4D/RCS Reference Model Architecture for Unmanned Vehicle Systems

James S. Albus
Senior NIST Fellow
Anthony J. Barbera
Senior Scientist
National Institute of Standards and Technology

- **Hierarchical structure of goals and commands**
- **Representation of the world at many levels**
- **Planning, replanning, and reacting at many levels**
- **Integration of many sensors stereo CCD & FLIR, LADAR, radar, inertial, acoustic, GPS, internal**

What is RCS?

A reference model architecture

for Real-time Control System design

RCS has been used to design control systems for many intelligent systems including robots , machine tools, automation systems, and unmanned vehicles

What is 4D/RCS?

The latest version of RCS

A reference model architecture

designed to enable any desired level of intelligence

up to and including human level intelligence

What does 4D/RCS specify?

- Functions, entities, events, relationships
- Interaction and information flow between systems and subsystems
- Structures for representation of knowledge, goals, plans, tasks, schedules, intentions, beliefs, values
- Mechanisms for perception, attention, cognition
- Mechanisms for reasoning, modeling, and learning

Attributes of RCS

- Combines AI with control theory
- Hierarchical representation of tasks, space, & time
- Combines deliberative with reactive at many levels
- Depends strongly on sensing and perception
- Supports a rich dynamic world model at many levels
- Integrates prior knowledge with current observations
- Models functional architecture of the human brain
- Addresses the full range of human behavior
- Is mature with engineering tools and software libraries

Contrast 4D/RCS and SOAR

4D/RCS	SOAR
Bottom up	Top down
Control system	AI system
Inspired by modeling of cerebellum function	Inspired by modeling of human cognition
Knowledge: Iconic and symbolic	Knowledge: Symbolic
Hardware applications: robot manipulators machine tools automation systems unmanned ground vehicles	Simulation applications: semi-automated forces pilot's associate unmanned air vehicles battle management

Compare 4D/RCS and SOAR

4D/RCS	SOAR
Attempt to model human intelligence	Attempt to model human intelligence
Hierarchical task decomposition	Hierarchical task decomposition
Represent behavior as finite state machines	Represent behavior as finite state machines
Long history of successful applications	Long history of successful applications
Potential application to FCS	Potential application to FCS

Strong FCS interest in tactical behaviors

RCS Versions

1979 – RCS-1 Laboratory robot control

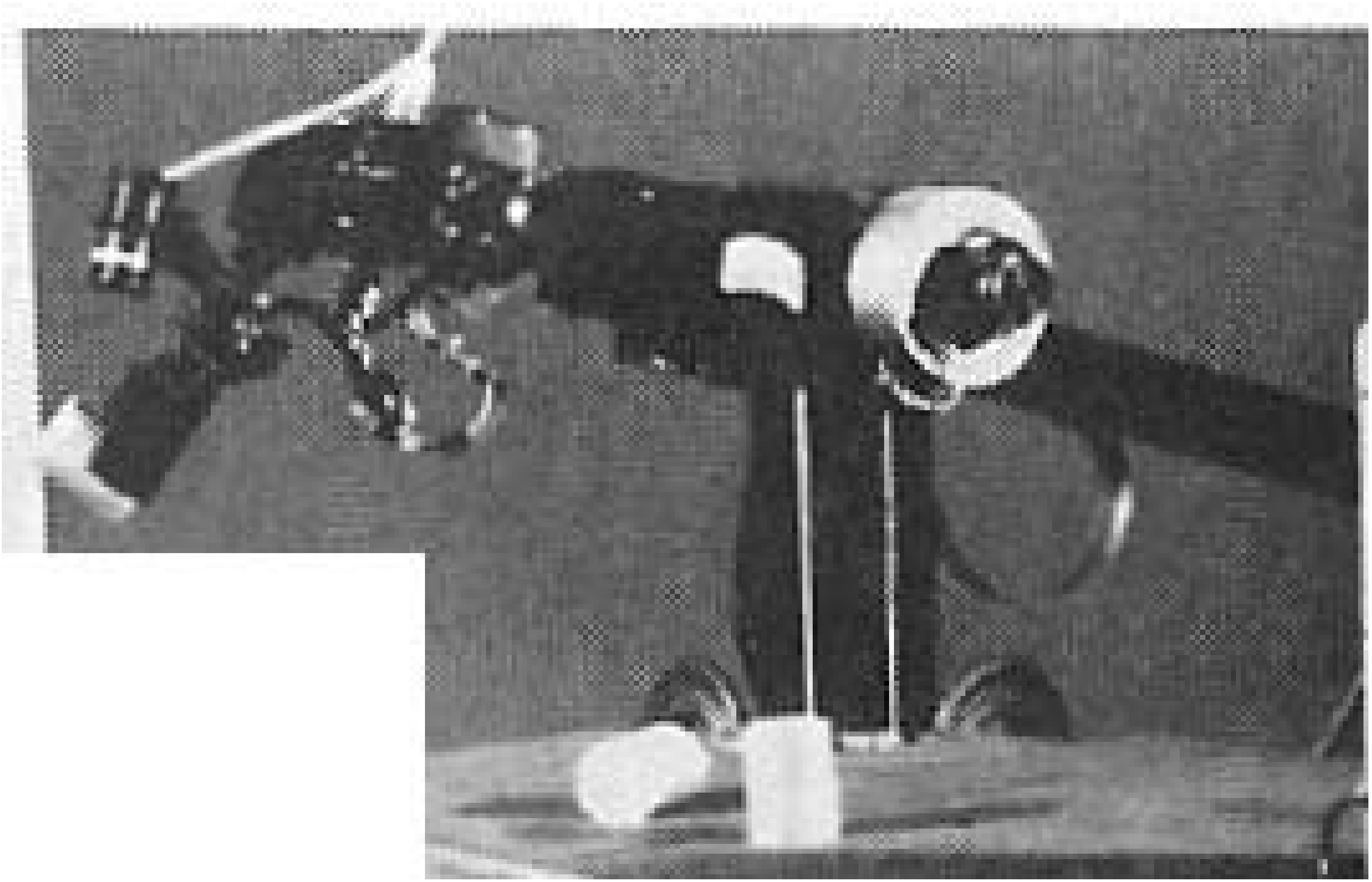
1981 – RCS-2 Automated Manufacturing

1987 – RCS-3 NASREM Space Telerobotics

1988 – RCS-4 DARPA Multiple AUVs

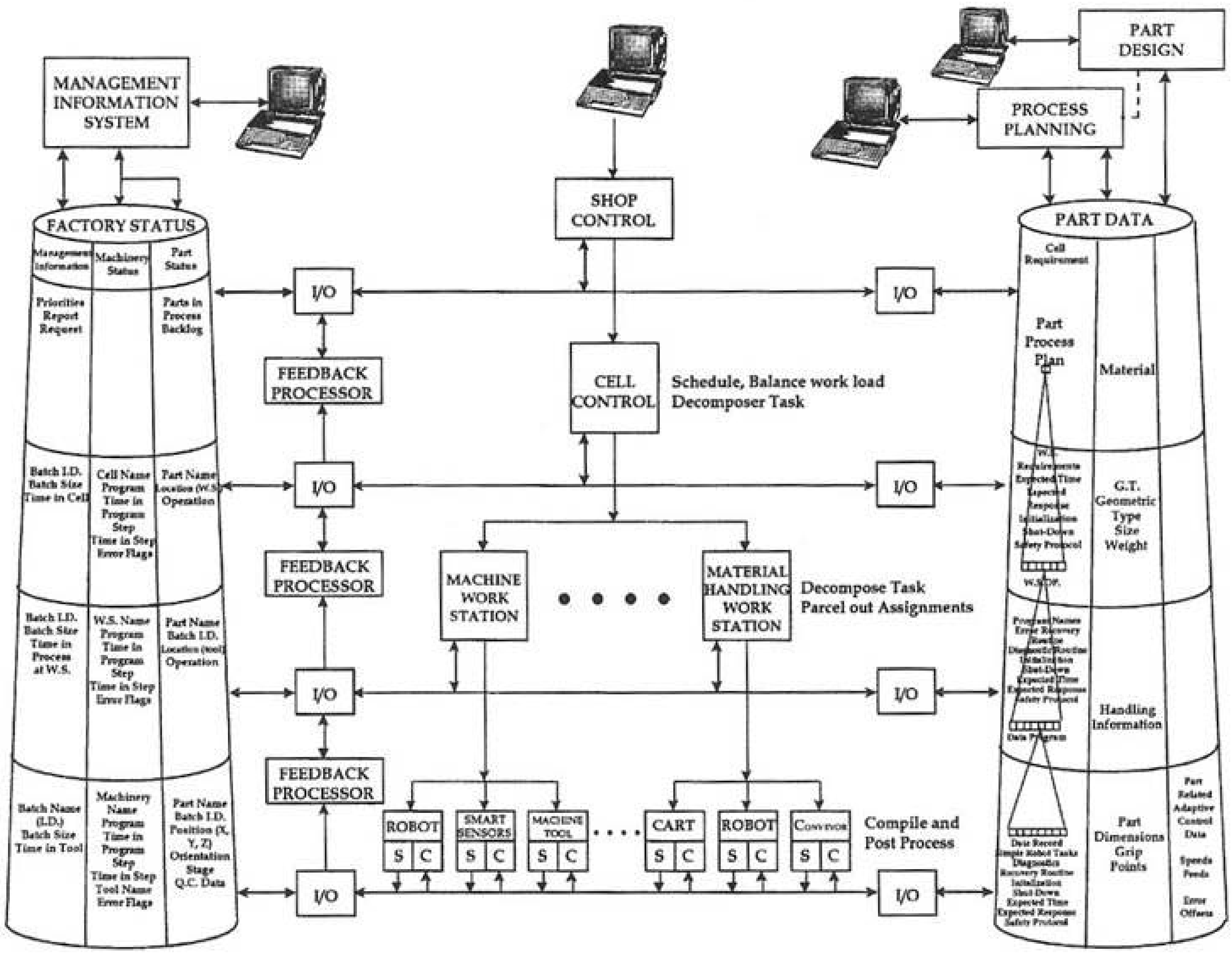
1998 – 4D/RCS Demo III Multiple UGVs

RCS-1 Robot Control with Camera and Line-of-Light Flash



1979

RCS-2 Automated Manufacturing Research Facility



1981

RCS-2 Machining Workstation

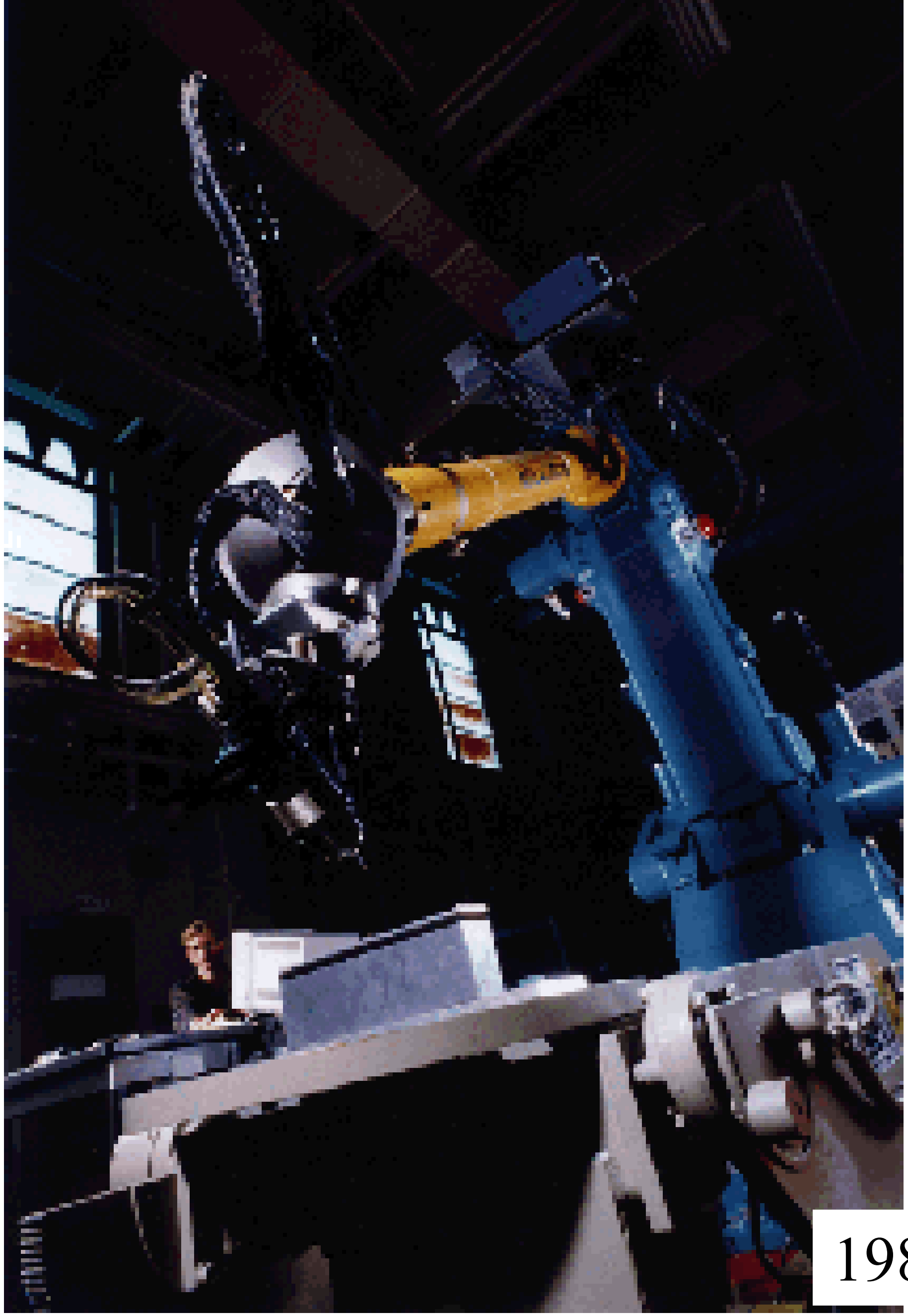
1981

RCS-2 Cleaning and Deburring Workstation

1982

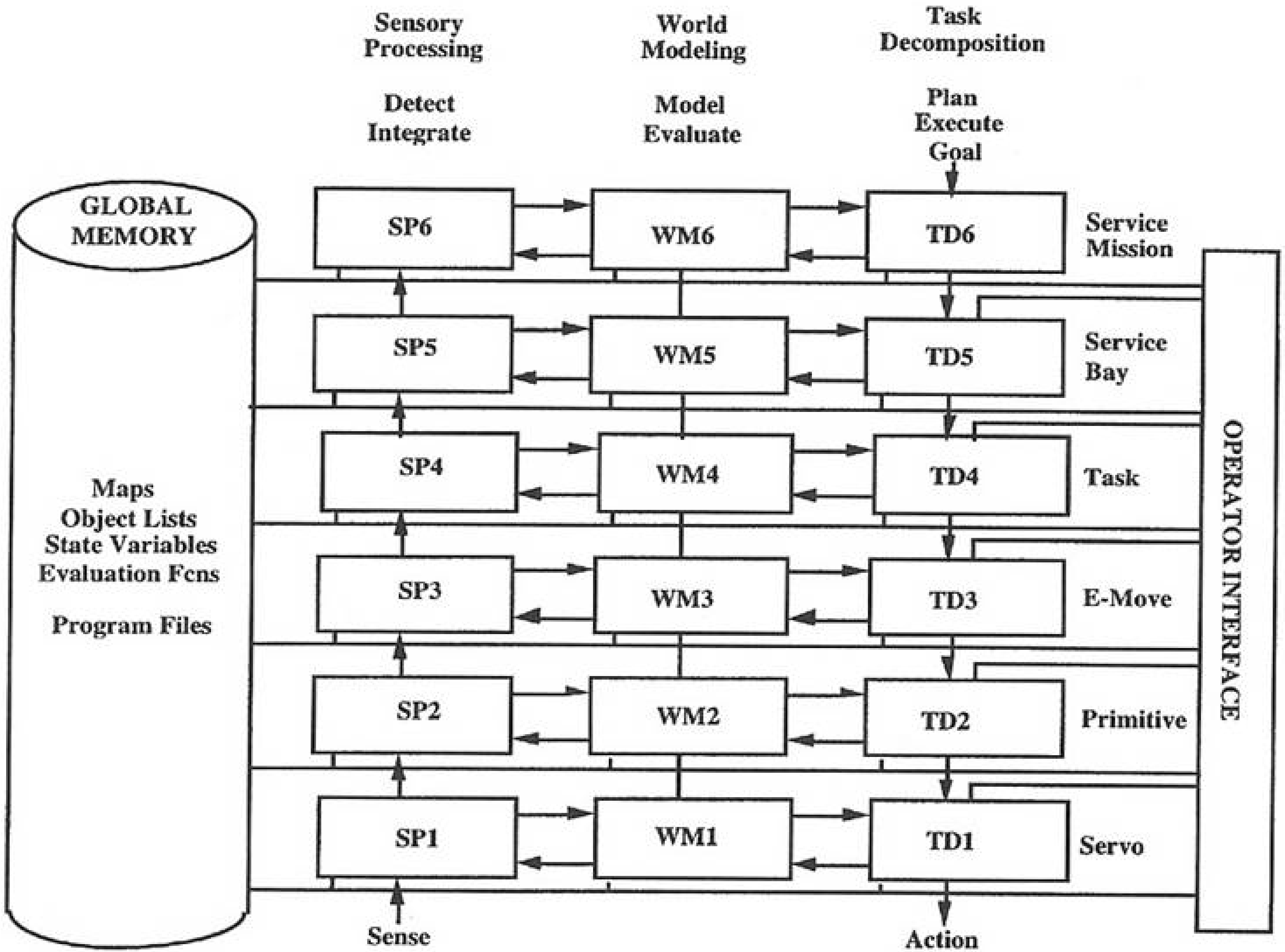
RCS-2 Automated Deburring & Chamfering System

RCS-2 Force Controlled Deburring & Chamfering Tool



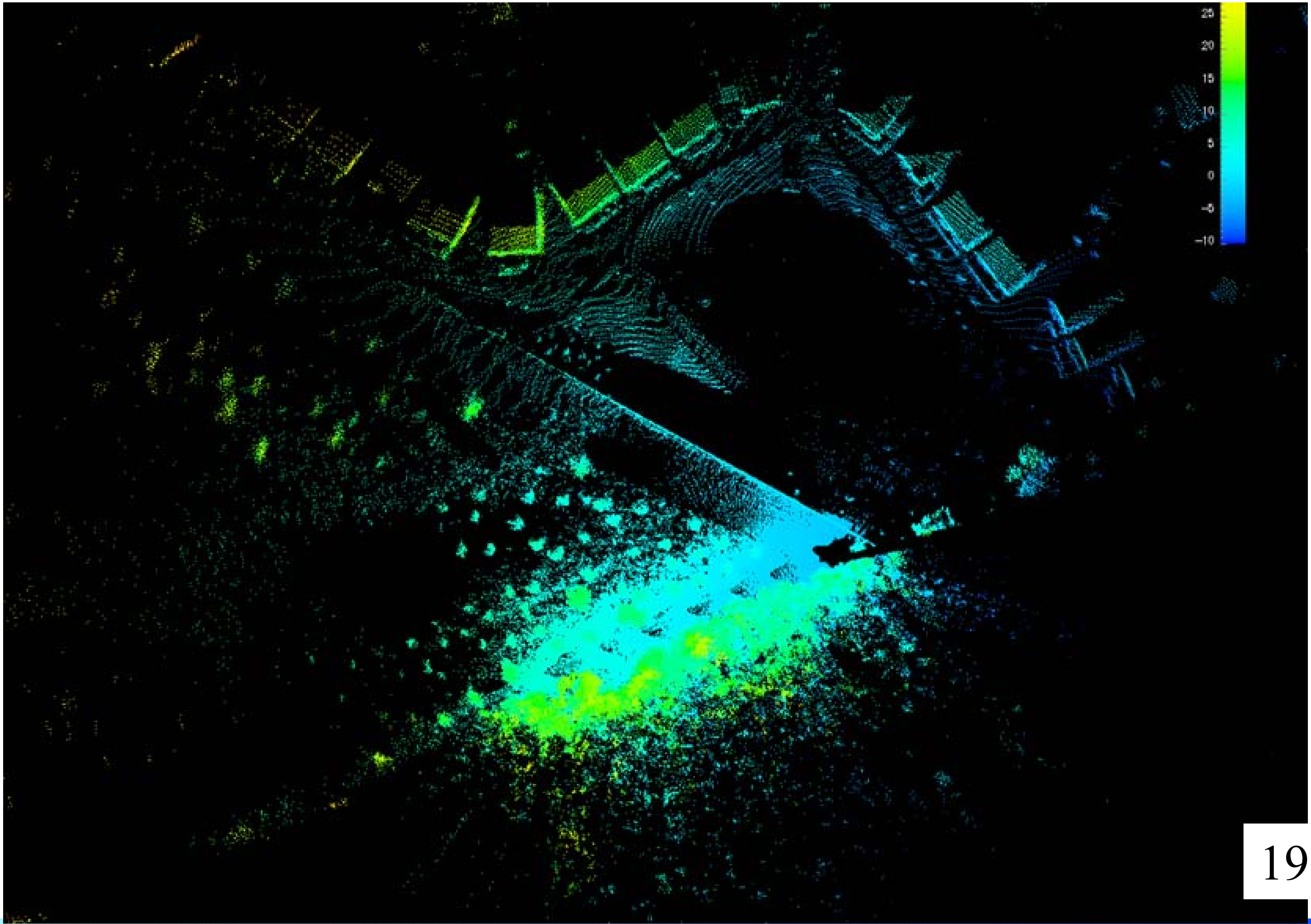
1986

RCS-3 NASREM
NBS/NASA Reference Model Architecture



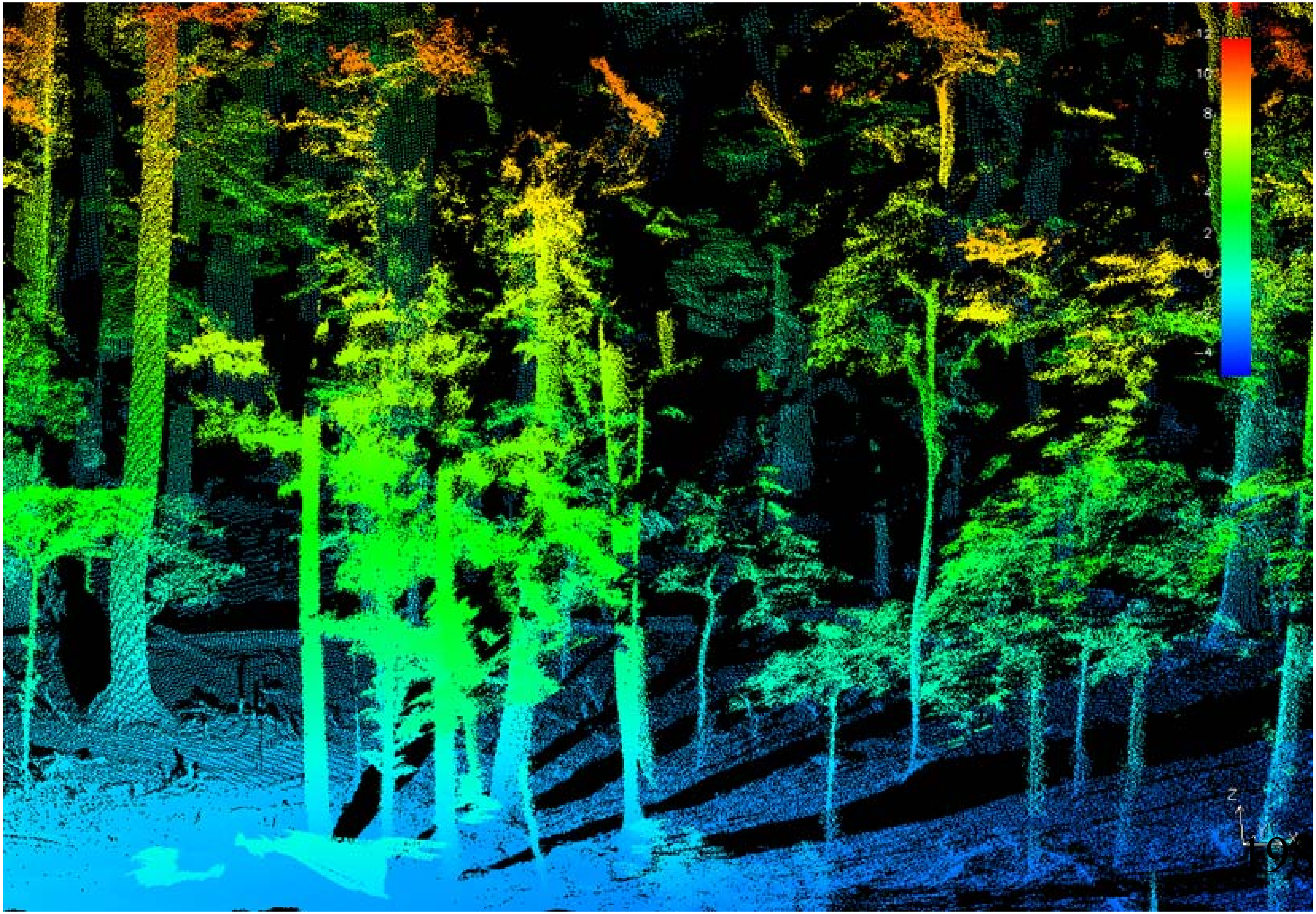
1987

Space Station Telerobotic Servicer



1987

RCS-3 Automated Coal Mining Machine



1998

RCS-4 Multiple Autonomous Undersea Vehicles

1989

RCS-4 Autonomous Vehicle Control

1993

4D/RCS Demo III
Experimental Unmanned Vehicle

1998

Integrated Combat Demo
Ft. Bliss
February 2003

RCS Application Summary

NBS/NIST – Robot control, Automated Manufacturing Research Facility
DARPA -- Multiple Unmanned Undersea Vehicles (MAUV)
DARPA -- Submarine Operational Automation System (SOAS)
GD Electric Boat -- Next generation nuclear submarine control
NASA - Space Station Flight Telerobotic Servicer (NASREM)
Bureau of Mines - Coal mine automation
U.S. Postal Service -- Stamp distribution center, General mail facility
Army - TEAM, TMAP, MDARS, Picatinny Interior UGV, Demo I and III
ARL Collaborative Technology Alliance, JAUGS, VTA
Navy – Double Hull Robot, Multiple UAV SWARM
General Motors – CNC & Inspection Control
Boeing – Cell Control, Riveting, Hi Speed machine tool
Commercial CNC - plasma & water jet cutting
DARPA – MARS, PerceptOR
FCS / Boeing - Autonomous Navigation System, Integrated Combat Demo
DOT -- Intelligent vehicle

4D/RCS Software on Demo III

- NML – Neutral Messaging Language that provides the basic communication services for the Demo III software

- LADAR image processing, terrain analysis, obstacle detection and avoidance, object classification

- World model map, real-time map generation and maintenance, object representation, iconic-symbolic pointers

- Path planning software – Real time cost/benefit optimization based on traversibility, risk, and mode (aggressive vs. stealthy)

4D/RCS Software in the Pipeline

•Vehicle level control

•Tactical behavior generation, value based planning

•Multi-vehicle tactical planning and behavior

•Next generation LADAR image processing, attention based fixation and tracking of important objects,

•Next generation world modeling, high resolution terrain maps,
•moving object representation, complex relationships

•Integration into real/virtual world of OneSAF

•Software development tools, simulation, and testing at all levels – from physics-based to force-on-force

What is the level of maturity?

