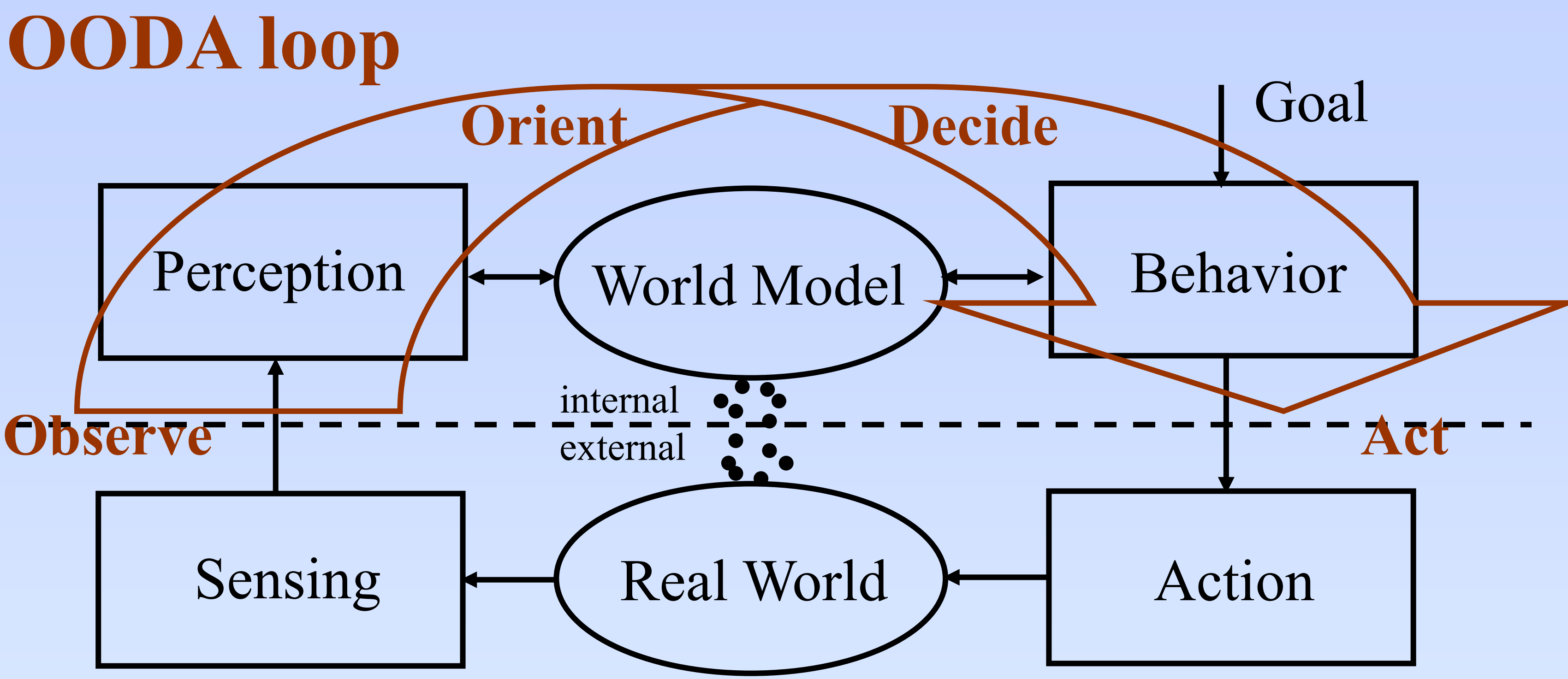
# Intelligent Control and Tactical Behaviors for Unmanned Ground Vehicles

**James Albus, Senior NIST Fellow (Retired)**
Intelligent Systems Division
National Institute of Standards and Technology

**Senior Robotics Scientist**
**Robotic Technology Inc.**

# Basic Intelligent System

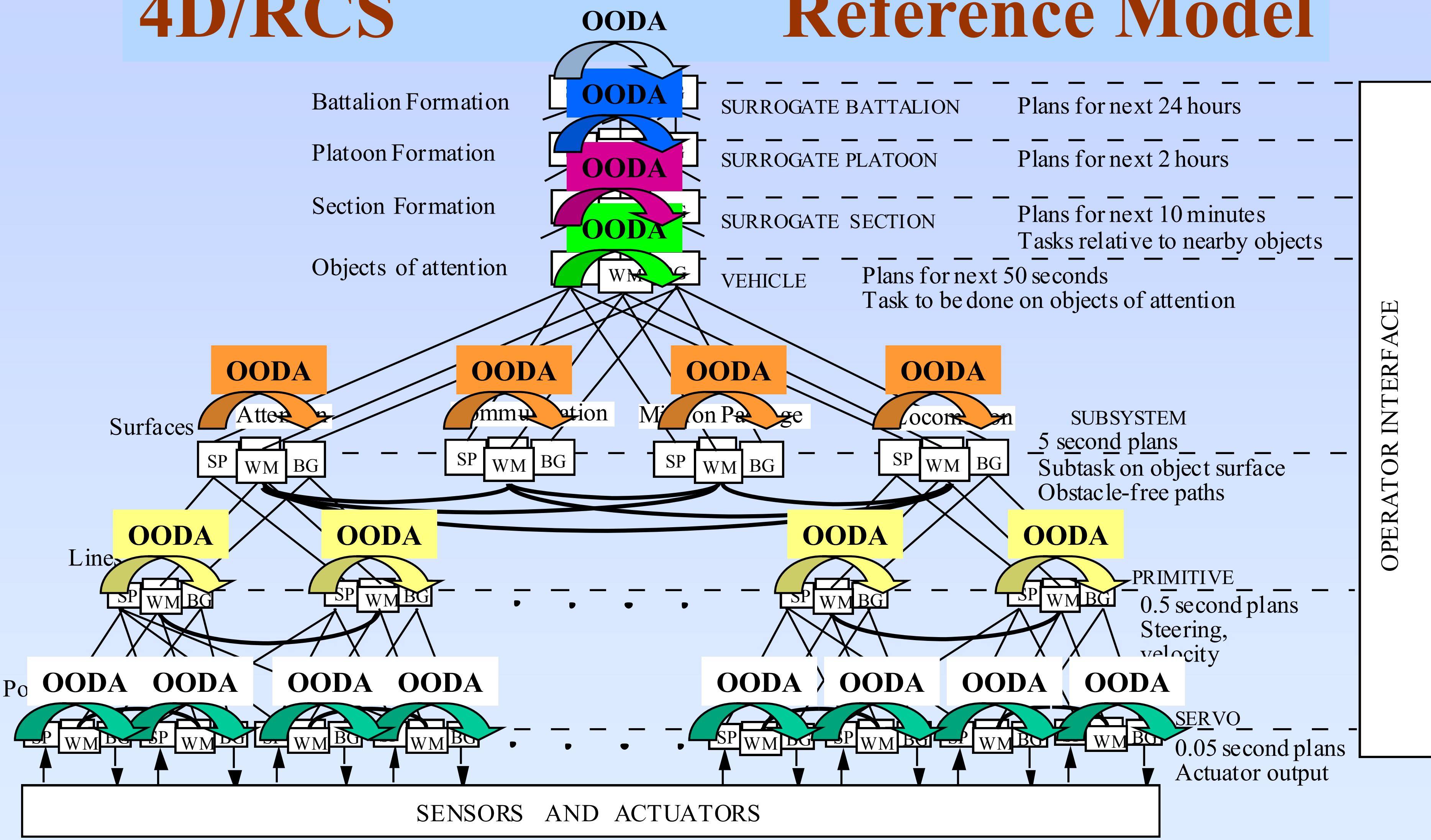



**Perception establishes correspondence between internal world model and external real world**

**Behavior uses world model to generate action to achieve goals**

# Intelligent System Architecture

## 4D/RCS Reference Model



# 4D/RCS Reference Model Architecture for Unmanned Vehicle Systems

**Adopted by GDRS for FCS Autonomous Navigation System**
**Adopted by TARDEC for Vetronics Technology Integration**

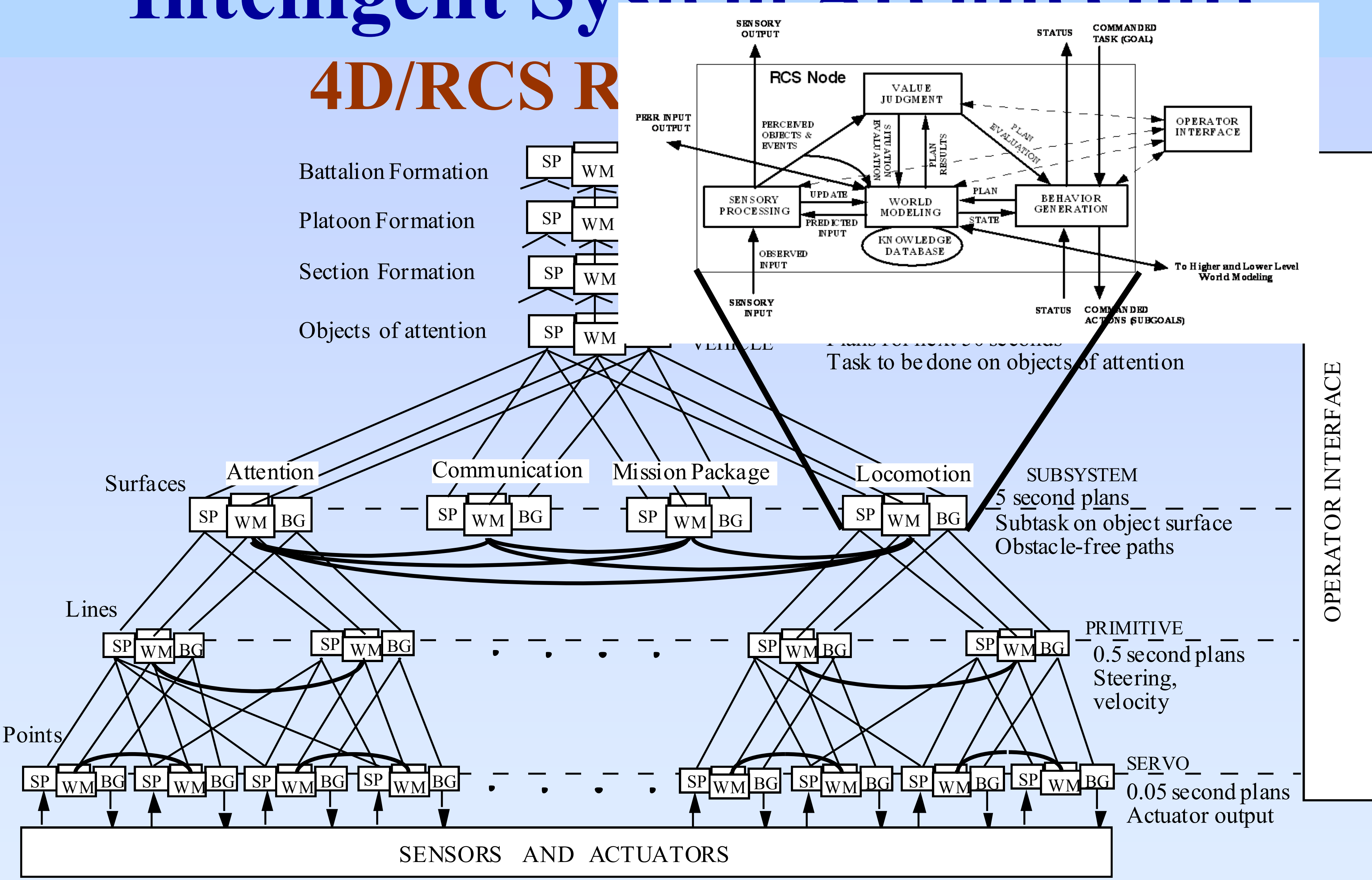
- **Hierarchical structure of goals and commands**
- **Representation of the world at many levels**
- **Planning, replanning, and reacting at many levels**
- **Integration of many sensors**
  **stereo CCD & FLIR, LADAR, radar, inertial, acoustic, GPS, internal**

# Attributes of 4D/RCS

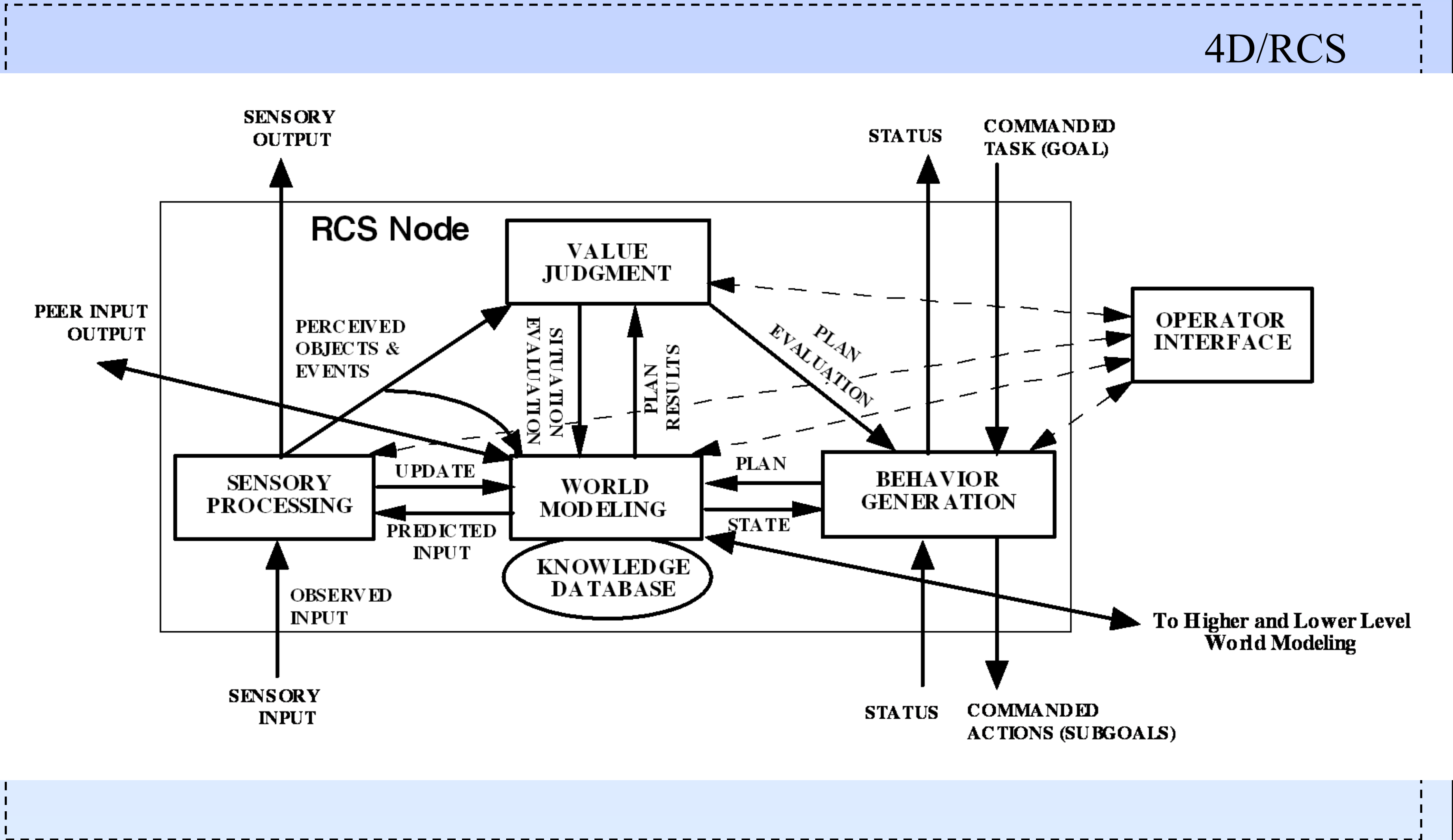
- Combines AI with control theory
- Hierarchical representation of tasks, space, & time
- Combines deliberative with reactive at many levels
- Depends strongly on sensing and perception
- Supports a rich dynamic world model at many levels
- Integrates prior knowledge with current observations
- Models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

