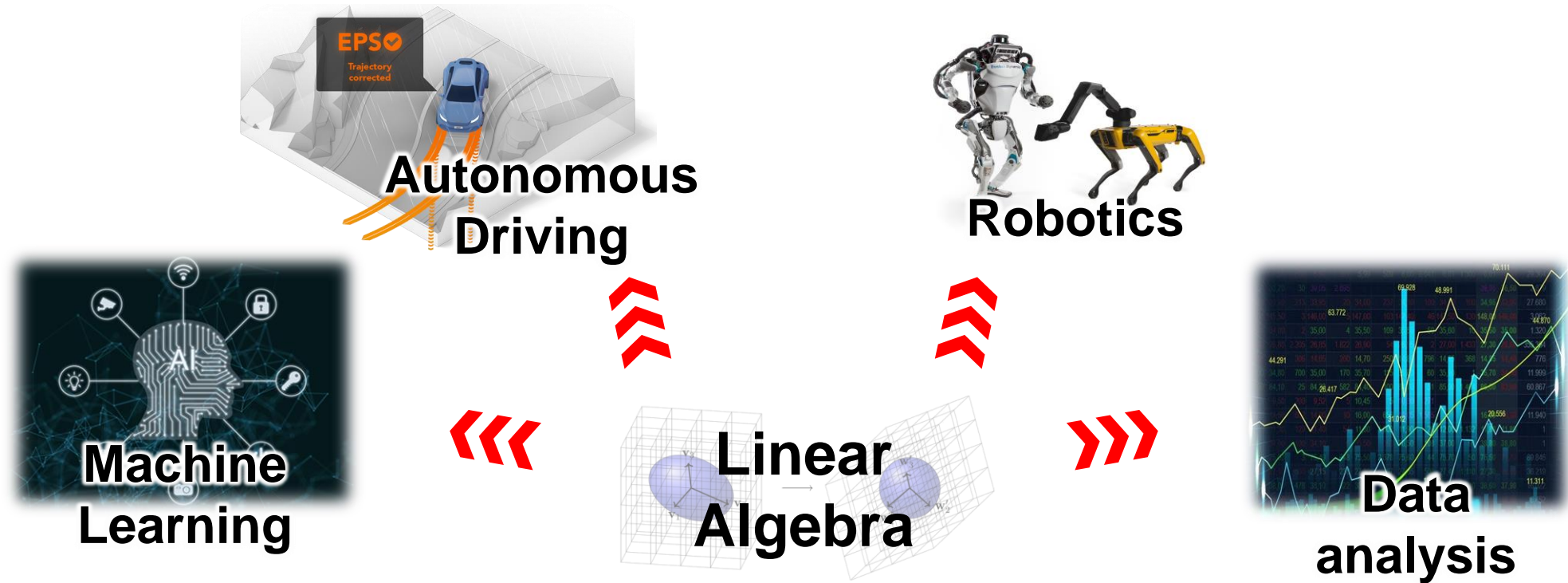
# Course background

# History of Linear Algebra

- In the 17th century in the West and much earlier in China, **matrices** were just used to provide a compact notation for storing sets of numbers.
- In the 20th century, **matrices** and **vectors** were used for **multivariate mathematics** including calculus, differential equations, physics, and economics.
- But most people didn't need to care about matrices until fairly recently.
  - ▶ Here's the thing: **computers** are extremely efficient at working with matrices.
  - ▶ And so, modern **computing** gave rise to modern linear algebra.



# Why Learn Linear Algebra



**Linear algebra** is utilized in various fields for machine learning, autonomous driving, robotics, and others.

*To understand how the algorithms work and to develop them, you should learn **Linear Algebra** !*

# Trend 1: Software Defined Vehicle (SDV)

- **SDV (Software-Defined Vehicle)** refers to vehicles whose functions and services are defined primarily by software.
- **Continuous feature improvements** are possible throughout the vehicle's lifespan via **software updates**.