- A free public domain software library
- A variety of software development tools
- A variety of process visualization tools for analysis, debugging, control, and human interface design
- Documentation and training materials

NIST provided an initial version of most of the Perception and Autonomous Mobility Software on the Demo III Unmanned Ground Vehicles

4D/RCS Documentation

Version 0.1 Issued with Demo III RFP, 1997
Version 1.0 January 1999
Version 2.0 August 2002

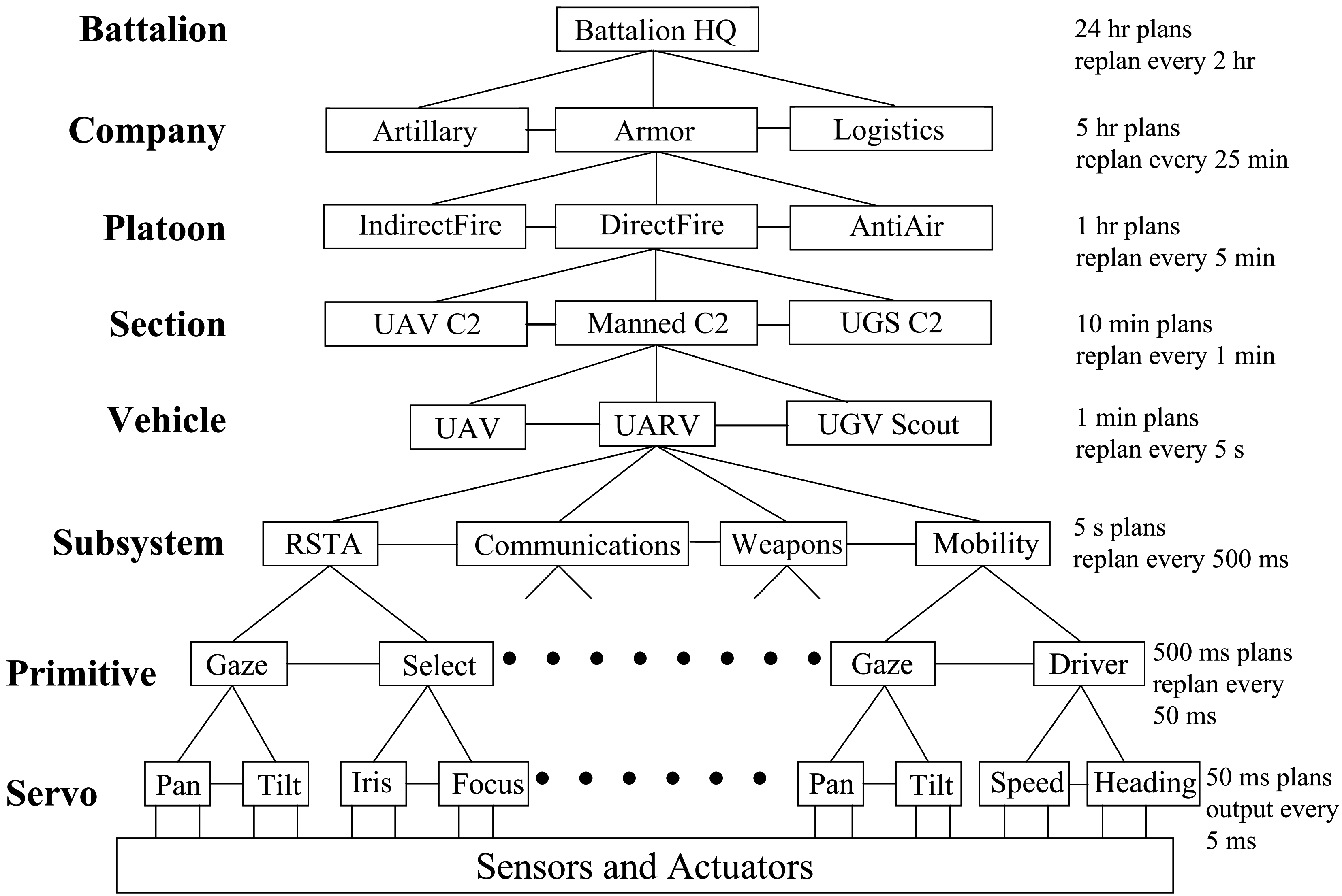
Books:

Engineering of Mind - Wiley, 2001
RCS Handbook – Wiley, 2001
Intelligent Systems – Wiley, 2002

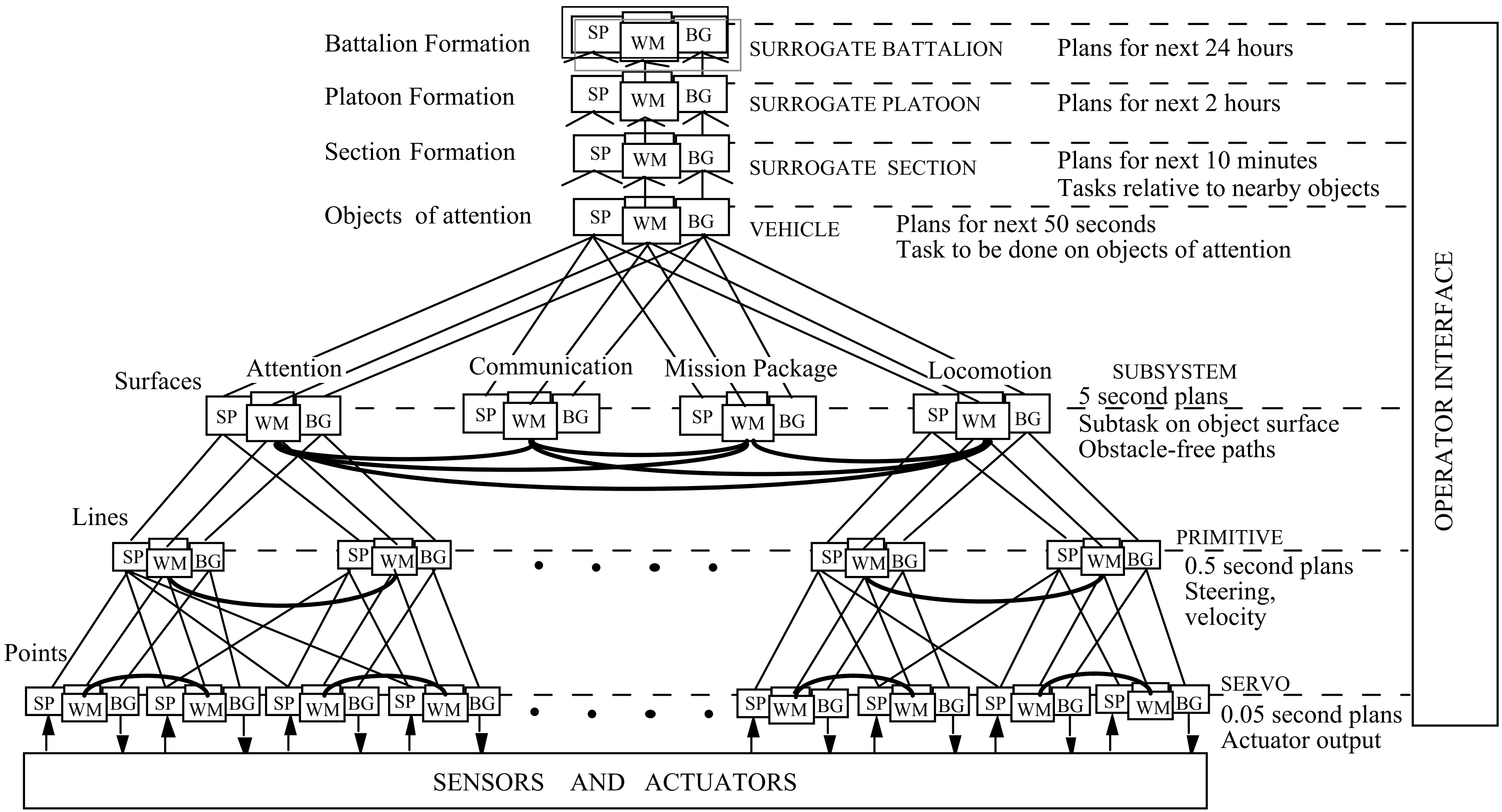
Numerous journal articles, reports, and conference papers