# Intelligent System Architecture

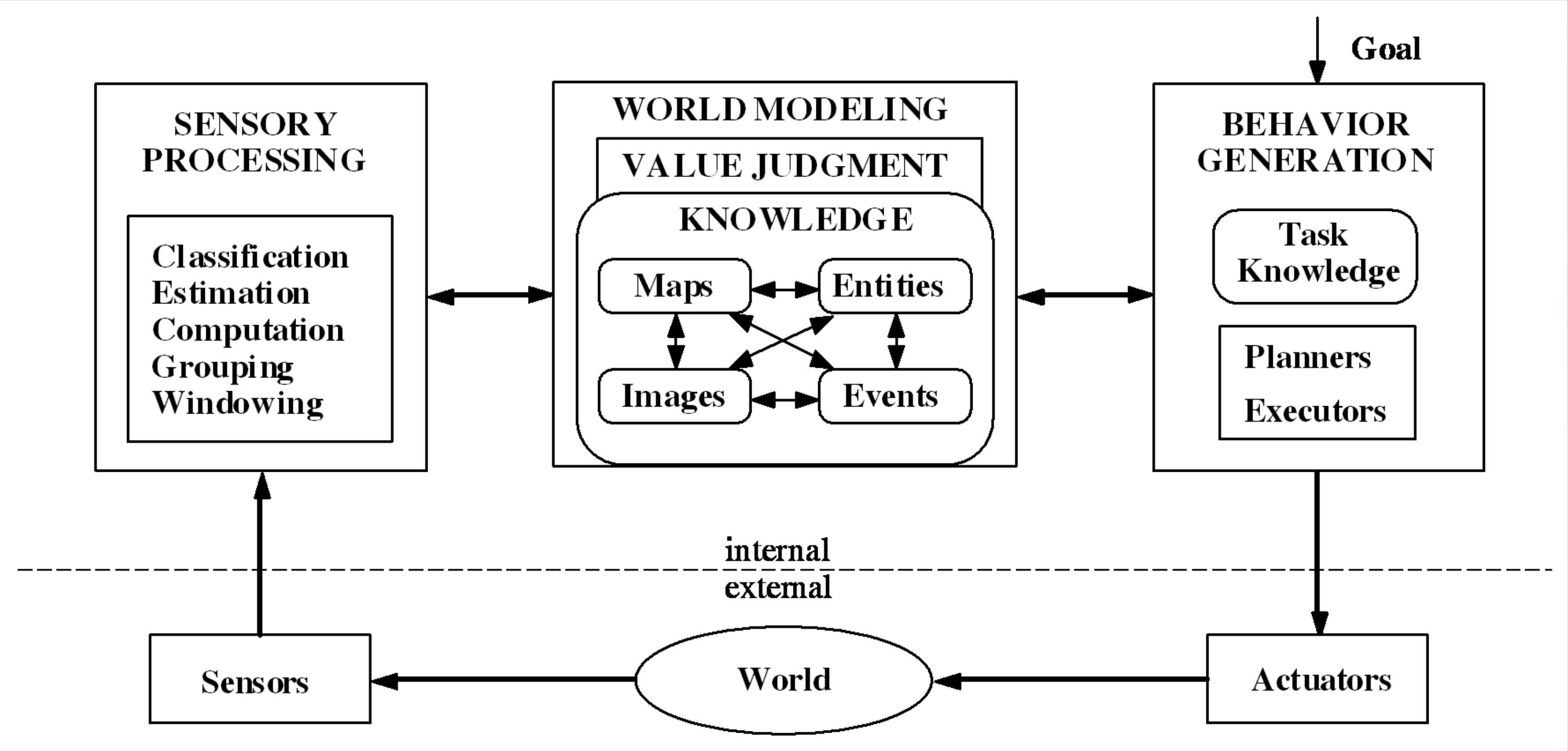
## 4D/RCS R



# A 4D/RCS Computational Node



# Knowledge is Central




4

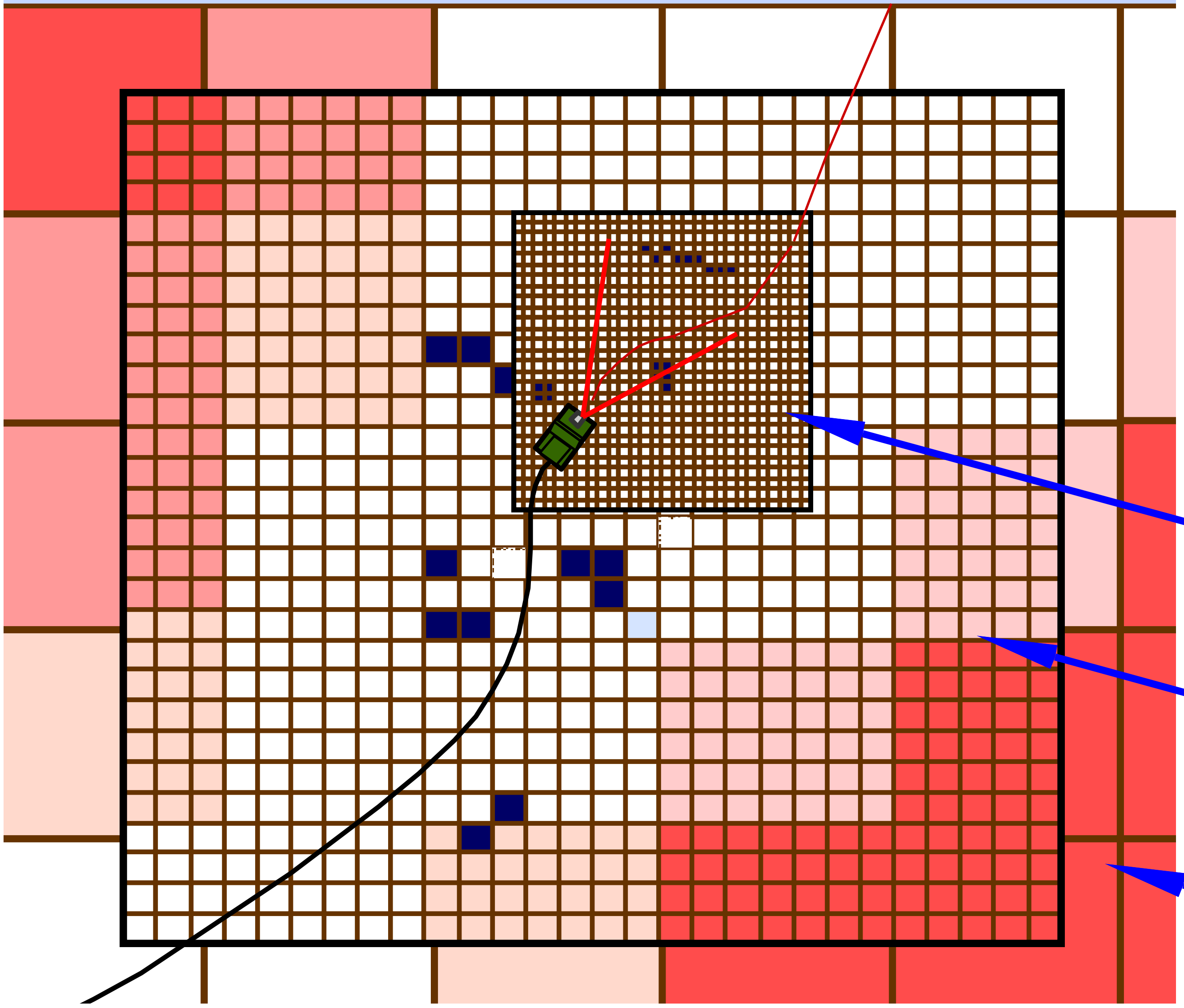
# Forms of Representation

**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

# MULTI-RESOLUTION MAPS



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

0.4 m grid
50 m wide

4 m grid
500 m wide

30 m grid
Terrain map

# Sensory Processing

*Classification

Compare group attributes with class prototype

Set pointers that define class membership

Computation of Group Attributes

e.g., size, shape, texture, motion

Recursive estimation of group attributes

*Segmentation and Grouping

Segment pixels that meet grouping criteria

Set pointers that define grouping relationships

Focus Attention

Direct sensors to region of interest

Window and track interesting entities and events

# Segmentation & Grouping

Spatial pixel patterns => Entities
Temporal signal patterns => Events

**Fundamental Problems:**
Any segmentation is a hypothesis. Needs confirmation.
2D images are ambiguous in range => infinite # of hypotheses

Segmentation criteria == Gestalt grouping hypotheses
- Proximity in space or time
- Similarity in brightness, color, shape, size, texture, etc.
- Symmetry, Smooth continuation

Bottom-up segmentation of optical images is notoriously poor.
Need to integrate top-down inputs

# Classification

**Fundamental Problem**

Object classification depends on:

1. accurate segmentation and grouping
2. dimensionality of object attribute vector
3. number of pixels on target (> 100)

Optical images are high in resolution, but ambiguous in range.
Therefore, segmentation is hard

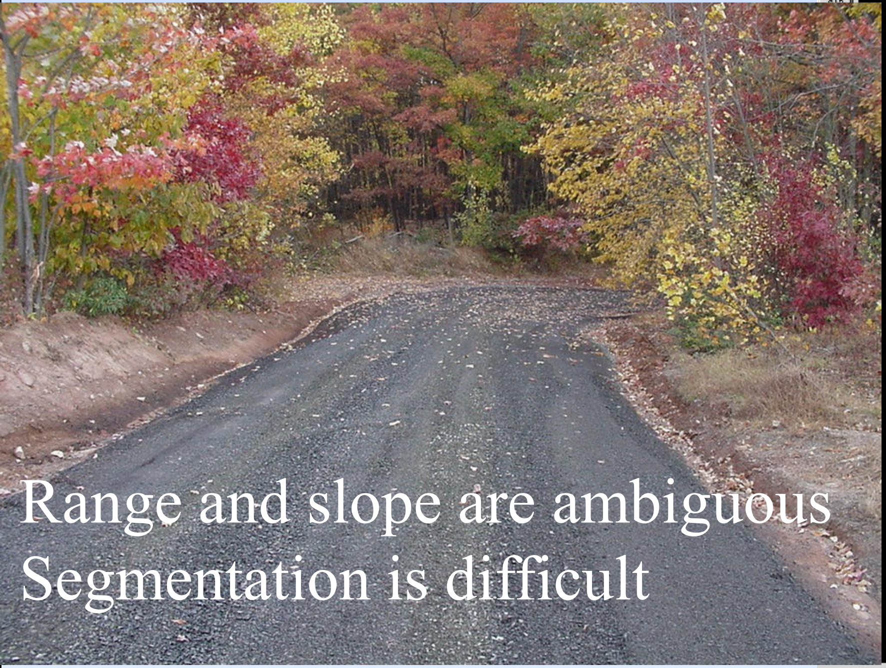
Range images are low in resolution
Therefore, not enough pixels on target

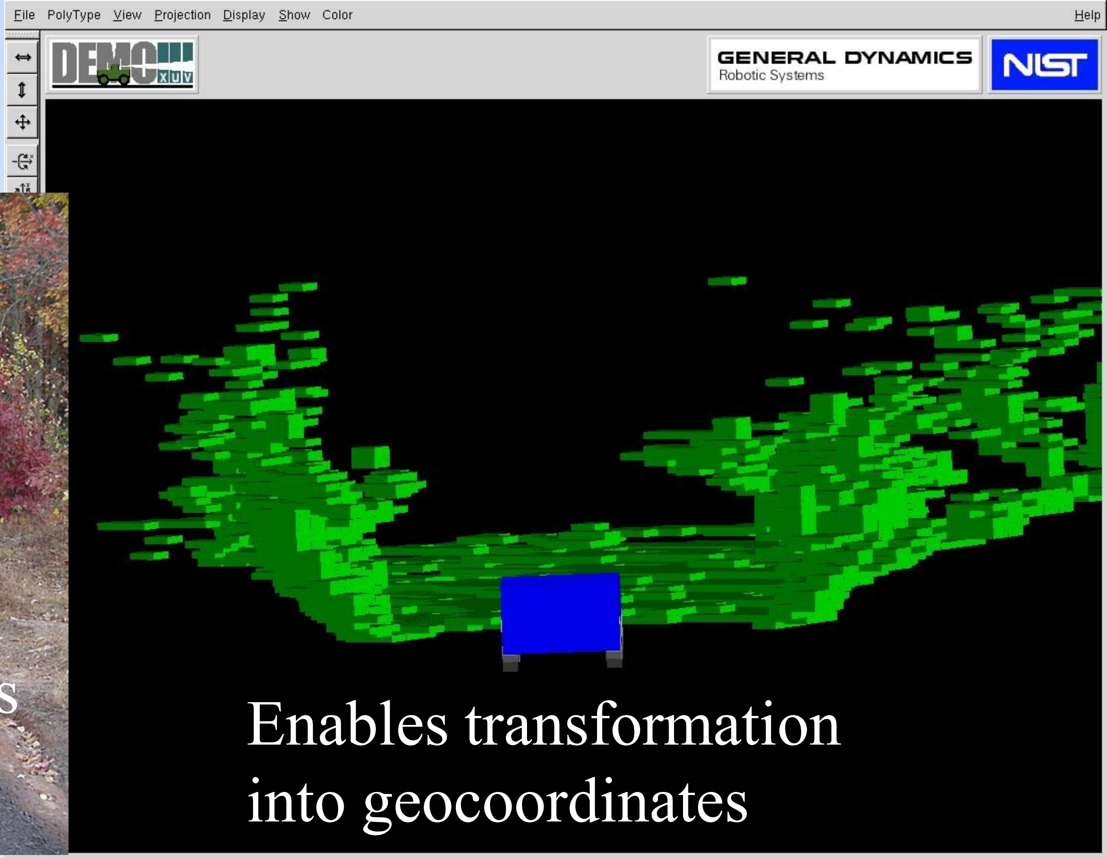
Data fusion helps
High-level context helps more

# LADAR is a Critical Break-Through

## 3D Range Image

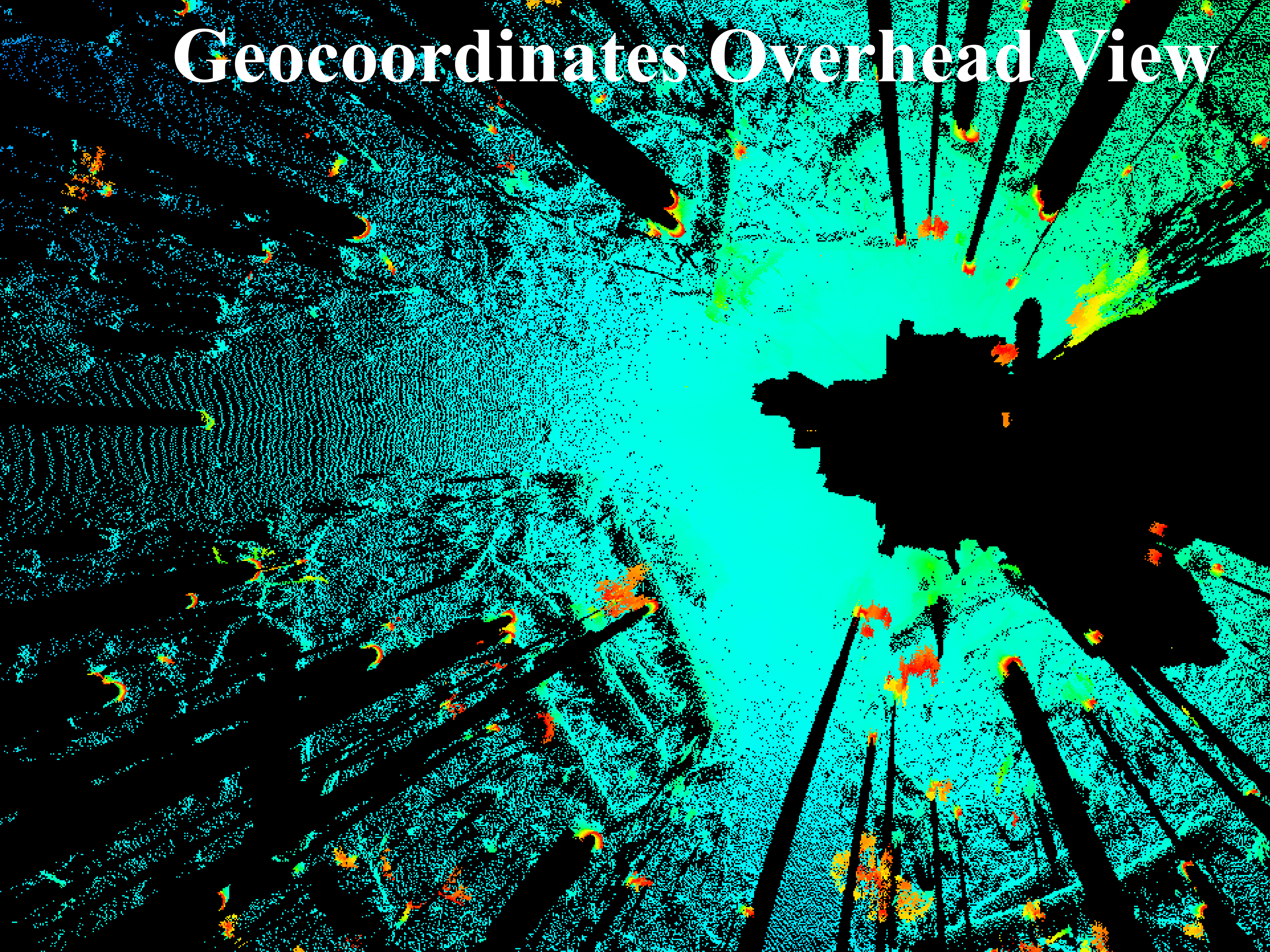
## 2D Color Image




DEMO

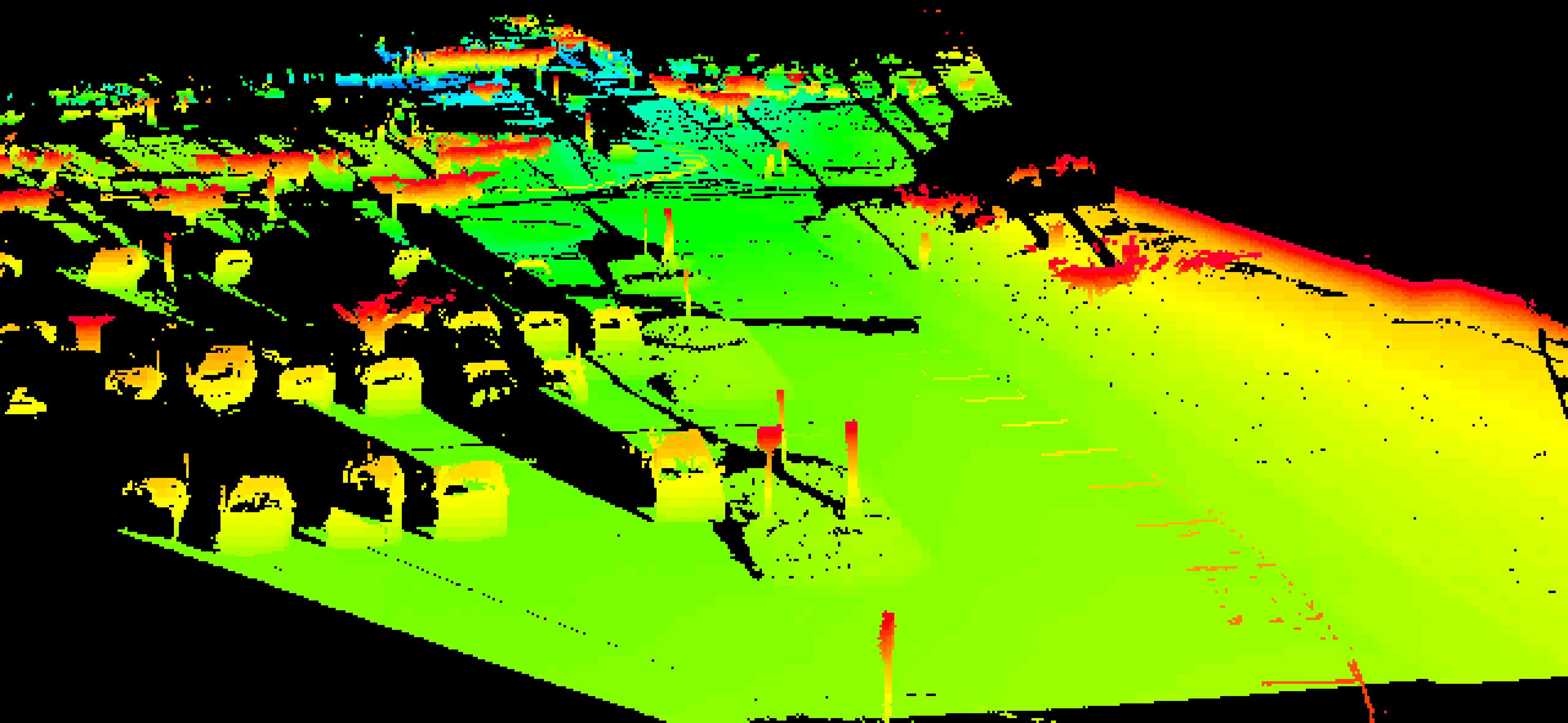
# Next Generation LADAR
## Intensity Image in the Woods


