Autonomous Vehicle Simulation (MDAS.ai)

Sridhar Lakshmanan

Department of Electrical & Computer Engineering

University of Michigan - Dearborn

Presentation for Physical Systems Replication Panel –
NDIA Cyber-Enabled Emerging Technologies Symposium

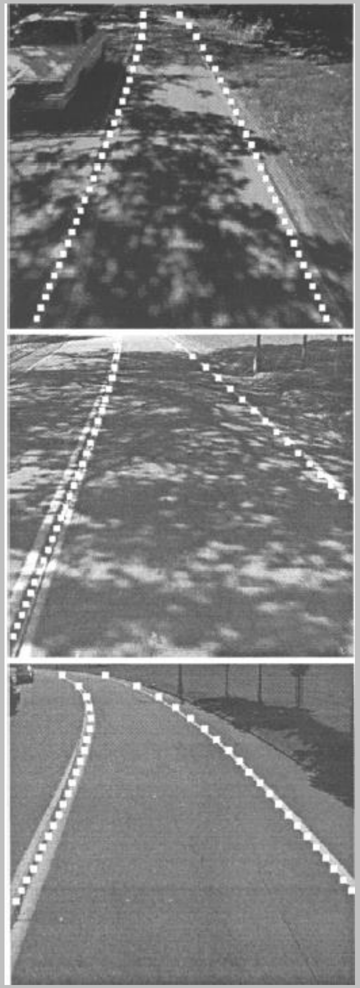
Core Areas of Expertise

	Key words	Faculty Involved
Perception Big Data	Machine learning Bayesian Inference Sensor fusion	Sridhar Lakshmanan Yi Lu Murphey Paul Watta
Intelligent Control	Autonomous vehicles UAV Industrial robots	Stan Baek Yu Zheng Samir Rawashdeh Michael Putty
Vehicle Communications	v2v v2i v2p	Paul Richardson Weidong Xiang Chun-Hung Liu
Standards	SAE On-Road Automated Vehicle Systems (J3016) / Functional Safety (ISO 26262) RVSWG → 20 light and medium trucks standard	Steve Underwood Mark Zachos
Cybersecurity	Fingerprinting ECU's IDS	Hafiz Malik Di Ma
Power Electronics	Solid state convertors Electric drives Charging	Kevin (Hua) Bai Maggie Wang Taehyung Kim Wencong Su
Sensors & Chips	Chip Design / SOC Nano technology Solid state optics	Riadul Islam Alex Yi

AUTONOMY

ELECTRONICS

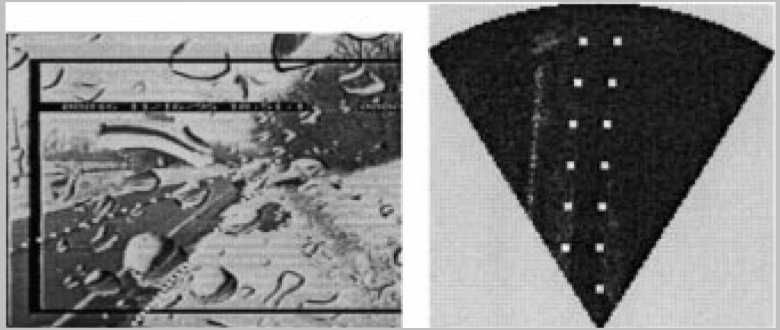
Autonomous Navigation: Army ATD



Sridhar Lakshmanan
Ph.D. Electrical & Computer Eng. (UMass-Amherst)
Associate Professor
University of Michigan–Dearborn
Office: SFC 212
313.593.5516 (O)
734.646.8920 (M)
lakshman@umich.edu
linkedin.com/in/slakshmanan
researchgate.net/profile/Sridhar_Lakshmanan
http://www.MDAS.ai