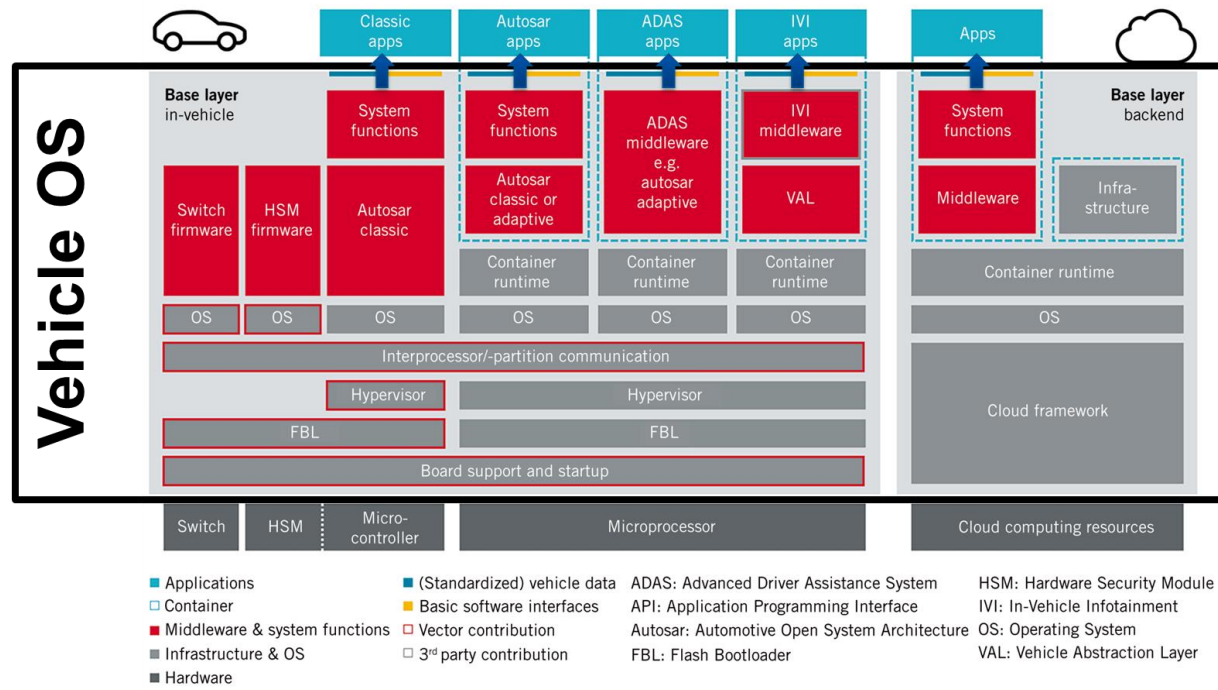


FIGURE 2 Architecture and building blocks of the base layer (© Vector Informatik)





# Automobile in the 1st and 2nd Industrial Revolutions

## ■ Mechanization and mass production ( - 1980)



# Automobile in the 3rd Industrial Revolution

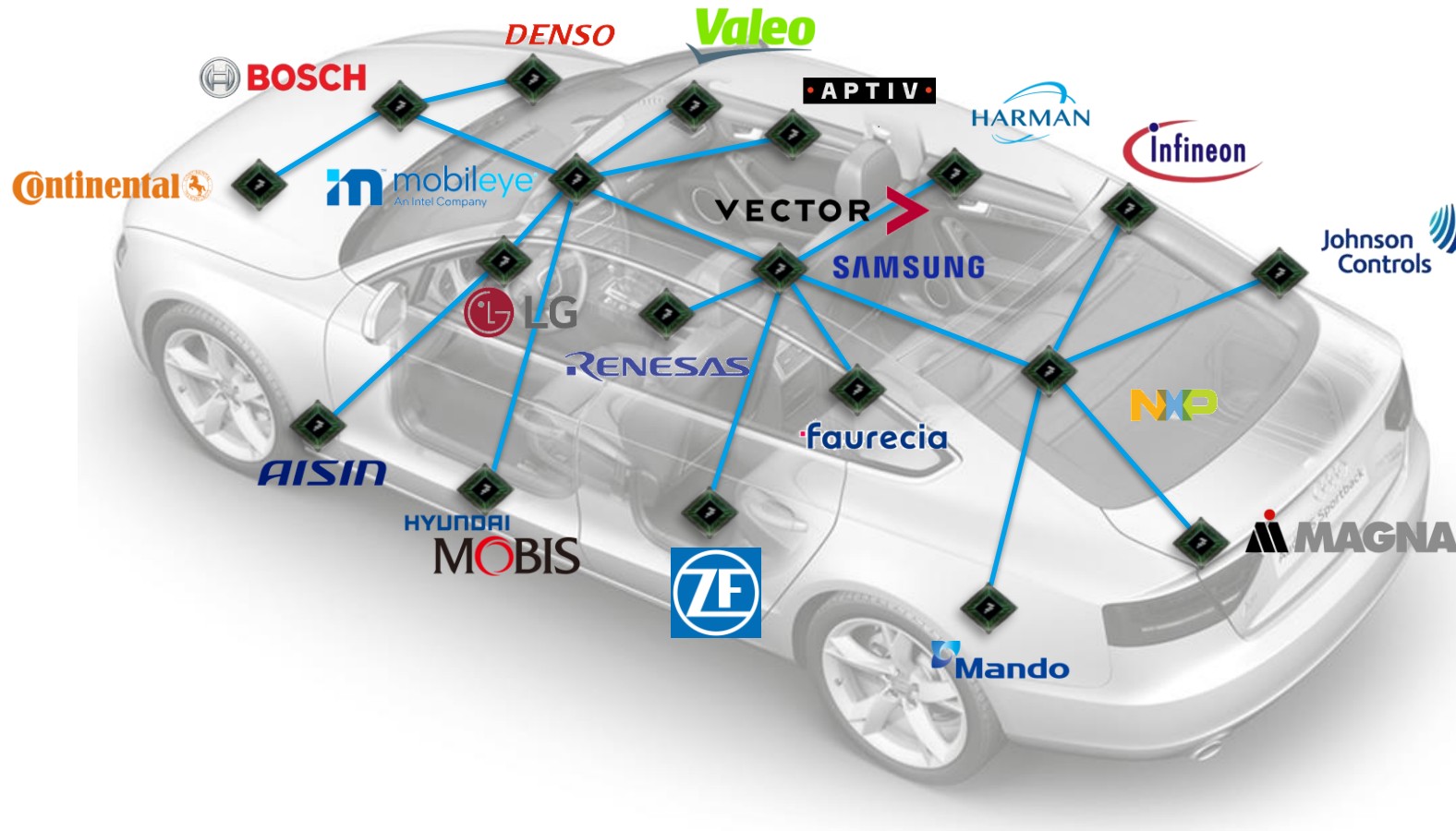
- Electronics and embedded system (1980 – )
- From this time on, **software** were applied to the automotive functions.