Geocoordinates Overhead View

# Range Image Oblique View

# High Resolution LADAR

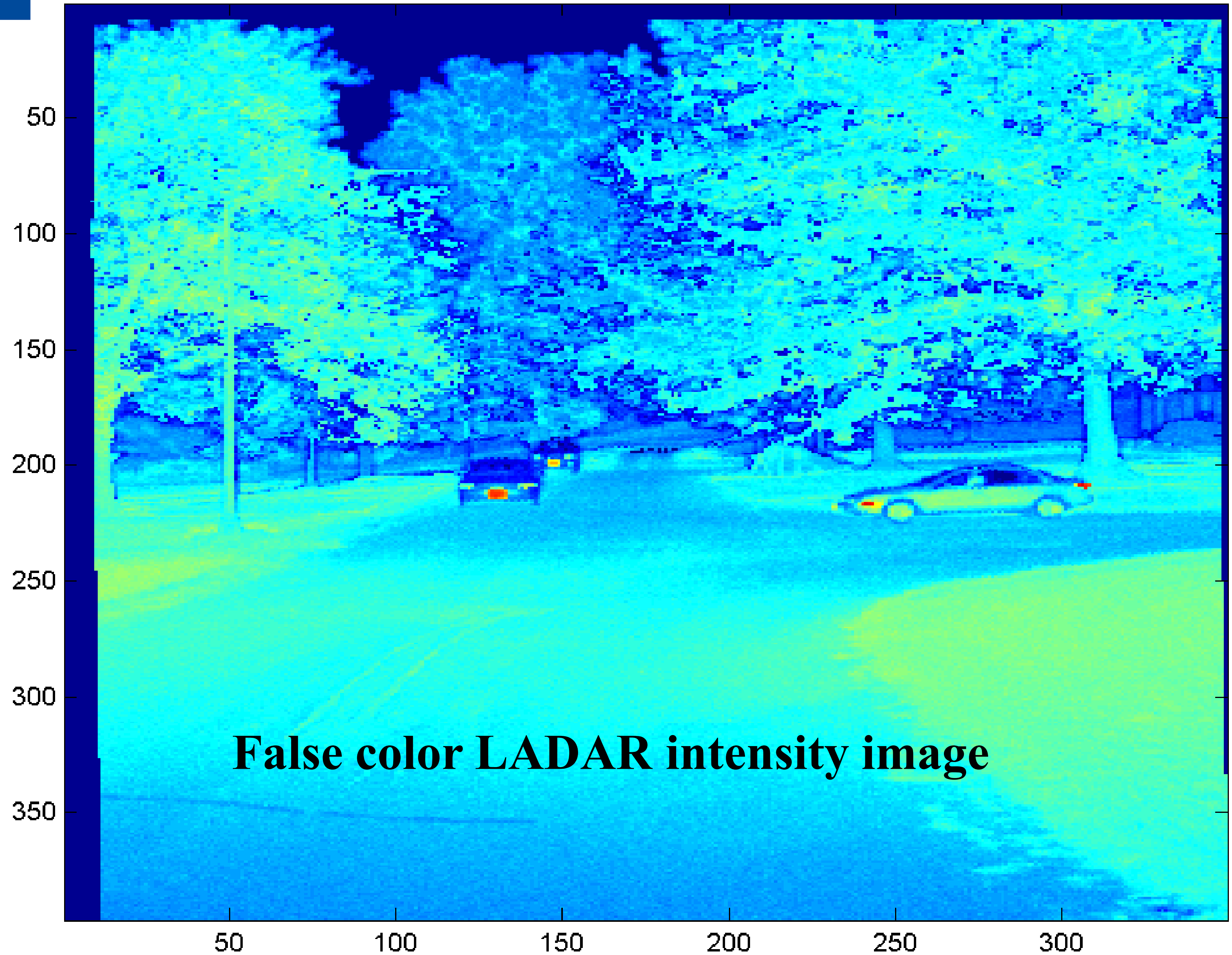
**.02 degree angular resolution**
**2 cm range resolution**
**5 x 80 degree field of view**



# Segmentation based on Fusion of Color & LADAR Images

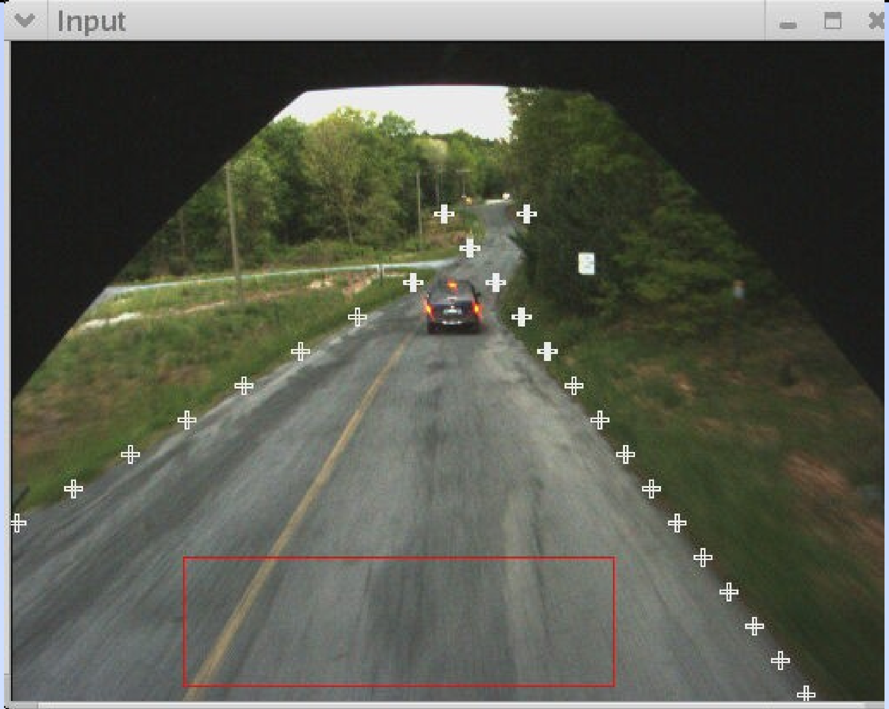
**James Albus**
**Tsai Hong**
**Mike Shneier**
**Gerry Cheok**
**Tommy Chang**

**National Institute of Standards and Technology**
**U. S. Department of Commerce**

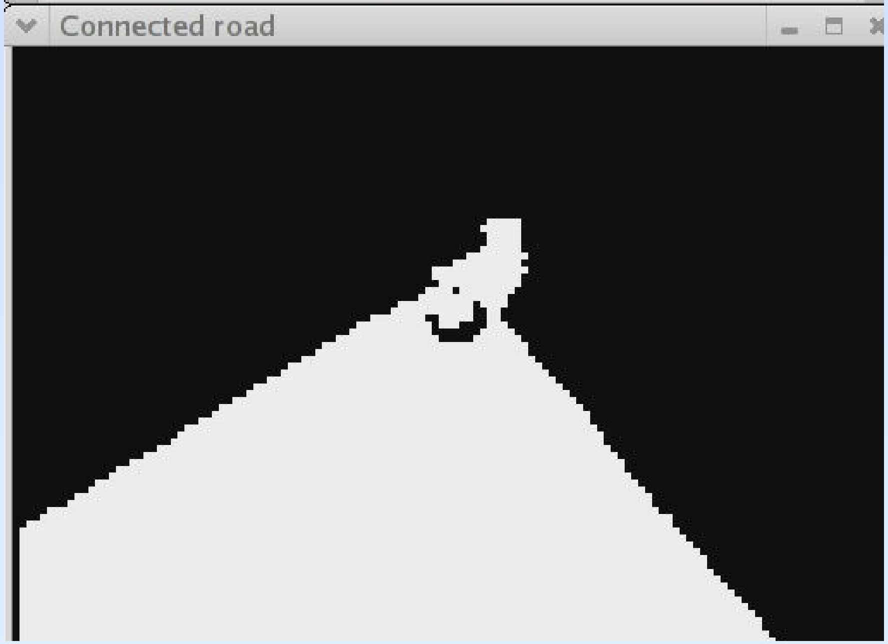

50
100
150
200
250
300
350
50
100
150
200
250
300
False color LADAR intensity image