Miniature Robots: Army SBIR

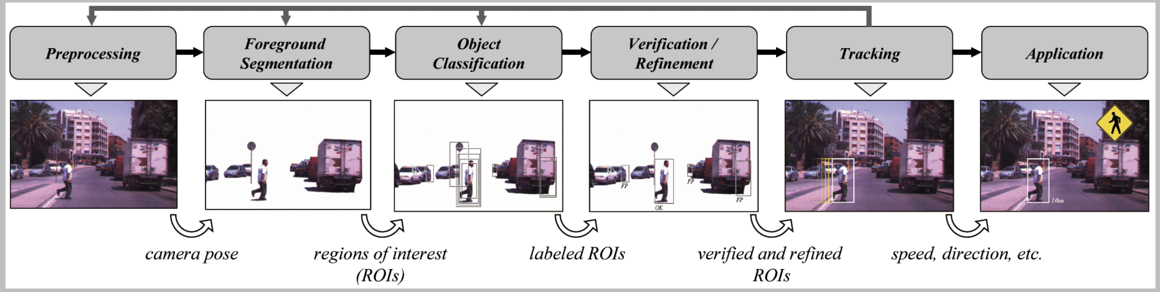
Sensor Fusion: DARPA



Lane Detection: Army

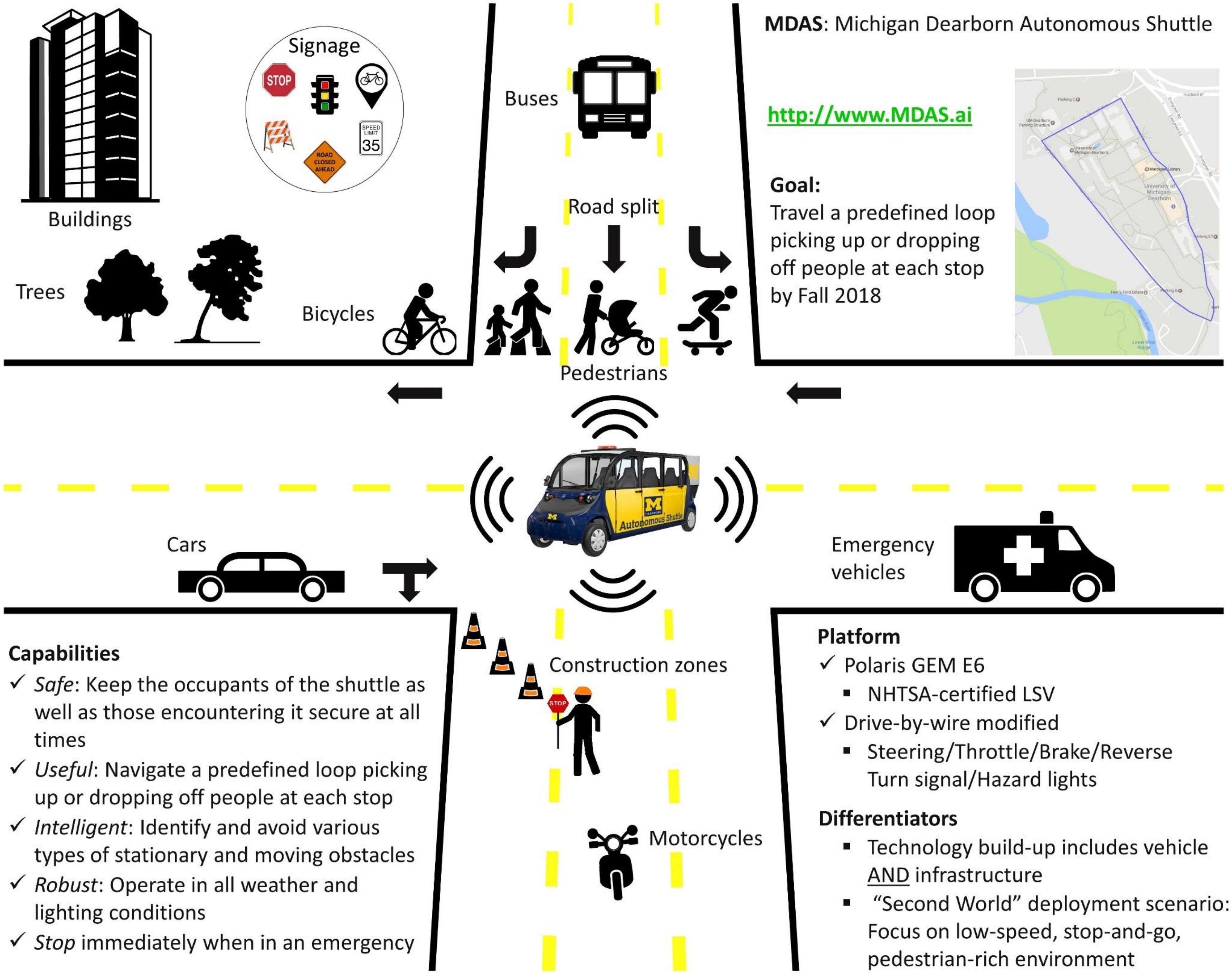
Driver Monitoring: NHTSA

Pedestrian Detection: Ford URP



MDAS: Michigan Dearborn Autonomous Shuttle

http://www.MDAS.ai



Goal:
Travel a predefined loop picking up or dropping off people at each stop by Fall 2018

Capabilities

- ✓ *Safe*: Keep the occupants of the shuttle as well as those encountering it secure at all times
- ✓ *Useful*: Navigate a predefined loop picking up or dropping off people at each stop
- ✓ *Intelligent*: Identify and avoid various types of stationary and moving obstacles
- ✓ *Robust*: Operate in all weather and lighting conditions
- ✓ *Stop* immediately when in an emergency

Platform

- ✓ Polaris GEM E6