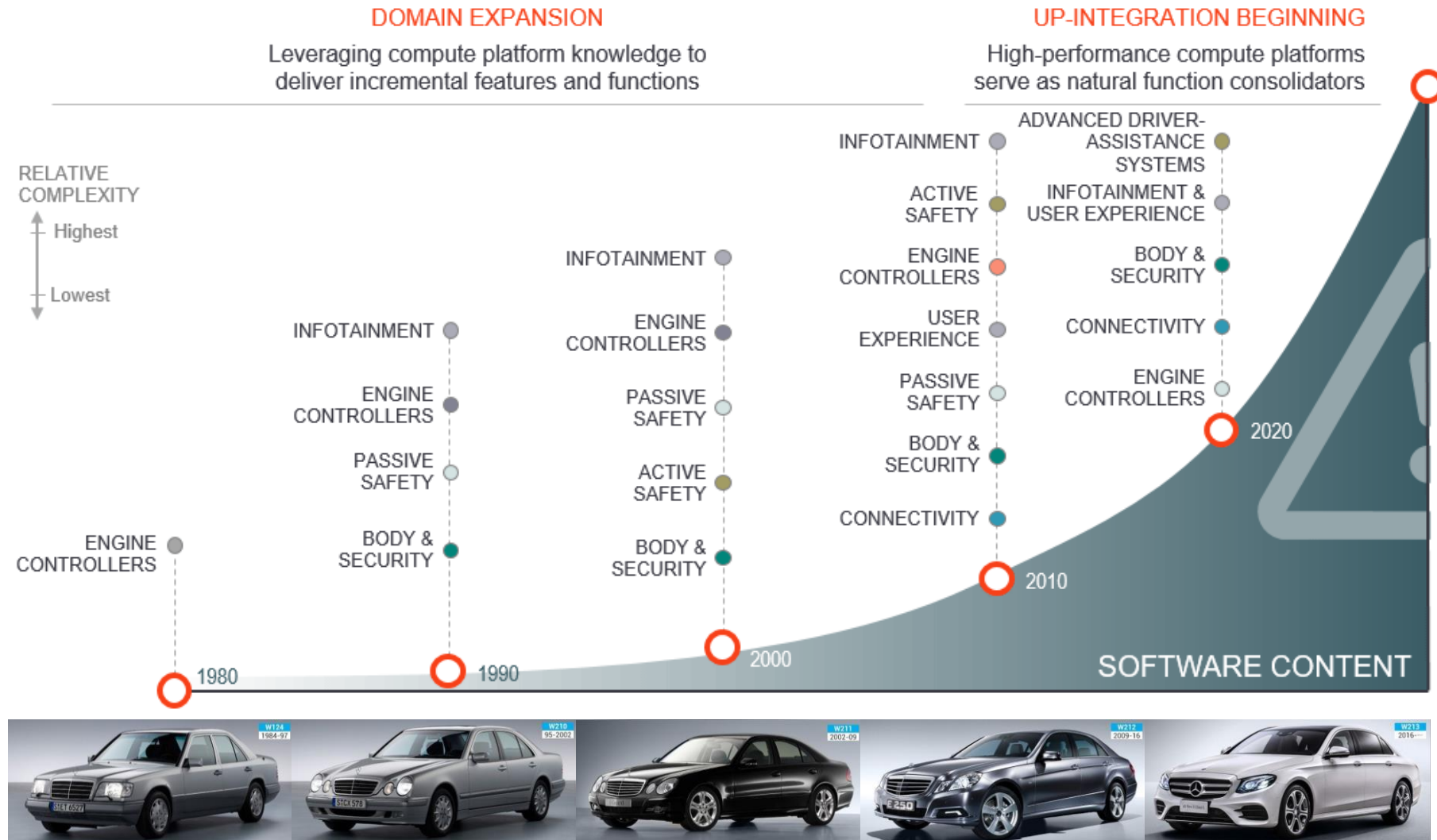
# Current Automotive Industry

## ■ OEM – Supplier structure

- ▶ HW-SW tightly coupled.
- ▶ Each supplier develops its own systems (sensors, computers, actuators) and the SW that relies on them.