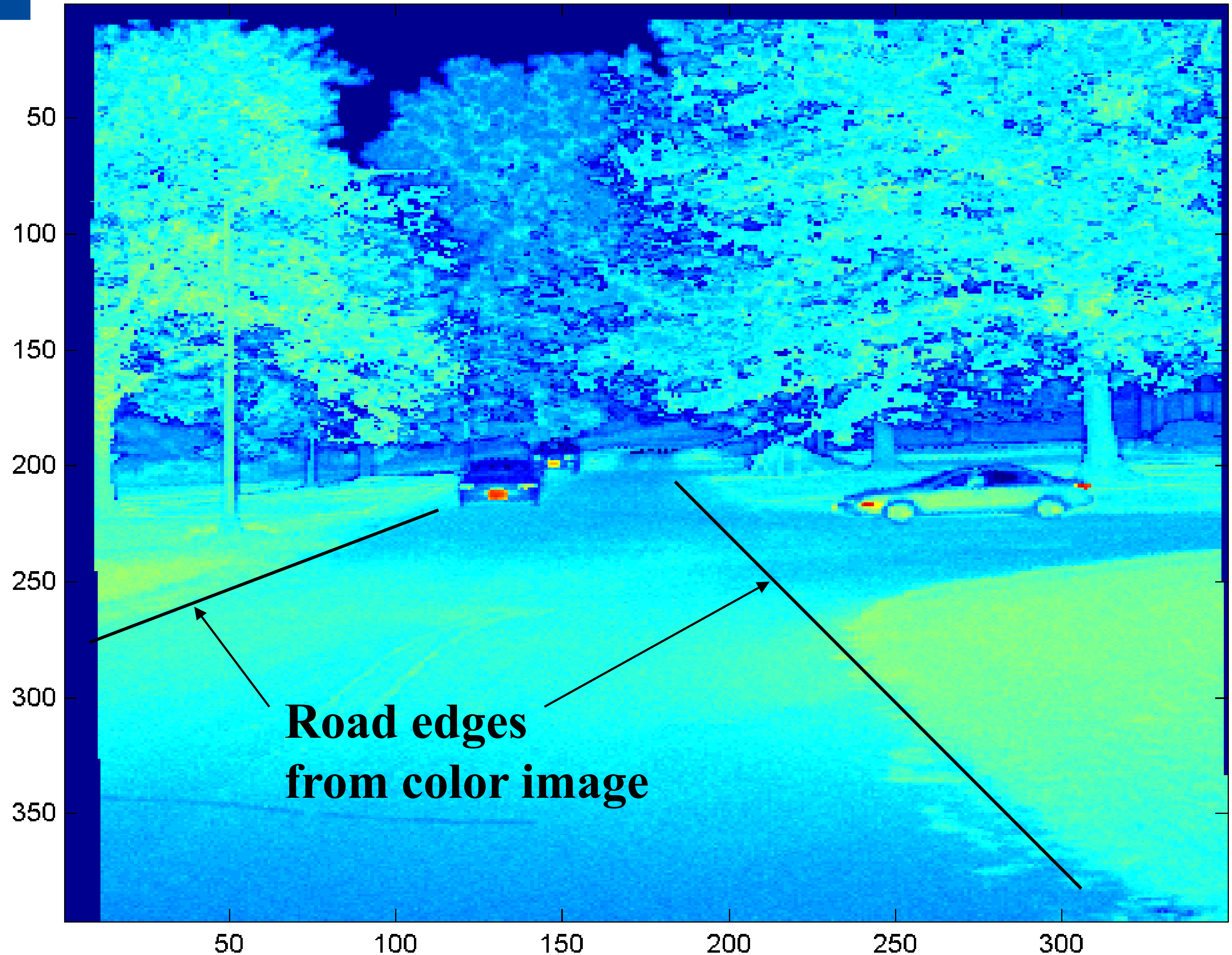
# Color Segmentation

Road Edge Detection

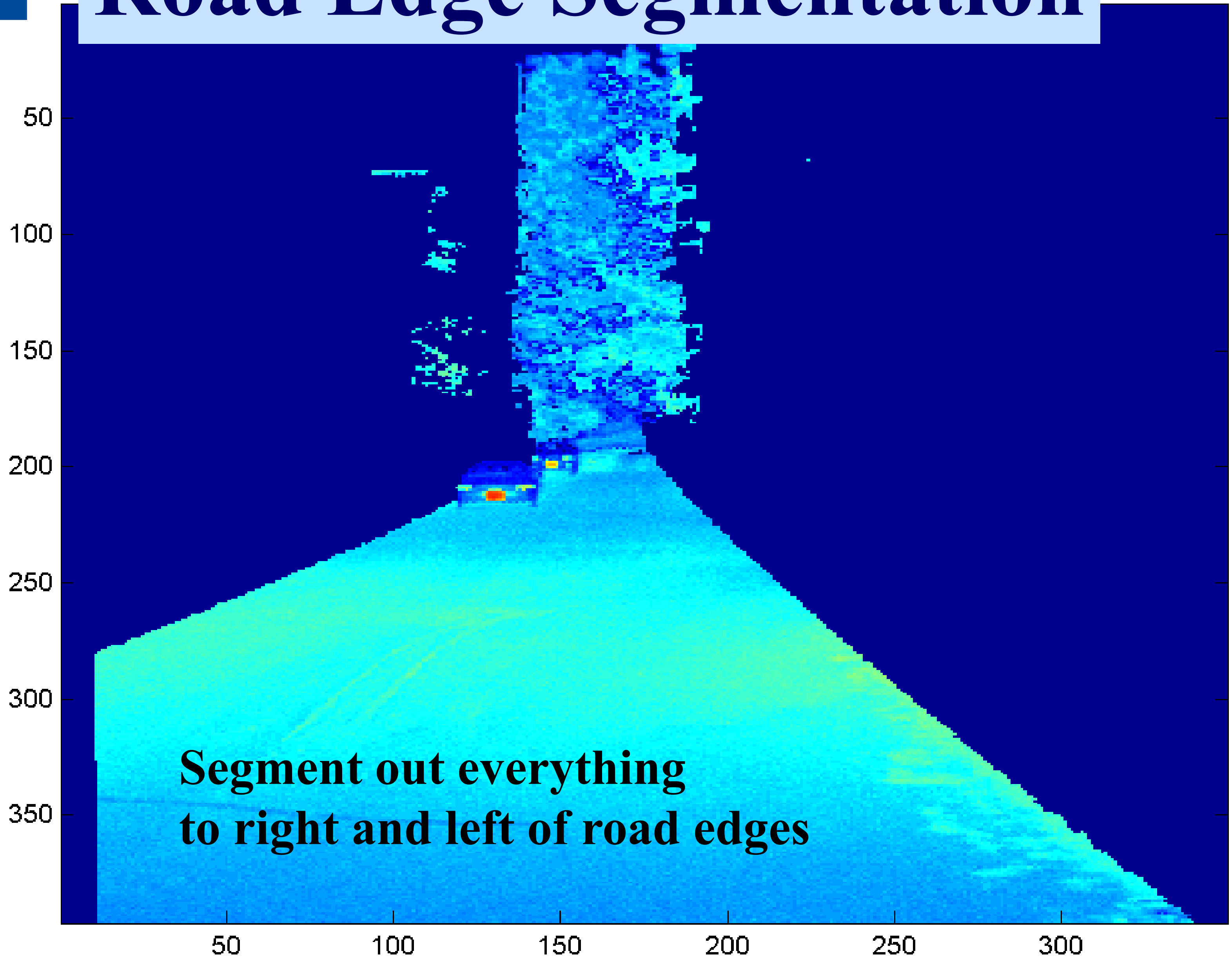



Road Detection

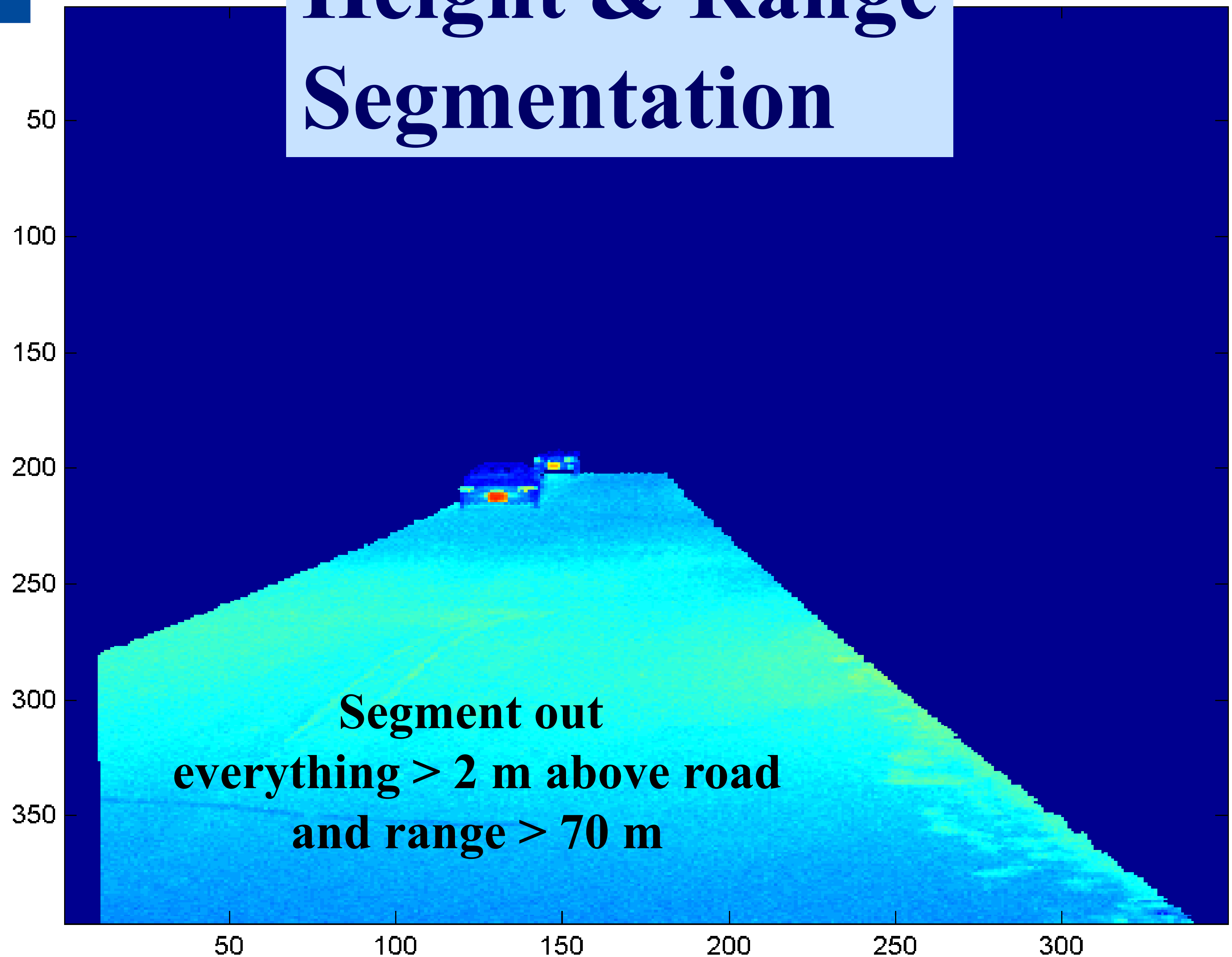



50
100
150
200
250
300
350
50
100
150
200
250
300
Road edges
from color image

# Road Edge Segmentation

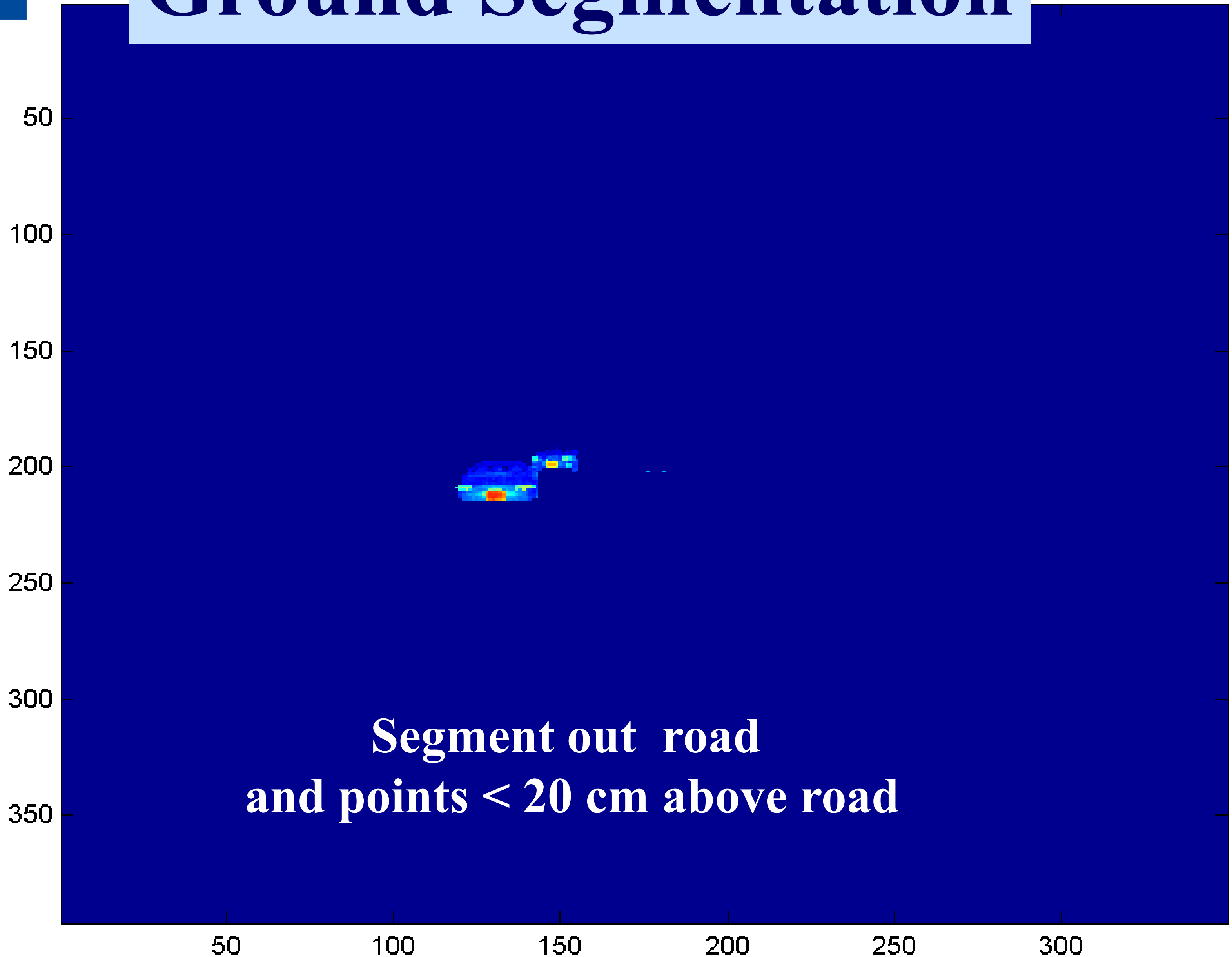


# Height & Range Segmentation




**Segment out everything > 2 m above road and range > 70 m**

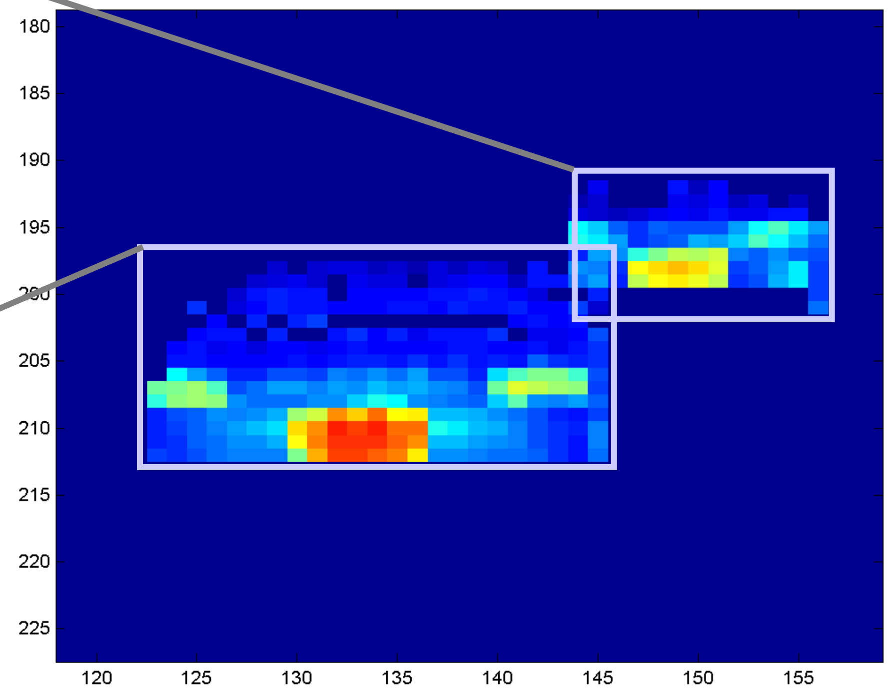
# Ground Segmentation



# Compute Attributes of Segmented Cars

**Object2**
**Range = 62 m**
**Closing speed = 2 m/sec**
**Width = 162 cm**
**Height = 140 cm**

**Object1**
**Range = 41 m**
**Closing speed = 2 m/sec**
**Width = 176 cm**
**Height = 128 cm**




**Classify based on height, width, and closing speed**

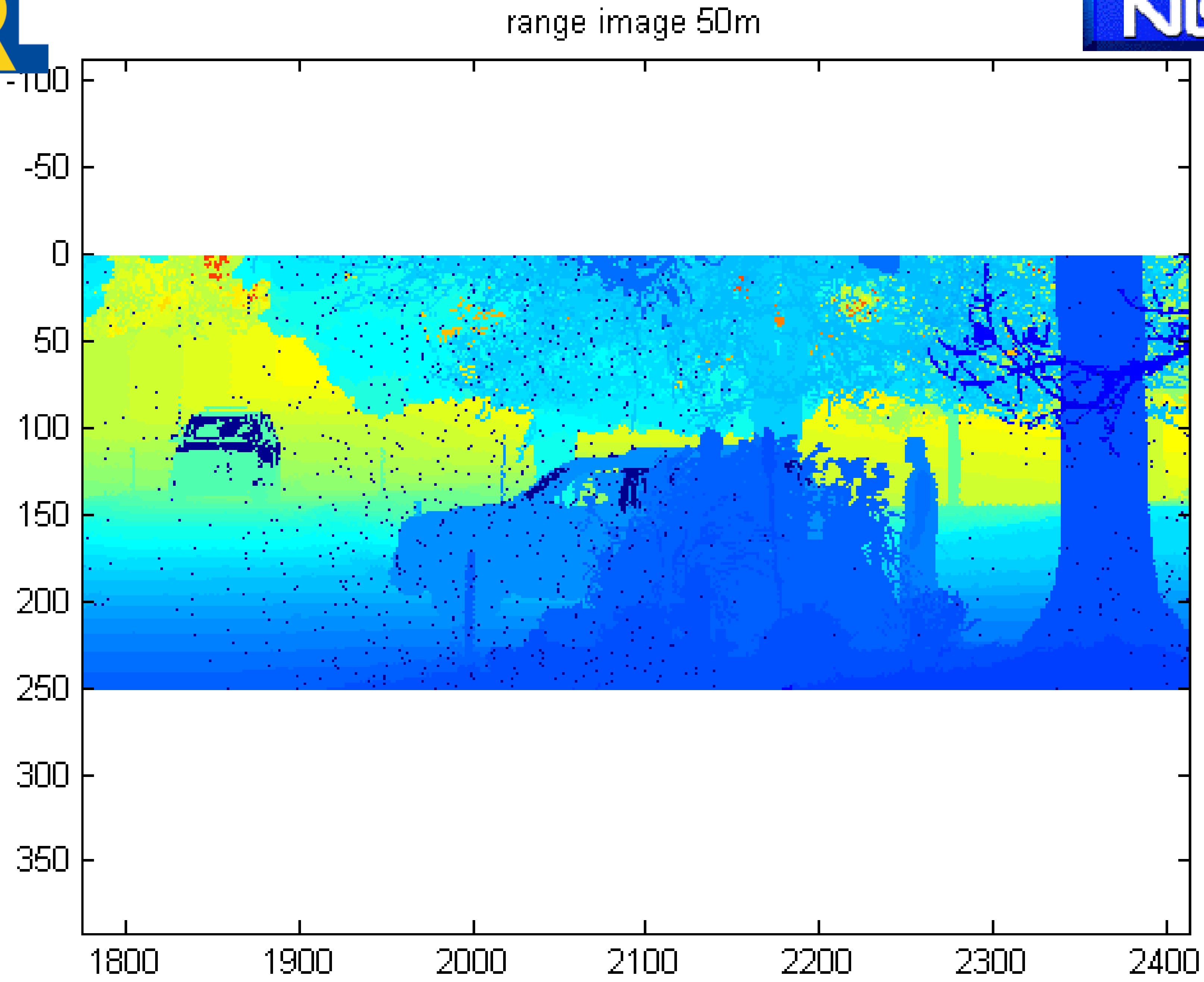
# Image Processing
# of
# High Resolution Range Images

# Human Detection
# in a
# Cluttered Environment

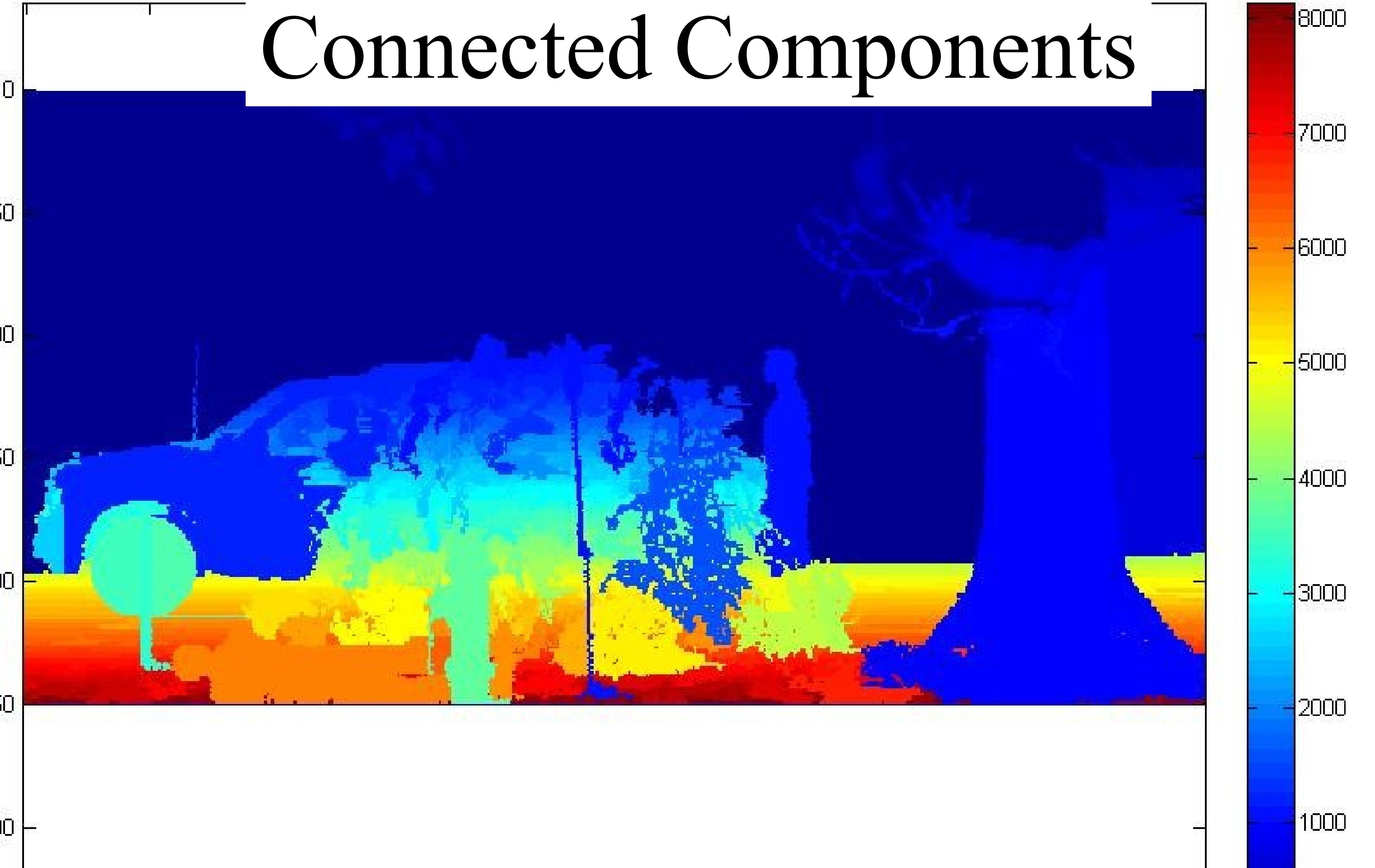
at 50 meters

**Jim Albus, Tsai Hong,**
**Will Shackelford, Tommy Chang, Gary Haas**

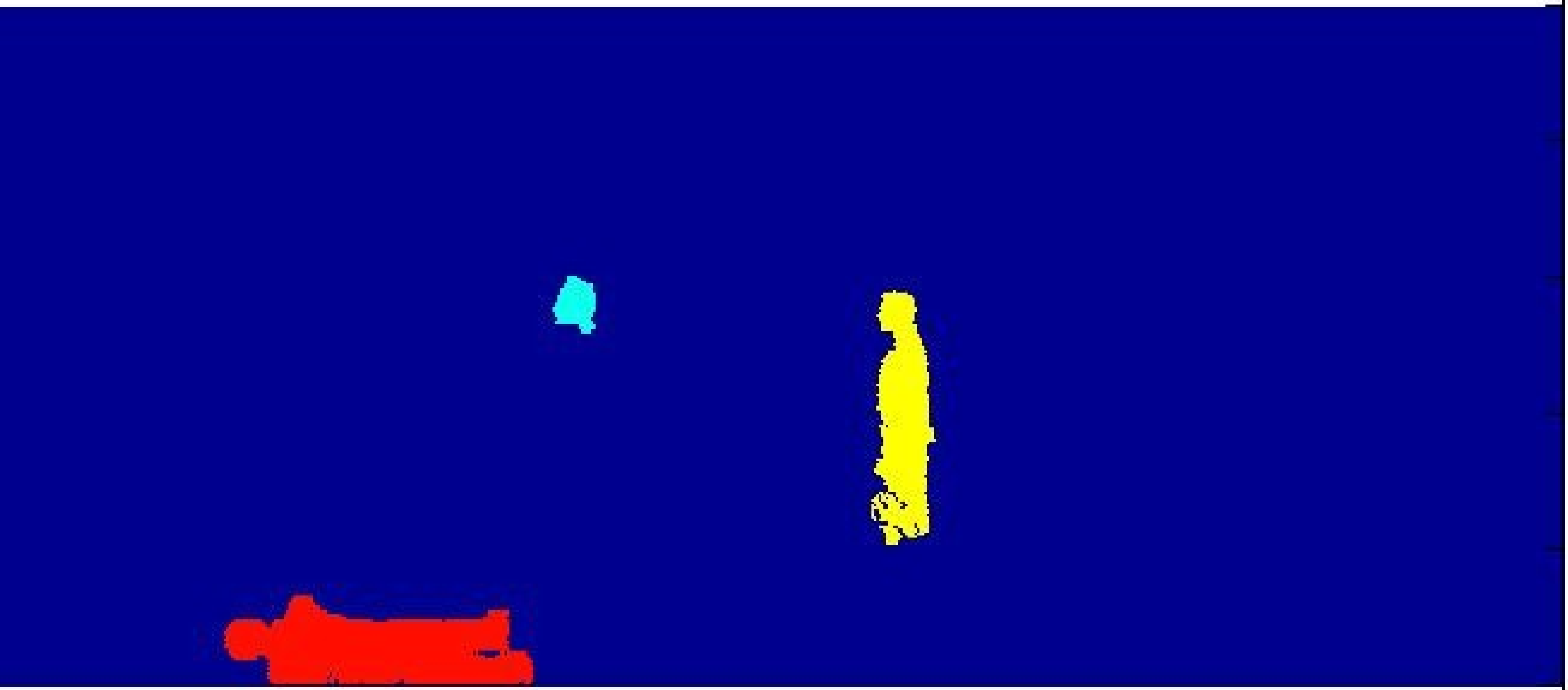
# Three Mannequins

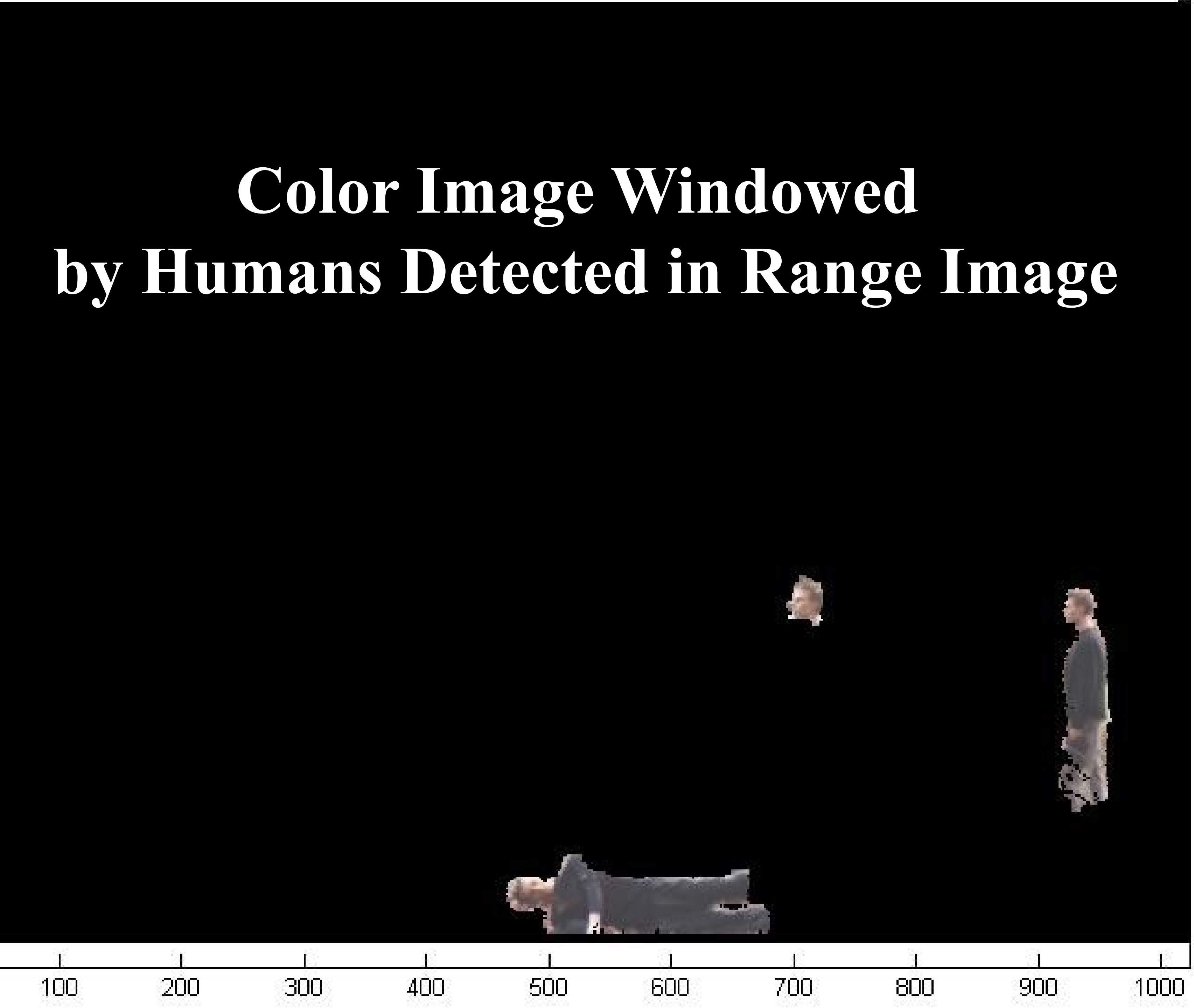

range image 50m
-100
-50
0
50
100
150
200
250
300
350
1800
1900
2000
2100
2200
2300
2400