Extensive software library   http://www.isd.mel.nist.gov/projects/rcslib

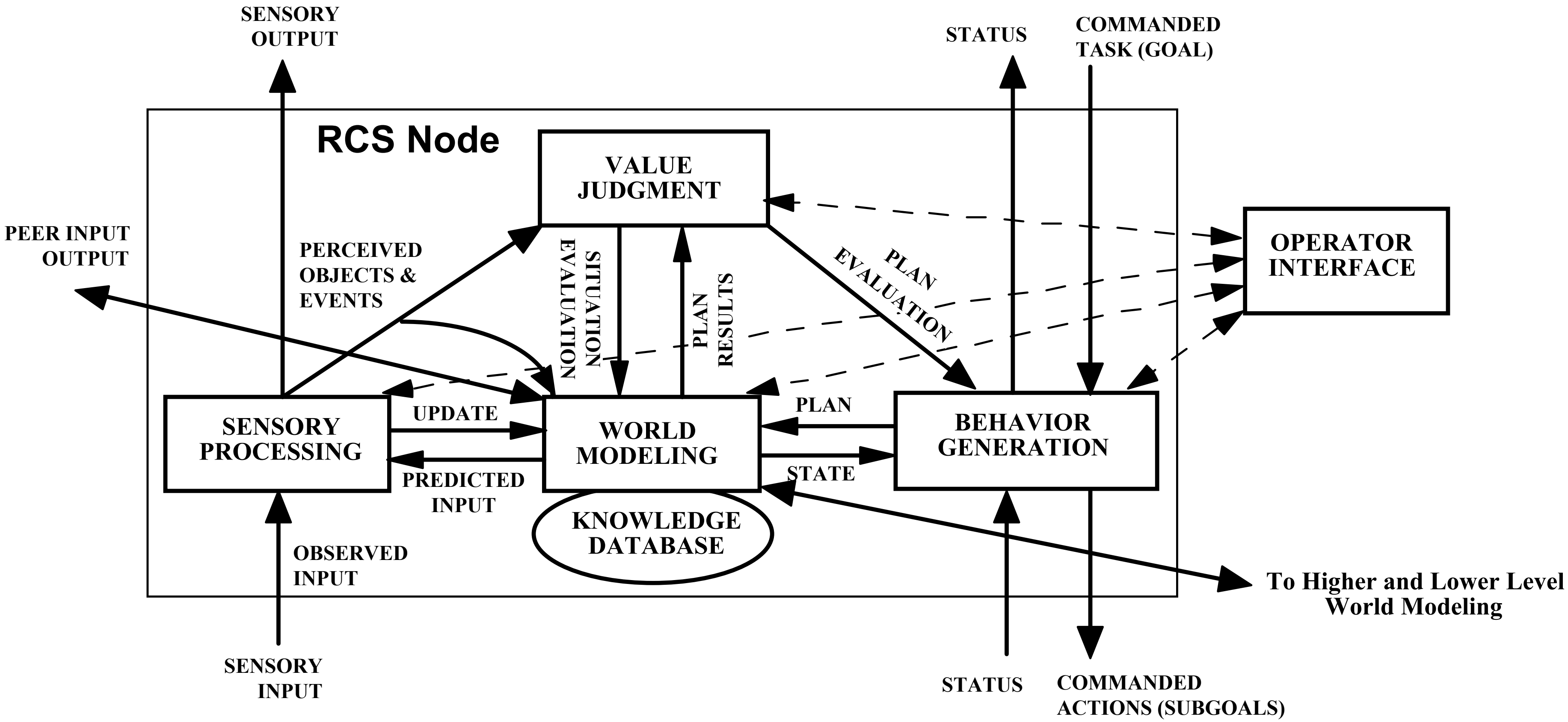
4D/RCS for Future Combat System



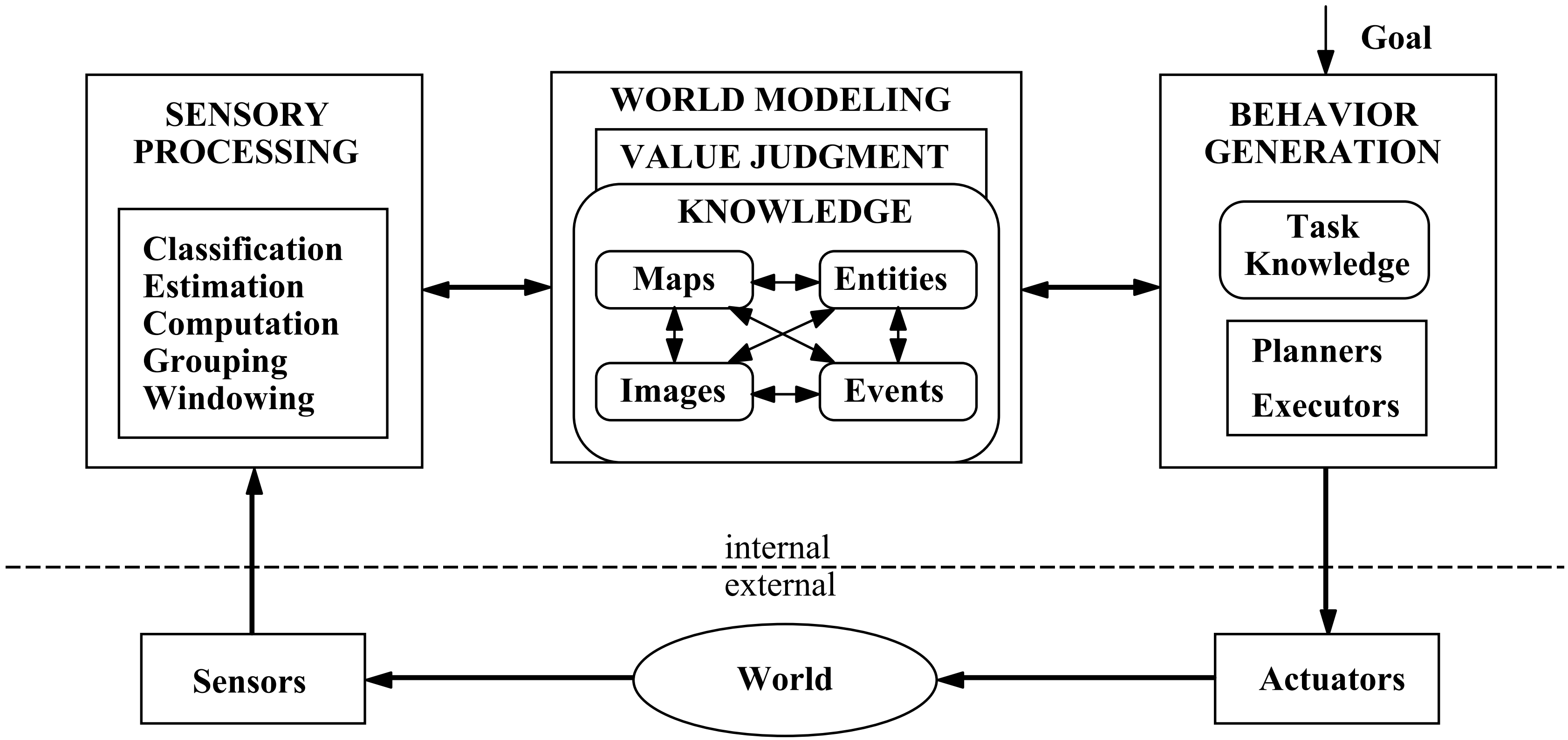
4D/RCS Reference Architecture



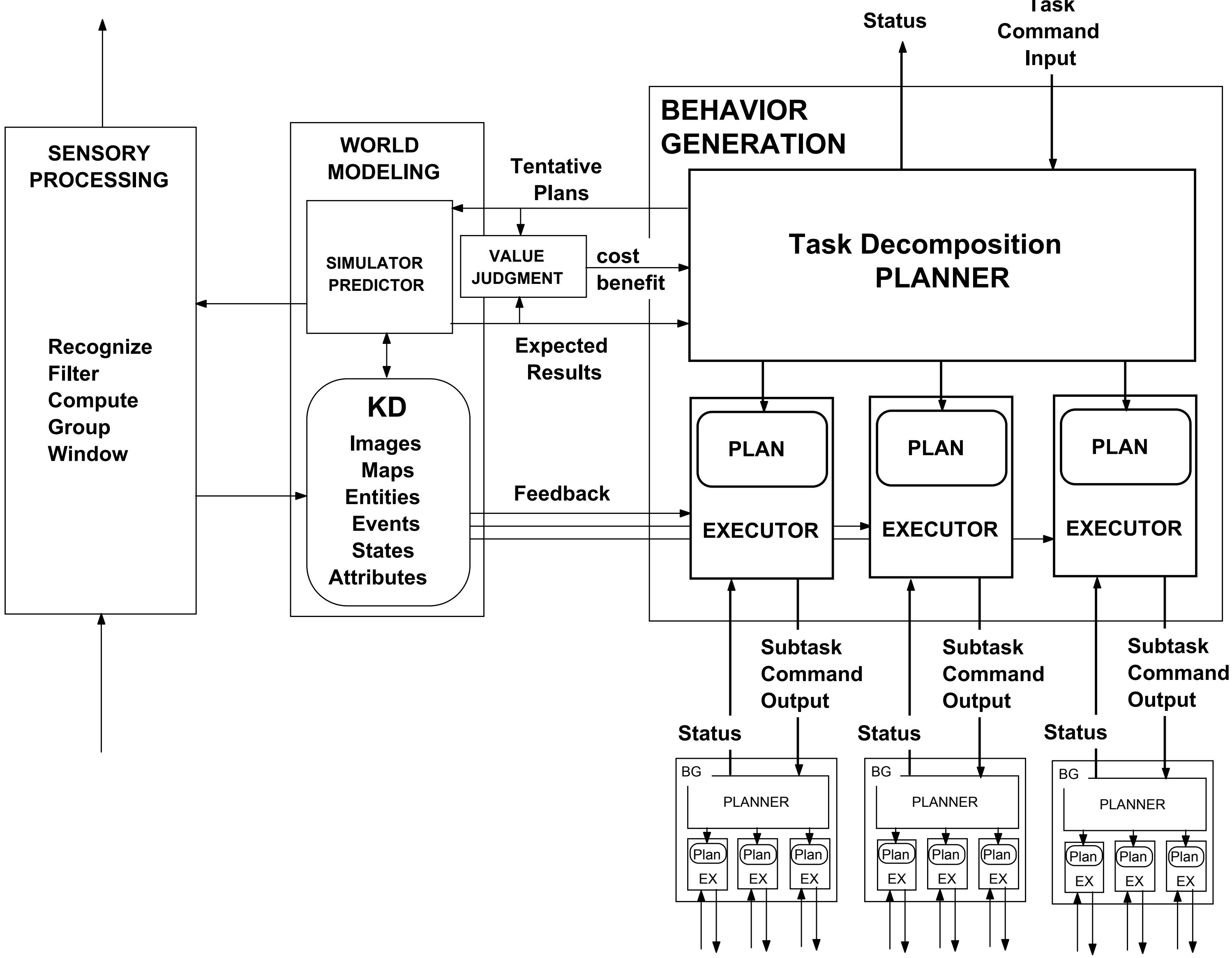
4D/RCS Computational Node



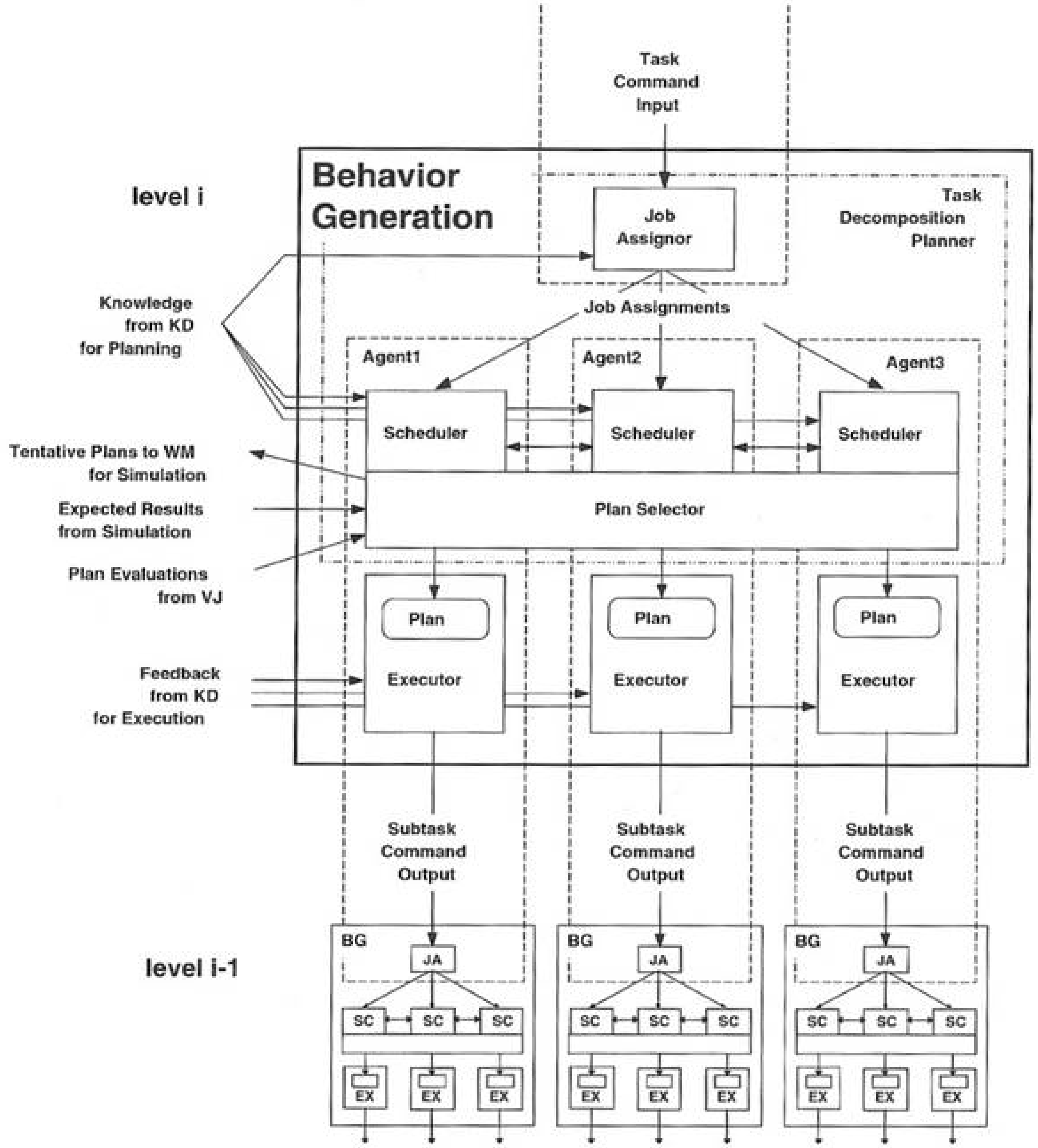
4D/RCS Computational Node



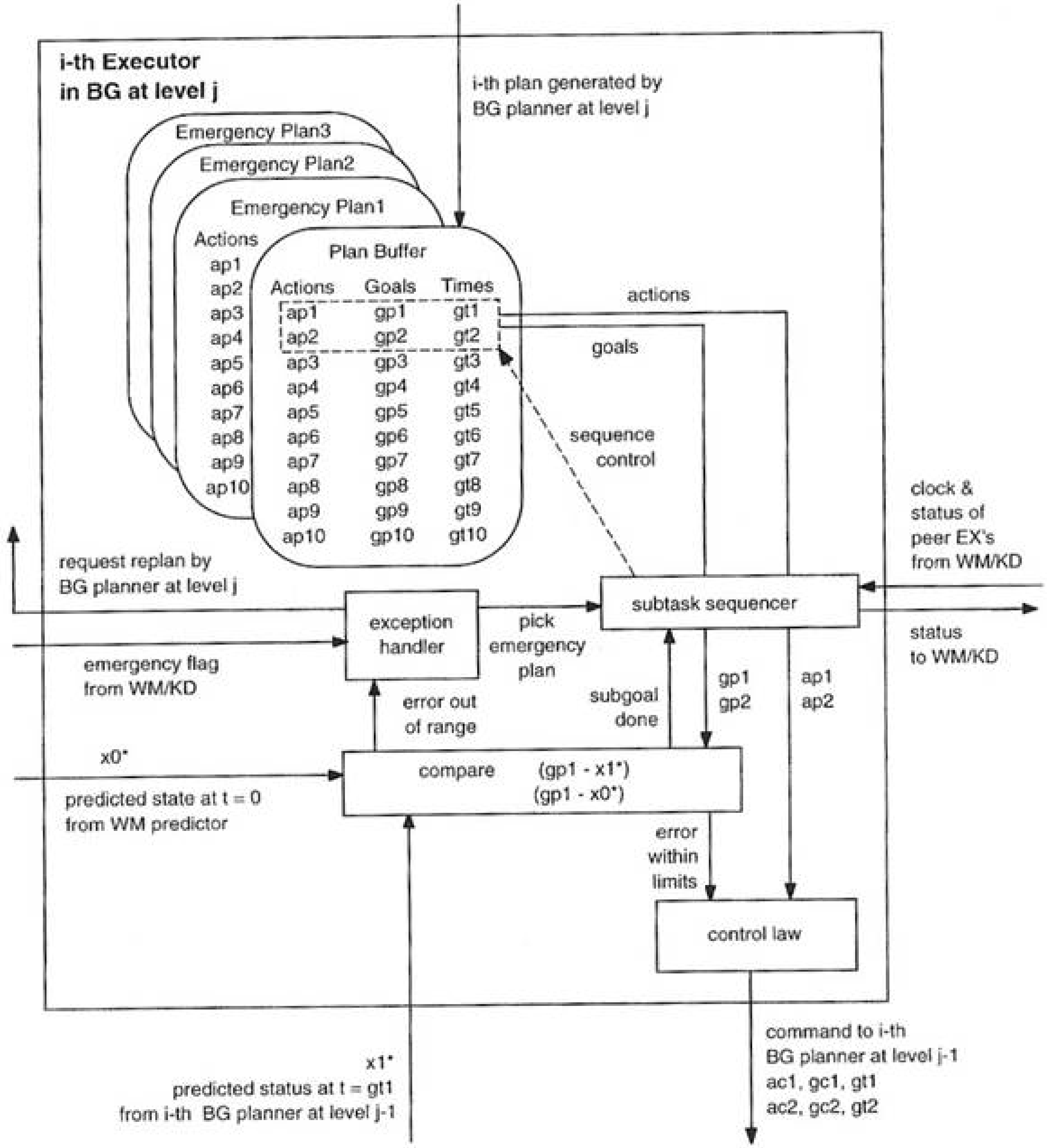
4D/RCS Computational Node



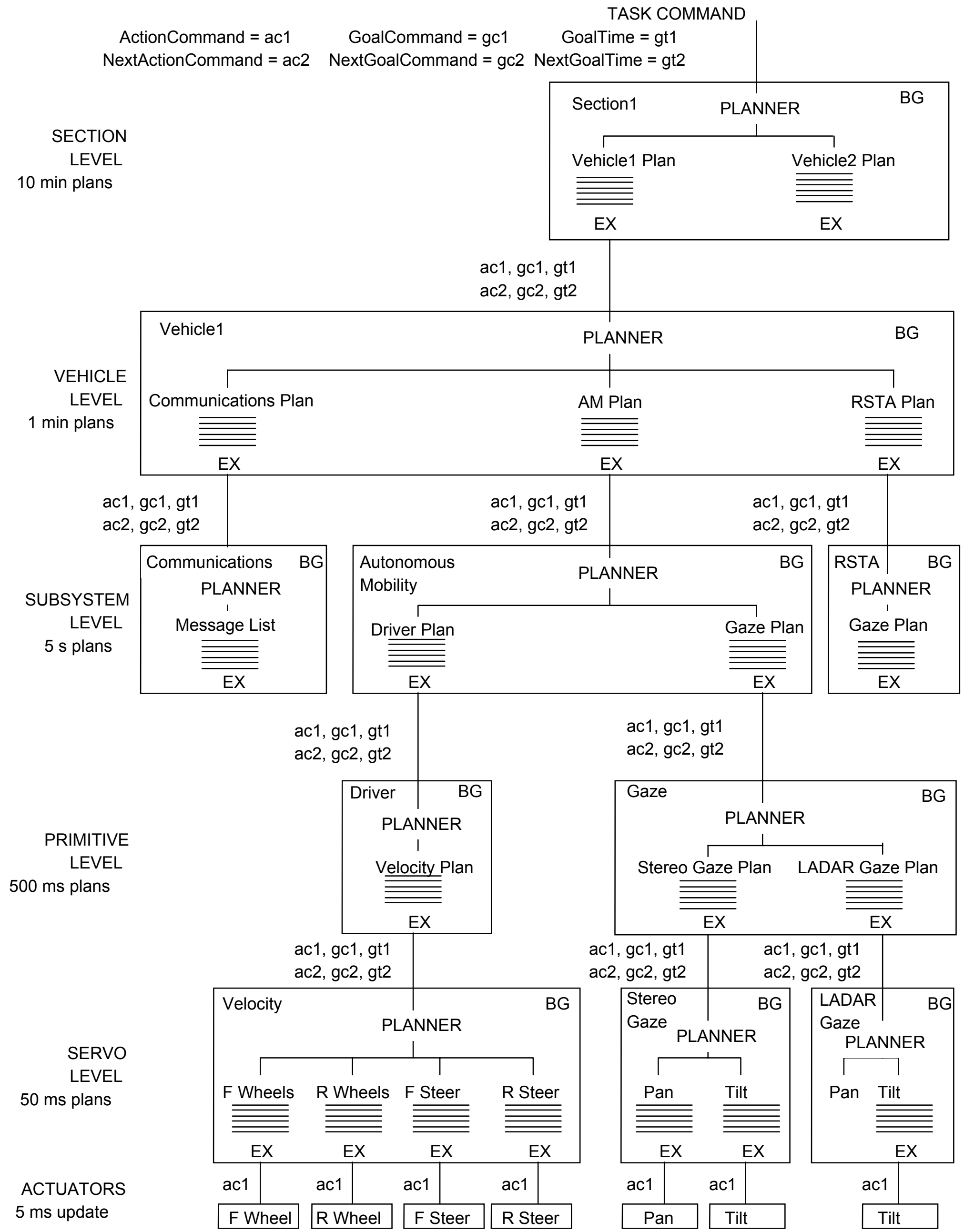
Behavior Generation Process



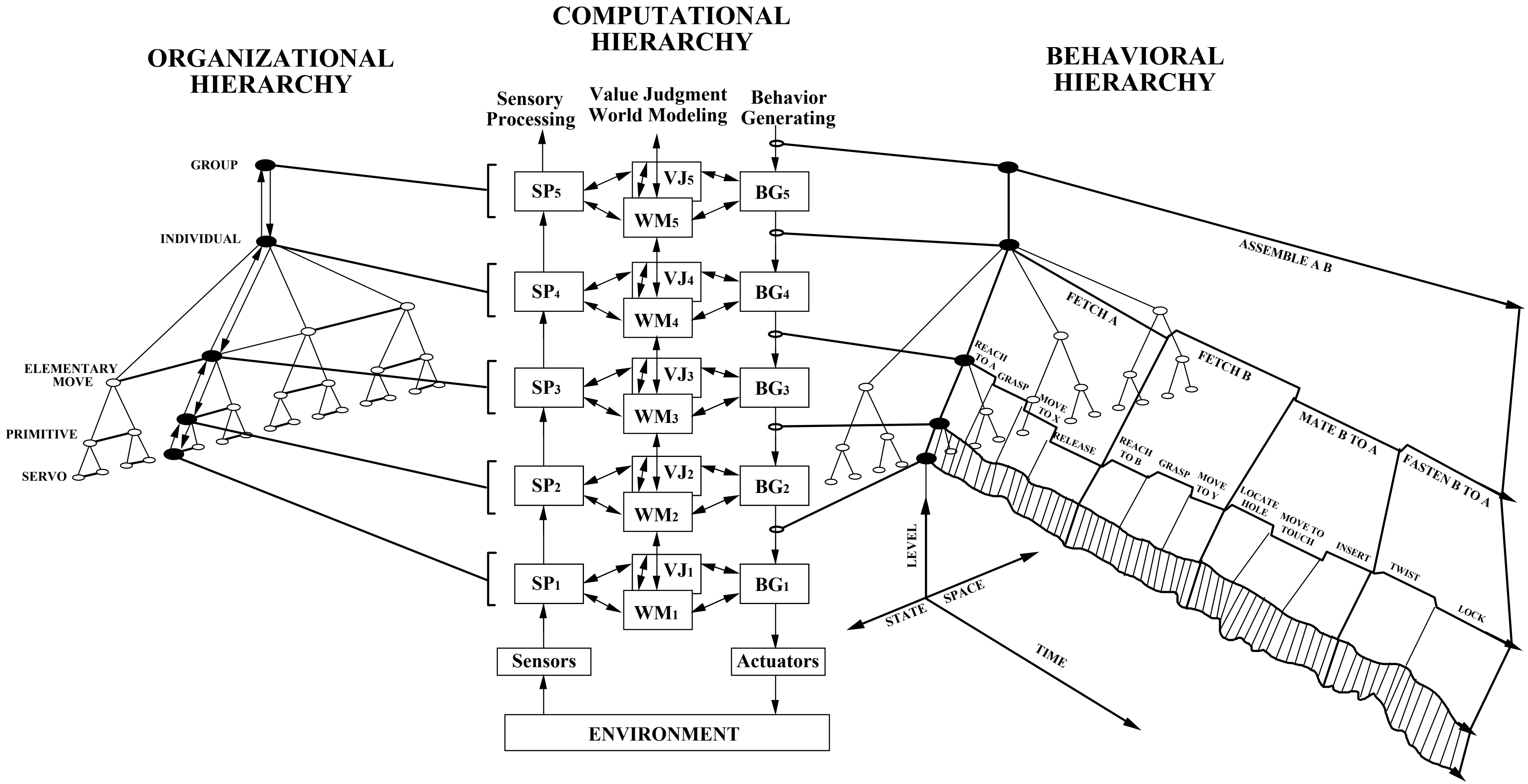
Internal Structure of an Executor



BG Hierarchy for Demo III

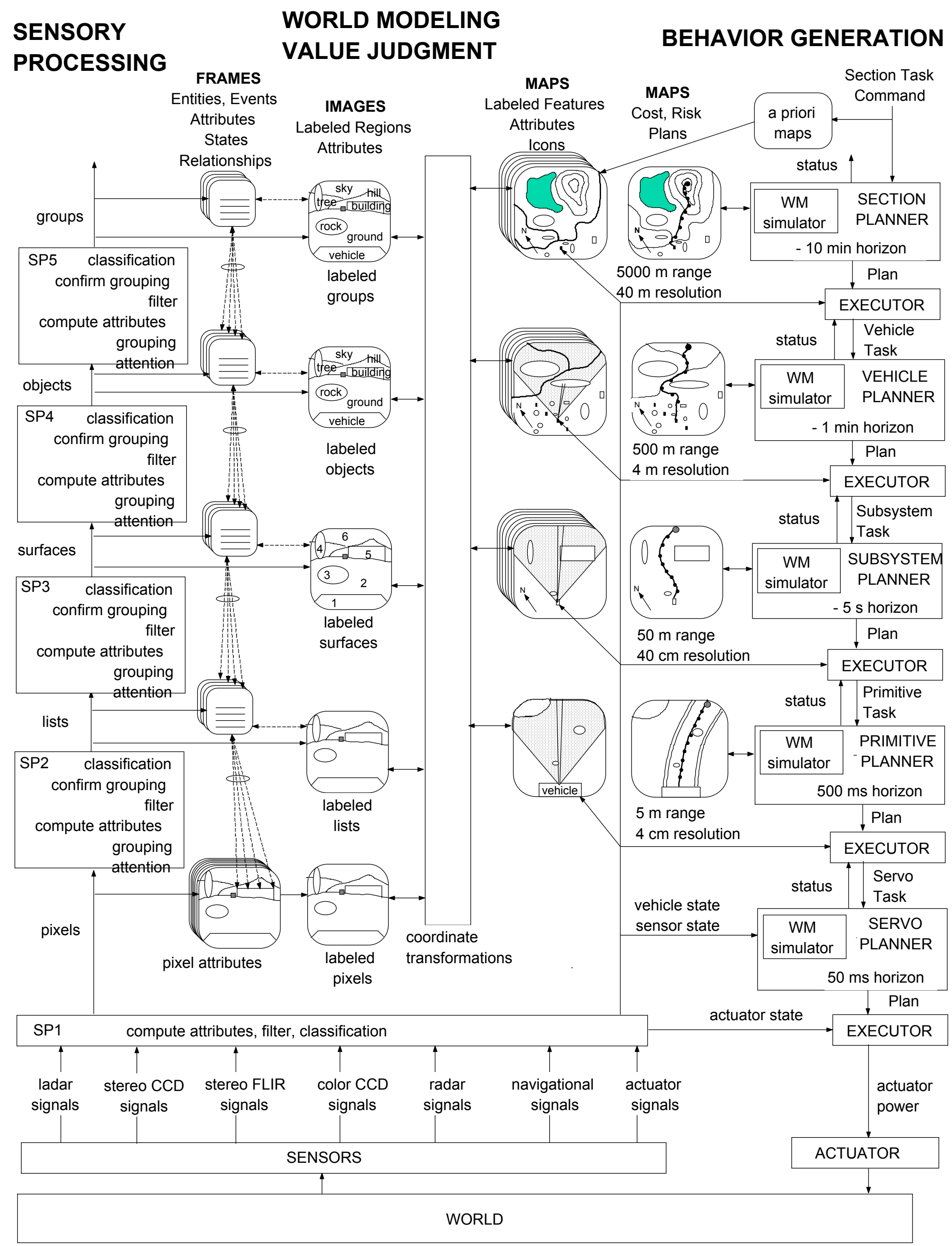


Three Aspects of 4D/RCS



4D/RCS for Demo III

Computational Hierarchy



RCS Timing

4D/RCS Reference Model

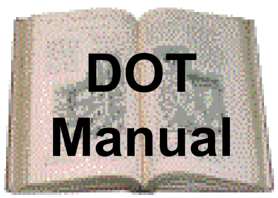
⇩

System Engineering Guidelines

- Software development methodology
- Software library and development tools
- Hardware design and testing experience
- Test and evaluation methods and procedures
- Integration and testing methodology
- Field experiments and operational testing results

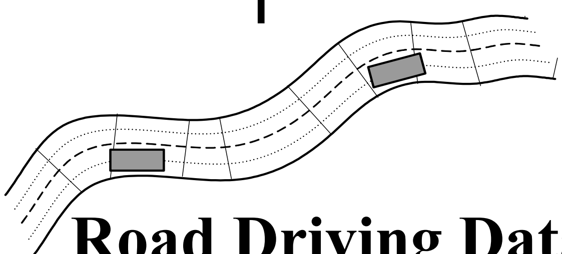
Driving Task Analysis

1) Analyze Autonomous Driving Tasks & Develop a Task Architecture



Task Analysis

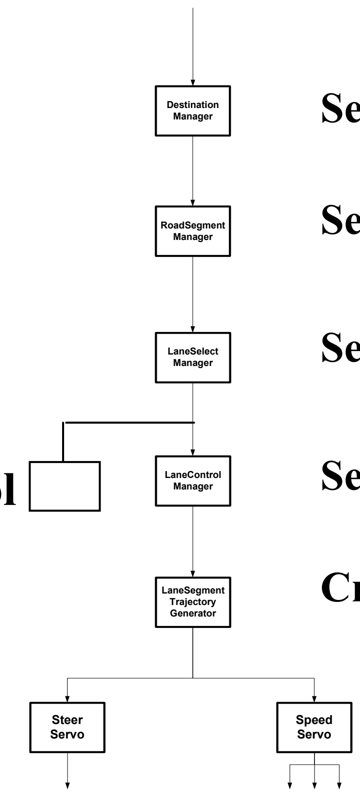
+



Road Driving Data Representations

=>

Gaze Control



Selects Destination 2 hour plan

Selects Road Segments 10 min plans

Selects Lane 1 min plans

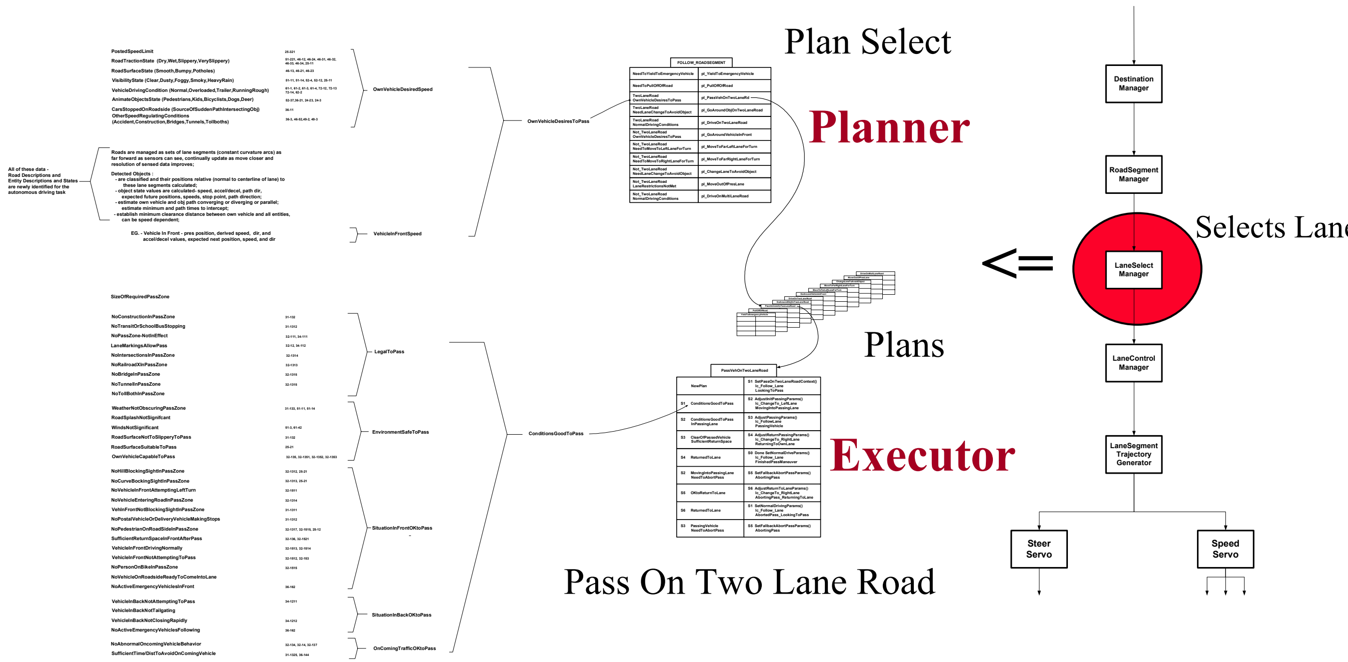
Selects Pos & Speed In Lane 10 sec plans

Creates Trajectory 1 sec plans

Servos Steer & Speed 100 ms plans

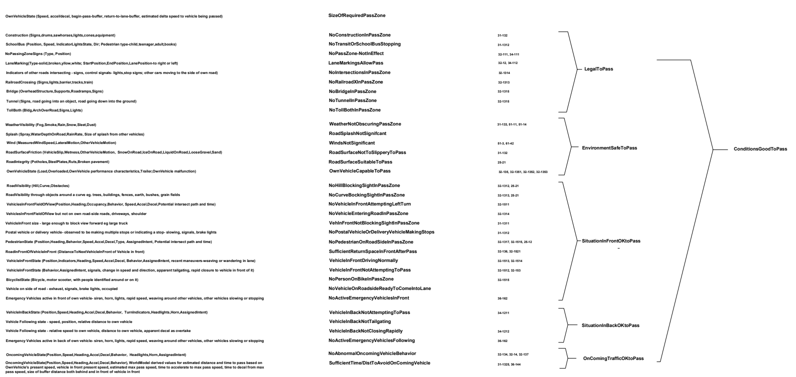
Driving Task Analysis

2) From task architecture, derive dependencies on World Model Situations and Value Judgments



World Model Situation | **Value Judgment** | **Behavior Generation**

Driving Task Analysis



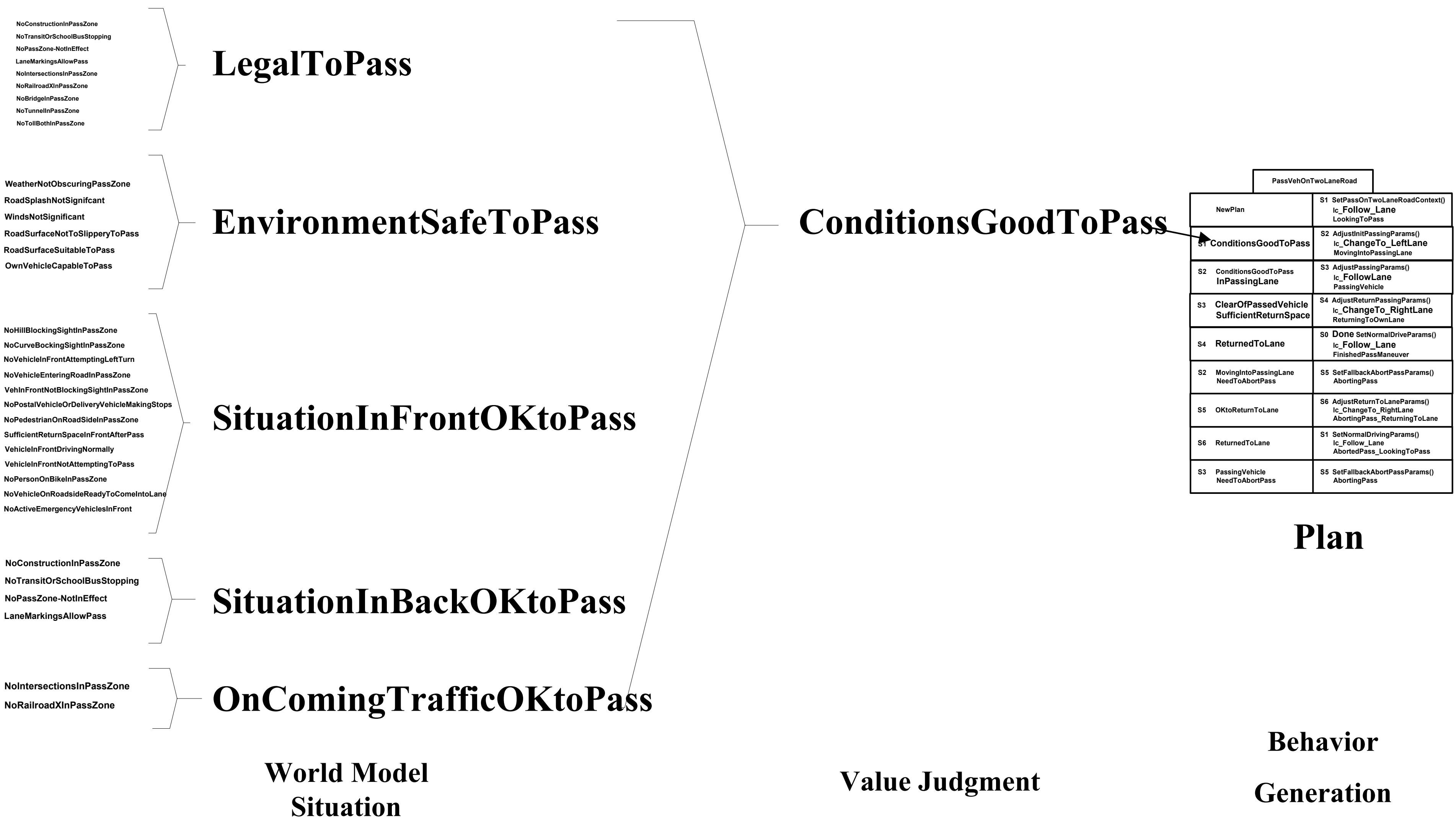
World Model Situation

Value Judgment

Pass on Two Lane Road	
NewPlan	S1 SetPassOnTwoLaneRoadContext() lc_Follow_Lane LookingToPass
S1 ConditionsGoodToPass	S2 AdjustInitPassingParams() lc_ChangeTo_LeftLane MovingIntoPassingLane
S2 ConditionsGoodToPass InPassingLane	S3 AdjustPassingParams() lc_FollowLane PassingVehicle
S3 ClearOfPassedVehicle SufficientReturnSpace	S4 AdjustReturnPassingParams() lc_ChangeTo_RightLane ReturningToOwnLane
S4 ReturnedToLane	S0 Done SetNormalDriveParams() lc_Follow_Lane FinishedPassManeuver
S2 MovingIntoPassingLane NeedToAbortPass	S5 SetFallbackAbortPassParams() AbortingPass
S5 OKtoReturnToLane	S6 AdjustReturnToLaneParams() lc_ChangeTo_RightLane AbortingPass_ReturningToLane
S6 ReturnedToLane	S1 SetNormalDrivingParams() lc_Follow_Lane AbortedPass_LookingToPass
S3 PassingVehicle NeedToAbortPass	S5 SetFallbackAbortPassParams() AbortingPass

Plan

Driving Task Analysis



Driving Task Analysis

NoConstructionInPassZone
NoTransitOrSchoolBusStopping
NoPassZone-NotInEffect
LaneMarkingsAllowPass
NoIntersectionsInPassZone
NoRailroadXInPassZone
NoBridgeInPassZone
NoTunnelInPassZone
NoTollBothInPassZone

LegalToPass

WeatherNotObscuringPassZone
RoadSplashNotSignifcant
WindsNotSignificant
RoadSurfaceNotToSlipperyToPass
RoadSurfaceSuitableToPass
OwnVehicleCapableToPass

EnvironmentSafeToPass

NoHillBlockingSightInPassZone
NoCurveBockingSightInPassZone
NoVehicleInFrontAttemptingLeftTurn
NoVehicleEnteringRoadInPassZone
VehInFrontNotBlockingSightInPassZone
NoPostalVehicleOrDeliveryVehicleMakingStops
NoPedestrianOnRoadSideInPassZone
SufficientReturnSpaceInFrontAfterPass
VehicleInFrontDrivingNormally
VehicleInFrontNotAttemptingToPass
NoPersonOnBikeInPassZone
NoVehicleOnRoadsideReadyToComeIntoLane
NoActiveEmergencyVehiclesInFront

SituationInFrontOKtoPass

NoConstructionInPassZone
NoTransitOrSchoolBusStopping
NoPassZone-NotInEffect
LaneMarkingsAllowPass

SituationInBackOKtoPass

NoIntersectionsInPassZone
NoRailroadXInPassZone

OnComingTrafficOKtoPass

ConditionsGoodToPass

PassVehOnTwoLaneRoad

NewPlan	S1 SetPassOnTwoLaneRoadContext() lc_Follow_Lane LookingToPass
S1 ConditionsGoodToPass	S2 AdjustInitPassingParams() lc_ChangeTo_LeftLane MovingIntoPassingLane
S2 ConditionsGoodToPass InPassingLane	S3 AdjustPassingParams() lc_FollowLane PassingVehicle
S3 ClearOfPassedVehicle SufficientReturnSpace	S4 AdjustReturnPassingParams() lc_ChangeTo_RightLane ReturningToOwnLane
S4 ReturnedToLane	S0 Done SetNormalDriveParams() lc_Follow_Lane FinishedPassManeuver
S2 MovingIntoPassingLane NeedToAbortPass	S5 SetFallbackAbortPassParams() AbortingPass
S5 OKtoReturnToLane	S6 AdjustReturnToLaneParams() lc_ChangeTo_RightLane AbortingPass_ReturningToLane
S6 ReturnedToLane	S1 SetNormalDrivingParams() lc_Follow_Lane AbortedPass_LookingToPass
S3 PassingVehicle NeedToAbortPass	S5 SetFallbackAbortPassParams() AbortingPass

World Model Situation

Value Judgment

Behavior Generation

Estimated ~2000 situations

Estimated ~150 FSAs

Driving Task Analysis

3) Define Entities, Events, Attributes, Resolutions, and Tolerances

CrossBuck(pos)
Lights(pos, state)
Crossing Gate(pos)
Signs(pos, facing-dir, text and graphics)
Tracks(pos, dir)
Train(pos, dir)
Lanes(pos, dir, width, curvature)
PassingZone(veh speeds, safety buffer, accel)
eg. All attributes must be recognizable out to 600 to 800 feet

NoConstructionInPassZone
NoTransitOrSchoolBusStopping
NoPassZone-NotInEffect
LaneMarkingsAllowPass
NoIntersectionsInPassZone
NoRailroadXInPassZone
NoBridgeInPassZone
NoTunnelInPassZone
NoTollBothInPassZone

LegalToPass

WeatherNotObscuringPassZone
RoadSplashNotSignifcant
WindsNotSignificant
RoadSurfaceNotToSlipperyToPass
RoadSurfaceSuitableToPass
OwnVehicleCapableToPass

EnvironmentSafeToPass

NoHillBlockingSightInPassZone
NoCurveBockingSightInPassZone
NoVehicleInFrontAttemptingLeftTurn
NoVehicleEnteringRoadInPassZone
VehInFrontNotBlockingSightInPassZone
NoPostalVehicleOrDeliveryVehicleMakingStops
NoPedestrianOnRoadSideInPassZone
SufficientReturnSpaceInFrontAfterPass
VehicleInFrontDrivingNormally
VehicleInFrontNotAttemptingToPass
NoPersonOnBikeInPassZone
NoVehicleOnRoadsideReadyToComeIntoLane
NoActiveEmergencyVehiclesInFront

SituationInFrontOKtoPass

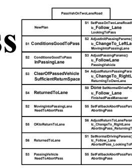
NoConstructionInPassZone
NoTransitOrSchoolBusStopping
NoPassZone-NotInEffect
LaneMarkingsAllowPass

SituationInBackOKtoPass

NoIntersectionsInPassZone
NoRailroadXInPassZone

OnComingTrafficOKtoPass

ConditionsGoodToPass

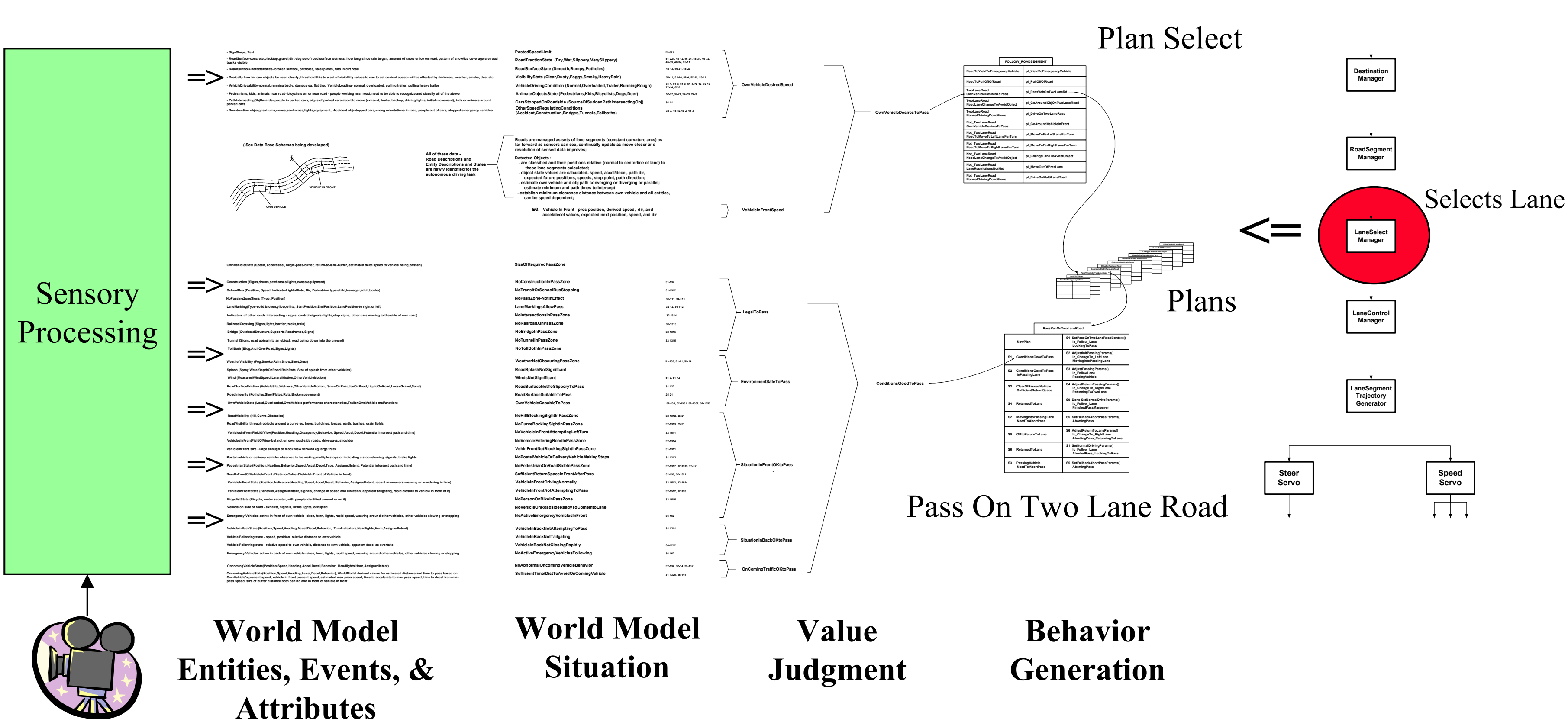


Plan

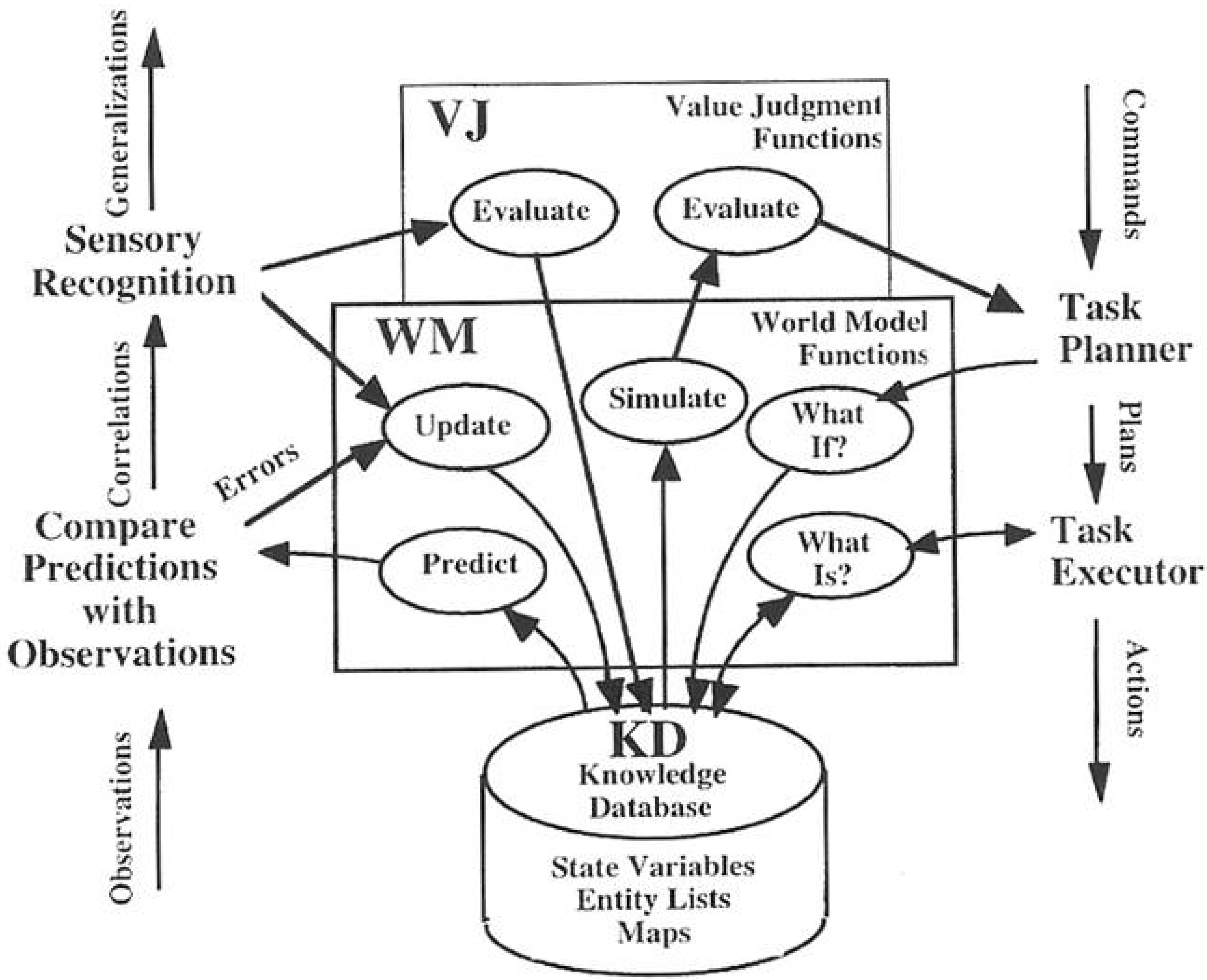
World Model Entities, Events, and Attributes | **World Model Situation** | **Value Judgment** | **Behavior Generation**