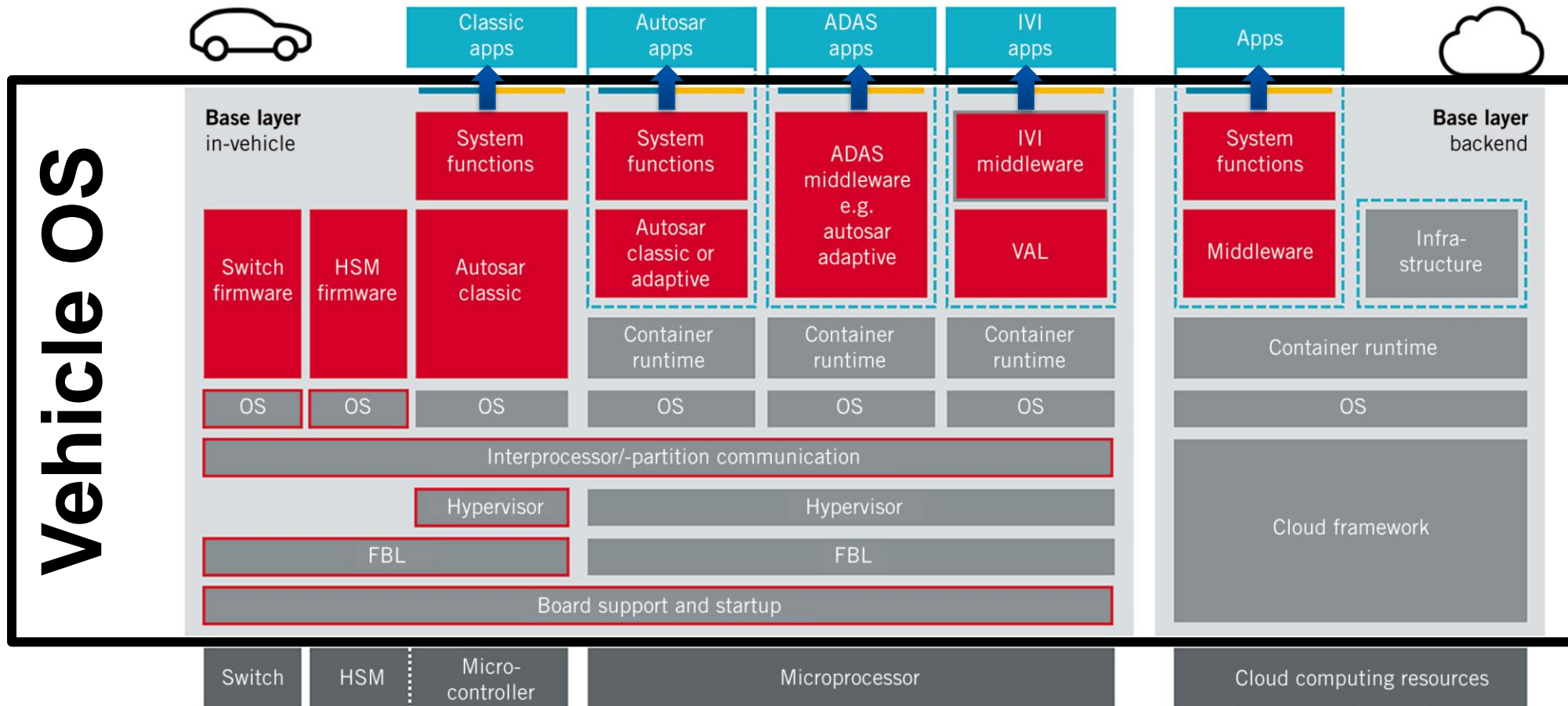
# Software Content in Automobile



[출처: Aptiv]



# Vehicle OS (Software Platform)



■ Applications

□ Container

■ Middleware &amp; system functions

■ Infrastructure &amp; OS

■ Hardware

■ (Standardized) vehicle data

■ Basic software interfaces

□ Vector contribution

□ 3<sup>rd</sup> party contribution

ADAS: Advanced Driver Assistance System

API: Application Programming Interface

Autosar: Automotive Open System Architecture

FBL: Flash Bootloader

HSM: Hardware Security Module

IVI: In-Vehicle Infotainment

OS: Operating System

VAL: Vehicle Abstraction Layer

FIGURE 2 Architecture and building blocks of the base layer (© Vector Informatik)



# The Future of the Automotive SW Industry as seen on Cell Phones (I)

## Features defined by Hardware



# The Future of the Automotive SW Industry as seen on Cell Phones (II)

Features defined by  
**Hardware**



Features defined by  
**Software**



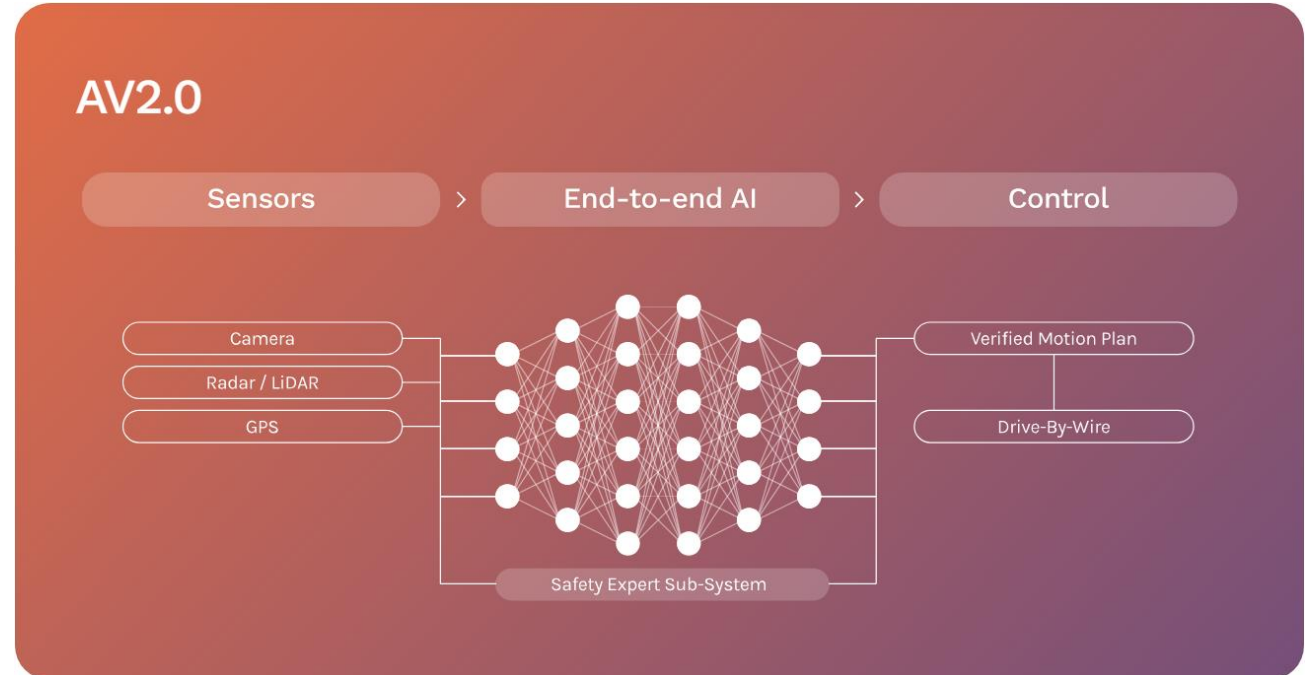
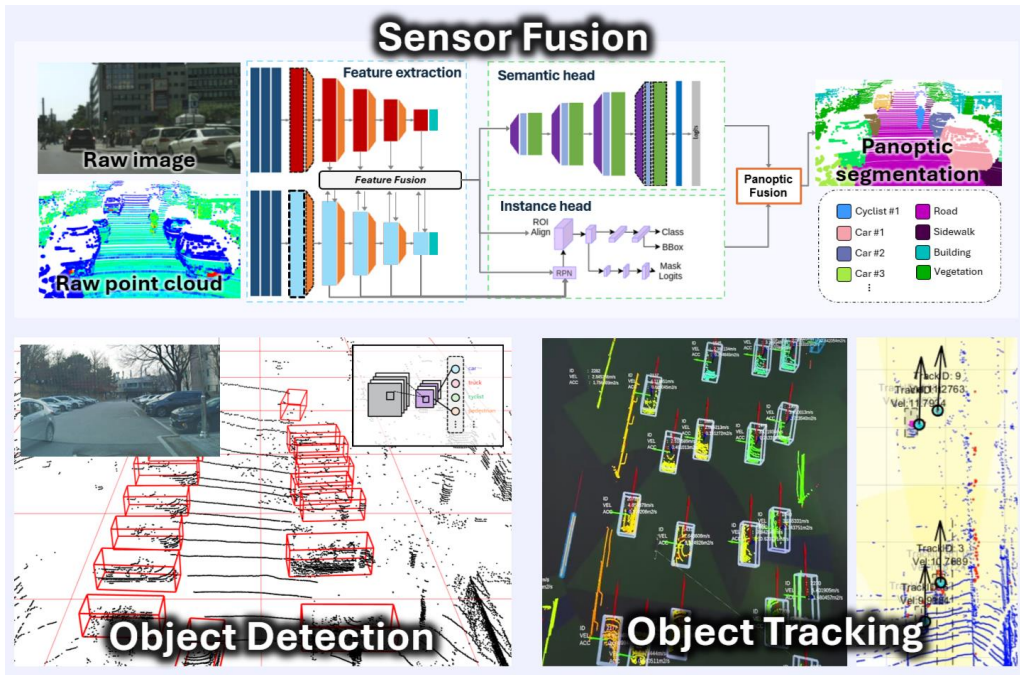