# Segmentation based on Connected Components

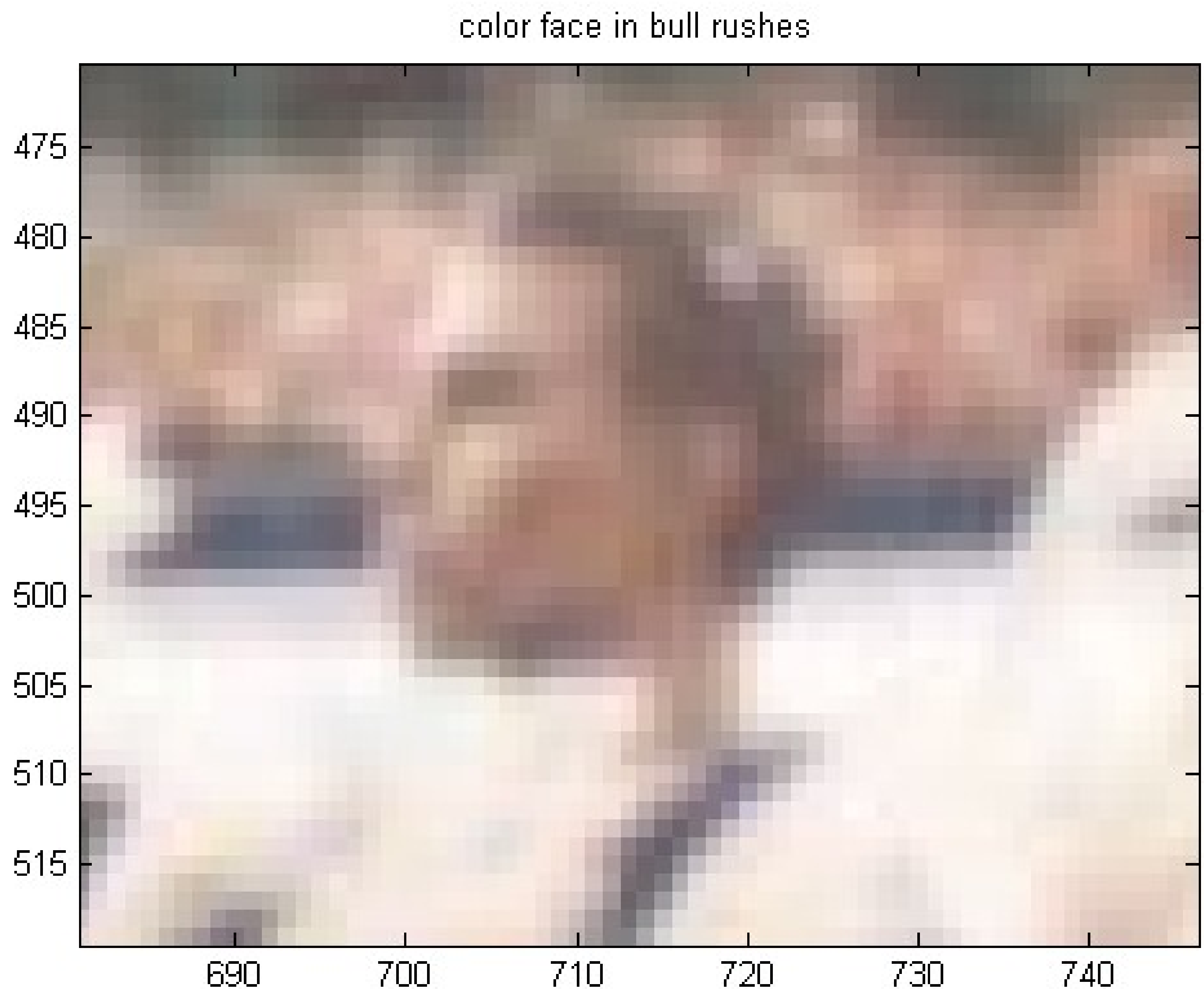


# Humans Detected




Color Image Windowed
by Humans Detected in Range Image
100
200
300
400
500
600
700
800
900
1000


full res color face in bull rushes
470
475
480
485
490
495
500
505
510
515
520
690
700
710
720
730
740

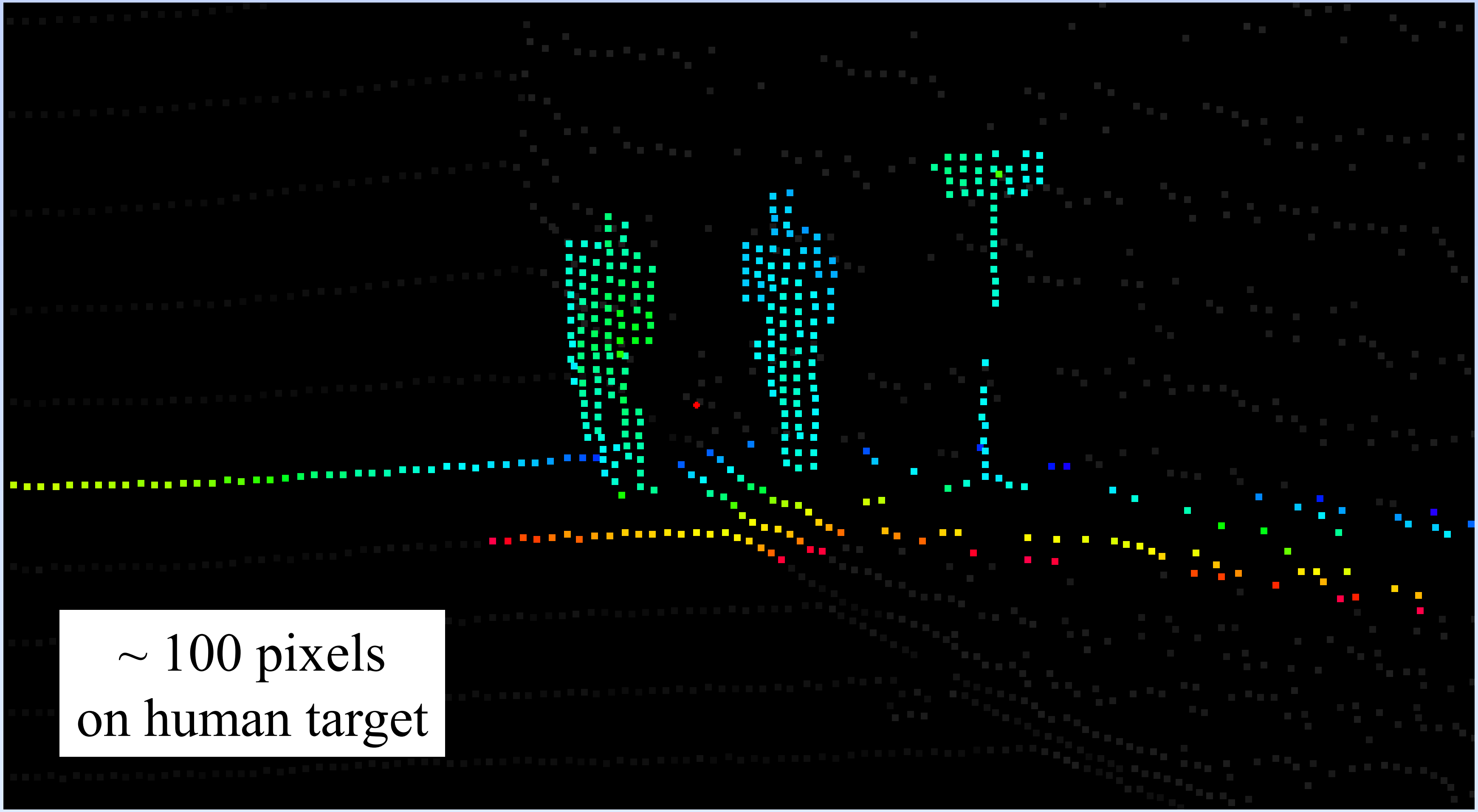

color face in bull rushes
475
480
485
490
495
500
505
510
515
690
700
710
720
730
740

# Classification
# requires pixels on target

**Translates into resolution required at distance**

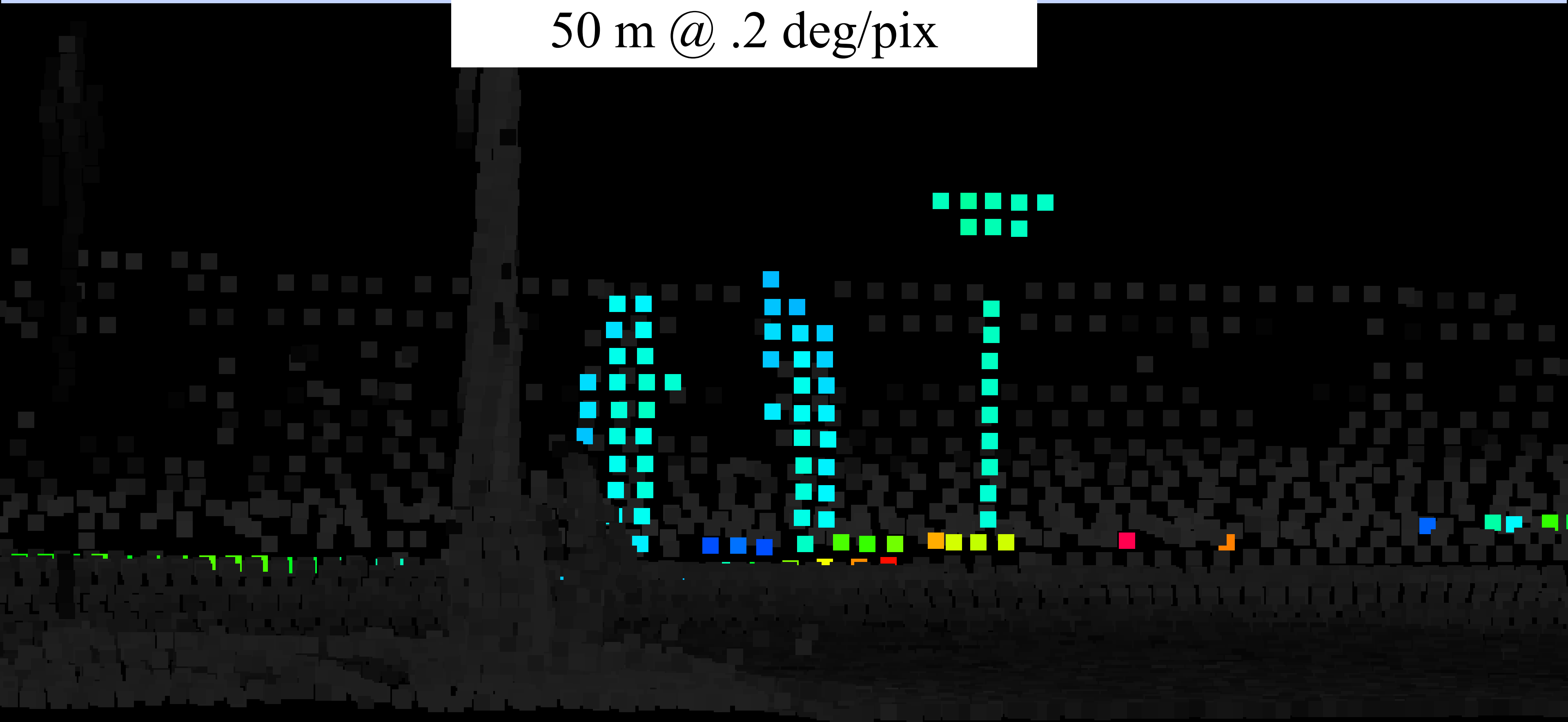

Color Video Camera
25 m @ .02 deg/pix
~ human vision
CENTER DR
~ 10,000 pixels
on human target