Driving Task Analysis

4) Use defined World Model Situations, Entities, Events, and Attributes as the Requirements for Perception



World Modeling



Forms of Representation

Iconic
- signals, images, maps (arrays)
- Support communication, geometry, and navigation
- Have range and resolution in space and time

Symbolic
- objects, events, classes (abstract data structures)
- Support mathematics, logic, and linguistics
- Have vocabulary and ontology

Links
- relationships (pointers)
- Support syntax, grammar, and semantics
- Have direction and type

Types of Knowledge

- About the environment – places, conditions, situations
- About things – entities, states, attributes, classes, relationships
- About actions – tasks, skills, motives, plans, behaviors
- About feelings – experiences, tastes, beliefs, emotions, pain, pleasure, grief, hope, fear, guilt, need
- About experiences – events, situations, scenarios, sights, sounds, smells, tastes
- About rules – logic, mathematics, geometry, language, physics
- About models – dynamics, kinematics, simulation, visualization

Types of Representation

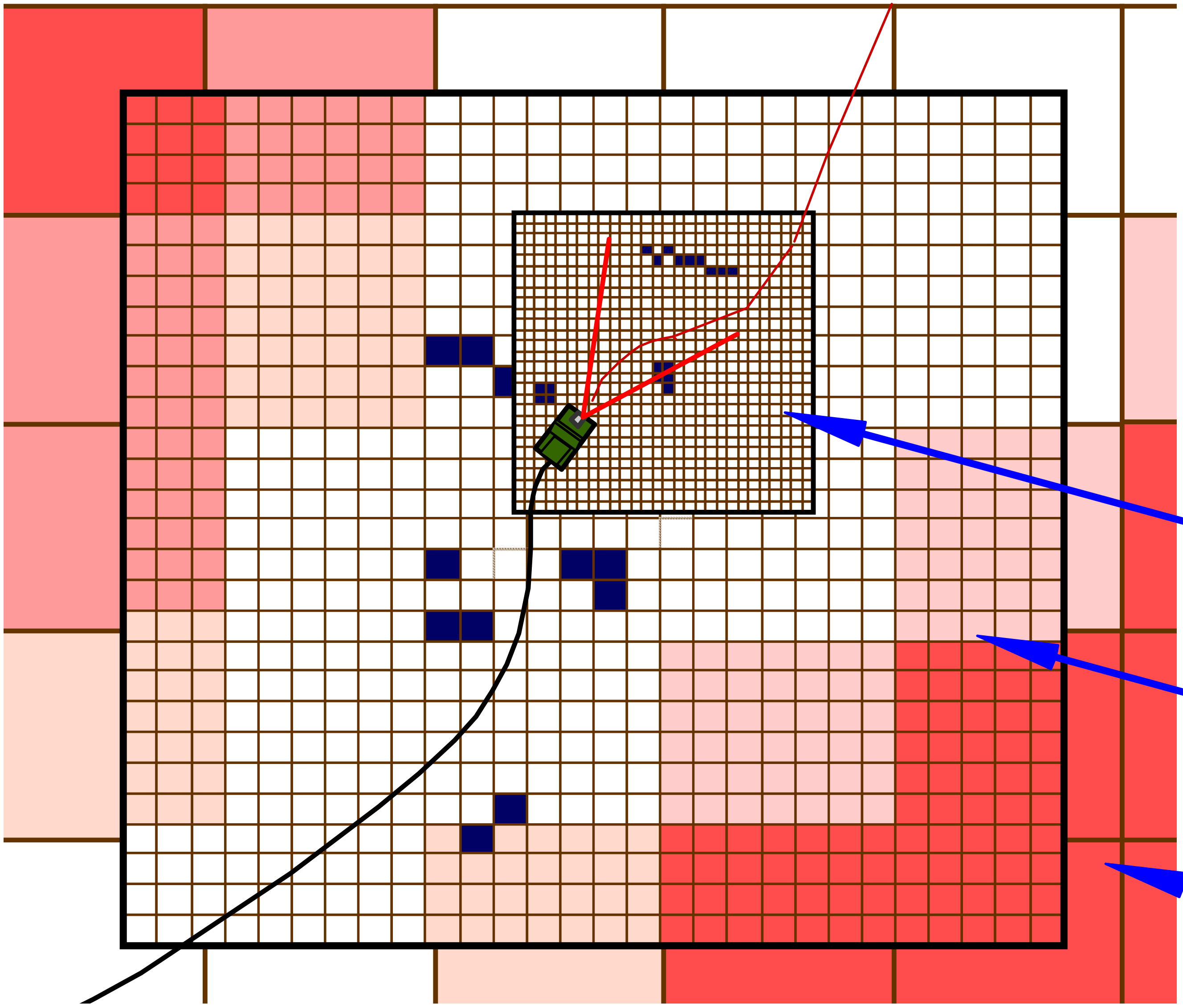
- Immediate experience < 100 ms, transitory
- Short term memory – seconds to minutes, volatile
- Long term memory – indefinite, non-volatile

- Prediction of future conditions
  - ~ immediate experience for perception
  - ~ short term memory for planning

- Entities – things that occupy space
- Events – things that occupy time
  - attributes and relationships of entities and events

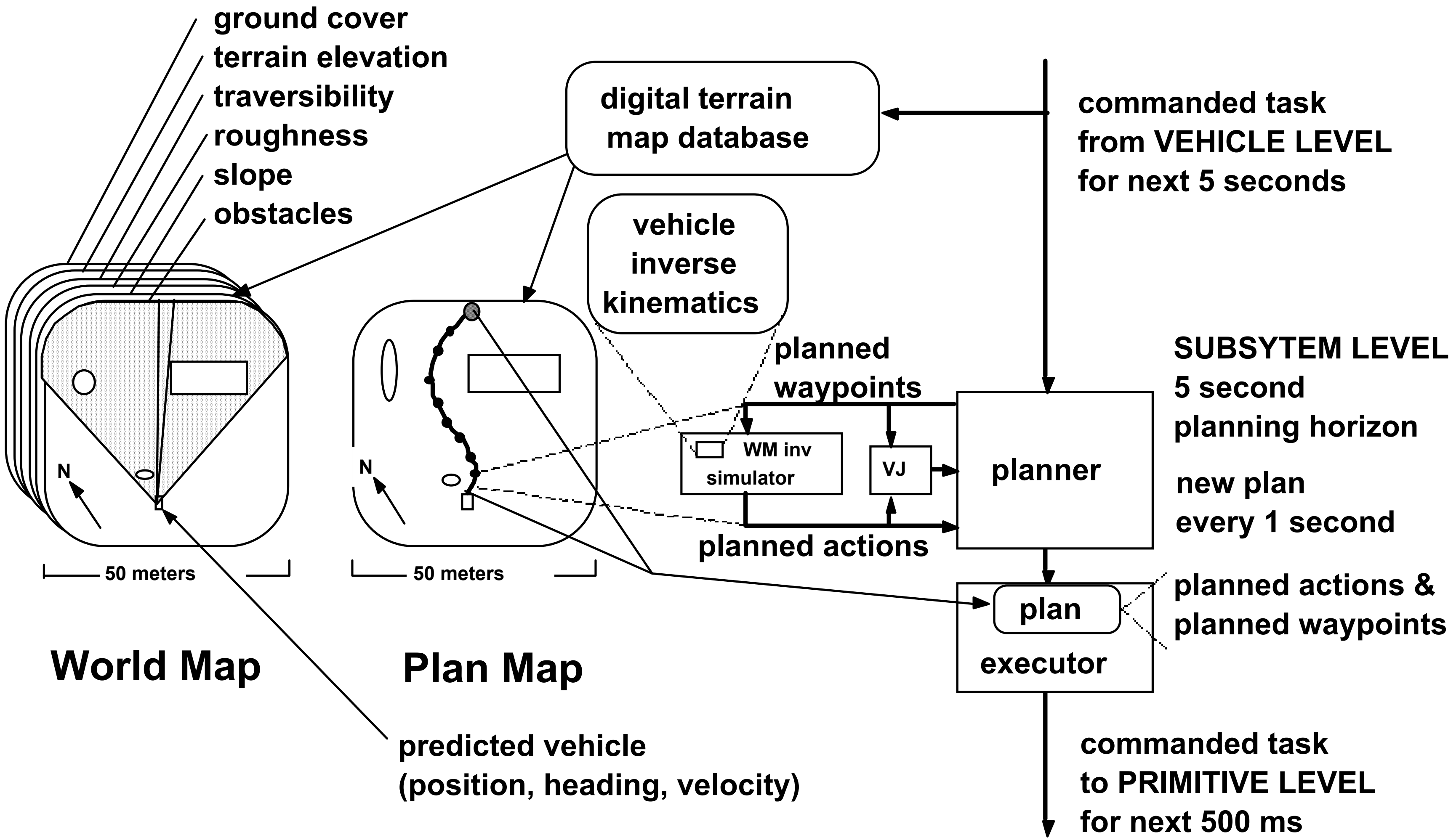
- Skills – knowledge of how to act so as to achieve goals

MULTI-RESOLUTION MAPS

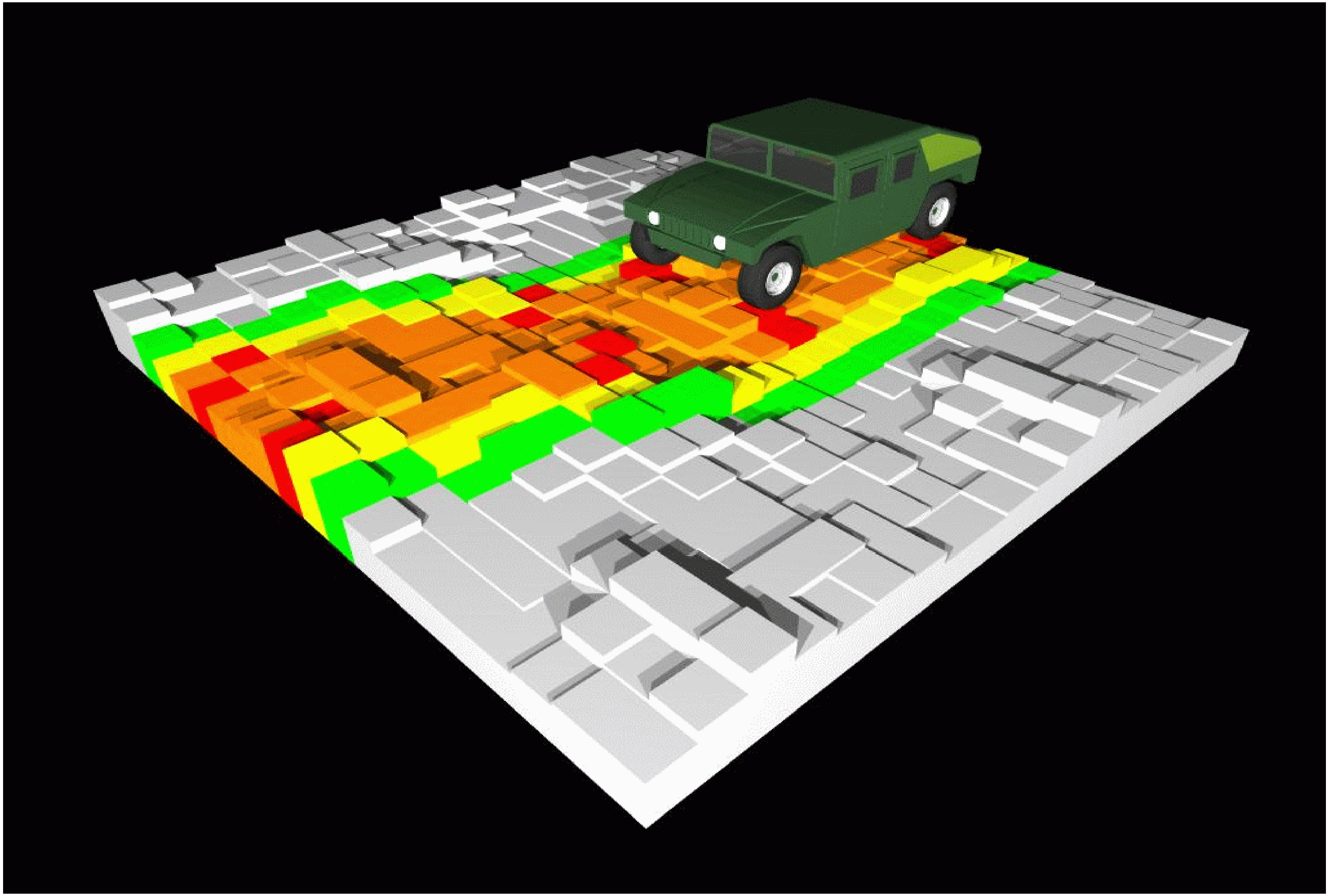


- Data flows up and down between the different maps
- Path planning occurs at each level