  - NHTSA-certified LSV
- ✓ Drive-by-wire modified
  - Steering/Throttle/Brake/Reverse Turn signal/Hazard lights

Differentiators

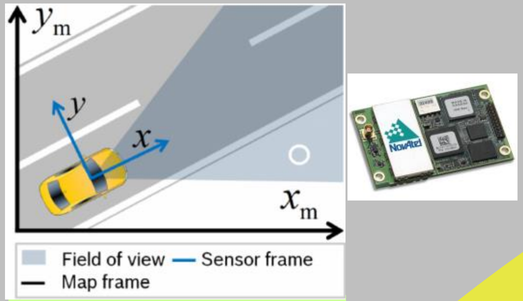
- Technology build-up includes vehicle AND infrastructure
- "Second World" deployment scenario: Focus on low-speed, stop-and-go, pedestrian-rich environment

- ✓ Design, Build & Test
- ✓ Why simulate?
  - *Bring data back*
  - *Requirements*
  - *Failure modes*

DEARBORN

MDAS.ai Timeline & Ecosystem

Deep learning: Nvidia GPU



Localization

- Sub-cm accuracy
- GPS+

Drive-by-wire conversion

- Power-assisted steer
- Linear brake
- Analog throttle

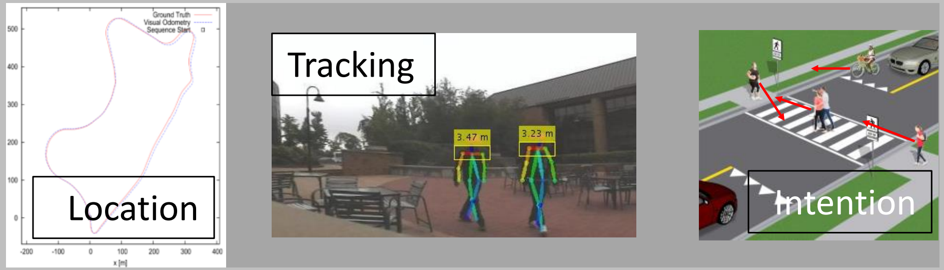
April '18: MI Robotics Day

Aug '18: UMD-MEDC Showcase

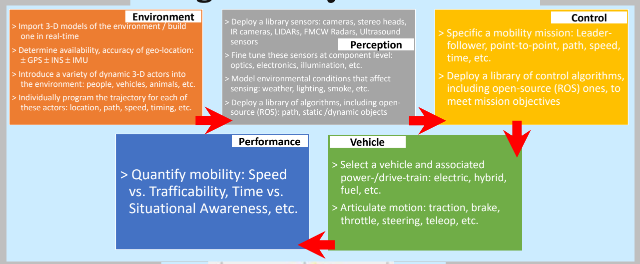
Dec '18: v1.0 Shuttle (Straightaway)