# Software for Automobile



# Trend 2: AI for Automobile

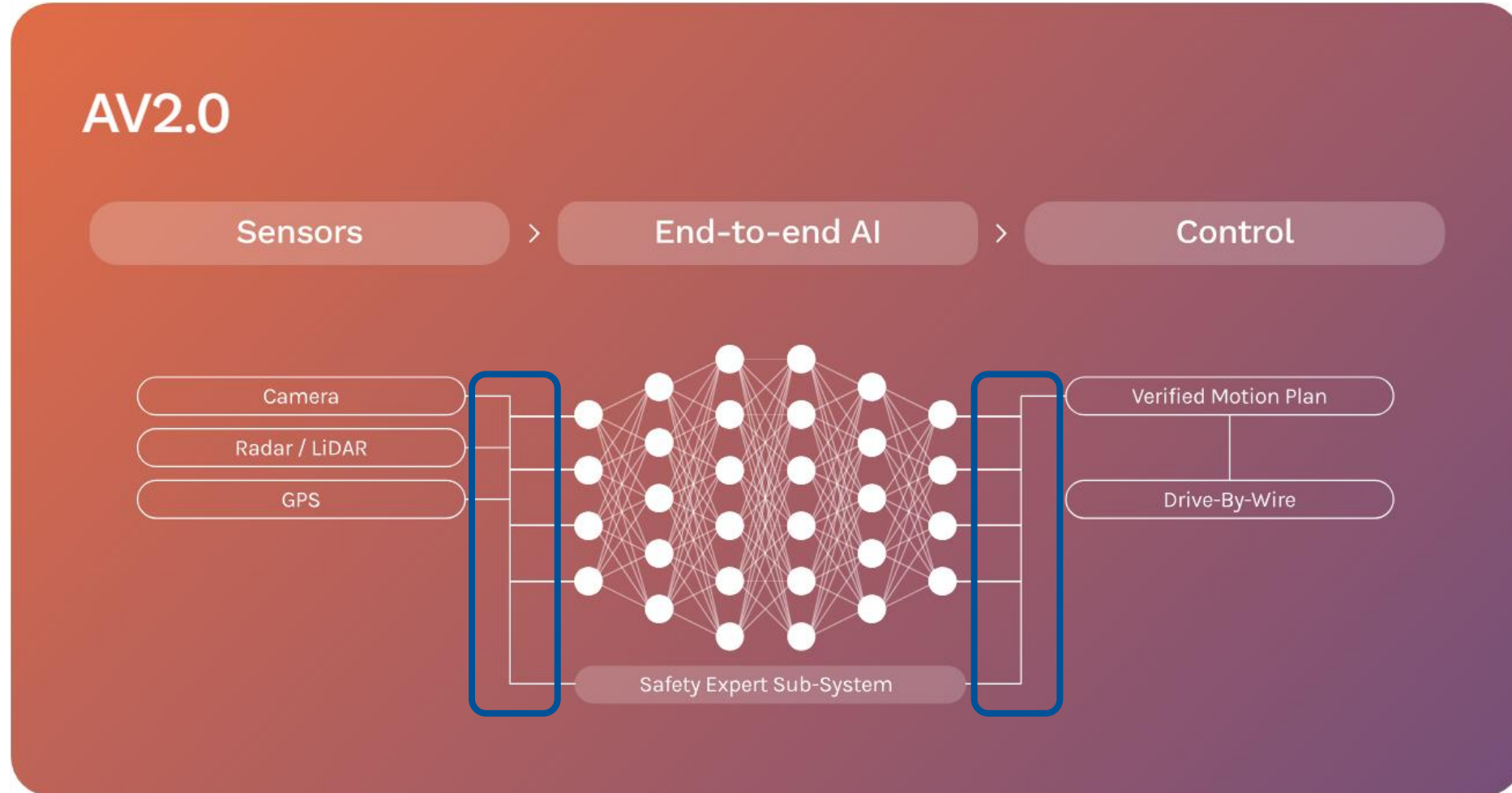
■ AI is being used for many functions in automobiles!