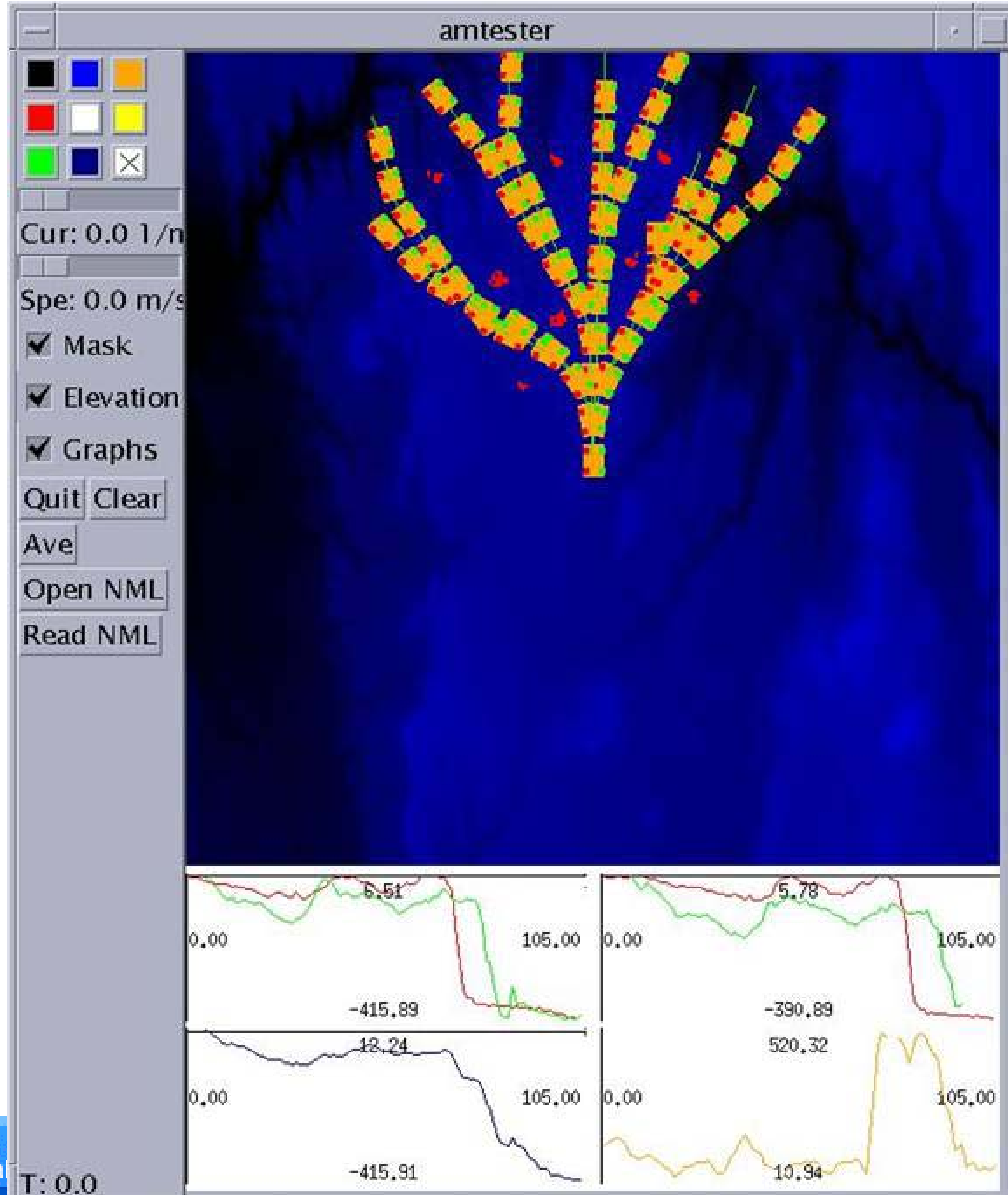
Planning at Subsystem Level



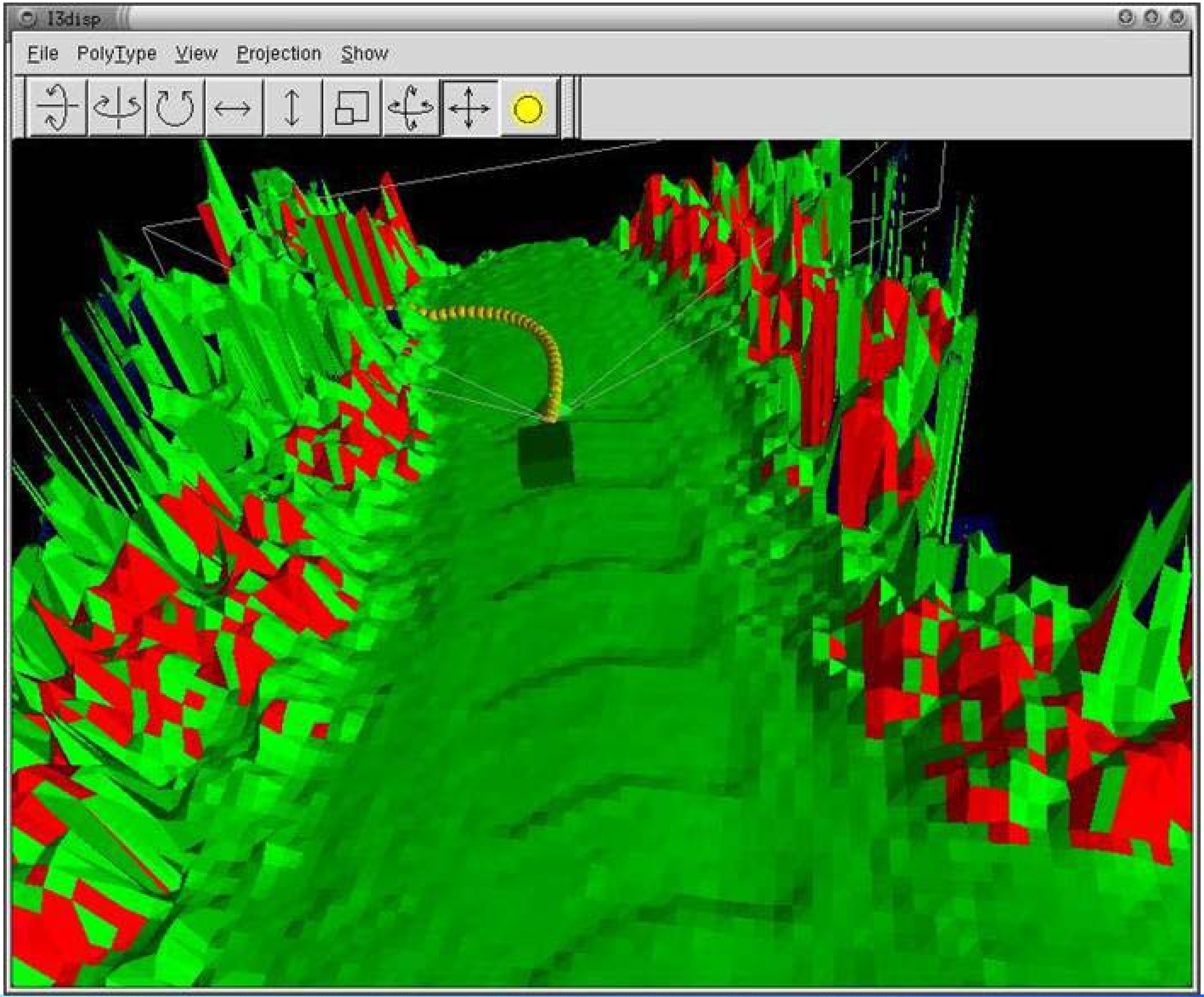
3-D Terrain Traversability



Path Cost Evaluation



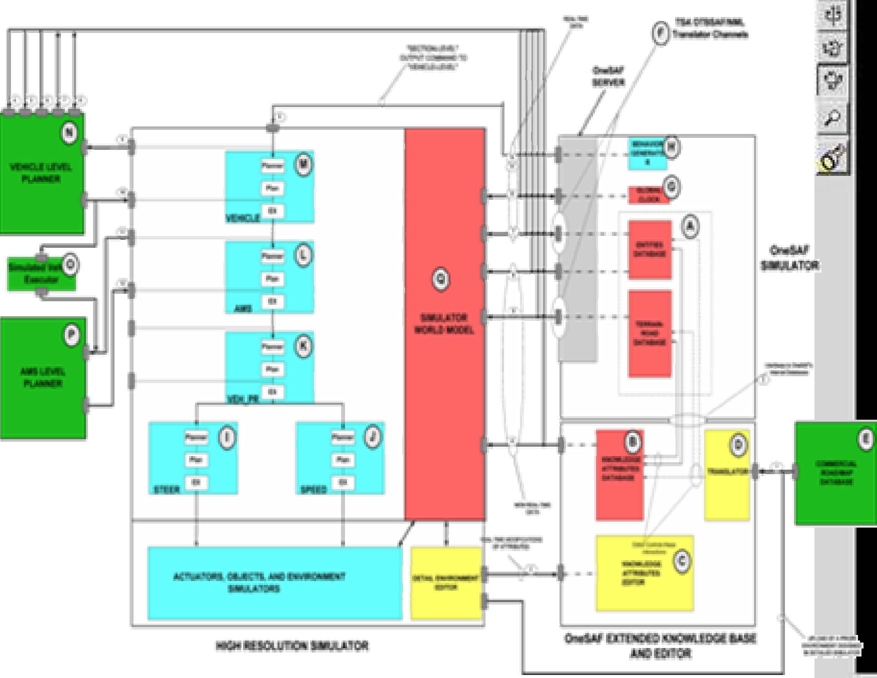
Planning to Turn Off-Road



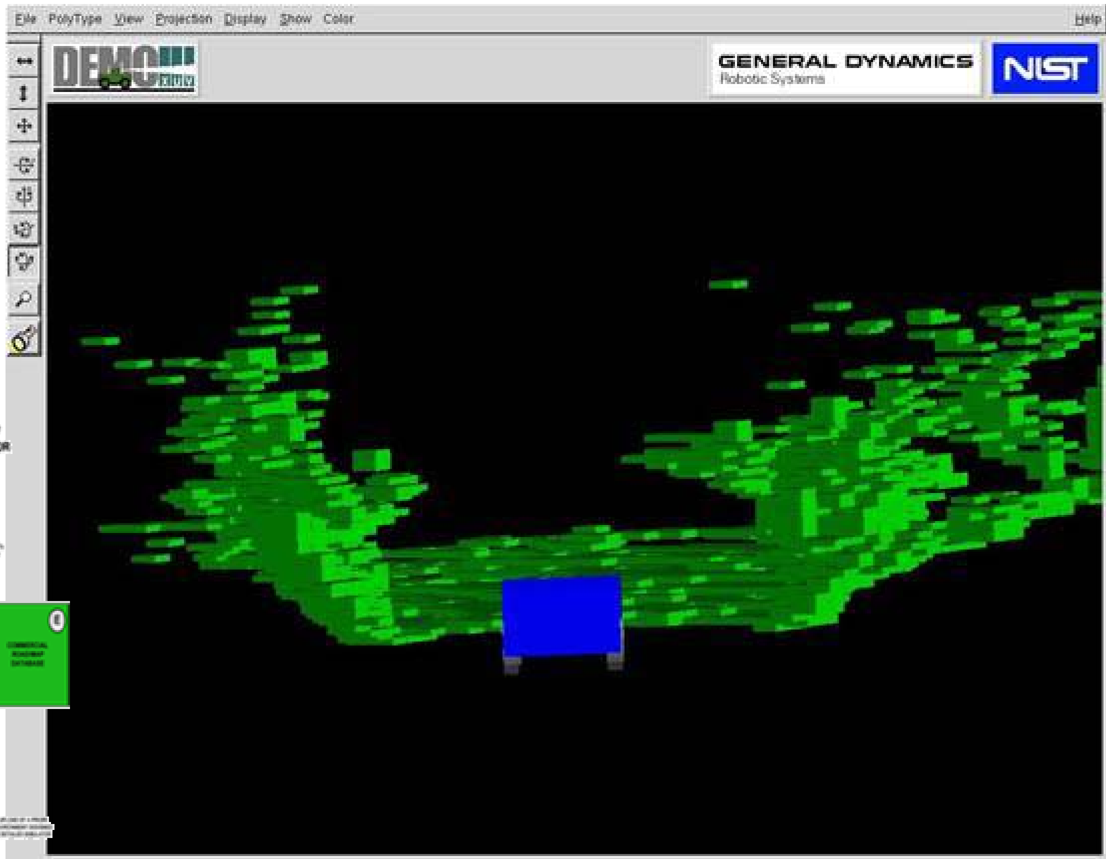

DEMO

LADAR is a Critical Break-Through

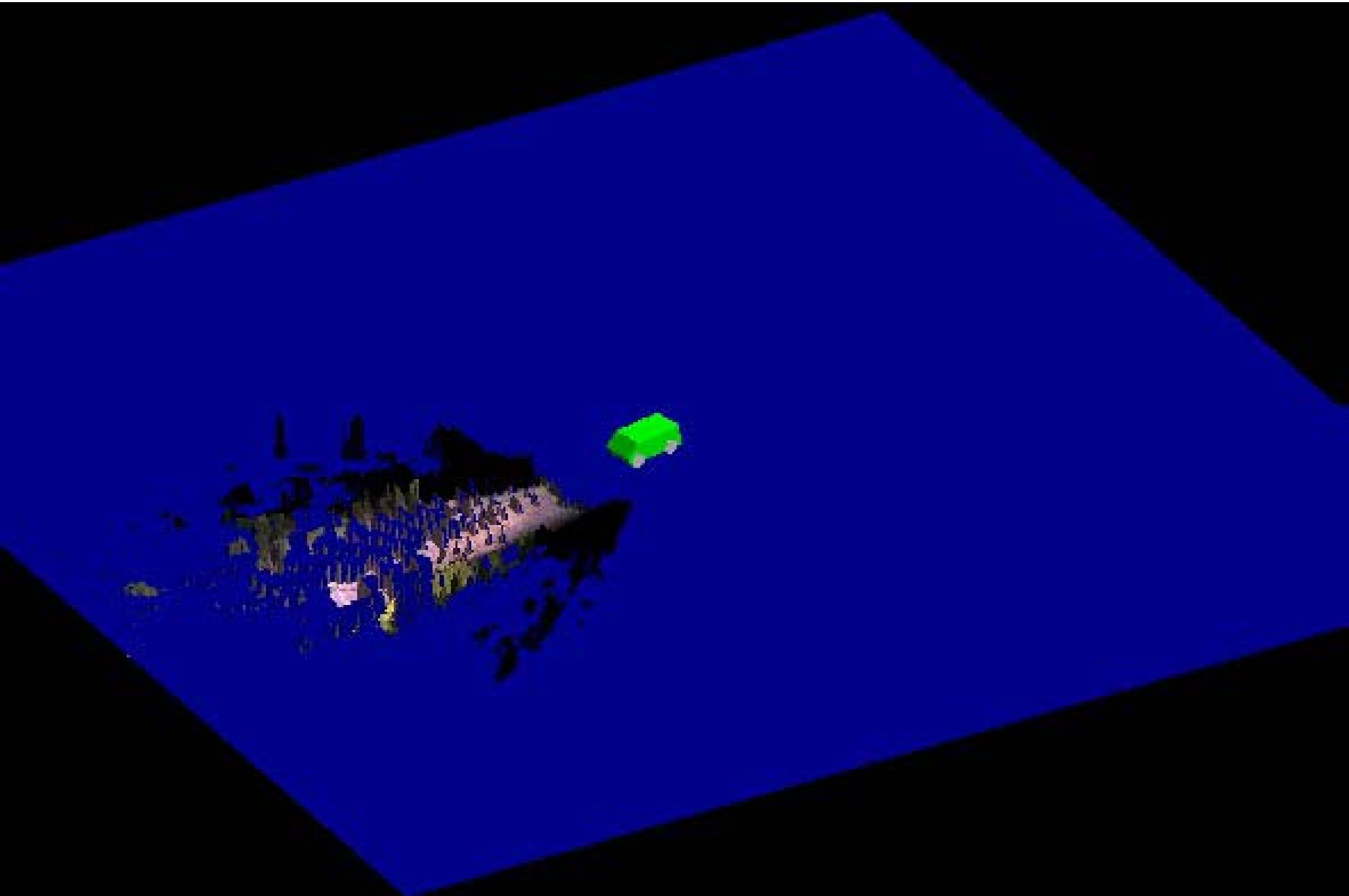
Color Image



Range Image



Color overlaid on LADAR



ARL/NIST LADAR Testbed

Color Camera

Hi Speed LADAR

Color Stereo

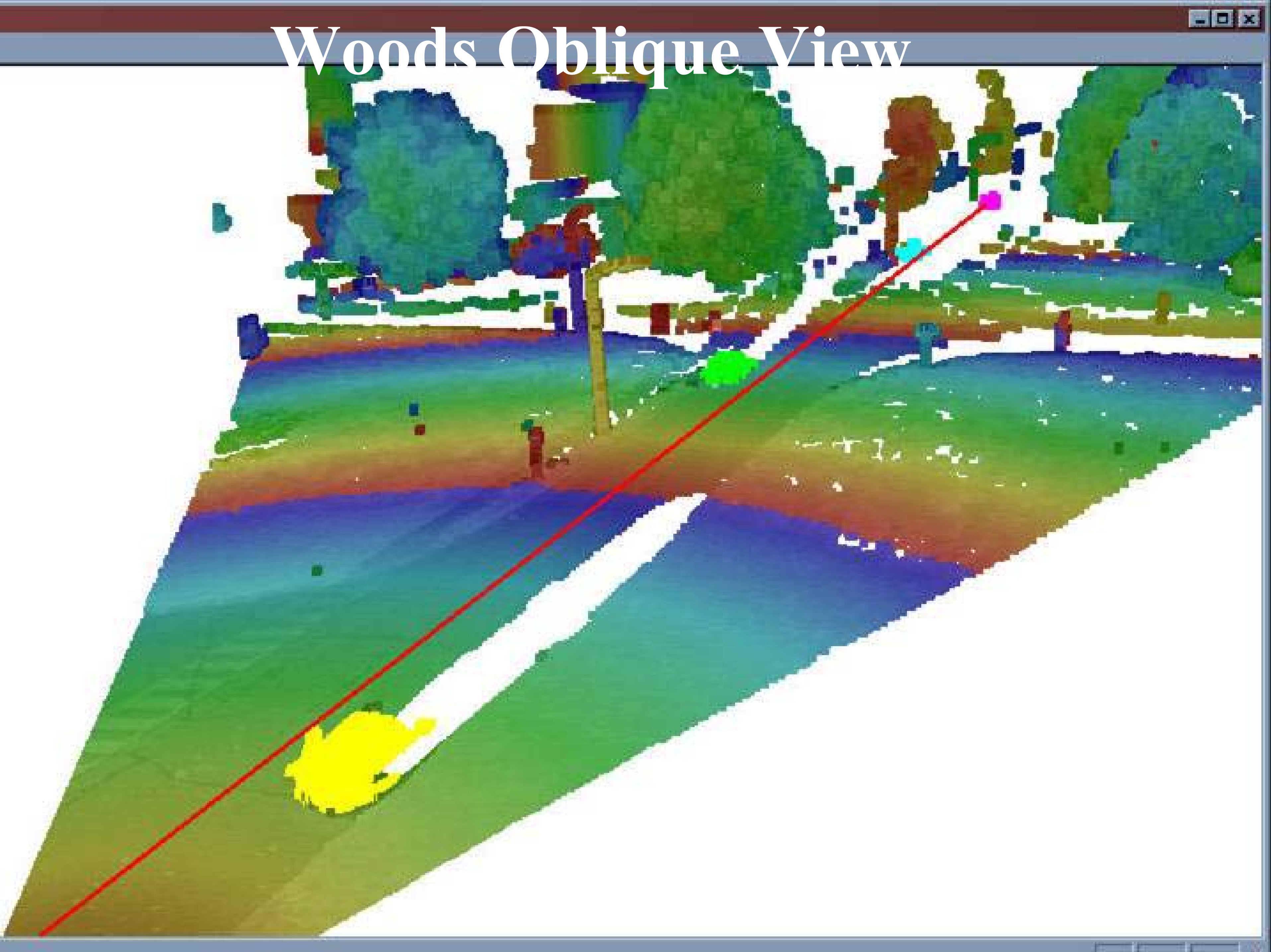
Hi Resolution LADAR

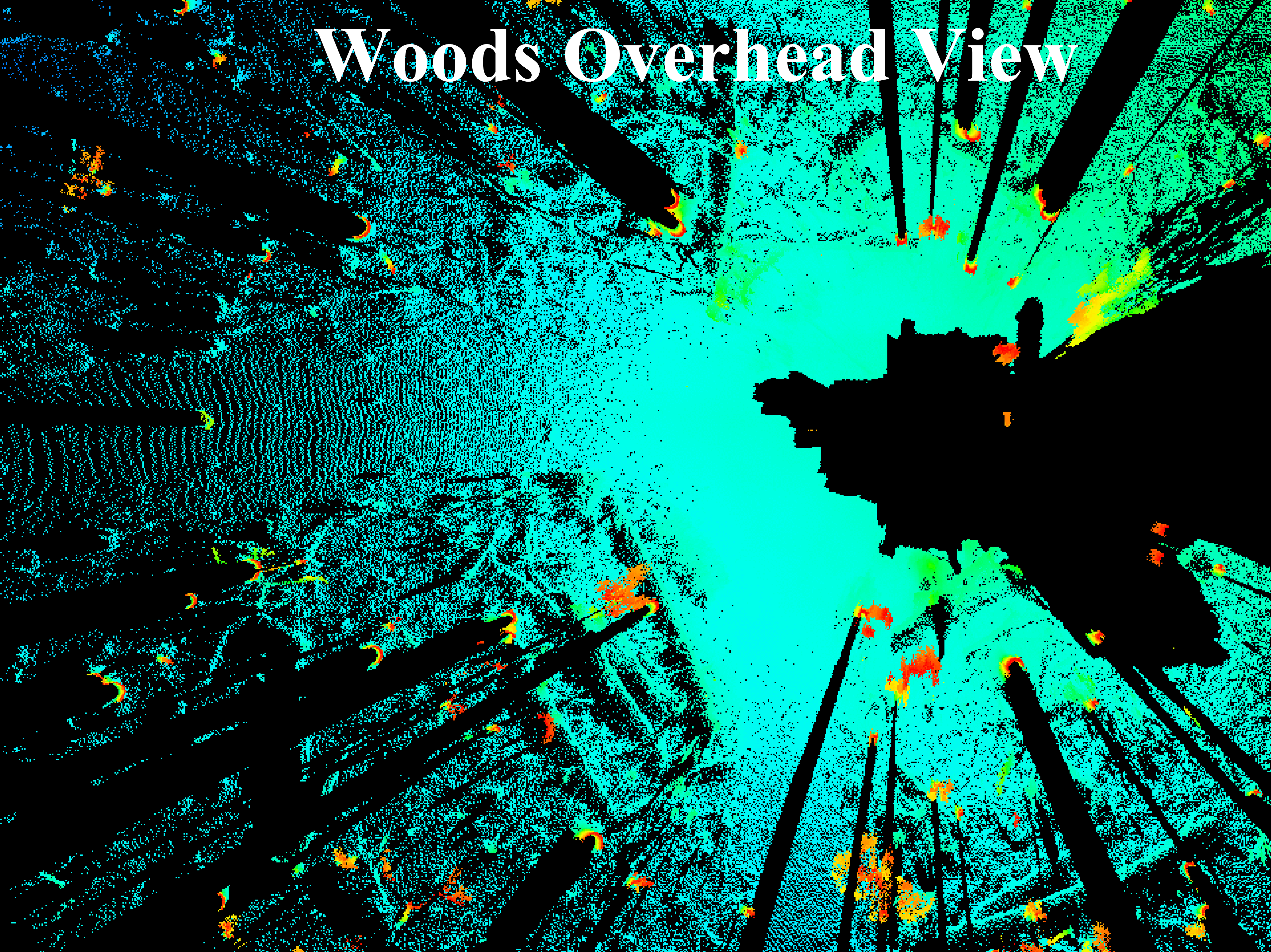
GPS

INS

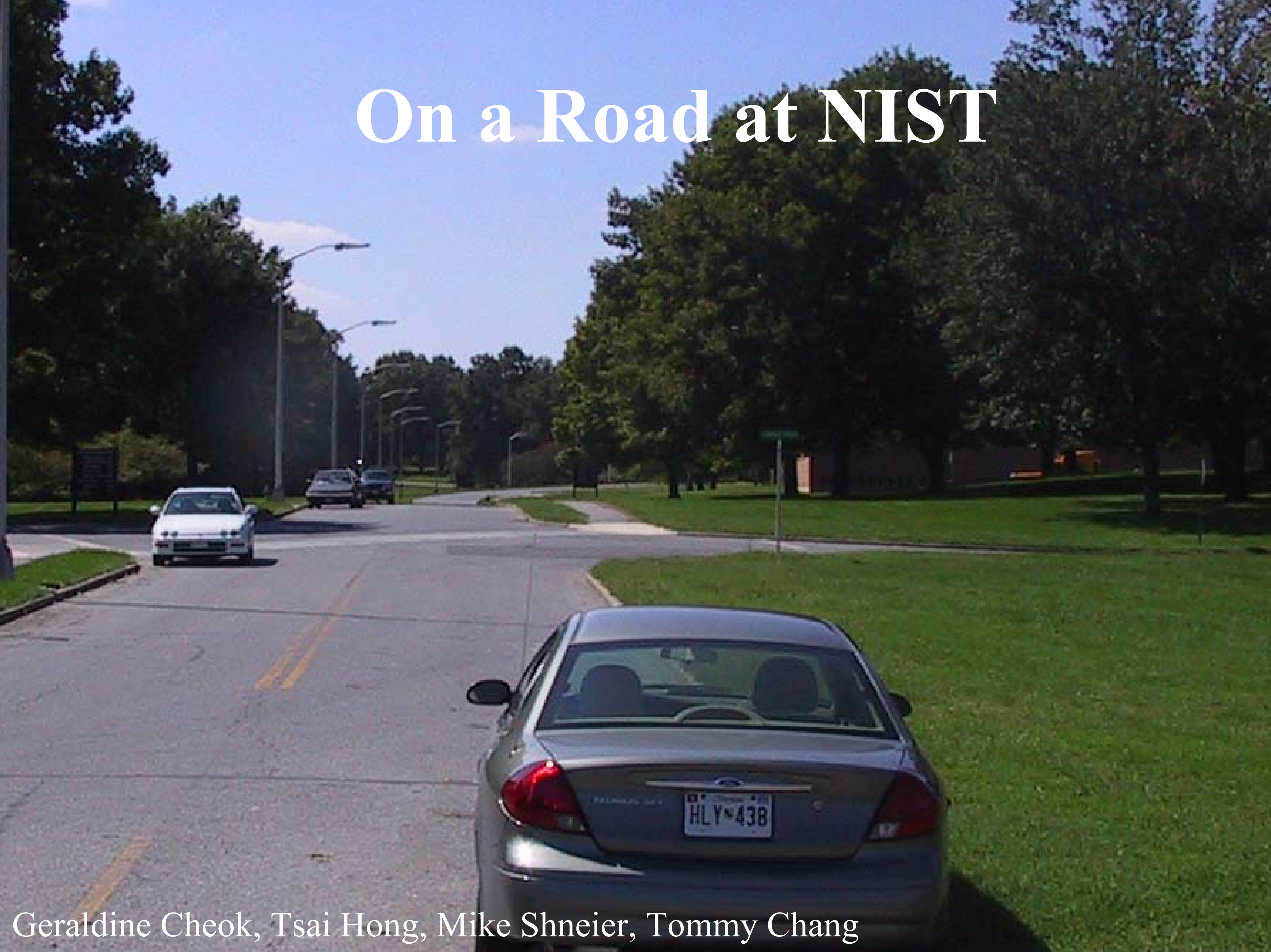
LADAR Intensity Image in the Woods

Woods Oblique View

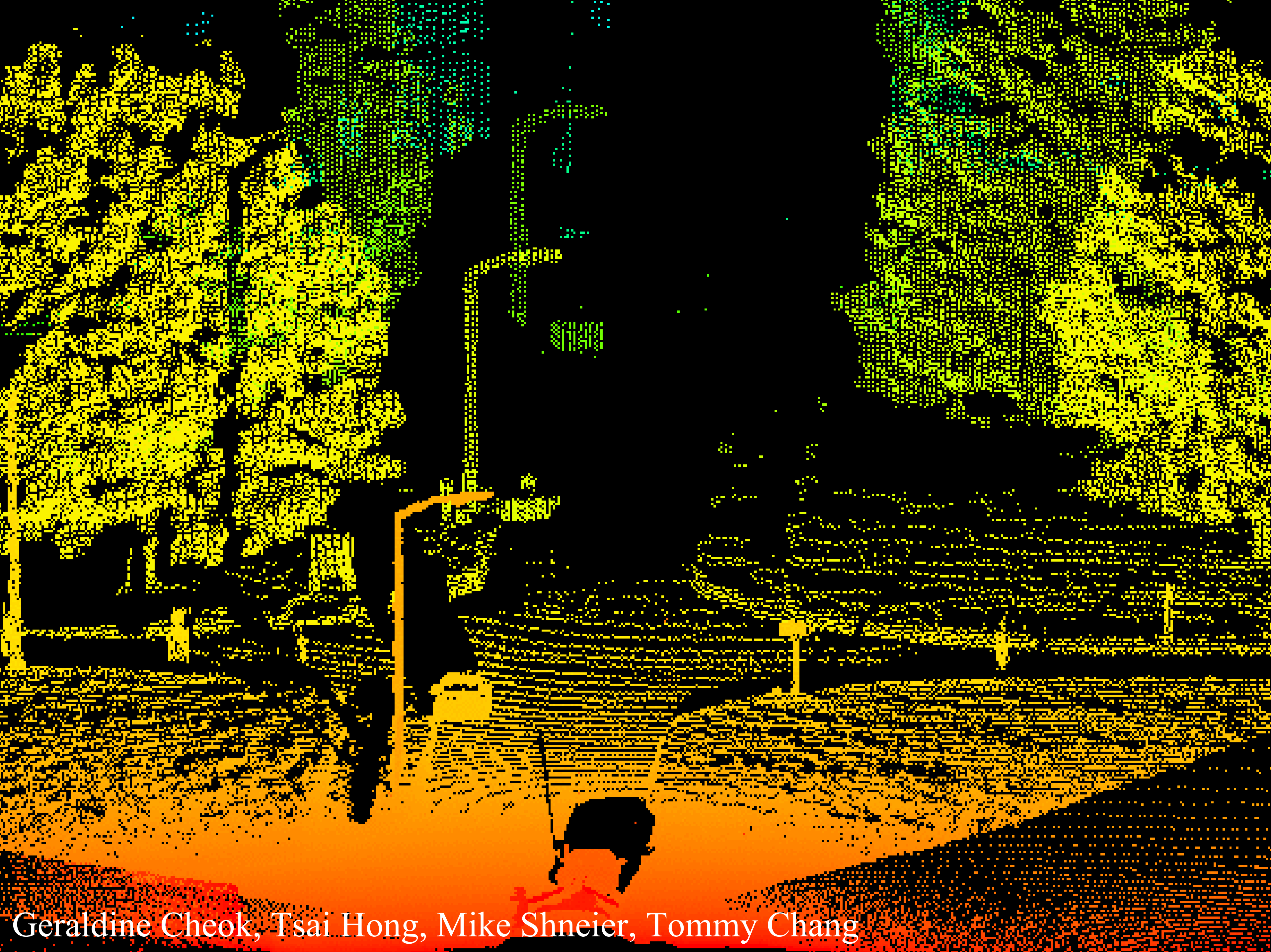



Woods Overhead View

On a Road at NIST



Geraldine Cheok, Tsai Hong, Mike Shneier, Tommy Chang

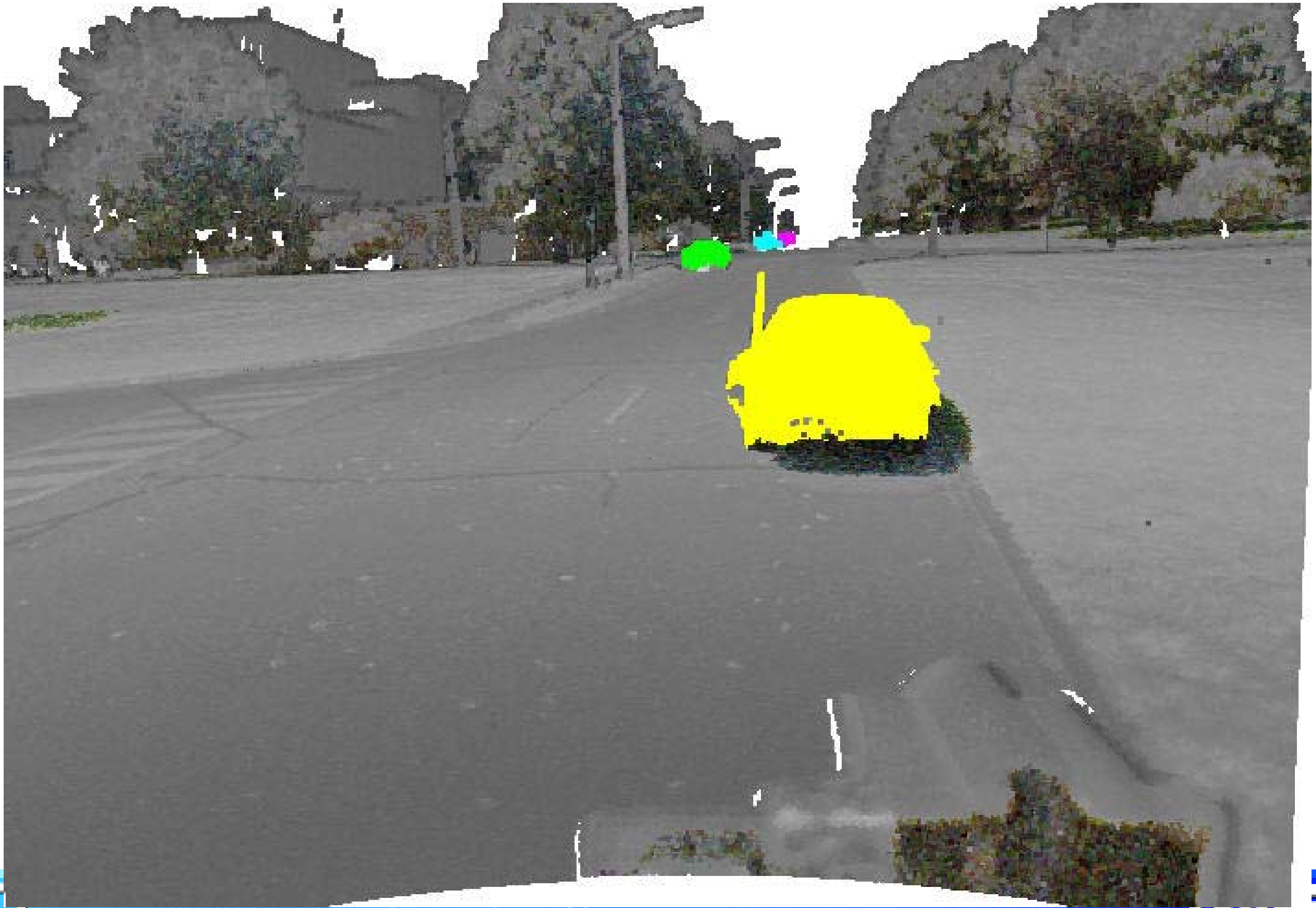

Geraldine Cheok, Tsai Hong, Mike Shneier, Tommy Chang

Segmentation
Remove Ground Plane

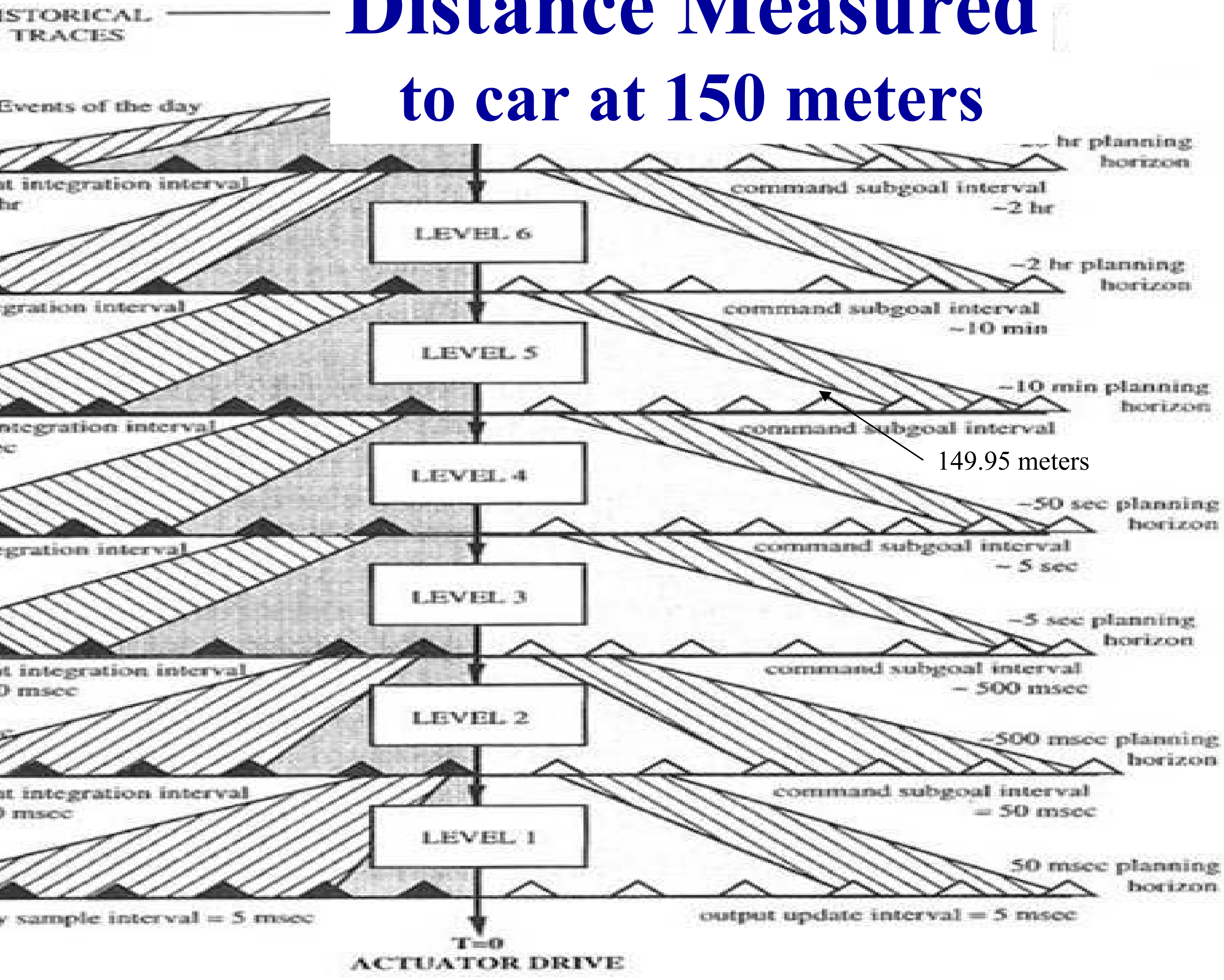
Repeating False Color Range Image

Segmentation of Cars

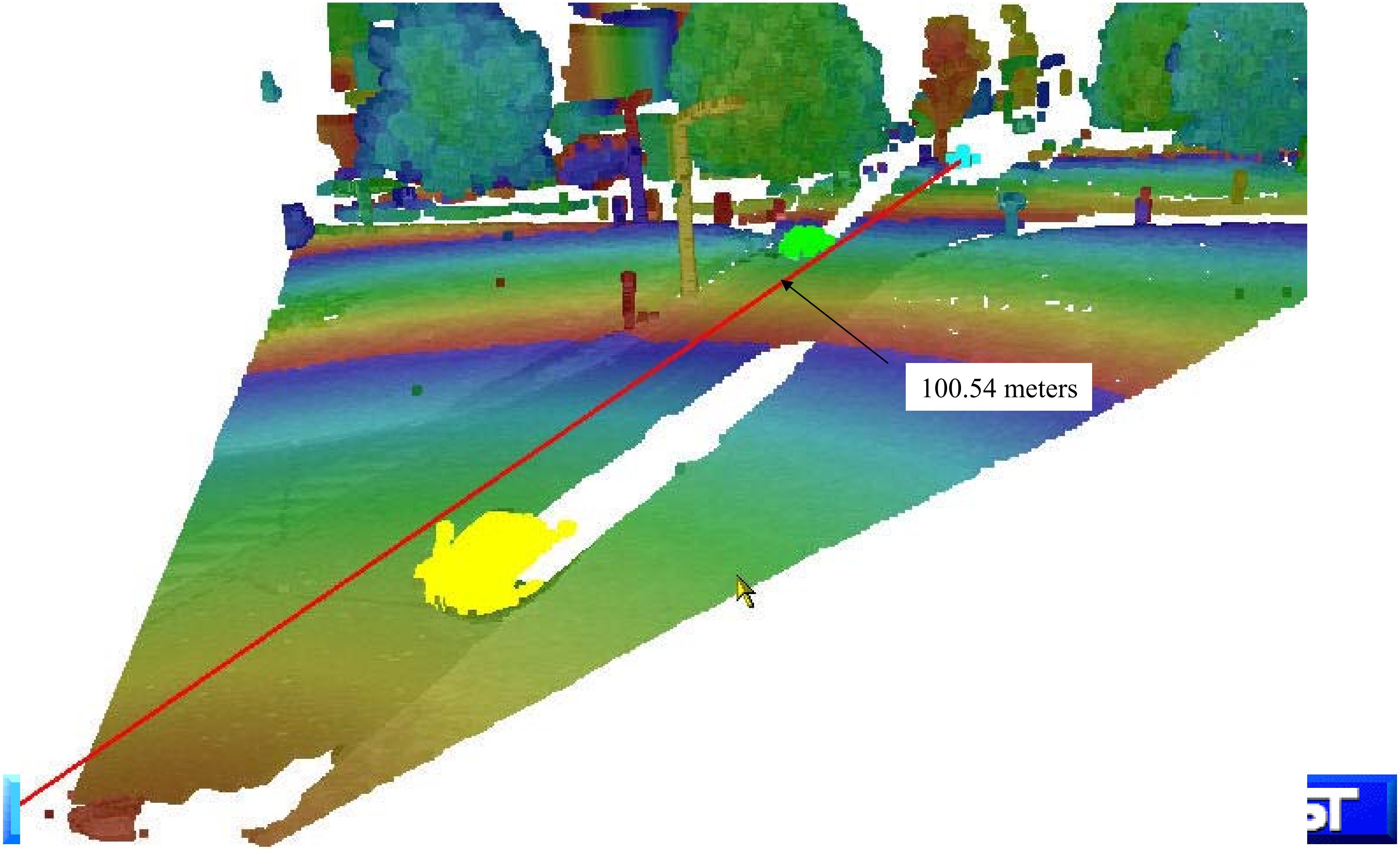
False Color Range



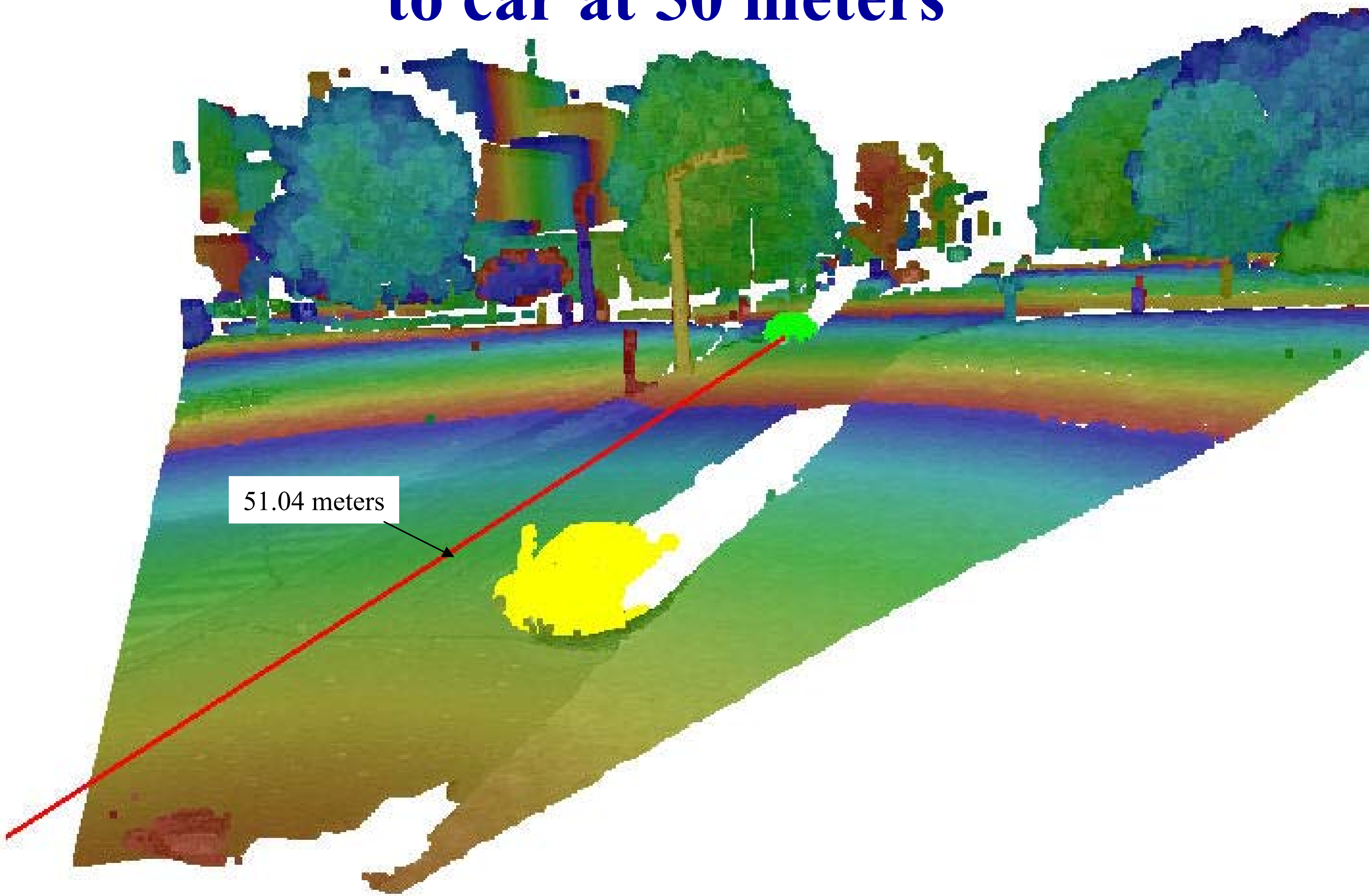
Distance Measured to car at 150 meters



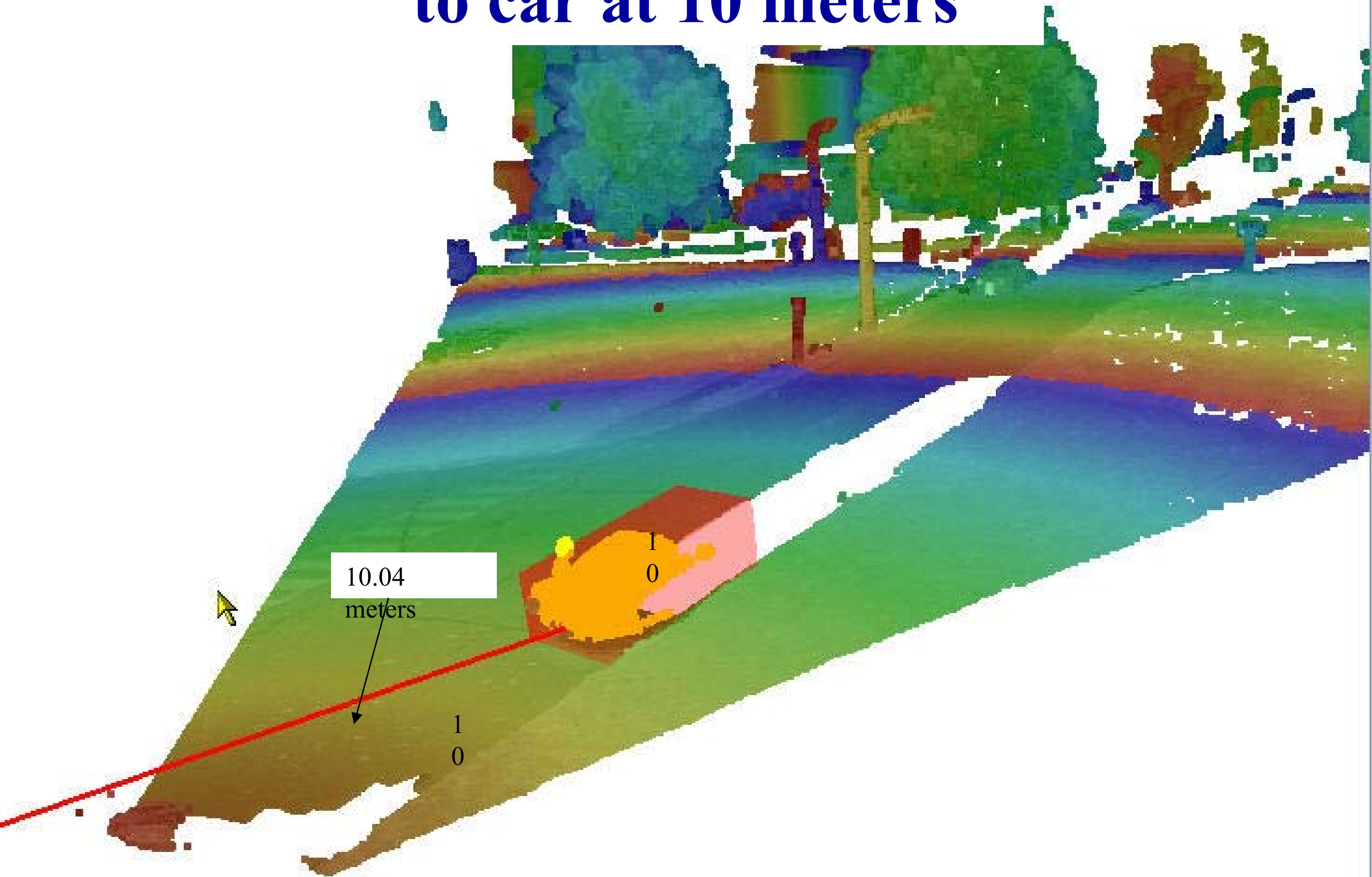
Distance Measured
to car at 100 meters



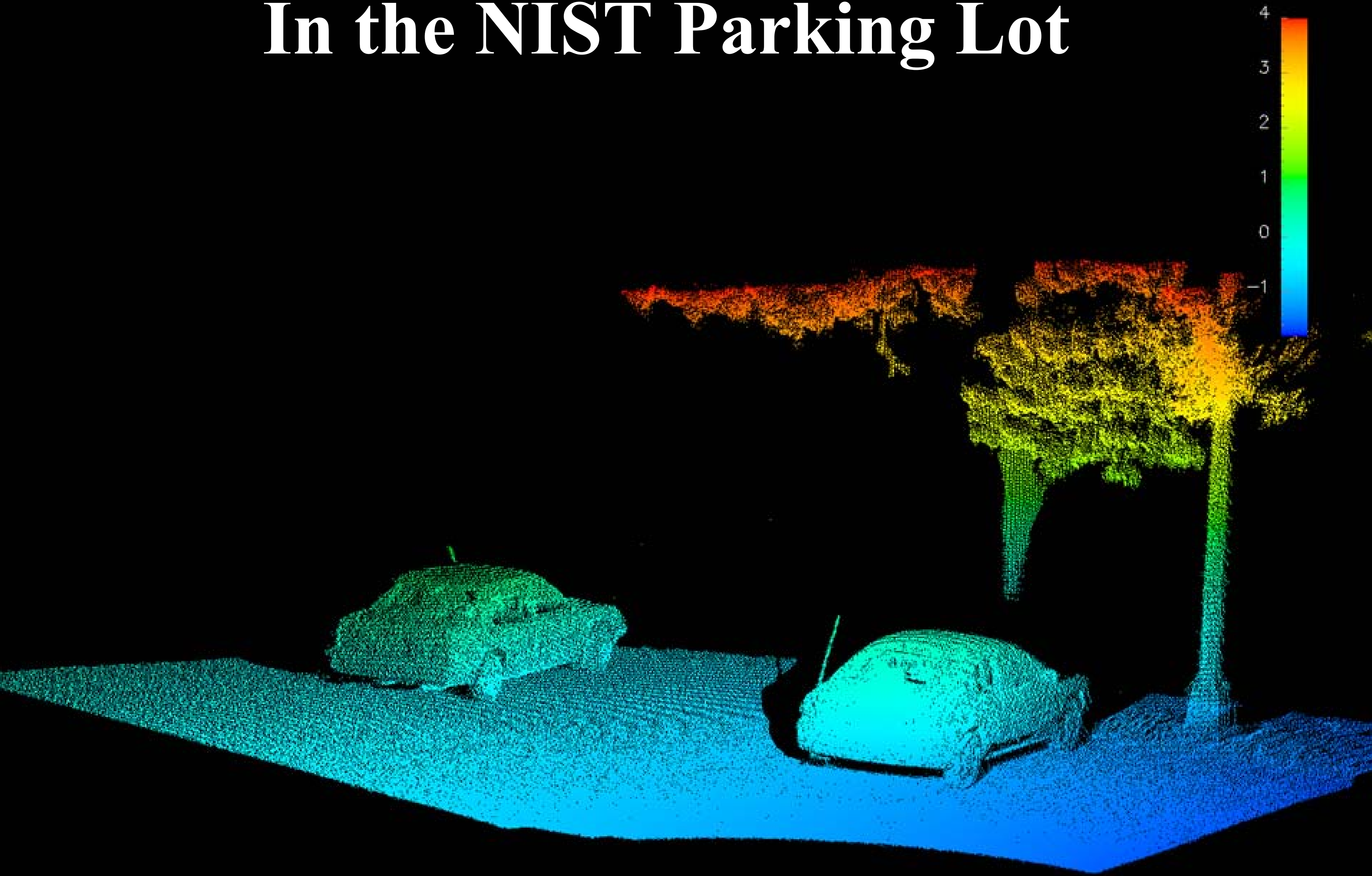
Distance Measured
to car at 50 meters



Distance Measured