Current real-time LADAR
25 m @ .2 deg/pix



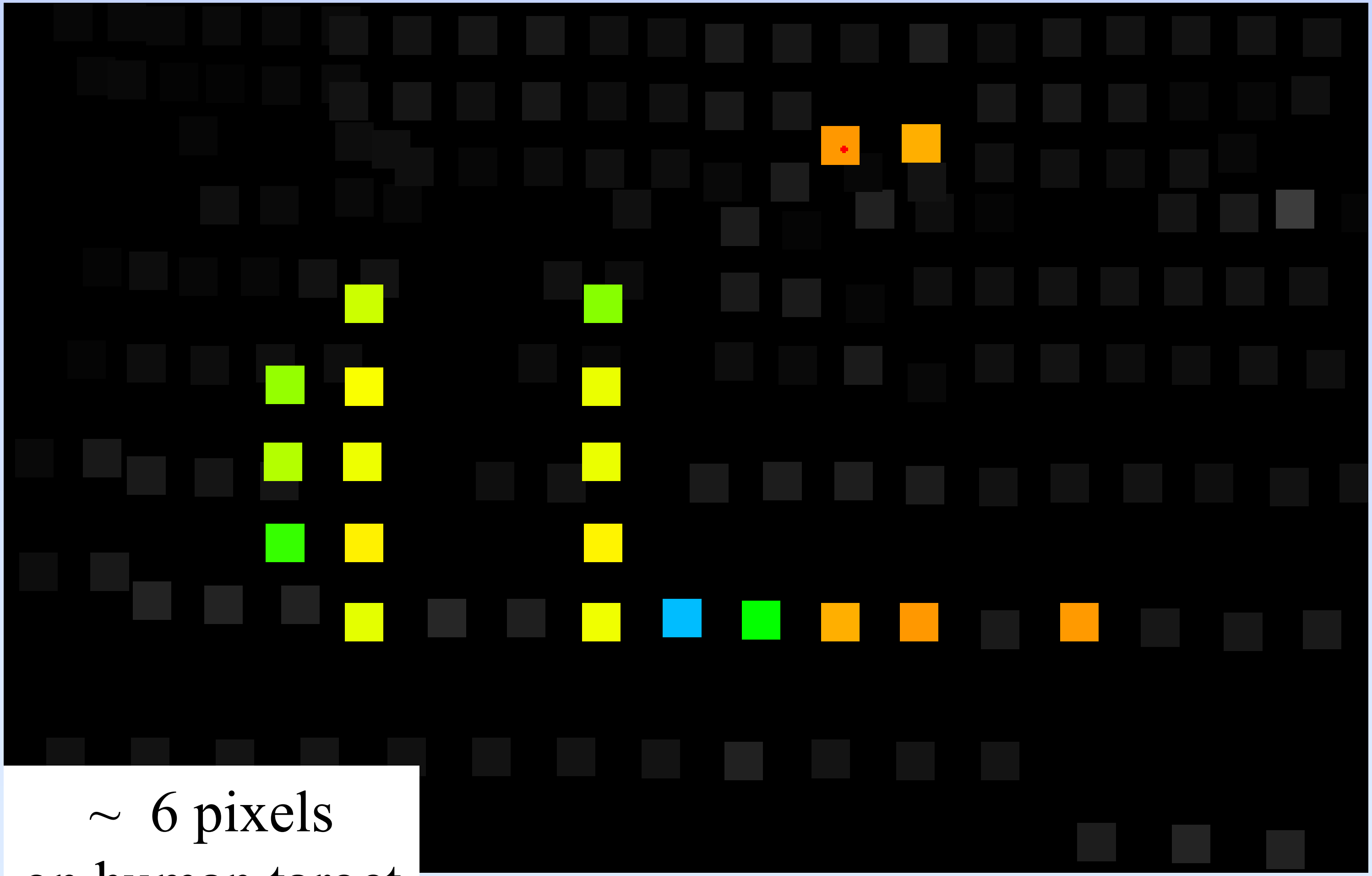
~ 100 pixels
on human target

# Current real-time LADAR
50 m @ .2 deg/pix



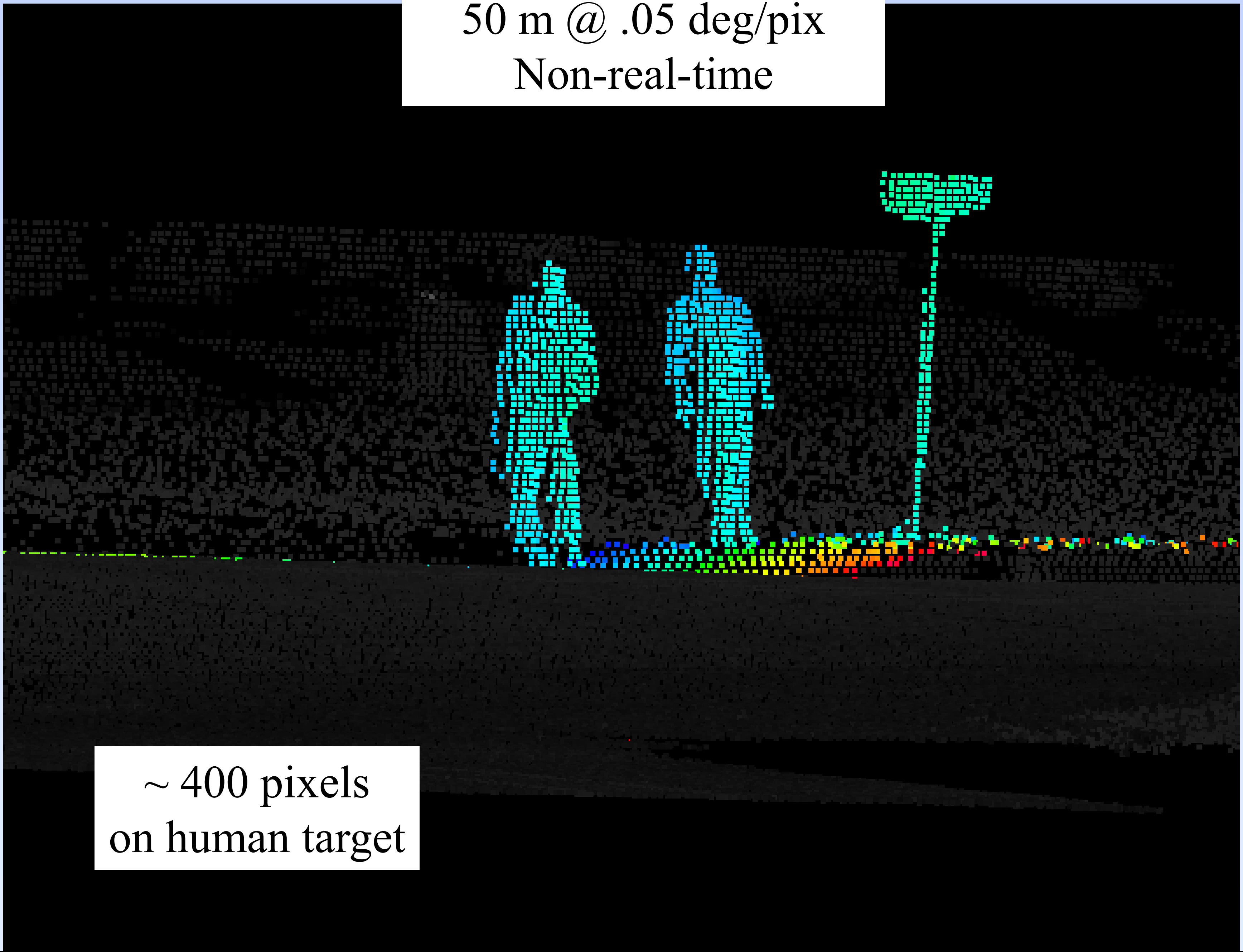
~ 25 pixels
on human target

Current real-time LADAR
100 m @ .2 deg/pix



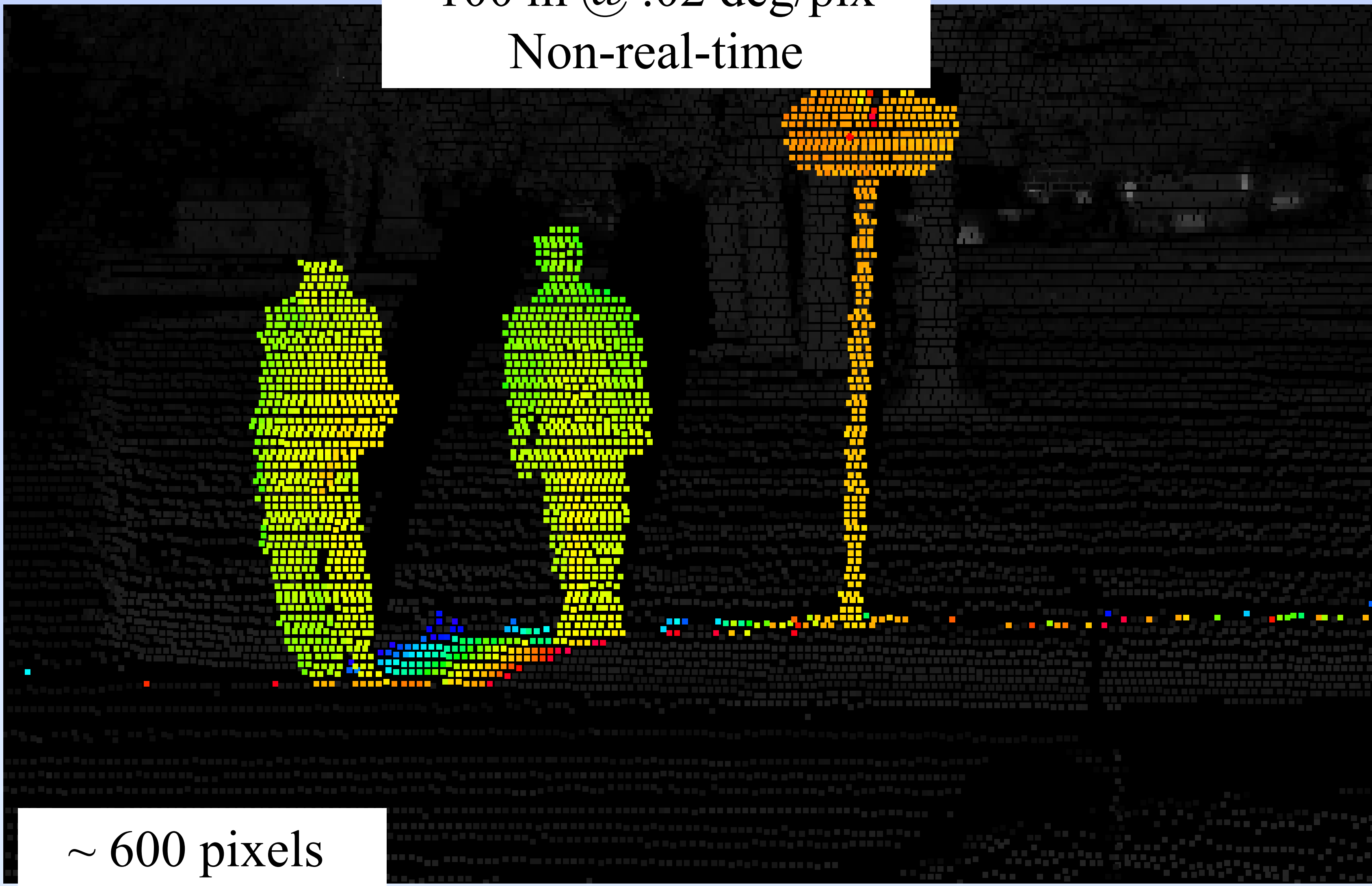
~ 6 pixels
on human target

High resolution LADAR
50 m @ .05 deg/pix
Non-real-time



~ 400 pixels
on human target

High resolution LADAR
100 m @ .02 deg/pix
Non-real-time



~ 600 pixels
on human target

# LADAR Resolution Required to Recognize Human Form at Various Distances

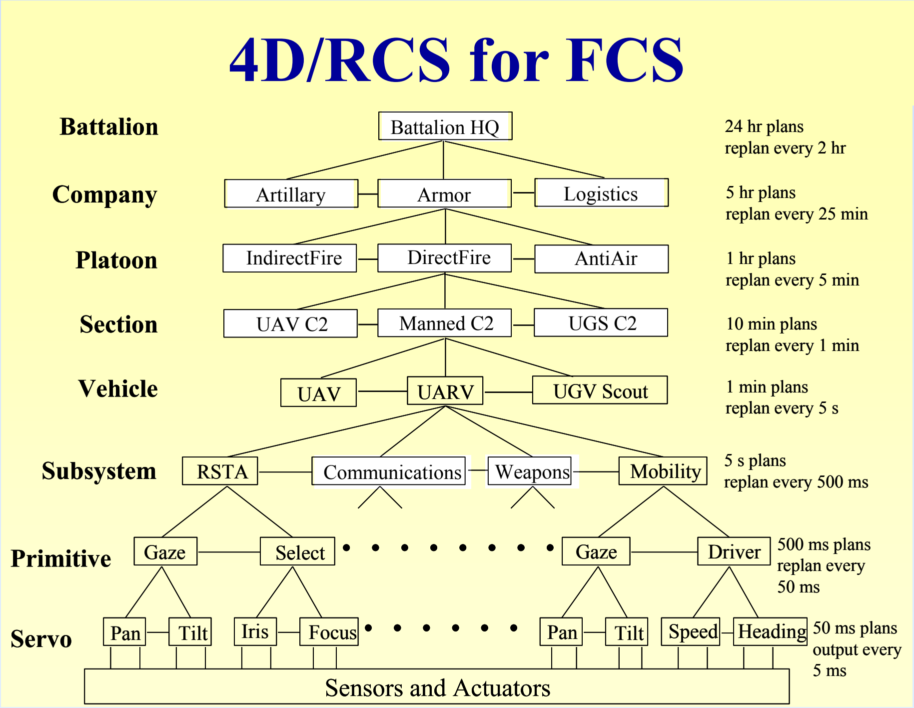
| Distances | Resolutions |
|---|---|
| 25 m | .2 deg/pix |
| 50 m | .05 deg/pix |
| 100 m | .02 deg/pix |

# 4D/RCS Methodology for Tactical Behaviors

**The ability to perform tactical behavior is the reason the Army is interested in robotics**

Company level – 30 to 40 vehicles
Platoon level – 8 to 10 vehicles
Section level – 2 to 4 vehicles
Vehicle level – single vehicle

**Manned/Unmanned collaboration**
**UGV/UAV/UGS collaboration**



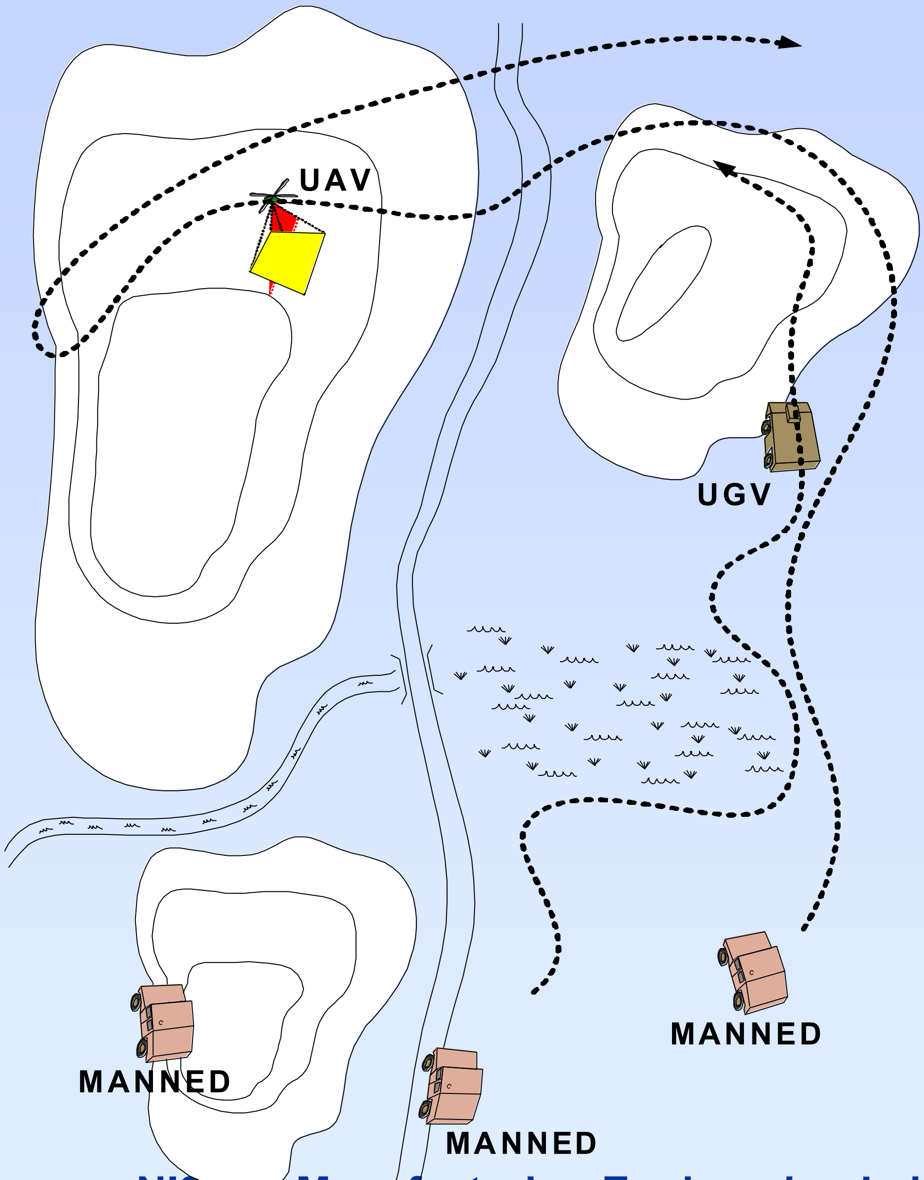
# An Example Scenario

A Light Cavalary Troop receives a command to perform a tactical road march to an assembly area

This results in a command to a Scout Platoon to perform a route reconnaissance of the road

The scout platoon is composed of three sections, each containing three manned HMMWVs, one unmanned ground vehicle (UGV), and one unmanned aerial vehicle (UAV.)

# A Section Scenario




Scout section is conducting a route reconnaissance

HMMWV reconnoitering the right flank comes upon an unexpected water obstacle

Center HMMWV discovers a bridge

The two vehicle commanders report their findings to the section leader

The section leader then might command the manned vehicles to take up overwatch positions for near-side security

The section leader also commands the UAV to look for a route around the water obstacle  UAV sends hi-resolution color images data back to the section leader for manual viewing, and/or by scanning the ground with a LADAR to assess the topography

Once a potential by-pass to the marsh is located, the UAV is commanded to search the far side of the marsh and the region beyond the next terrain feature for evidence of enemy forces

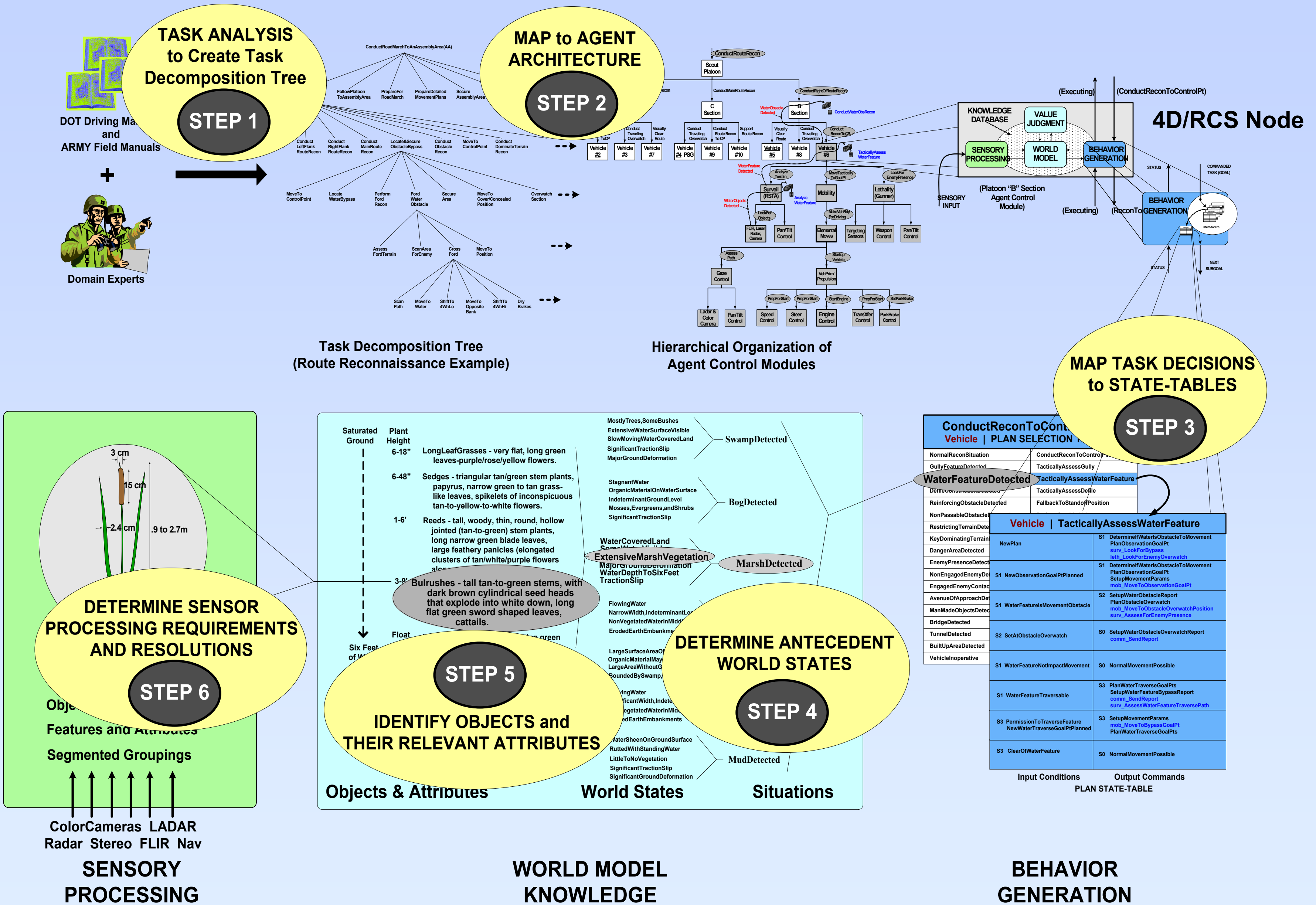
UGV might then be commanded to proceed through the bypass and establish an overwatch position on the far side of the next terrain feature

The UGV path can be automatically generated from the data returned from the UAV and approved by the section leader before being executed

Once the UGV is set in position, the UAV continues scanning for enemy activity further along the route

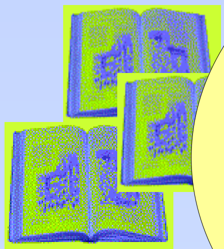
Manned elements perform manual reconnaissance of the marsh by-pass, and/or assess the load carrying capacity of the bridge.