# Applications of AI for Automotive Functions

## ■ End-to-End AI Autonomous Driving

- ▶ Data processing technology is essential!





# Course objectives

# Course Objectives

- Understand linear algebra as an **engineering tool**, **not as mathematics**.
- Understand linear algebra through **programming** **rather than solving problems by hand**.
- Actively utilizes **geometric visual examples** through programming to aid understanding.
- Improve your **programming skills** through **real world problems**.

# Traditional and Modern Linear Algebra

## ■ Traditional and Modern

### Traditional

- Abstract.
- Learned through proofs and equations.
- Emphasizes proofs and abstract concepts, often with little relevance to practical applications or implementations.

### Modern

- Computational.
- Learned through code and applications.
- Emphasizes geometric intuition and implementation of linear algebra concepts in practical applications.

# Mathematical Proofs vs. Visualization and Examples

## ■ The two ways to understand math

### ▶ Mathematical proofs

- A proof in mathematics is a sequence of statements showing that a set of assumptions leads to a logical conclusion.
- Rigor but rarely intuition.

### ▶ Visualizations and examples

- Clearly written explanations, diagrams, and numerical examples help you gain intuition for concepts and operations in linear algebra.
- More intuition than mathematical proofs.

## ■ This class will focus on **visualizations and examples** through MATLAB.

# This class will focus on

## Modern

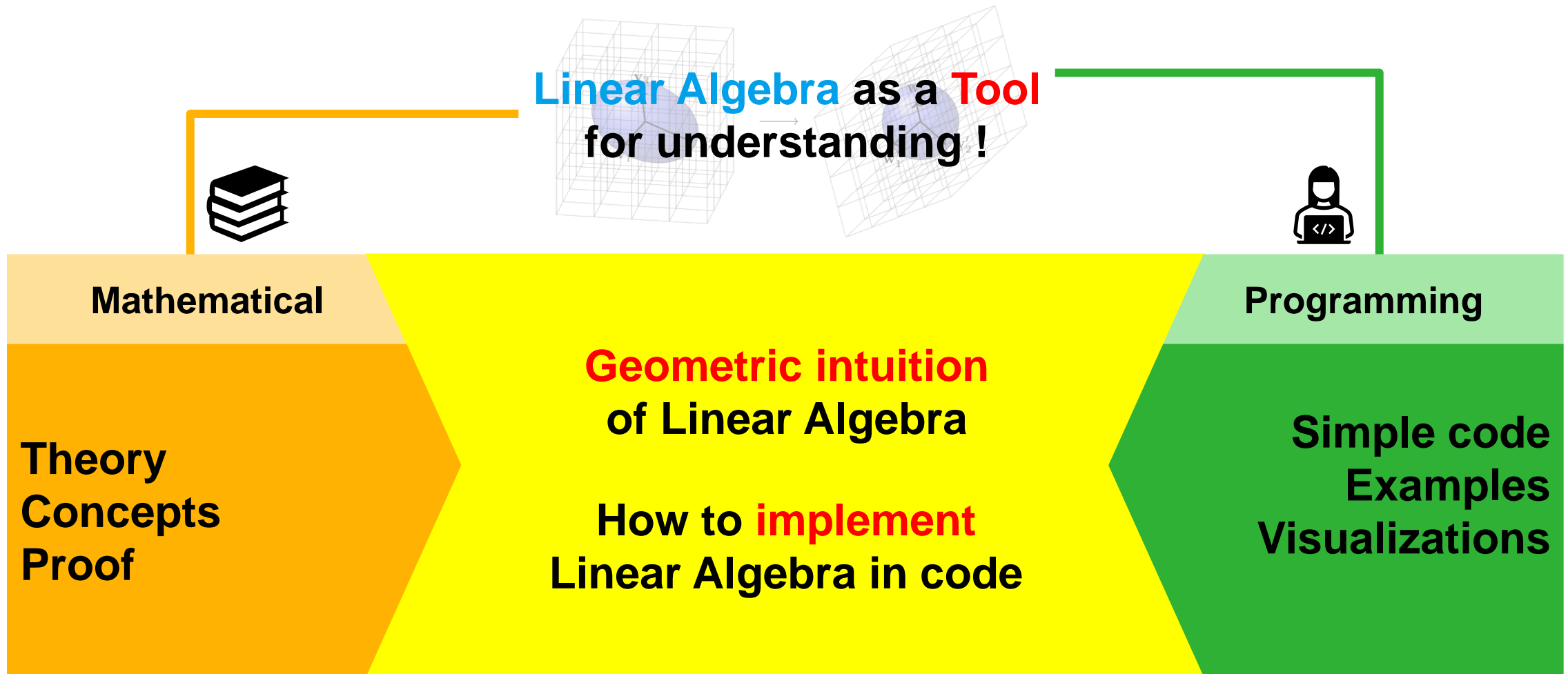
- Computational.
- Learned through code and applications.
- Emphasizes geometric intuition and implementation of linear algebra concepts in practical applications.