to car at 10 meters

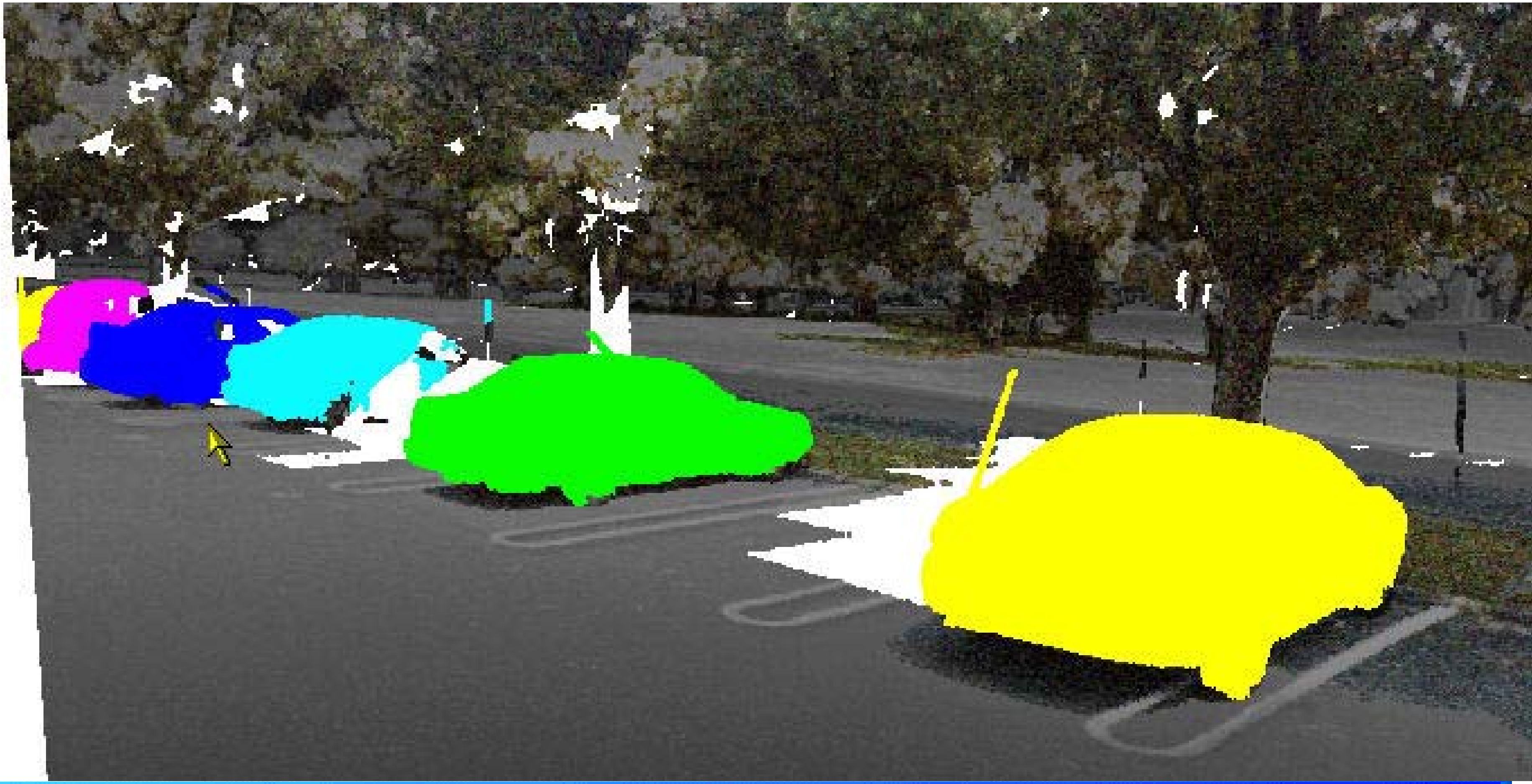


In the NIST Parking Lot

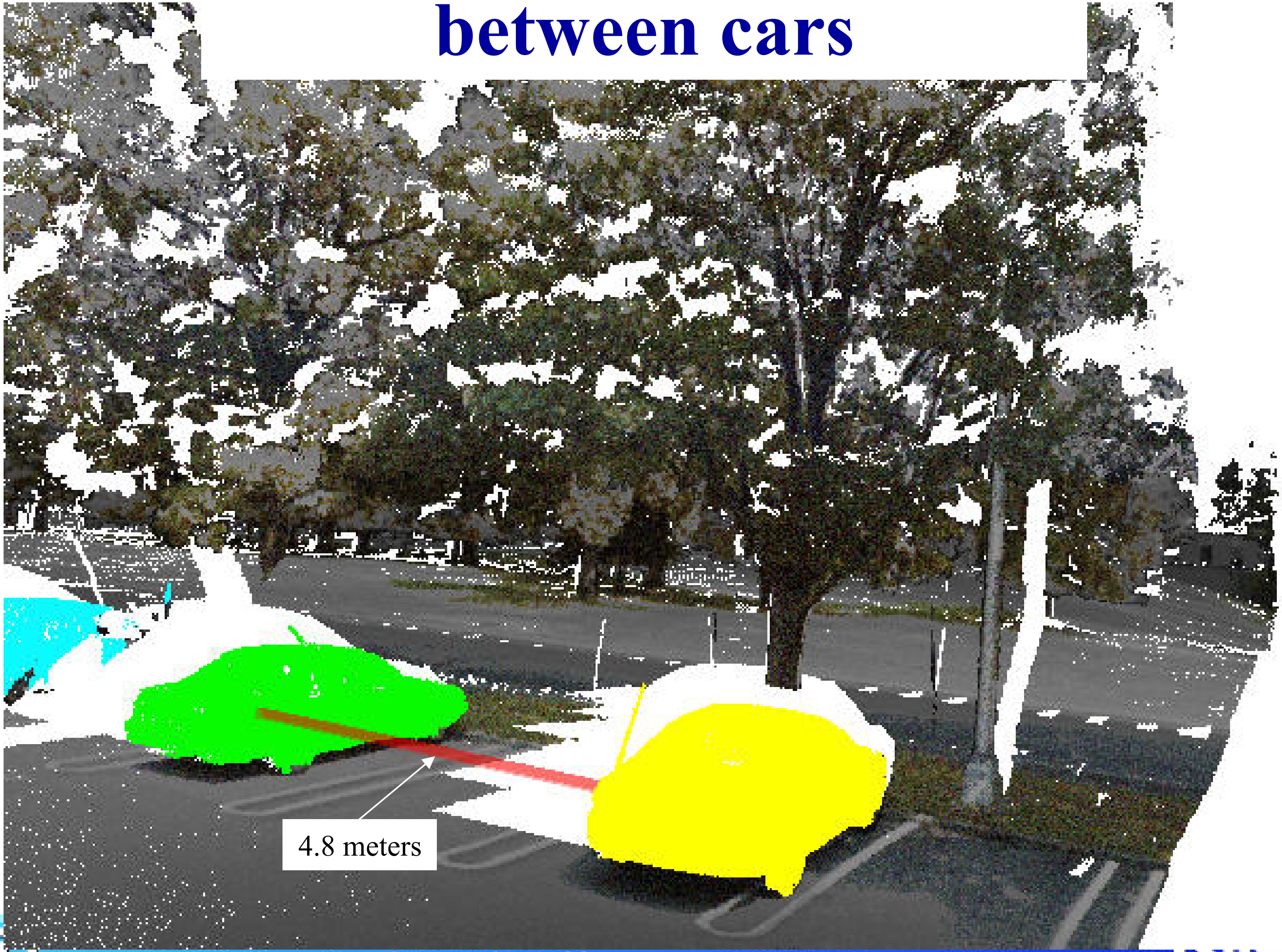


Geraldine Cheok, Tsai Hong, Mike Shneier, Tommy Chang

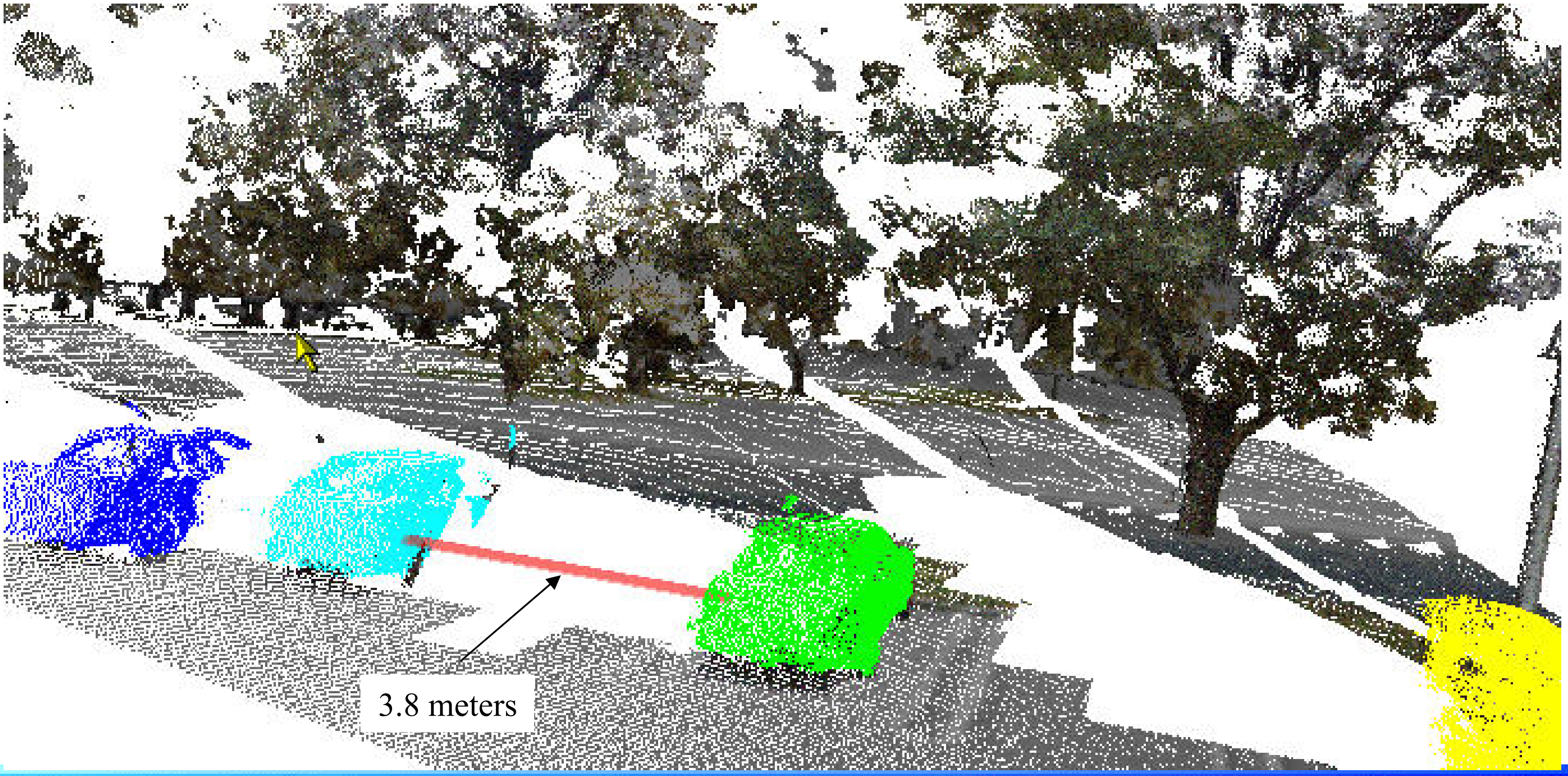
Segmentation of Cars
in parking lot



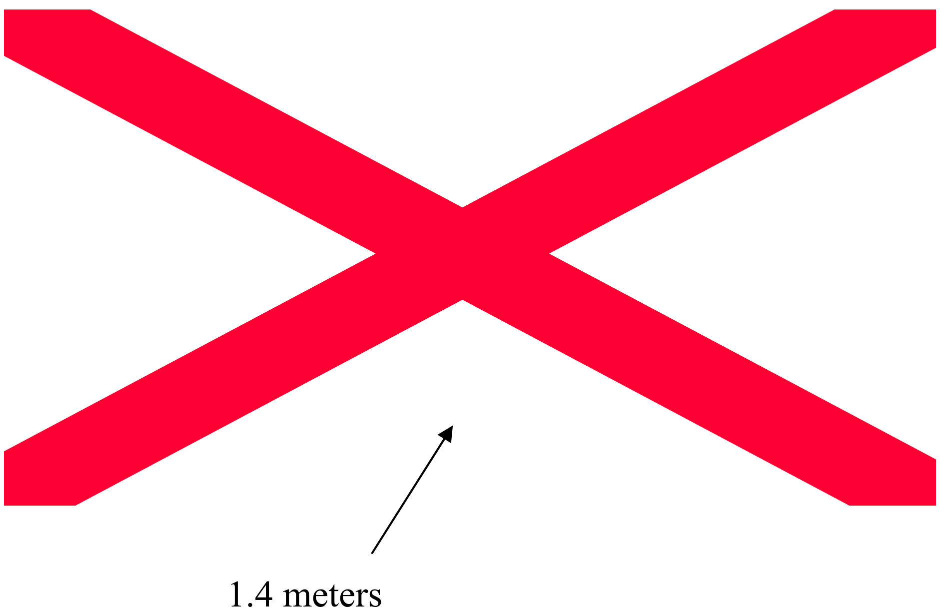
Measured Separation
between cars



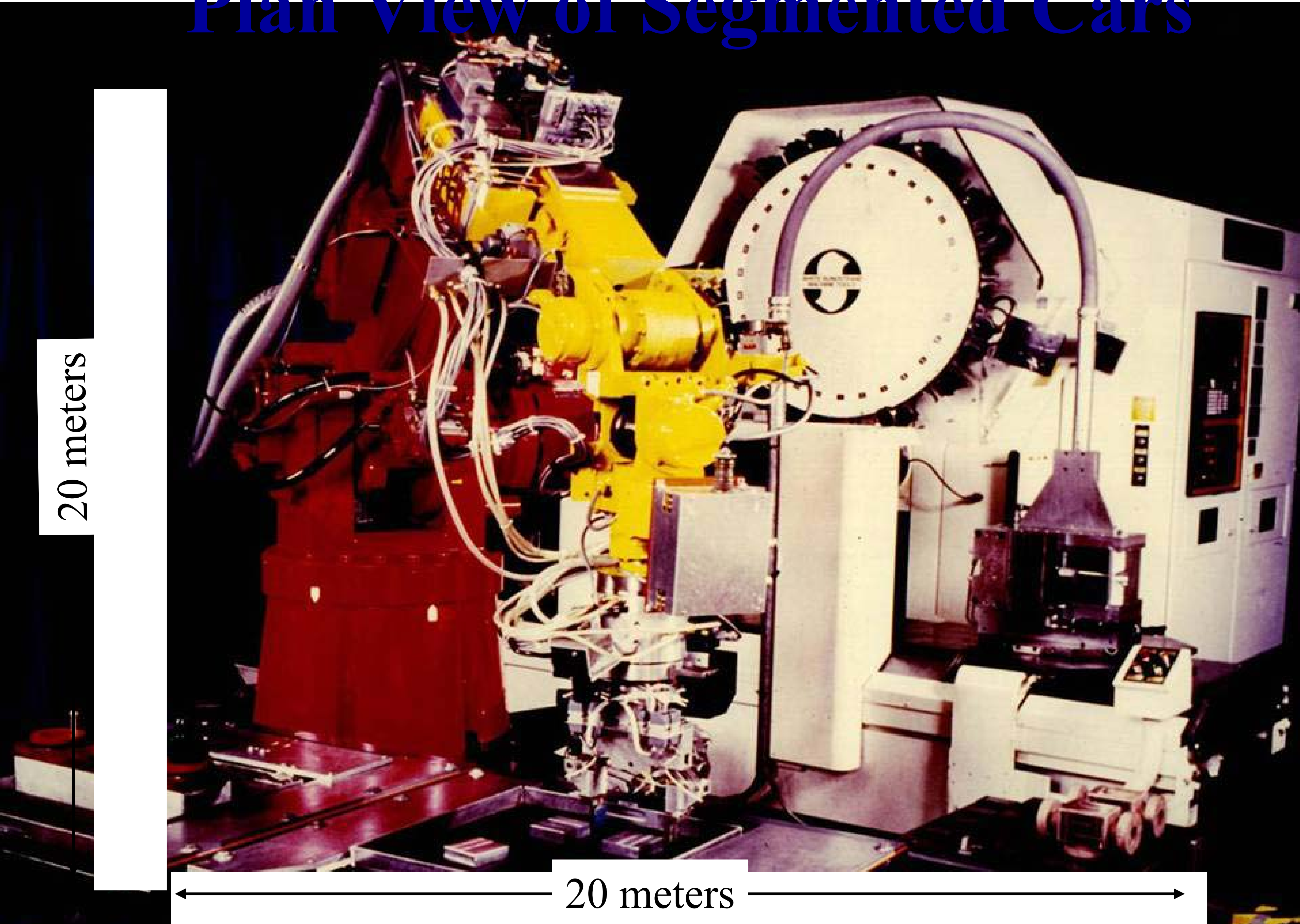
Measured Separation
between second and third car



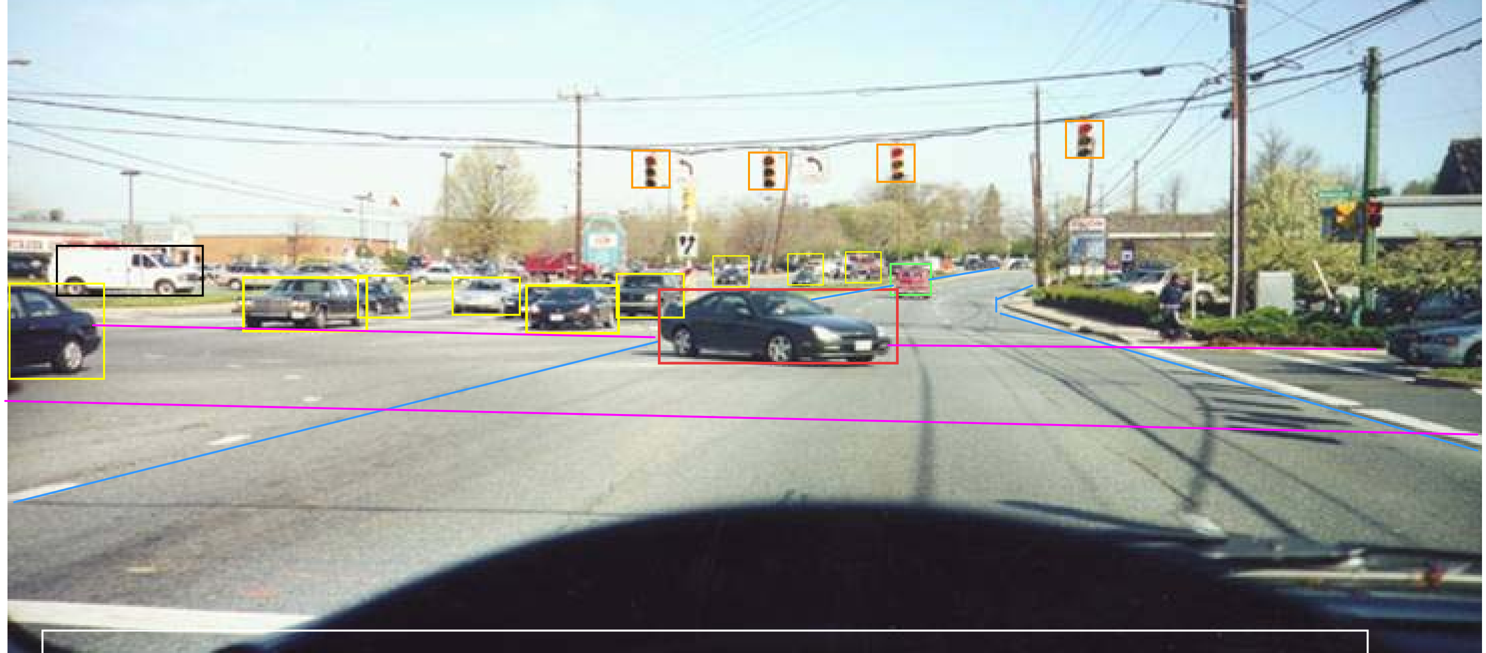
Measured Separation
between third and fourth car



Plan View of Segmented Cars

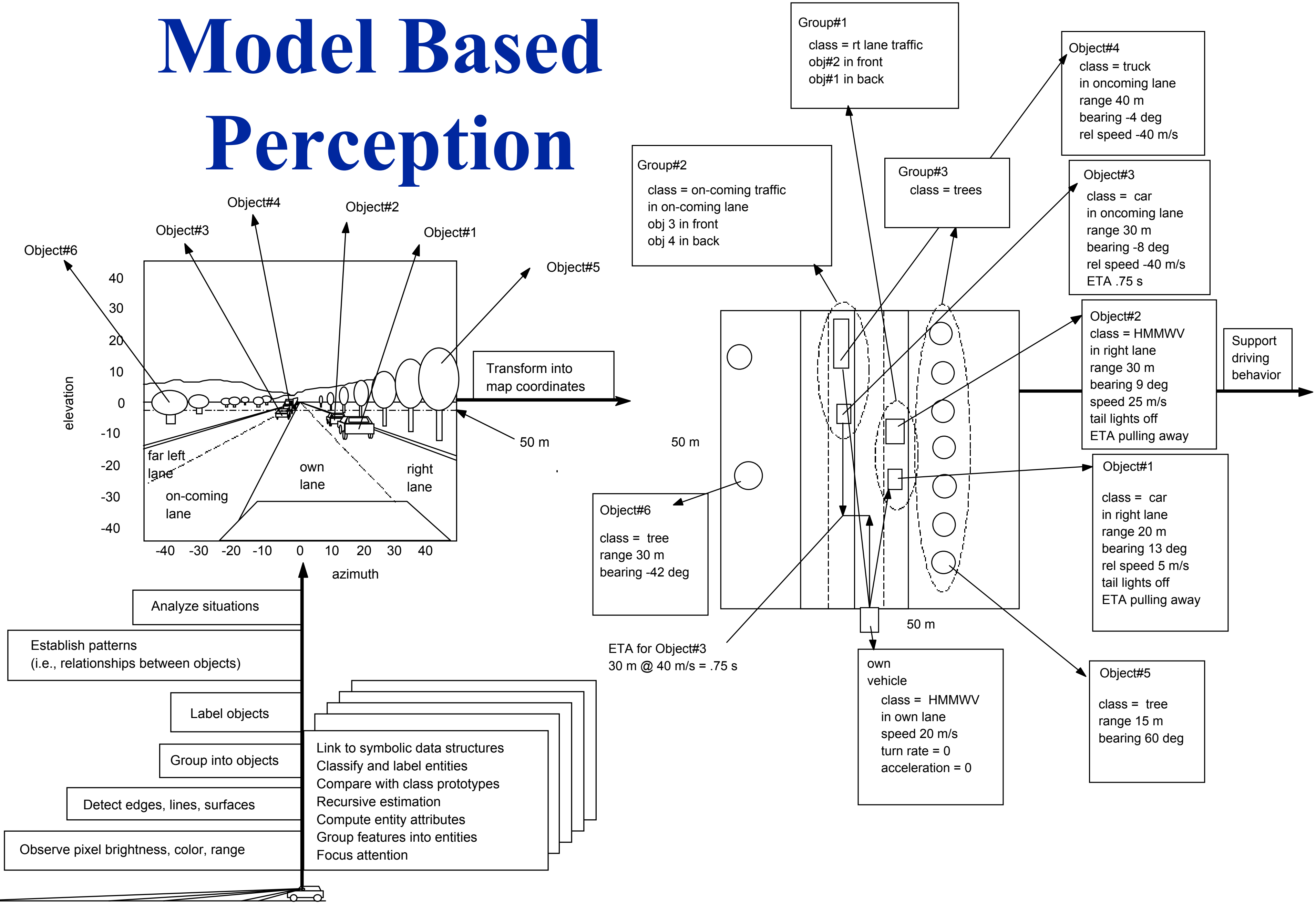


Situation Assessment



Car turning left (position, velocity)
Oncoming cars (position, velocity)
Traffic signals (stop)
Truck on own road (position, velocity)
Own road edges (Old Georgetown Road, heading North)
Intersecting road edges (Democracy Boulevard, to West)
Self in lane 2 (position, velocity) intent (go straight)

Model Based Perception



New Perception of What is Possible

Autonomous ground vehicles with human level performance are achievable within the FCS time frame

Useable autonomous driving could be achieved by:
- 2008 for convoy, leader-follower, mule
- 2010 for smoke, point-man, indirect fires, scout

Near human level performance could be achieved by:
- 2015 for driving (on-road and off-road)
- 2020 for tactical behaviors

Performance superior to humans in all areas by 2025

Why now?