# Capture Behavioral Knowledge



# TASK ANALYSIS to Create Task Decomposition Tree

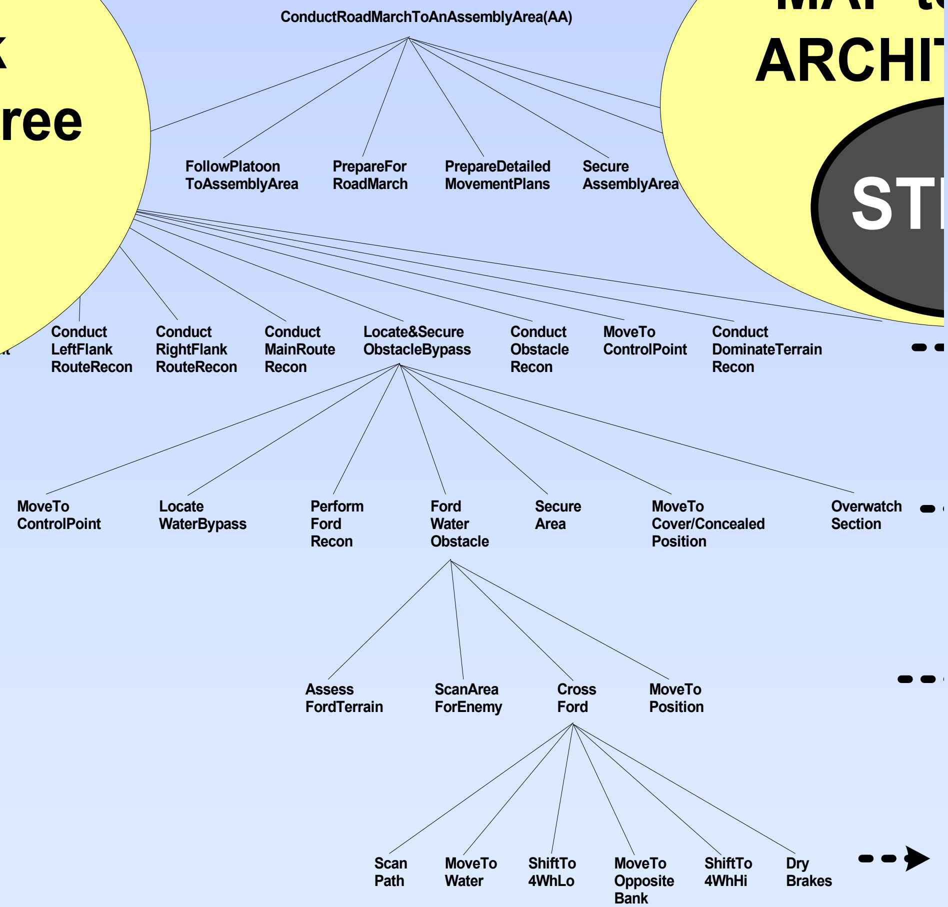
**STEP 1**



DOT Driving Ma
and
ARMY Field Manuals

+

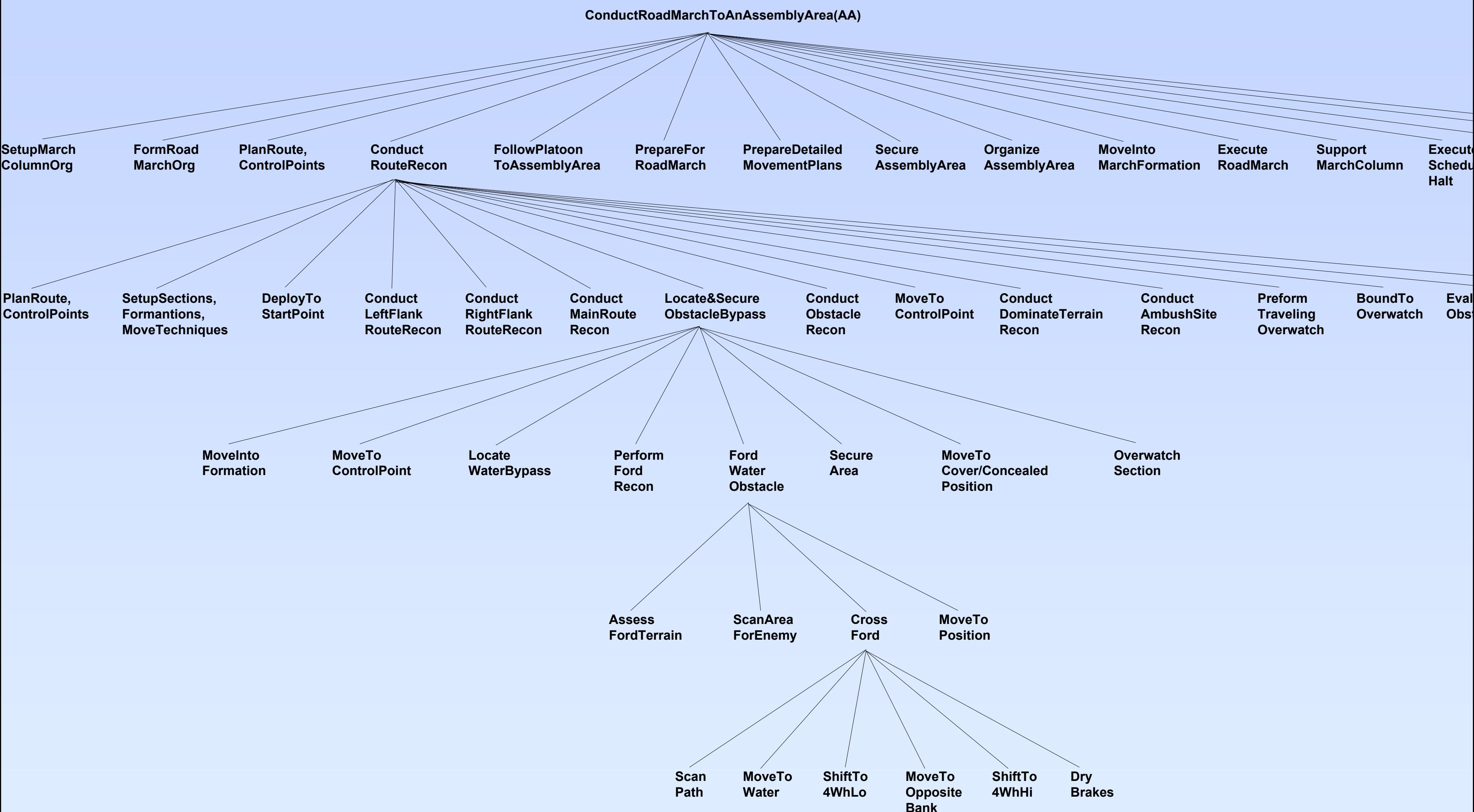
Domain Experts

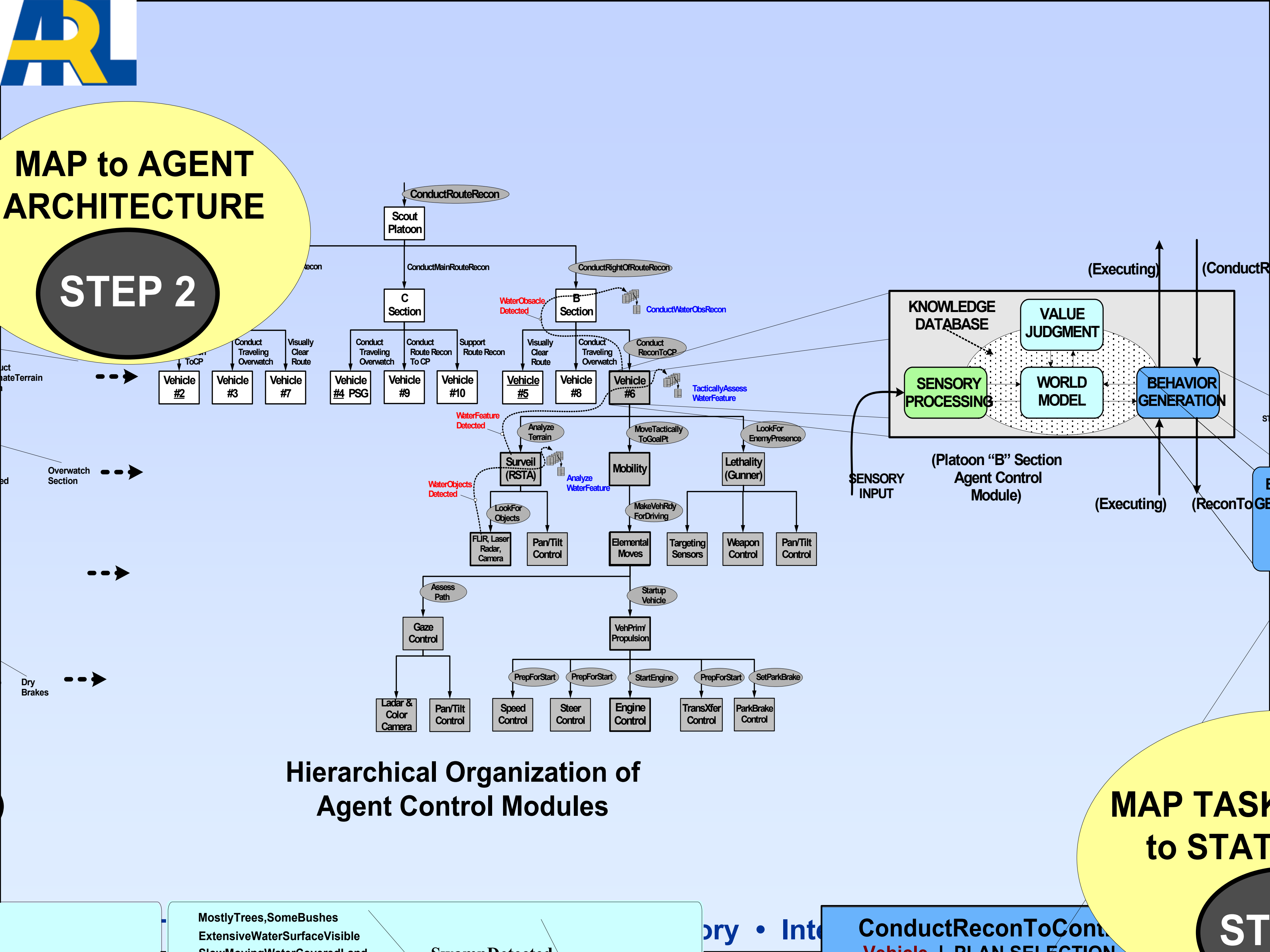



Task Decomposition Tree

MAP t
ARCHI
ST

# Task Vocabulary at Each Echelon

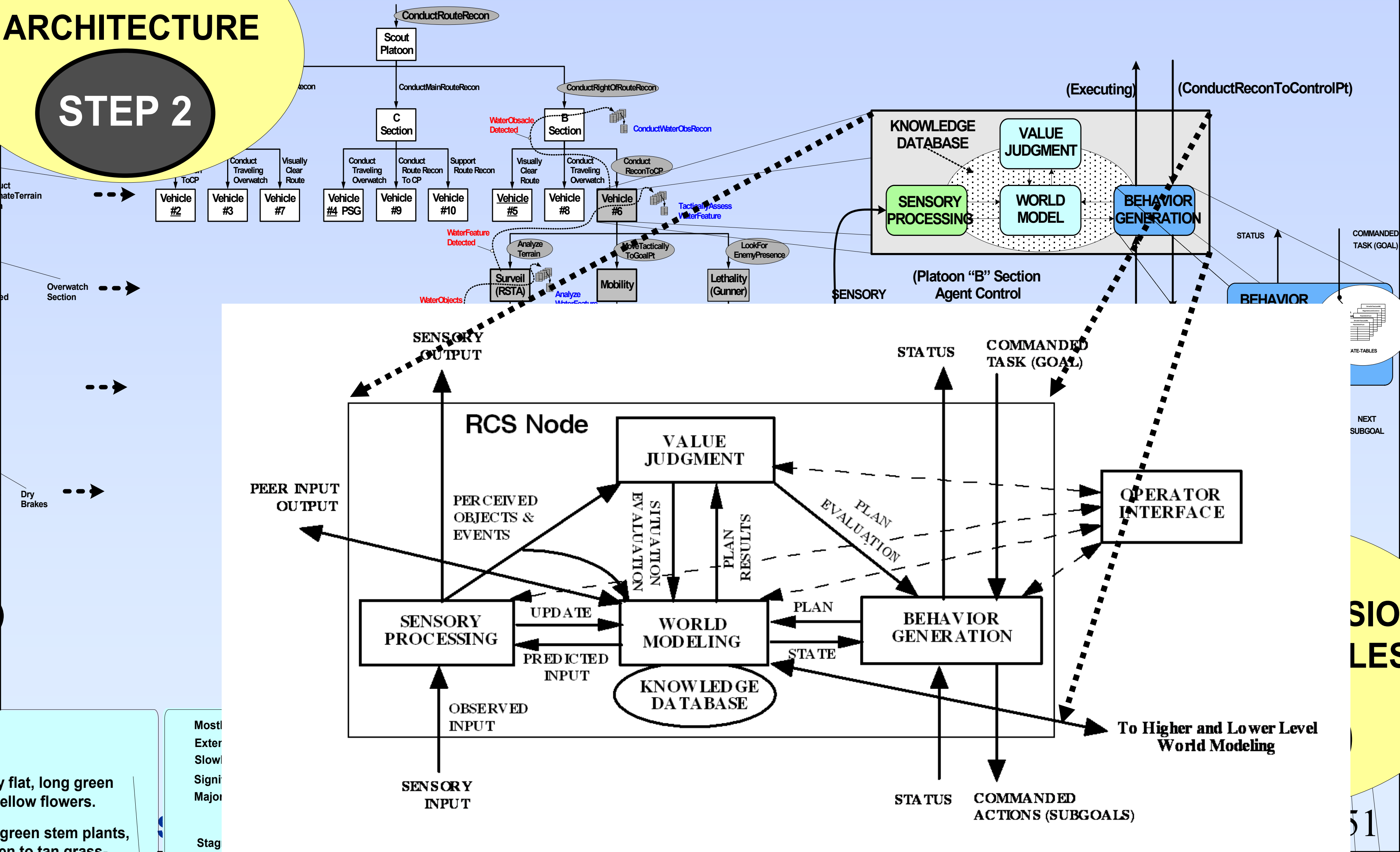




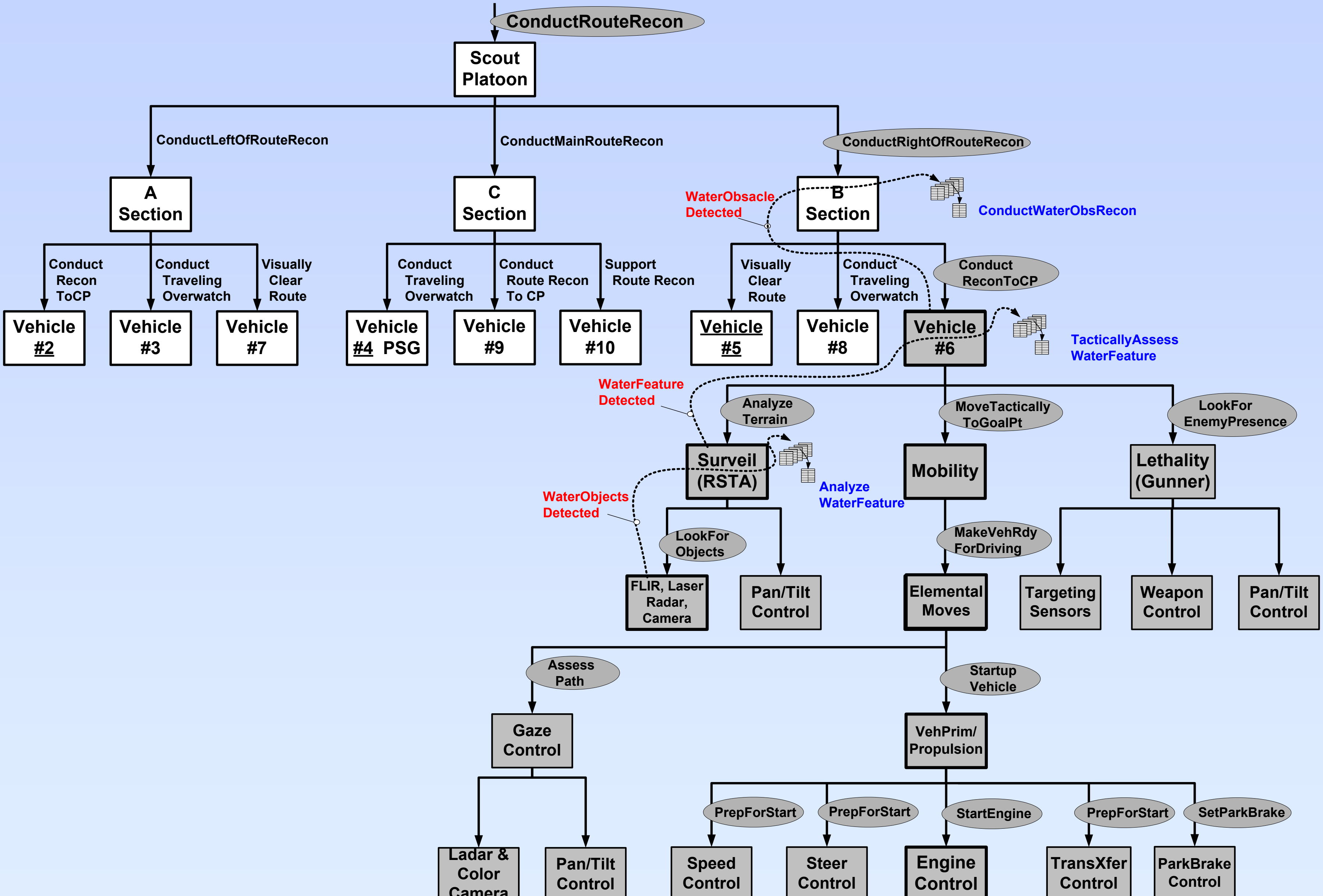
MAP to AGENT ARCHITECTURE
STEP 2
ConductRouteRecon
Scout Platoon
ConductMainRouteRecon
ConductRightOfRouteRecon
C Section
B Section
WaterObsacle Detected
ConductWaterObsRecon
Conduct Traveling Overwatch
Visually Clear Route
Conduct Route Recon To CP
Support Route Recon
Conduct ReconToCP
Vehicle #2
Vehicle #3
Vehicle #7
Vehicle #4 PSG
Vehicle #9
Vehicle #10
Vehicle #5
Vehicle #8
Vehicle #6
TacticallyAssess WaterFeature
WaterFeature Detected
Analyze Terrain
MoveTactically ToGoalPt
LookFor EnemyPresence
Surveil (RSTA)
Mobility
Lethality (Gunner)
WaterObjects Detected
Analyze WaterFeature
LookFor Objects
MakeVehRdy ForDriving
FLIR, Laser Radar, Camera
Pan/Tilt Control
Elemental Moves
Targeting Sensors
Weapon Control
Assess Path
Startup Vehicle
Gaze Control
VehPrim/ Propulsion
PrepForStart
StartEngine
SetParkBrake
Ladar & Color Camera
Speed Control
Steer Control
Engine Control
TransXfer Control
ParkBrake Control
Hierarchical Organization of Agent Control Modules
KNOWLEDGE DATABASE
VALUE JUDGMENT
SENSORY PROCESSING
WORLD MODEL
BEHAVIOR GENERATION
SENSORY INPUT
(Executing)
(Platoon "B" Section Agent Control Module)
Overwatch Section
Dry Brakes
MAP TASK to STAT
MostlyTrees,SomeBushes
ExtensiveWaterSurfaceVisible
ConductReconToCont

# MAP to AGENT ARCHITECTURE

## STEP 2



# Agent Architecture



# MAP TASK DECISIONS to STATE-TABLES

## STEP 3

**ConductReconToCont**
**Vehicle | PLAN SELECTION T**

| | |
|---|---|
| NormalReconSituation | ConductReconToControlP |
| GullyFeatureDetected | TacticallyAssessGully |
| WaterFeatureDetected | TacticallyAssessWaterFeature |
| DefileConstrictionDetected | TacticallyAssessDefile |
| ReinforcingObstacleDetected | FallbackToStandoffPosition |
| NonPassableObstacleD | |
| RestrictingTerrainDete | |
| KeyDominatingTerrainI | |
| DangerAreaDetected | |
| EnemyPresenceDetect | |
| NonEngagedEnemyDet | |
| EngagedEnemyContac | |
| AvenueOfApproachDet | |
| ManMadeObjectsDetec | |
| BridgeDetected | |
| TunnelDetected | |
| BuiltUpAreaDetected | |
| VehicleInoperative | |