Especially here!

**Linear Algebra** as a **Tool**  
for understanding  
data, statistics, deep  
learning, image  
processing, etc.!



# Instructional approach



# Proof based on Matlab

```
% Creating a and b matrix with 4 by 4
```

```
a = rand(4)
```

```
b = rand(4)
```

```
% proof  $a*b$  is not same as  $b*a$ 
```

```
 $a*b - b*a$ 
```

# Course schedule and evaluation

# Schedule (I)

## ■ Week 1 – 8:

Week	Date	1 <sup>st</sup> Class	2 <sup>nd</sup> Class
1	03-05, 03-07	<b>Orientation (lecture)</b> - Introduction of objective, evaluation, and class - Introduction of MATLAB	<b>Chapter 1</b> Vector, part 1 : vectors and basic operations
2	03-12, 03-14	<b>Chapter 1</b> Vector, part 1 : vectors and basic operations	<b>Chapter 2</b> Vector, part 2 : expand concept of vectors
3	03-19, 03-21	<b>Chapter 2</b> Vector, part 2 : expand concept of vectors	<b>Chapter 3</b> Vector applications
4	03-26, 03-28	<b>Chapter 4</b> Matrices, part 1 : matrices and basic operations	<b>Chapter 4</b> Matrices, part 1 : matrices and basic operations
5	04-02, 04-04	<b>Chapter 5</b> Matrices, part 2 : expand concept of matrices	<b>Chapter 5</b> Matrices, part 2 : expand concept of matrices
6	04-09, 04-11	<b>Chapter 5</b> Matrices, part 2 : expand concept of matrices	<b>Chapter 6</b> Matrix applications
7	04-16, 04-18	<b>Chapter 7</b> Matrix inverse	<b>Chapter 7</b> Matrix inverse
8	04-23, 04-25	<b>Midterm Exam</b>	<b>Midterm Exam</b>

# Schedule (II)

## ■ Week 9 – 16:

Week	Date	1 <sup>st</sup> Class	2 <sup>nd</sup> Class
9	04-30, 05-02	<b>Chapter 8</b> Orthogonal Matrices and QR Decomposition	<b>Chapter 8</b> Orthogonal Matrices and QR Decomposition
10	05-07, 05-09	<b>Chapter 9</b> Row reduction and LU Decomposition	<b>Chapter 9</b> Row reduction and LU Decomposition
11	05-14, 05-16	<b>Chapter 10</b> General Linear Models and Least Squares	<b>Chapter 11</b> Least Squares applications
12	05-21, 05-23	<b>Chapter 11</b> Least Squares applications	<b>Chapter 12</b> Eigen Decomposition
13	05-28, 05-30	<b>Chapter 12</b> Eigen Decomposition	<b>Chapter 13</b> MATLAB Onramp – Linear Algebra
14	06-04, <b>06-06</b>	<b>Chapter 14</b> Singular value Decomposition	-
15	06-11, 06-13	<b>Chapter 14</b> Singular value Decomposition	<b>Chapter 15</b> Eigen Decomposition and SVD applications
16	06-18, 06-20	<b>Final Exam</b>	<b>Final Exam</b>



# Evaluation



10%: attendance



30%: assignment



30%: mid-term project and/or exam.



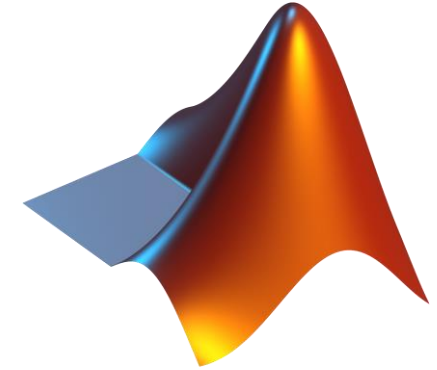
30%: final project and/or exam.

# Assignment

# MATLAB Onramp

## ■ MATLAB?

- ▶ **MATLAB** is a programming and numeric computing platform used by millions of engineers and scientists to analyze data, develop algorithms, and create models.



## ■ MATLAB Onramp

- ▶ MATLAB Onramp is a course that teaches you the basics of using MATLAB.
- ▶ Onramp consists of exercises to answer the questions correctly by entering commands directly and video lecture.

### MATLAB



#### MATLAB Onramp

15개 모듈 | 2시간 | 언어

MATLAB의 기본 사항을 빠르게 학습할 수 있습니다.