We understand how to deal with complexity

- Hierarchical decomposition in time and space
- Multi-resolutional representations
- Multiple representations
  - Iconic: Signals, Images, Maps
  - Symbolic: Entities, Events
  - Relationships: Pointers, Classes
- 4D/RCS architecture validated by Demo III

We understanding how to acquire and use knowledge

- Model-based perception
- Model-based behavior

We understand how to make decisions

- Value-driven control

Summary

4D/RCS provides for:

Many sensors, a rich world model
High speed sensory processing
Deliberative and reactive behavior
Engineering tools and methodology

4D/RCS is a reference model architecture that is:

Open
Portable
Reliable
Intelligent
Mature

Conclusions

1. Useful autonomous on-road and off-road driving is feasible within this decade.

2. Human level performance in autonomous on-road and off-road driving is feasible by 2015

3. Future Combat System will provide the rational and funding to build intelligent vehicle systems

Conclusions

1. 4D/RCS shares many concepts with SOAR

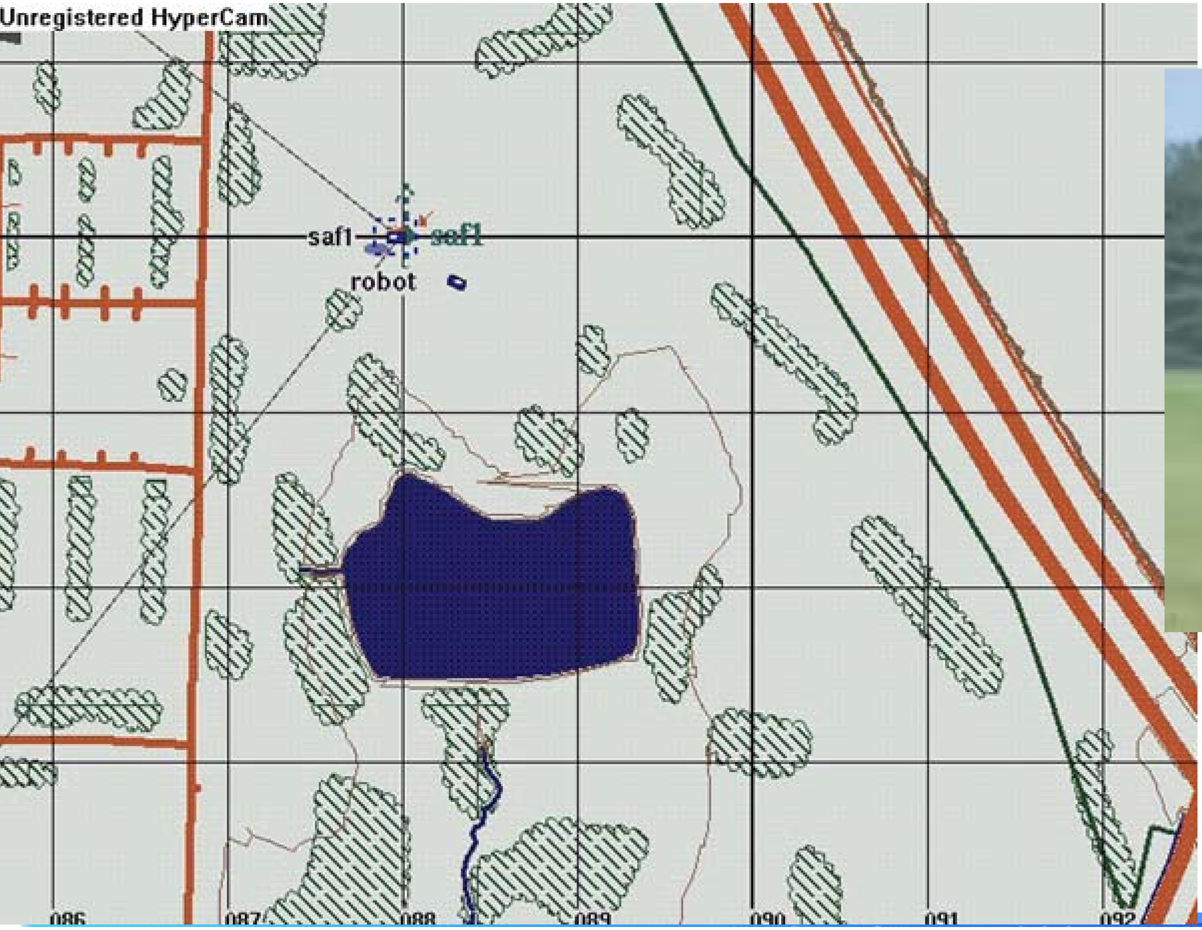
2. 4D/RCS is complementary to SOAR

3. 4D/RCS and SOAR should collaborate on tactical behaviors for FCS

Back Up Slides

Real/Virtual Environment for Off-Road Driving

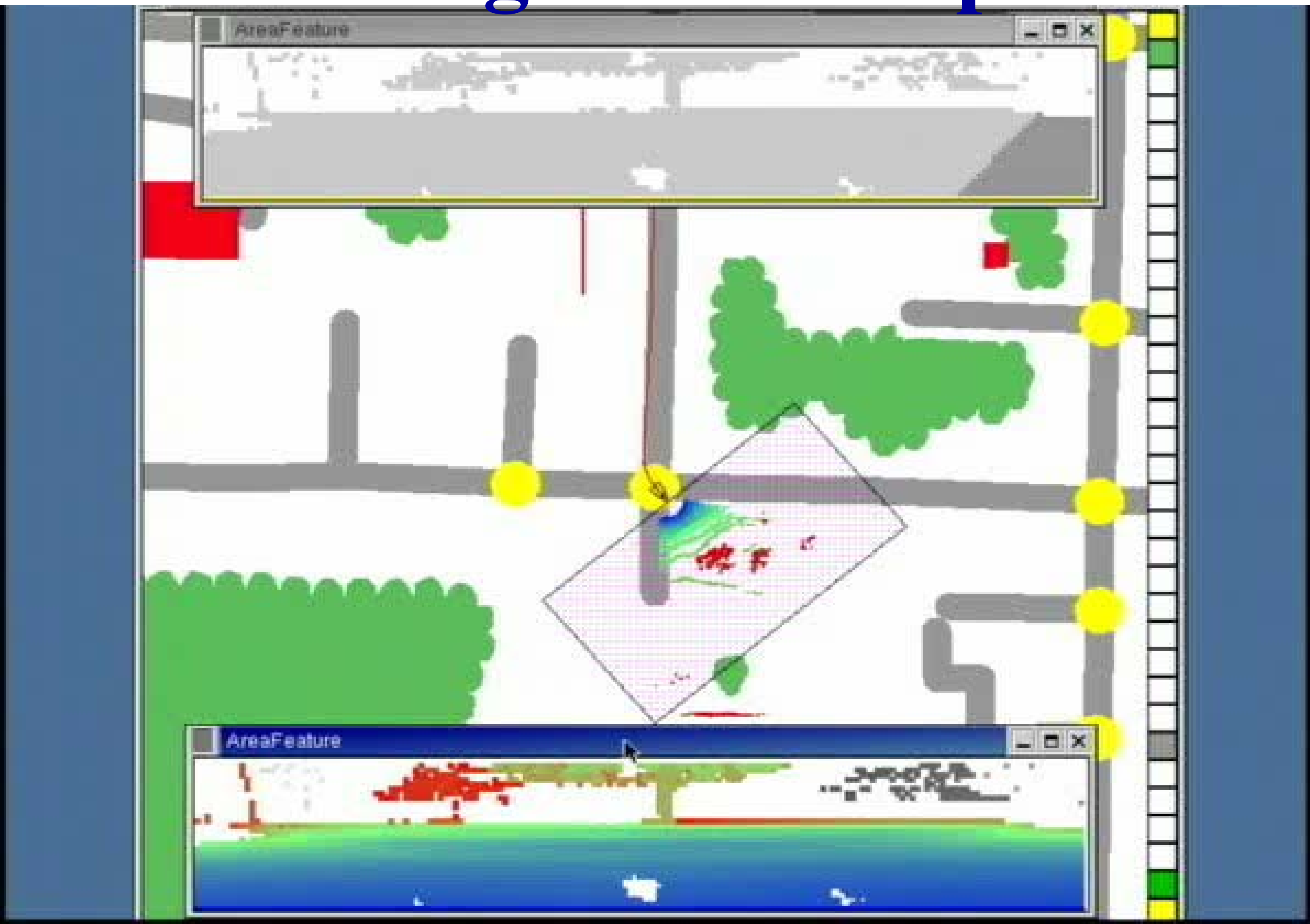
Steve Balakirsky



Follower Vehicle in OneSAF Virtual World

NIST HMMV in Real World

Prediction of road position in the image from map data



Job Assignor

1. Accepts task commands from an executor in a higher level BG processes, or from an operator

2. Decomposes each task into a set of jobs for subordinate BG processes

3. Transforms job into coordinate frame of reference appropriate for subordinate BG process

4. Allocates resources to subordinate BG processes

Schedulers

1. Accept jobs from Job Assignor

2. Decompose each job into a tentative sequence of planned actions and desired states to accomplish the assigned job

There is a scheduler for each subordinate BG process

The scheduler for each subordinate BG process coordinates its job plan with other schedulers

Plan Selector

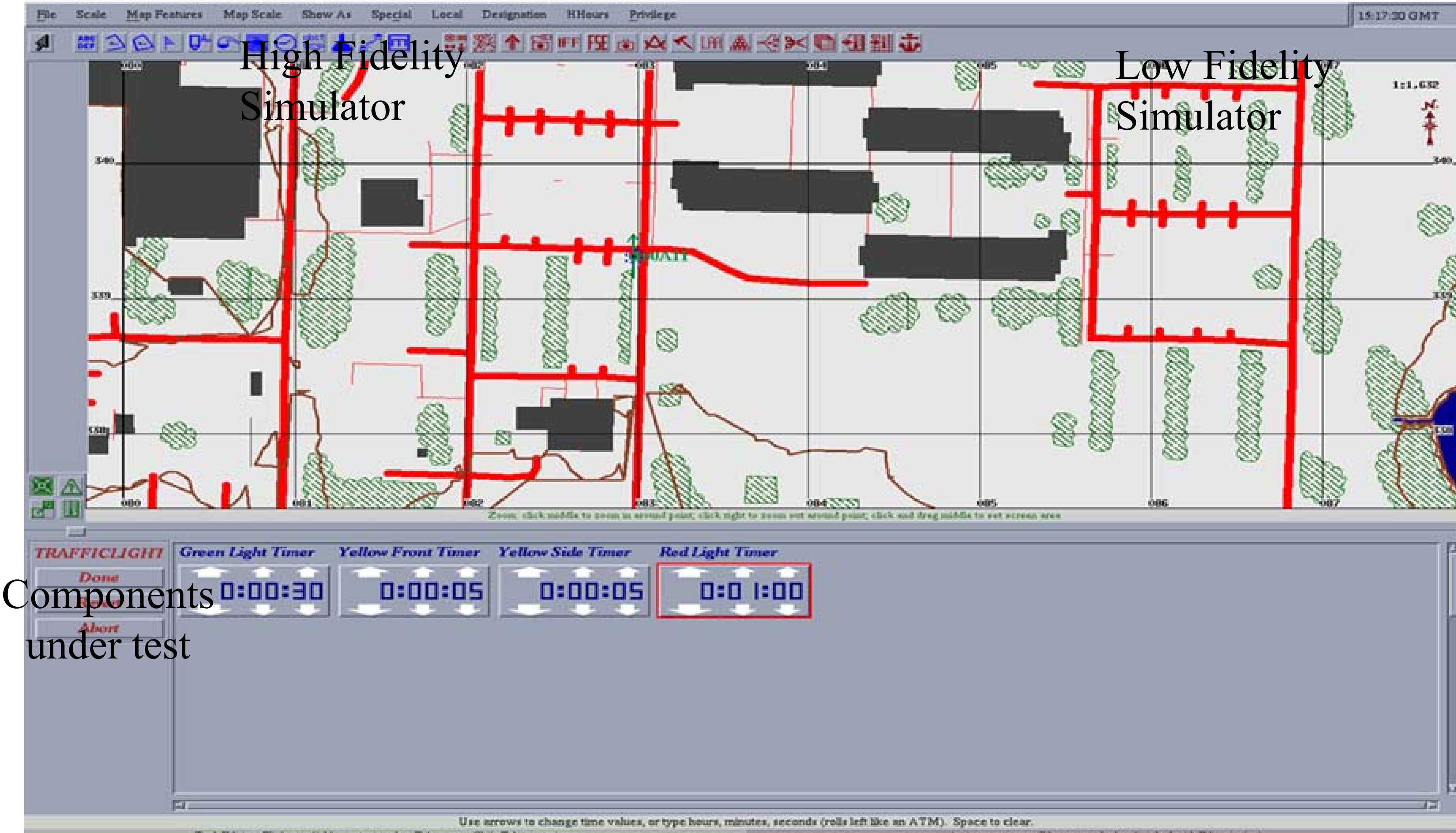
1. Submits tentative plans to Value Judgment for cost/benefit/risk evaluation

2. Stores the tentative plan with the best evaluation to date

3. At the end of each planning cycle, inserts the best job plan into a plan buffer for each Executor

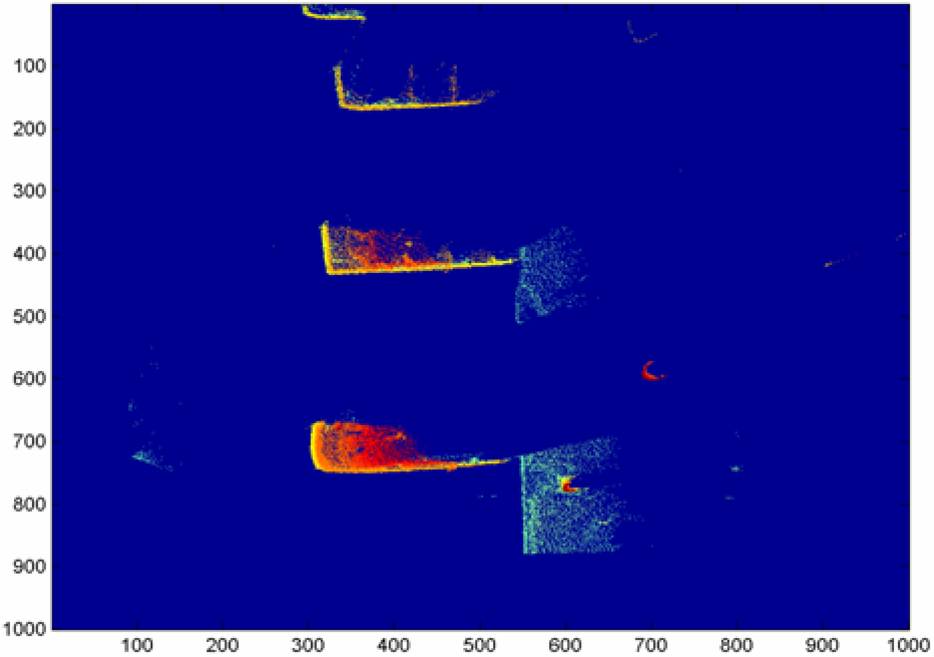
Real/Virtual Environment

Architecture



Low-Fidelity Simulator

- Based on OTBSaf
  - Maintains all of OTBSaf's behavior generation modules (e.g. convoy)
- New simulation capabilities added:
  - Static and dynamic traffic signals
  - NML channels
    - Traffic channel
    - Entity channel
    - Terrain feature channel
    - Master clock channel (under development)
  - High-level mysql editor (under development)



Advances Needed in LADAR

Most LADARs developed for ATR, air reconnaissance,
or construction site metrology

Need LADAR for driving on the ground
10 frames/sec, range and color, 1 - 200 meters

Need foveal / peripheral / wrap-around imaging, pan / tilt, neck

Need saccades, stabilization, tracking

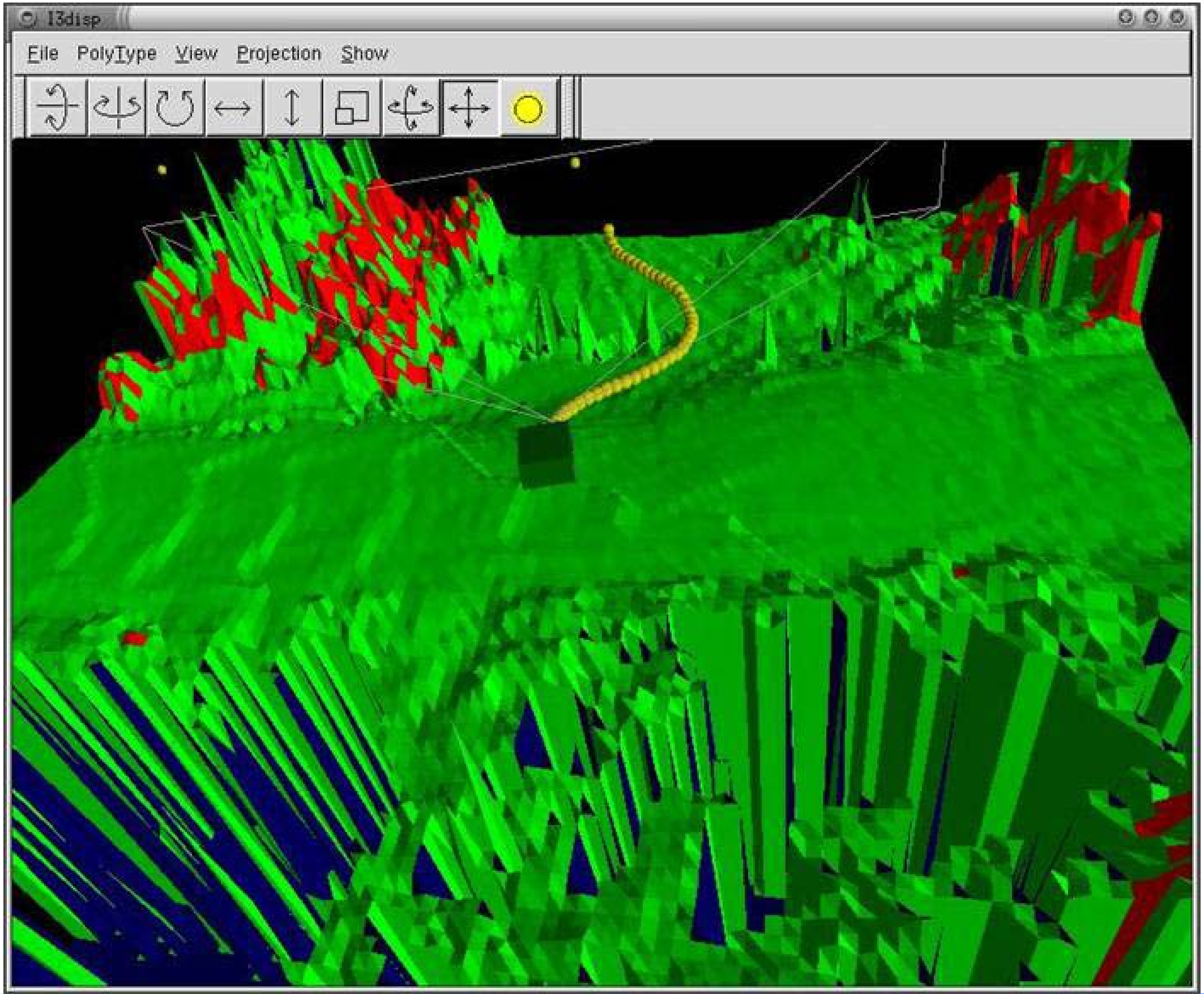
Need inexpensive, rugged systems

Need penetration of foliage, smoke, dust

Need to determine load bearing properties of ground
under tall grass, weeds, marsh, mud, snow, and water

BAA issued, 15 proposals evaluated, 4 funded

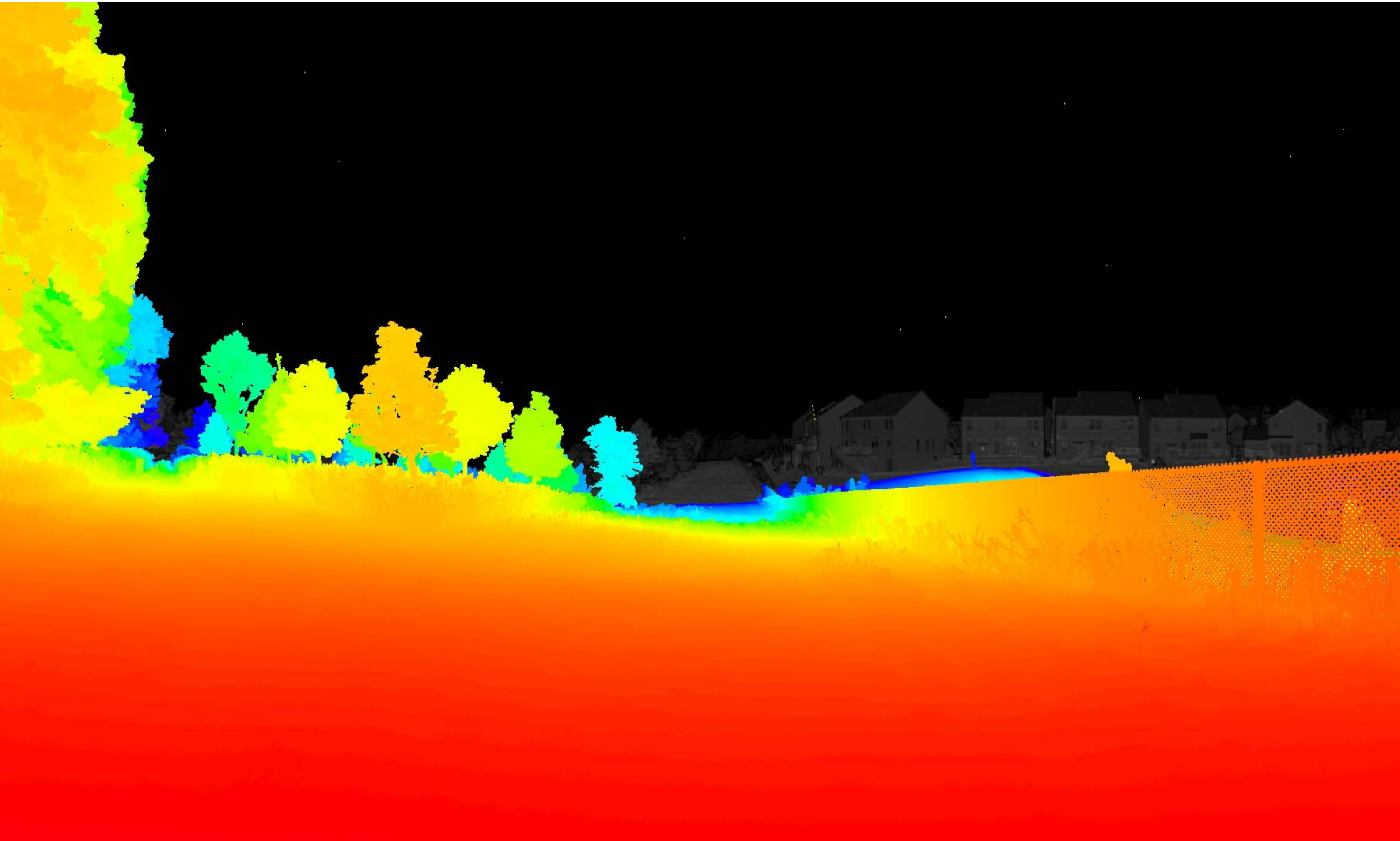
Replanning to Avoid Obstacle During Turn