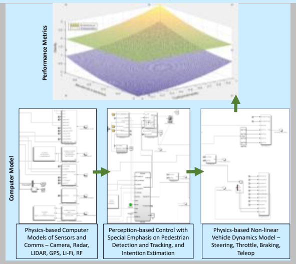
May '19: AutoSens-D

Fall '19: v2.0 Shuttle (Loop)

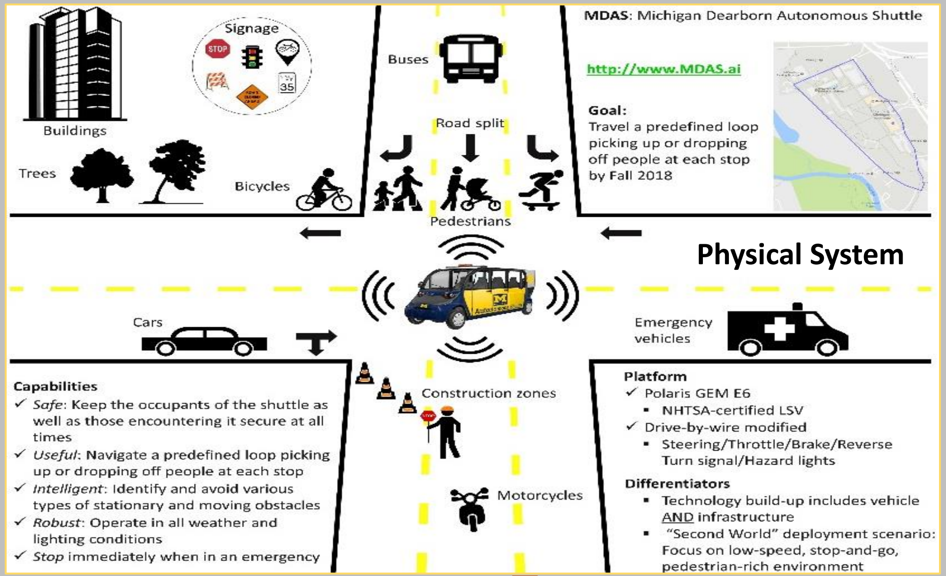


High-fidelity simulation



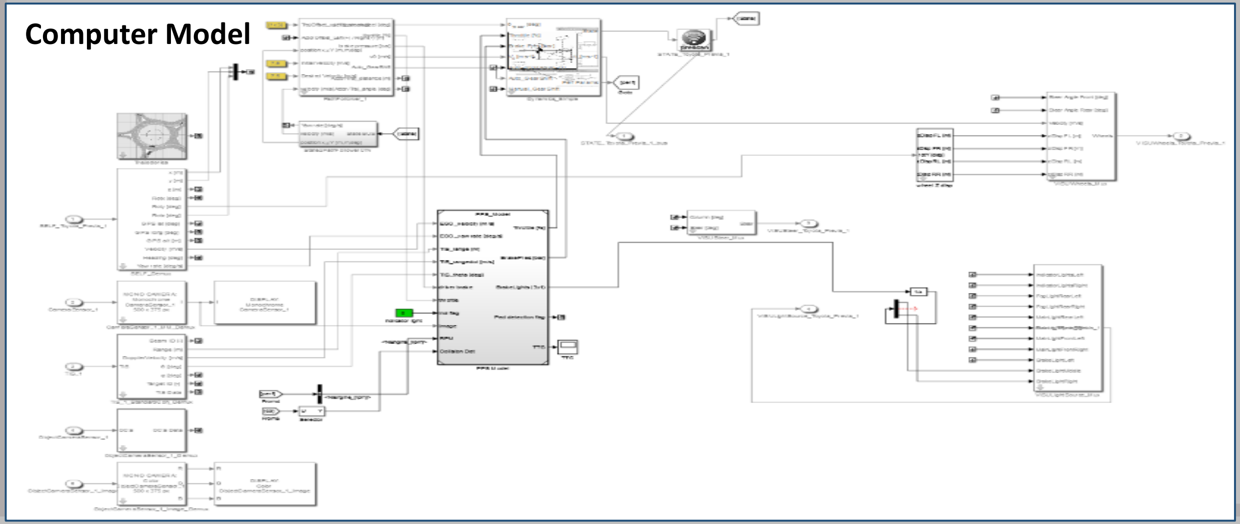


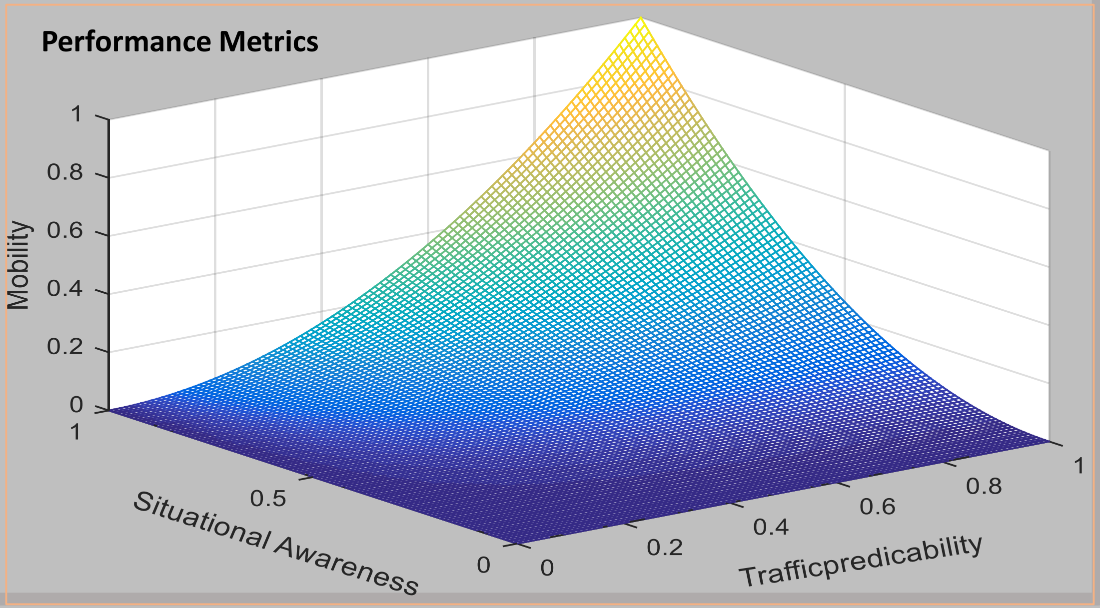
Mobility Model



Capabilities

- ➢ Campus mobility model is Physics-based and not based on empirical data (see next sheet)
- ➢ Special case of the Next-Generation NATO Reference Mobility Model (NG-NRMM)
- ➢ Computer model is validated by real data from the physical shuttle MDAS.ai, and conversely, computer model is used to improve on-road performance of the vehicle
- ➢ Model output is performance metrics such as – Mobility, Traversability, Repeatability, Reliability
- ➢ Model used to:
  - ✓ Assess and compare autonomous systems in campus/urban environments
  - ✓ Compare autonomous systems to baseline human-driven systems
  - ✓ Benchmark progression of autonomous systems from Level-0 to Level-5
  - ✓ Assess performance of Perception Systems and Control Strategies





High-Fidelity Simulation: System of Systems of Systems

Environment

> Import 3-D models of the environment / build one in real-time

> Determine availability, accuracy of geo-location: ± GPS ± INS ± IMU

> Introduce a variety of dynamic 3-D actors into the environment: people, vehicles, animals, etc.

> Individually program the trajectory for each of these actors: location, path, speed, timing, etc.

Perception

> Deploy a library sensors: cameras, stereo heads, IR cameras, LIDARs, FMCW Radars, Ultrasound sensors

> Fine tune these sensors at component level: optics, electronics, illumination, etc.

> Model environmental conditions that affect sensing: weather, lighting, smoke, etc.

> Deploy a library of algorithms, including open-source (ROS): path, static /dynamic objects

Control

> Specific a mobility mission: Leader-follower, point-to-point, path, speed, time, etc.

> Deploy a library of control algorithms, including open-source (ROS) ones, to meet mission objectives

Vehicle

> Select a vehicle and associated power-/drive-train: electric, hybrid, fuel, etc.

> Articulate motion: traction, brake, throttle, steering, teleop, etc.

Performance

> Quantify mobility: Speed vs. Trafficability, Time vs. Situational Awareness, etc.