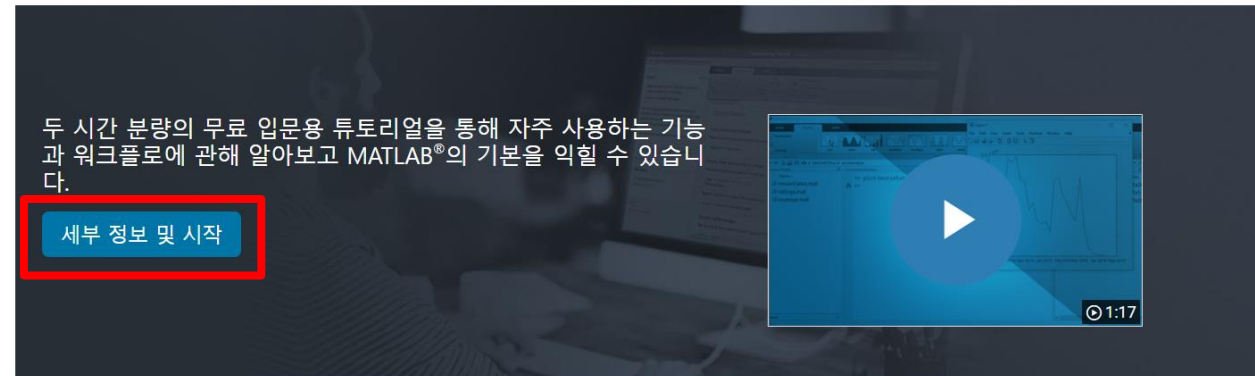
# MATLAB Onramp

## ■ Start Onramp on matlab web browser

▶ <https://kr.mathworks.com/learn/tutorials/matlab-onramp.html>



## MATLAB Onramp

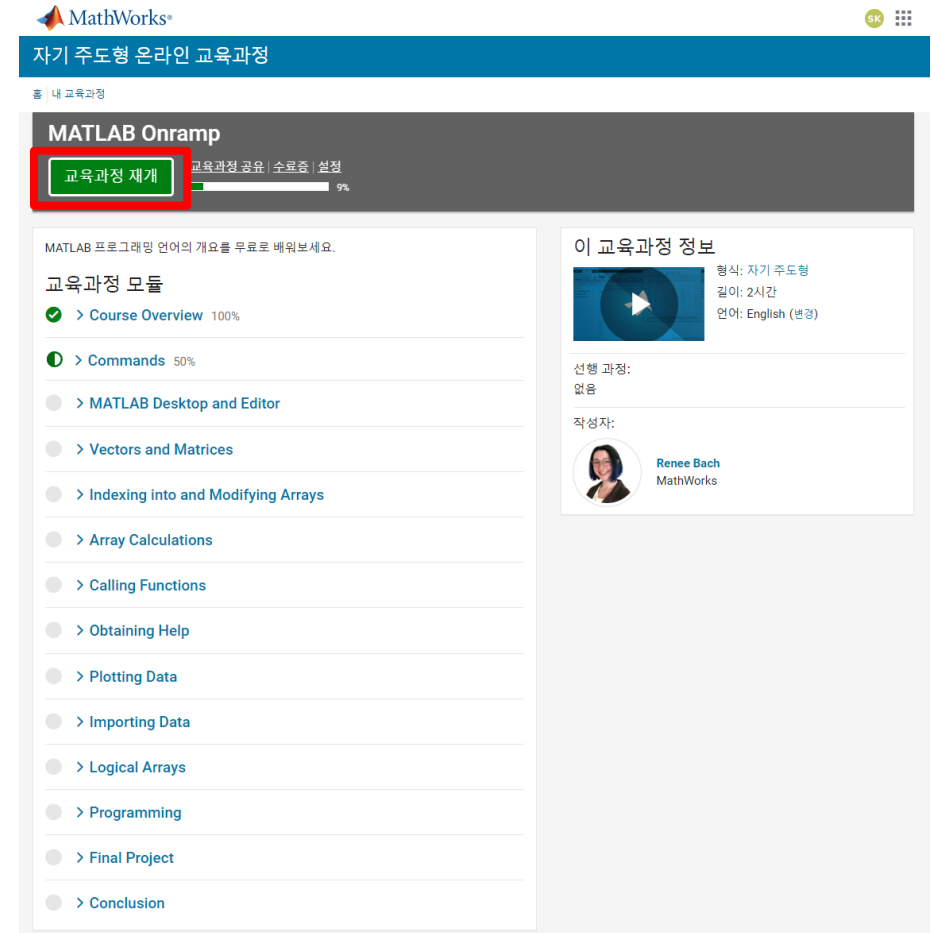


웹 브라우저를 통한 MATLAB 액세스

흥미로운 비디오 튜토리얼

자동 채점 및 피드백이 제공되는 실습 예제

한국어, 영어, 중국어, 스페인어, 일본어로 학습 가능



# MATLAB Onramp

- You can submit the practice by entering the appropriate code in the command line or script for each task.

**작업 1**

MATLAB is designed to work naturally with arrays. For example, you can add a scalar value to all the elements of an array.

```
x = [1 2 3];
y = x + 2
y =
    3    4    5
```

**작업**

Add `1` to each element of `v1` and store the result in a variable named `r`.

힌트 | 정답 보기 | 초기화

**제출하기** 다음 작업

테스트 결과: **정답입니다!**

- ✓ Did you make a variable named `r`?
- ✓ Did you add 1 to each element of `r`?

arrayops.mlx

## Performing Array Operations on Vectors

Instructions are in the task pane to the left. Complete and submit each task one at a time.

This code sets up the interaction.

```
1 load datafile
2 density = data(:,2);
3 v1 = data(:,3)
4 v2 = data(:,4)
```

**Task 1**

5 `r = v1 + 1`

**Task 2**

6

7

`v1 = 7×1`

```
4.0753
6.6678
1.5177
3.6375
4.7243
9.0698
5.3002
```

`v2 = 7×1`

```
0.5000
2.1328
3.6852
8.5389
10.1570
2.8739
4.4508
```

`r = 7×1`

```
5.0753
7.6678
2.5177
4.6375
5.7243
10.0698
6.3002
```



**THANK YOU  
FOR YOUR ATTENTION**