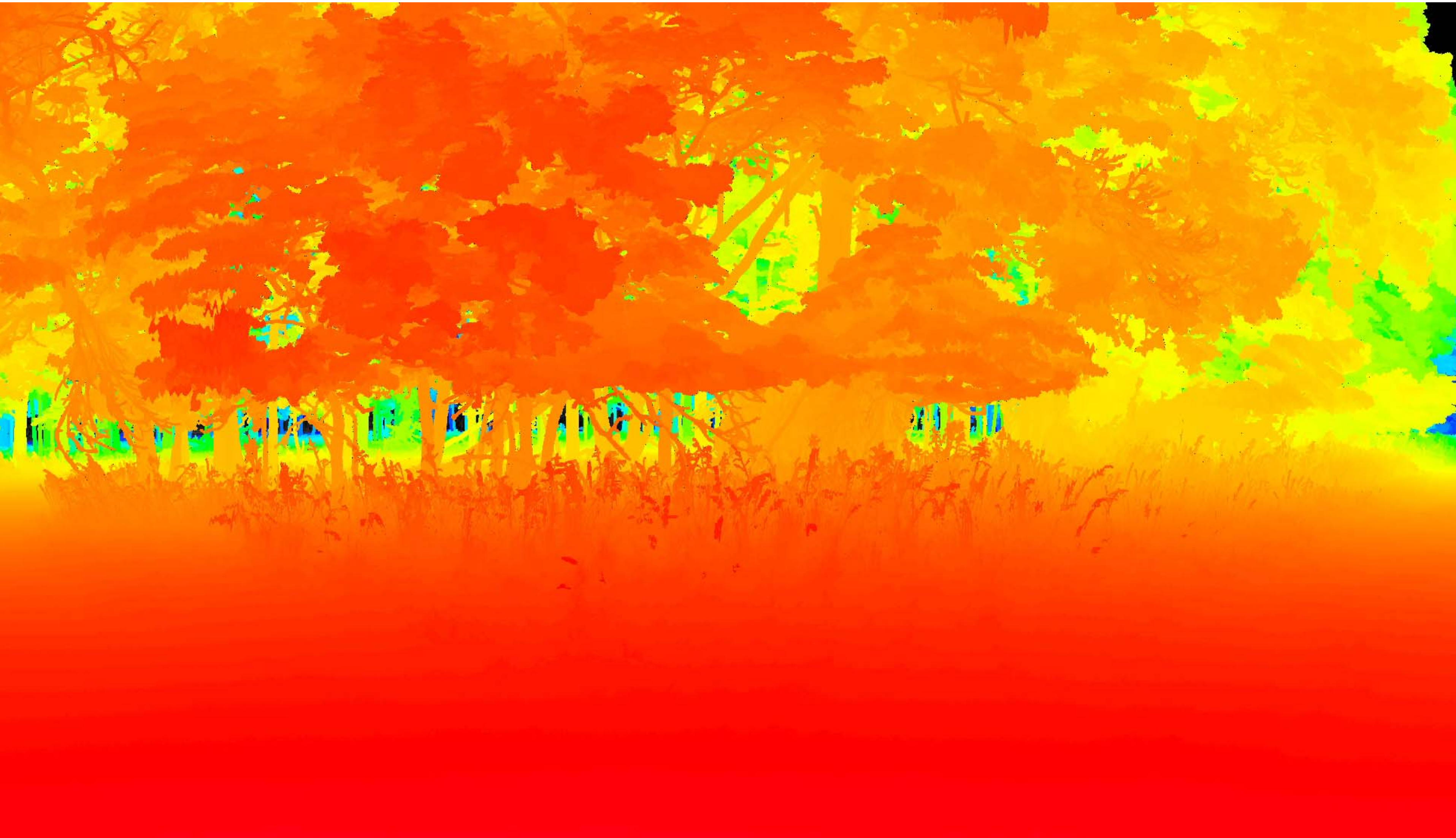
Next Generation LADAR

Grassy Knoll Ground Level View
Color Camera Mosaic

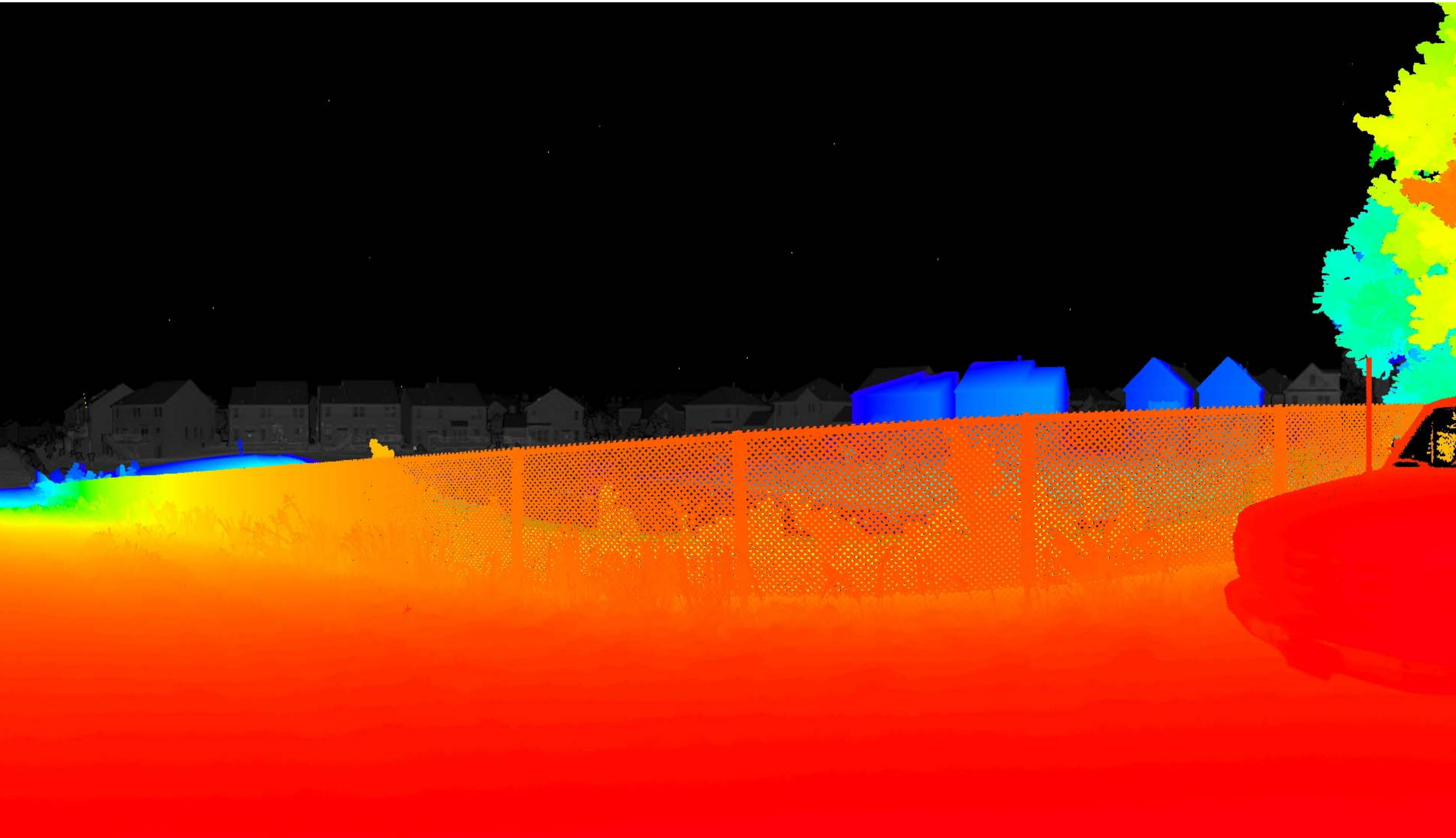
Grassy Knoll Ground Level View
LADAR range image



Look Left into Woods from Grassy Knoll LADAR range image



Look Right at Fence from Grassy Knoll
LADAR range image



Grassy Knoll Overhead Perspective

Five Level SP Hierarchy

1) Pixel entities

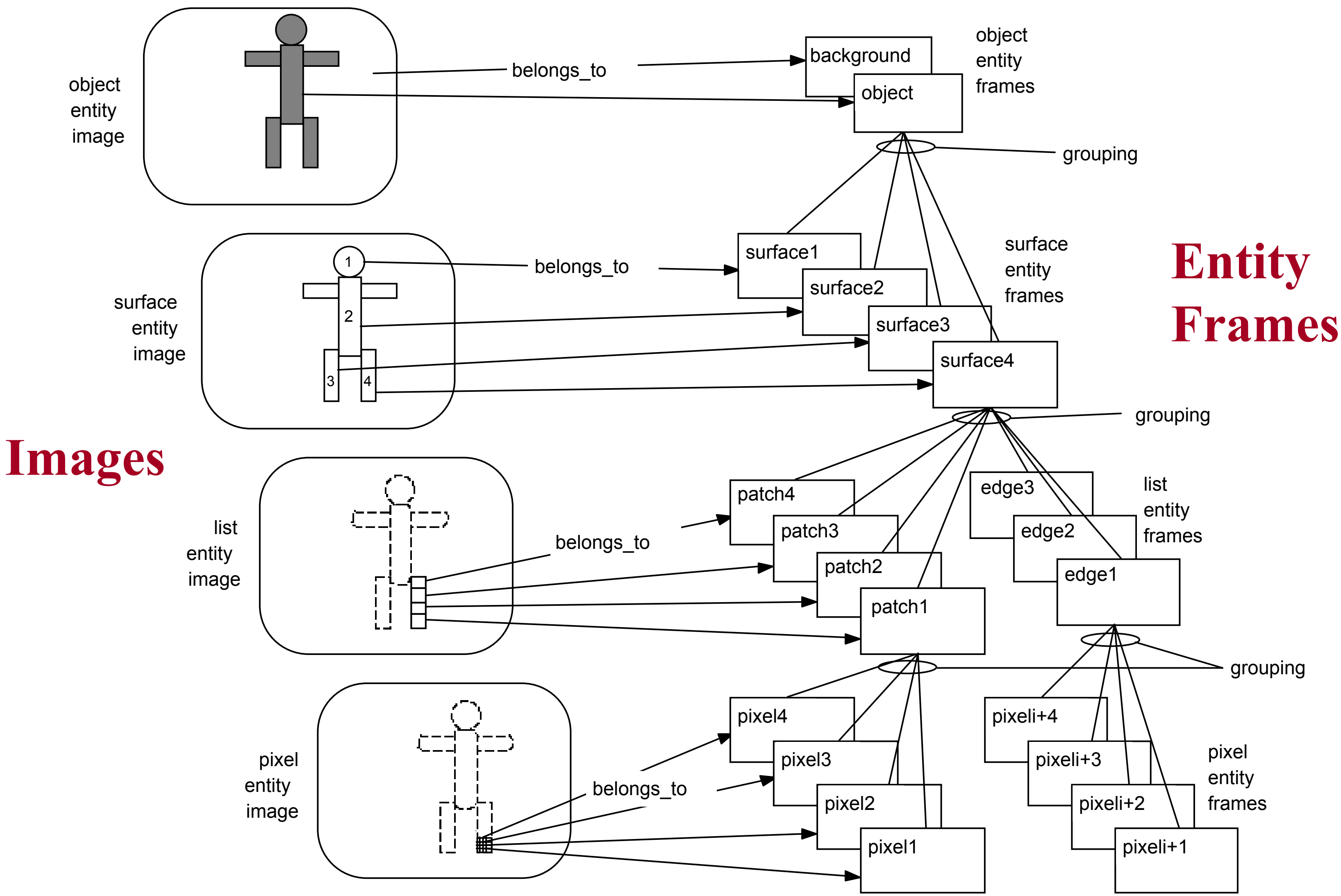
2) List entities (edges, vertices, surface patches)

3) Surface entities (boundaries, surfaces)

4) Object entities

5) Group entities

Perception Hierarchy



At each level in the SP hierarchy

1) Focus of attention (or windowing)

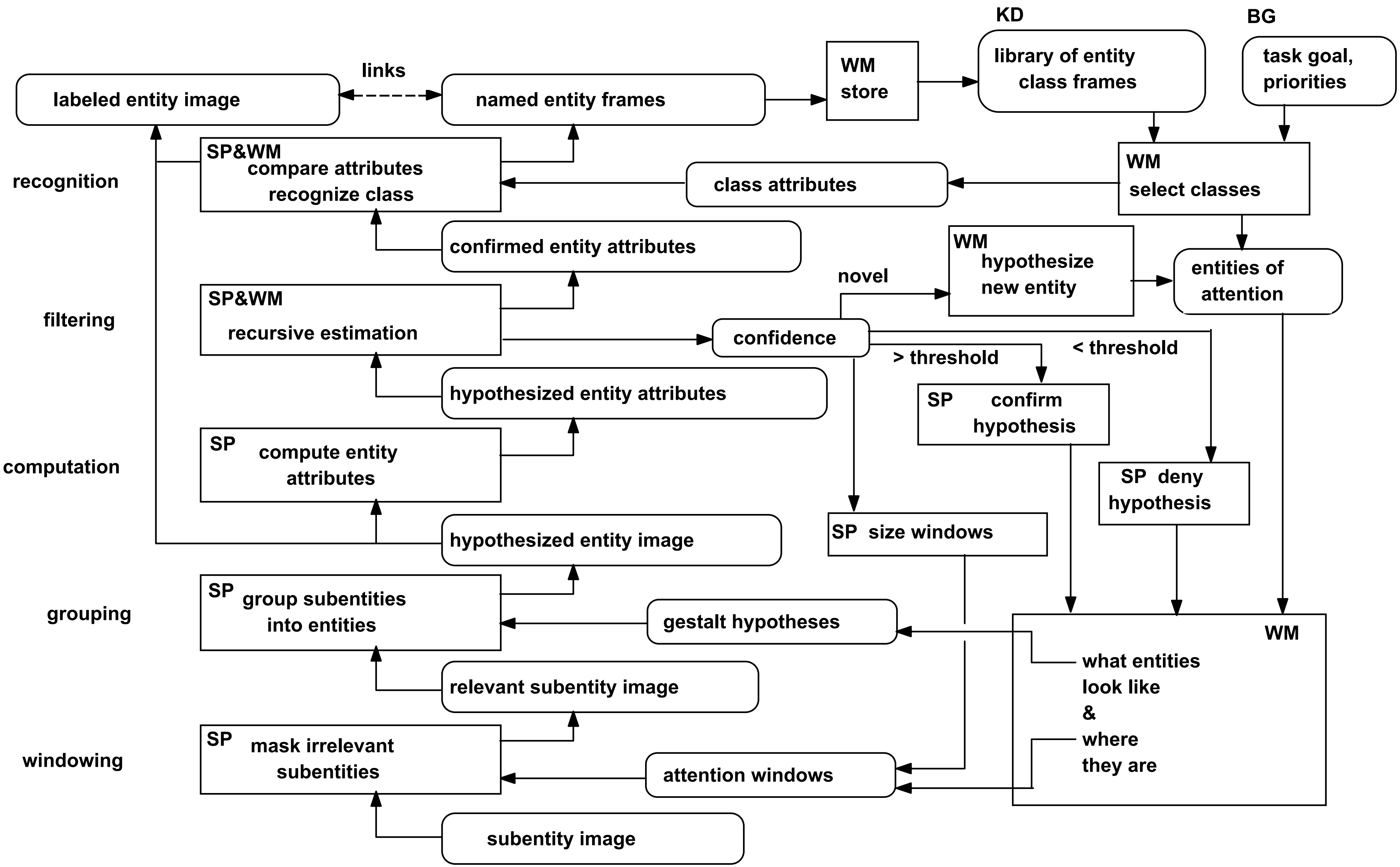
2) Grouping / Segmentation

3) Compute group attributes

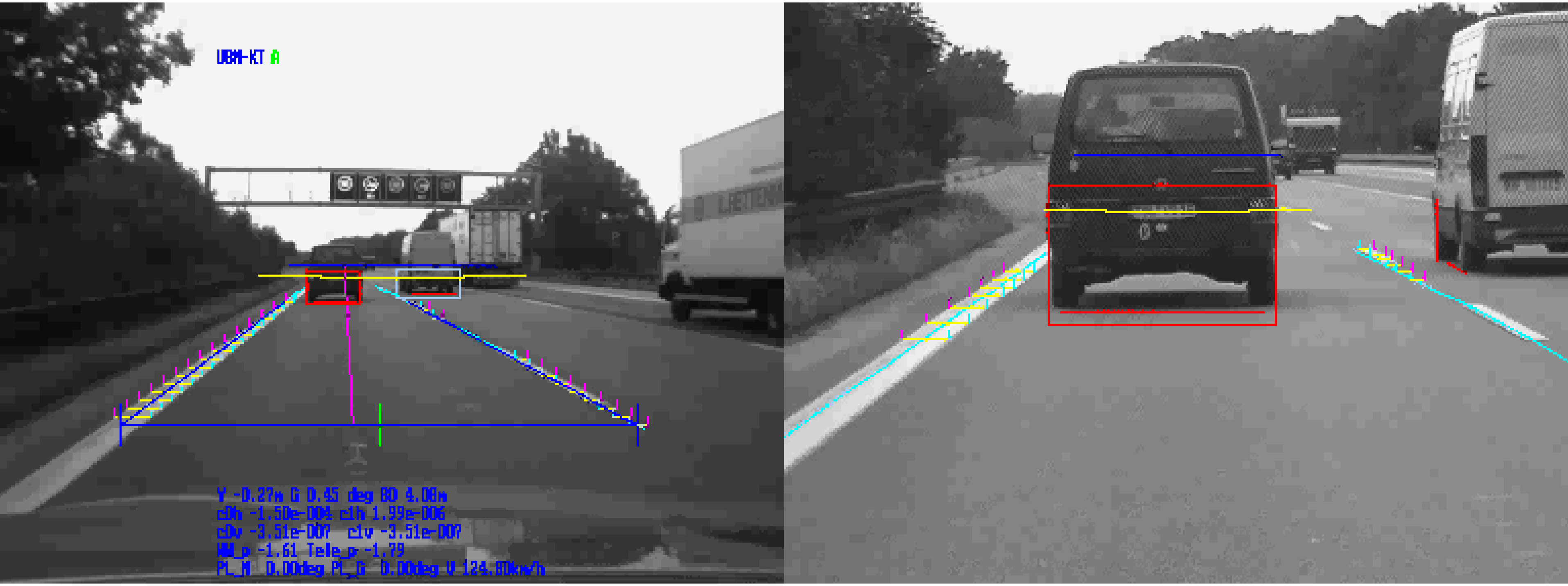
4) Filter (recursive estimation)

5) Classification, Recognition

Image Processing



The 4-D in 4D/RCS
Recursive Estimation in the Image

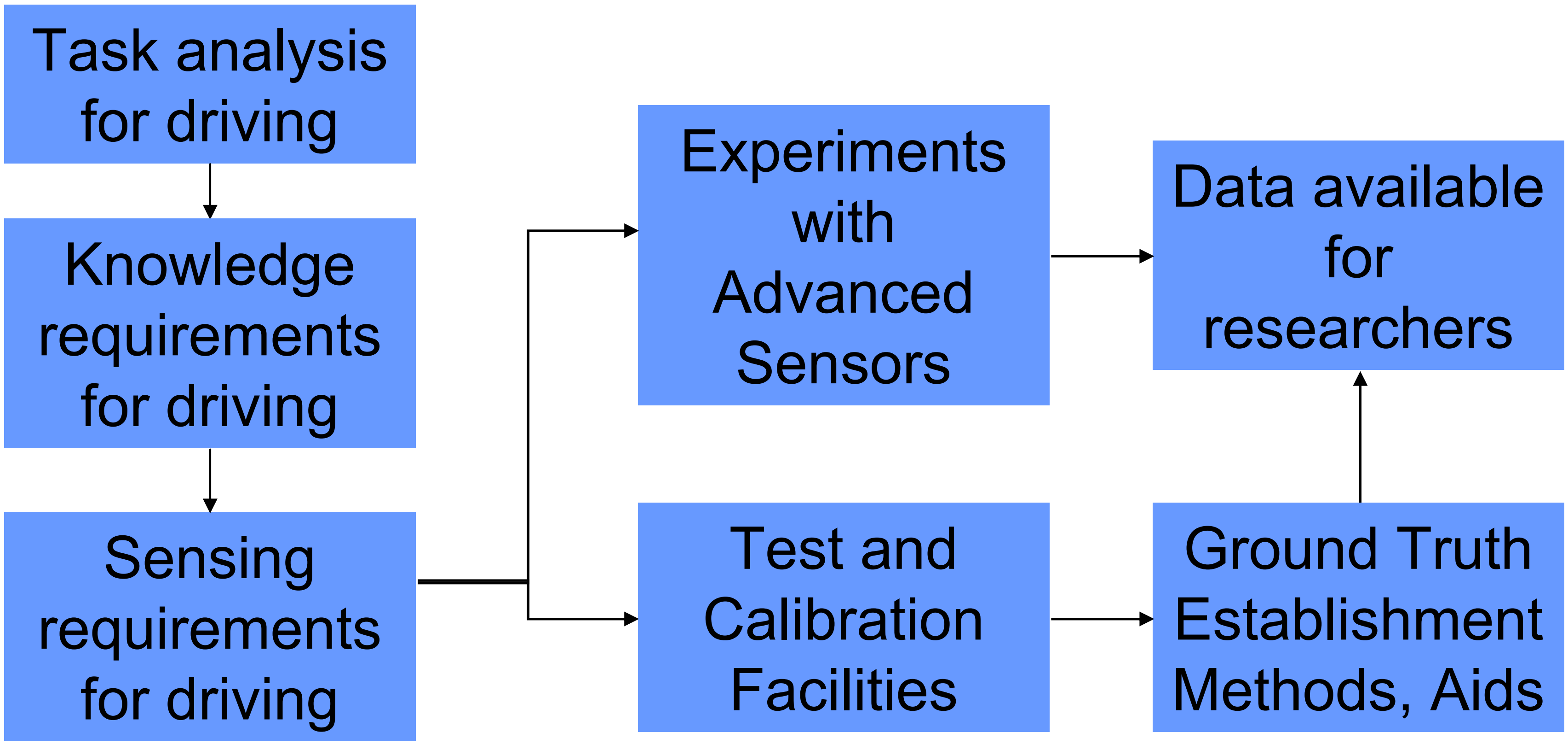


Developed at Universitat der Bundeswehr, Munich
by Ernst Dickmanns et al

Task Analysis for Driving

- Understanding of the problem scope and challenges grows through several investigations
  - Scoping:
    - Task analysis is going beyond behaviors
    - Define requirements for world model, knowledge base, perception, and sensing imposed by behaviors
  - e.g., perceive and plan in a world filled with moving objects
  - compare approaches for representation and planning
  - inject *a priori* data into system's world model

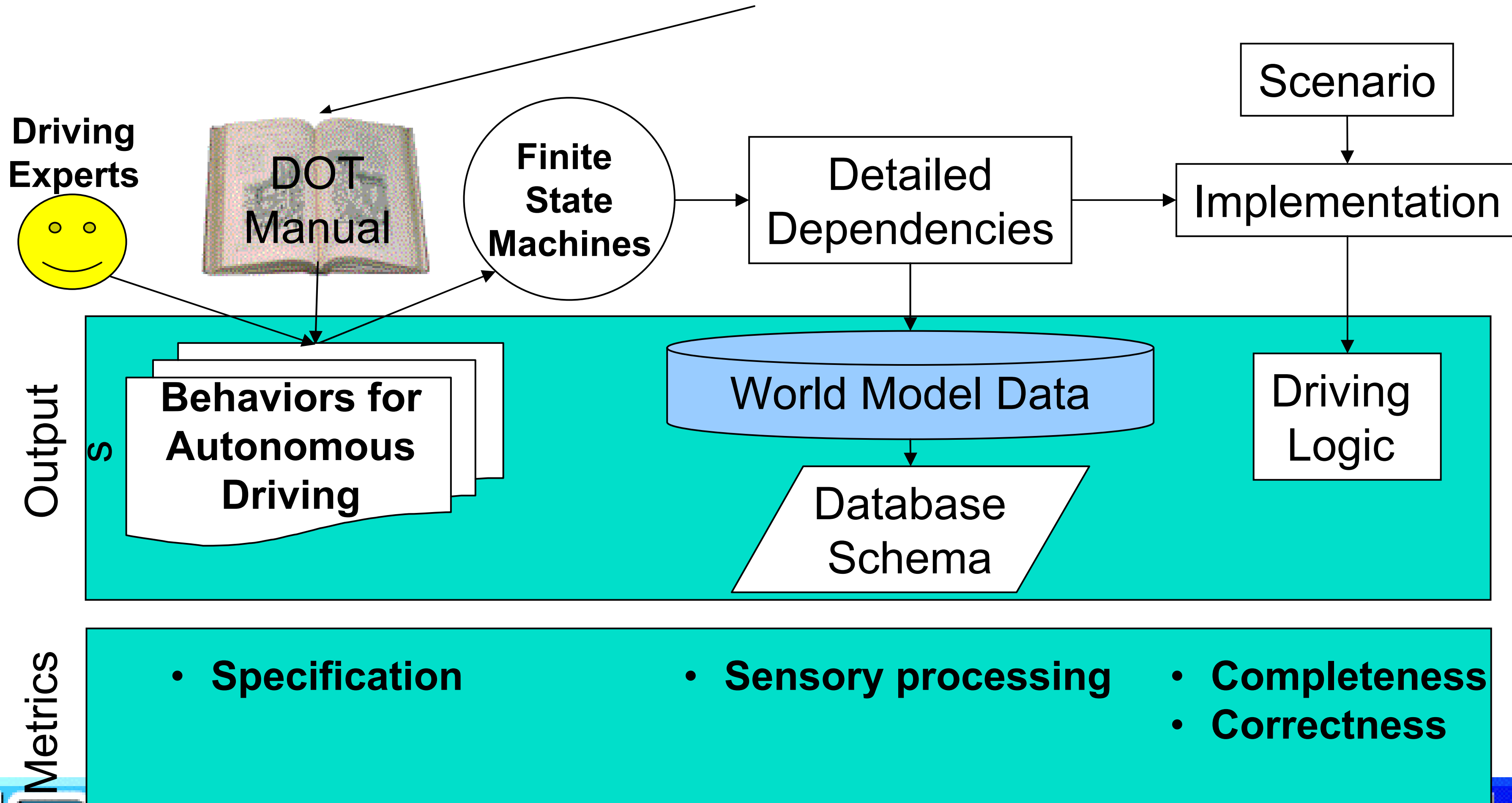
Technology Analysis



Capabilities of advanced sensors currently being used to help establish ground truth will be available for future on-board perception systems

Driving Task Analysis

Goal: Quantify the complexity of on-road autonomous driving tasks, in terms of design, execution, and resource complexity



Driving Task Analysis

1) Analyze Autonomous Driving Tasks & Develop a Task Architecture

2) From Task FSAs, define dependencies on World Model Situations and Value Judgments

3) From World Model Situations, define Entities, Events, Attributes and their Resolutions and Tolerances

4) From these World Model Data, define Requirements for Perception

Driving Task Analysis

Next Steps - Provide the following Deliverables:

1) Generate Task Analysis Document for Autonomous Driving

2) Develop the Task Dependencies to define the required World Model Situations, Entities, Events, and Attributes along with their Resolutions and Tolerances

3) Interact with other MARS contractors (PercepTek) to refine World Model Data Attributes and Schemas

4) Use this Definition of World Model Data as the Requirements for Perception for Autonomous Driving

5) Derive an initial Set of Performance Metrics for Perception, Representation, and Planning Algorithms

A Demonstration: Automated Driving to DARPA

1. **Drive around NIST grounds**

   curbs, moving obstacles, two lane roads, same lane and on-coming traffic, intersections, traffic signs, pedestrians, deer

2. **Drive to NIST-North and park**

   intersection with 4 lane road, traffic signals, cross traffic, find parking space

3. **Drive to Quince Orchard Plaza and park**

   turn into intersection, multiple traffic signals, left turn across multiple lanes, negotiate traffic in parking lot, choice of various routes

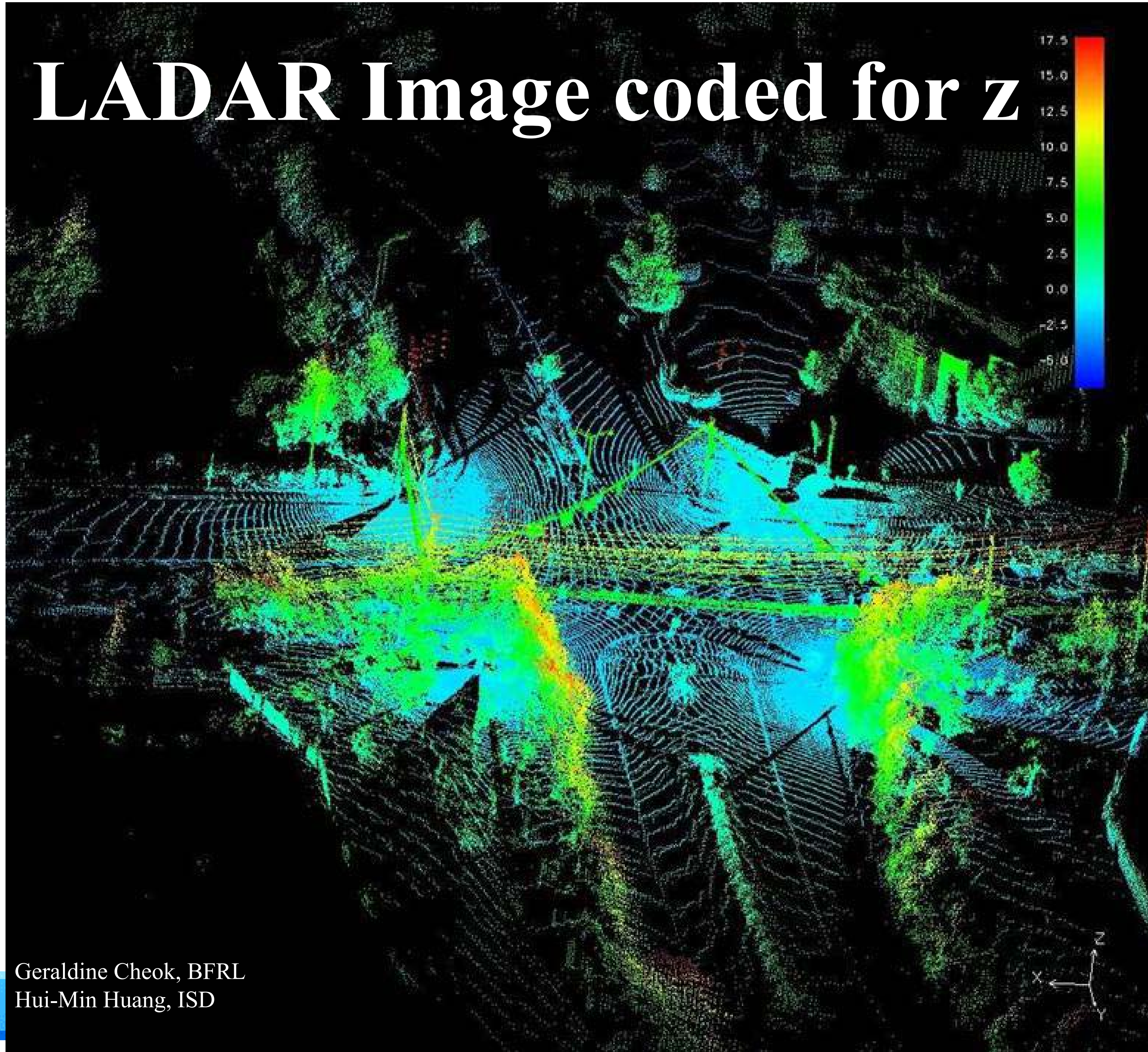
4. **Drive to Montgomery Mall and park**

   enter and exit freeway, high speed traffic, merge, lane change, passing, road signs, construction barriers, parking garage

5. **Drive to DARPA and park**

   multiple freeways, interchanges, city driving, dense traffic, difficult parking

LADAR Image coded for z



Geraldine Cheok, BFRL
Hui-Min Huang, ISD