**Vehicle | TacticallyAssessWaterFeature**

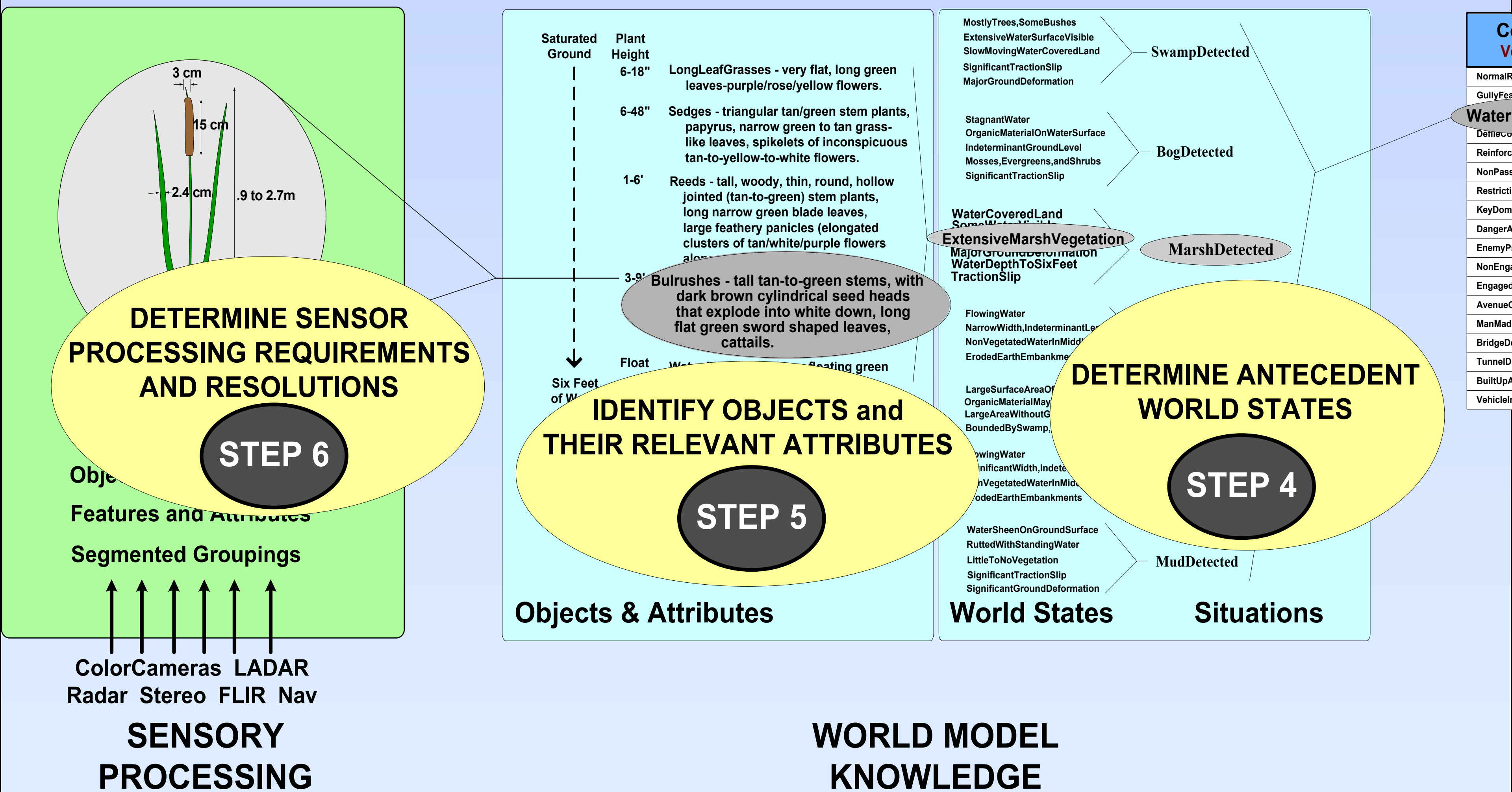
| Input Conditions | Output Commands |
|---|---|
| NewPlan | S1 DetermineIfWaterIsObstacleToMovement<br>PlanObservationGoalPt<br>surv_LookForBypass<br>leth_LookForEnemyOverwatch |
| S1 NewObservationGoalPtPlanned | S1 DetermineIfWaterIsObstacleToMovement<br>PlanObservationGoalPt<br>SetupMovementParams<br>mob_MoveToObservationGoalPt |
| S1 WaterFeatureIsMovementObstacle | S2 SetupWaterObstacleReport<br>PlanObstacleOverwatch<br>mob_MoveToObstacleOverwatchPosition<br>surv_AssessForEnemyPresence |
| S2 SetAtObstacleOverwatch | S0 SetupWaterObstacleOverwatchReport<br>comm_SendReport |
| S1 WaterFeatureNotImpactMovement | S0 NormalMovementPossible |
| S1 WaterFeatureTraversable | S3 PlanWaterTraverseGoalPts<br>SetupWaterFeatureBypassReport<br>comm_SendReport<br>surv_AssessWaterFeatureTraversePath |
| S3 PermissionToTraverseFeature<br>NewWaterTraverseGoalPtPlanned | S3 SetupMovementParams<br>mob_MoveToBypassGoalPt<br>PlanWaterTraverseGoalPts |
| S3 ClearOfWaterFeature | S0 NormalMovementPossible |

**PLAN STATE-TABLE**

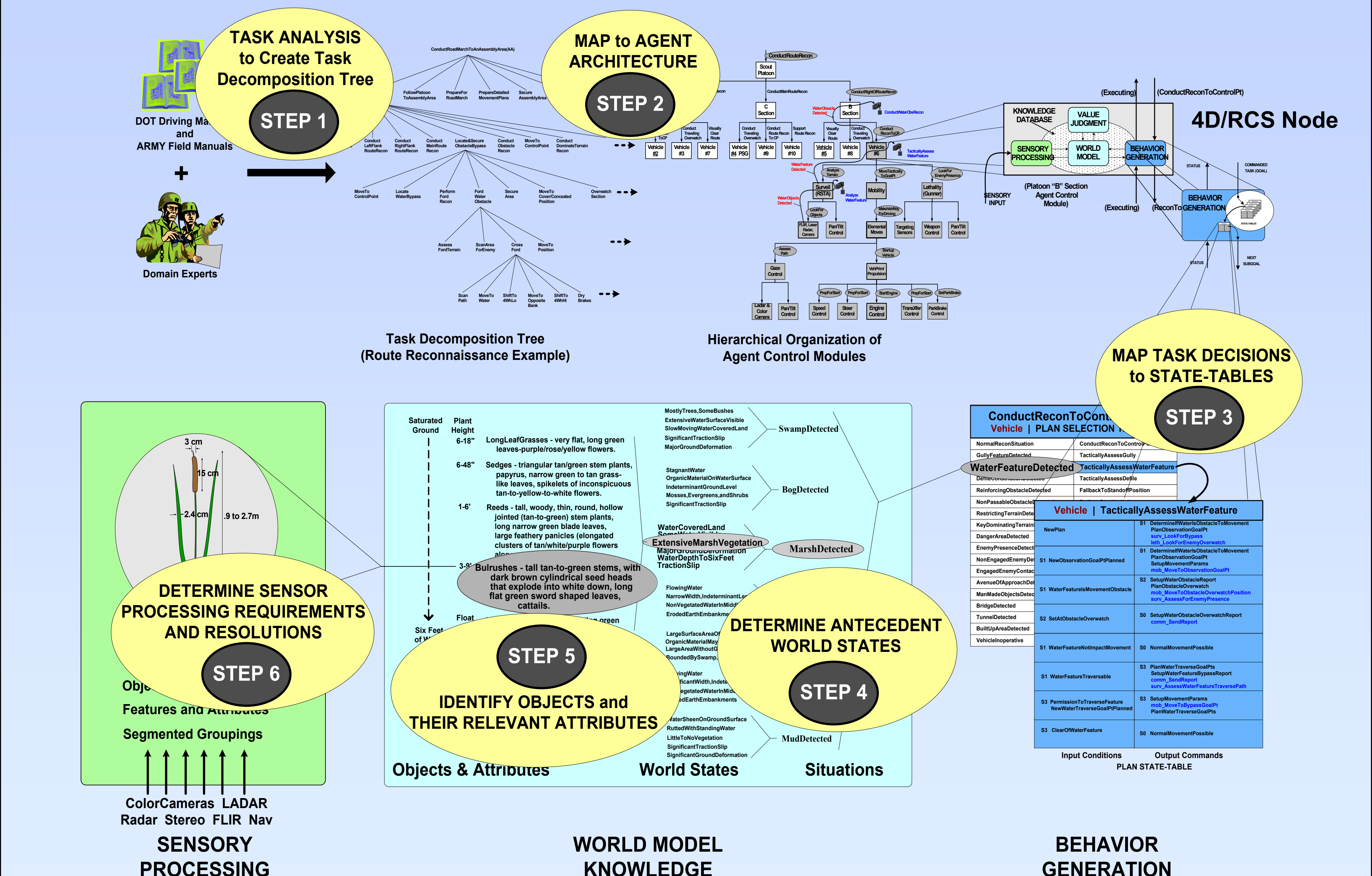
# Determine Transition Conditions

# Identify Objects and Attributes

# Determine Sensor Requirements



# Capture Behavioral Knowledge



## Activities

## Agent Hierarchy with Commands

### Section

Uses maps, traffic, weather, and construction reports to select active mapquest-like output command

**Destination Manager (Section)**

InitializeSytem
MakeVehOperational
ShutDownVehicle
TurnOffSystem
Goto_Destination
FollowVehicle

### Vehicle

Reads signs, detects road intersections and decides on real-time road changing maneuvers

**RouteSegment Manager (Vehicle)**

GoOn_TurnRightOnto_
FollowRoute
TurnRightAtIntersection
TurnRightUsingExitRamp
TurnRightUsingFork

InitializeSystems
StartupVehicle
TurnOffSystems
ShutDownVehicle

FollowVehicle
FollowRoute
TurnLeftAtIntersection
TurnLeftUsingExitRamp
TurnLeftUsingFork
TurnLeftUsingRightWithUTurn
TurnLeftUsingUWithRightTurn
TurnRightAtIntersection
TurnRightUsingExitRamp
TurnRightUsingFork
TurnIntoDrive

GoOn_TurnLeftOnto_
FollowRoute
TurnLeftAtIntersection
TurnLeftUsingExitRamp
TurnLeftUsingFork
TurnLeftUsingRightWithU Turn
TurnLeftUsingUWithRight Turn

RespondToEmerVeh
RespondToOwnVehEmer
RespondToSchoolBus
RespondToTrafficPerson
PullOntoRoad
BackUp
Make_U_Turn
GoOn_
GoOn_Becomes_
StopAt_

### Mobility

Detects and recognizes relevant vehicles and objects, and determines how they affect basic driving behaviors

**DriveBehavior Manager (Mobility)**

FollowRoad
PassVehInFront
DriveOnTwoLaneRd
DriveOnMultiLaneRd
PullOntoRoad
ChangeLaneToGoFaster
ChangeToGoalLane
AccomodatePassingVehicle
RespondToFollowingVeh
NegotiateLaneConstriction
NegotiateMovingConstriction
RespondToPedestrian
RespondToBicyclist
RespondToVehEnteringLane
DriveOnNarrowRoad
RespondToOncomingPassingVeh

CrossThru_Intersect
CrossThru_StopSign
CrossThru_YieldSign
CrossThru_SignalLight
CrossThru_UncontrolledInter
CrossThru_TrafficPerson
MergeInto_TravelLane
AccomodateMerge
Negotiate_RRCrossing
Negotiate_TollBooth
Negotiate_PedestrianCross
Negotiate_GateKeeper

InitSubsystems
StartUpVehicle
ShutDownVehicle
TurnOffSubsystems

TurnLeftAtInterTo_
TurnLeft_StopSign
TurnLeft_YieldSign
TurnLeft_SignalLight
TurnLeft_UncontrolledInter
TurnLeft_IntoDrive
TurnLeft_FromDrive
TurnLeft_IntoParkingSpace
TurnLeft_TrafficPerson

Fork_Right
Fork_Left
Merge_Right
Merge_Left
GoTo_RightExitRamp
GoTo_LeftExitRamp
BackOut_GoLeft
BackOut_GoRight

TurnRightAtInterTo_
TurnRight_StopSign
TurnRight_YieldSign
TurnRight_SignalLight
TurnRight_UncontrolledInter
TurnRight_IntoDrive
TurnRight_FromDrive
TurnRight_IntoParkingSpace
TurnRight_TrafficPerson

Backup_
BackupVehicle
BackupIntoParallelPark
BackupOutOfParkSpace

RespondTo_OwnVehEmer
Accommodate_SchoolBus
Accommodate_EmerVeh

BackRightTo_
BackRight_IntoLane
BackRight_IntoDrive
BackRight_IntoParkingSpace

Make_U_Turn
Do_U_TurnAtIntersection
Do_U_TurnThruAccess
TurnAroundUsingDrive
TurnAroundInRoad
TurnAround_TrafficPerson

BackLeftTo_
BackLeft_IntoLane
BackLeft_IntoDrive
BackLeft_IntoParkingSpace

### Dynamics

Receives goal lane with list of relevant vehicles and objects along immediate route and generates goal path to avoid collisions

**Elemental Maneuver Subsystem**

InitSubsystems
StartupVehicle
ShutDownVehicle
TurnOffSubsystems
FollowLane
PassOnLeft
PassOnRight
TurnRightTo_
TurnLeftTo_
StopAt

PullOff_OnLeftShoulder
PullOff_OnRightShoulder
GotoGap_LeftLane
GotoGap_RightLane
Premerge_LeftLane
Premerge_RightLane
ChangeTo_LeftLane
ChangeTo_RightLane
StopAtIntersection
AbortPass

CreepForward
PeekForPass
Backup
BackOut_ToGoLeft
BackOut_ToGoRight
BackInto_FromLeft
BackInto_FromRight
DoUTurn_AtInter
DoUTurn_MidRoad
Do3Pt_UTurn

CreepBackward
AllowVehToEnter_FromLeft
AllowVehToEnter_FromRight
YieldToPassingVeh
ReactToPassingVehAbort
PullOntoRd_FromLeftSh
PullOntoRd_FromRightSh

### Servo

Receives commanded goal path and calculates real-time dynamically feasible trajectories that adapt to side skid and traction slip

**GoalPath Trajectory**

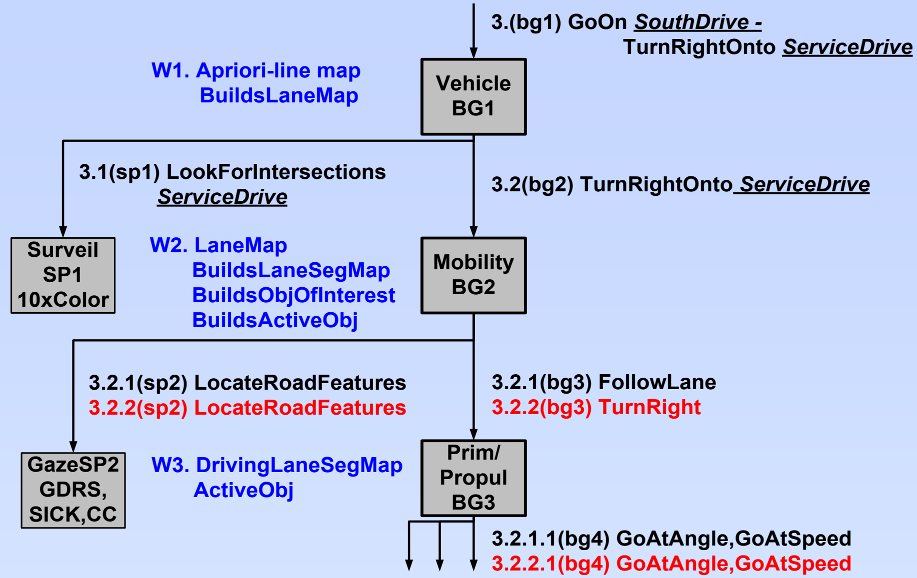
InitSubsystems
StartupVehicle
ShutDownVehicle
TurnOffSubsystems

Follow_StLine
Follow_CirArcCW
Follow_CirArcCCW

Stop/Halt
SetupForwardDirTraj
SetupReverseDirTraj

# On-road Driving

# On-road Driving Analysis

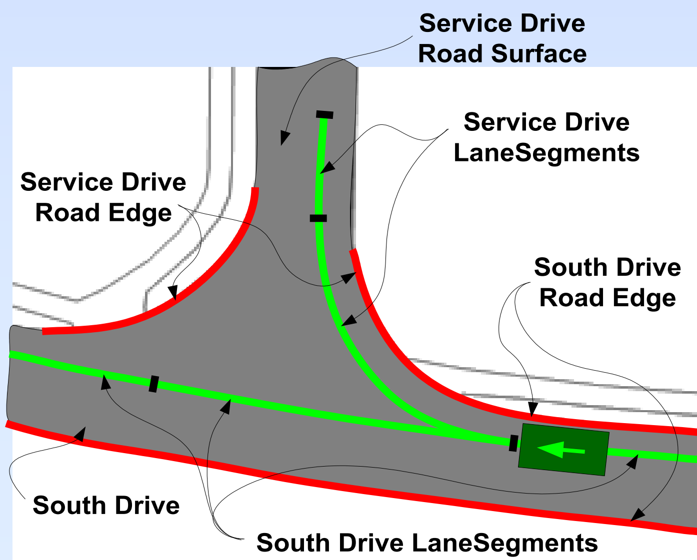



## NARRATIVE

Vehicle has detected Service Drive intersection and is now within the turning distance tolerance. This causes BG2 to send a TurnRight command to BG3 which will cause it to build goal paths along the lane segments forming the right turn onto Service Drive. This turn will be made at speed since the vehicle has the right-of-way at this intersection. This has been determined by the lack of detecting any traffic control devices for own lane of travel and the fact that Service Drive "tees" into South Drive which usually means that vehicles on South Drive will have the right-of-way.

BG2 will continuously update changes in the LaneSegments as measured by SP2 detecting road edges and surfaces. The curving road edges leading into Service Drive continue to be of interest and are, therefore, still in the BG3 ActiveObjectsTable and the SP2 ObjectsOfAttention list.

BG3 will continuously adapt the goal path to the LaneSegment changes from BG2 by controlling the real-time trajectory vector to Steer and Speed Control in BG4.




WORLD MODEL

# RCS Methodology

**This is a tedious process.**

**There are many tasks in the command library at each level**

**There are many parameters for each task**

**There are many objects that must be recognized**

**There are many situations that must be understood**

**But, the numbers are not infinite. They are, in fact, quite modest. (One of the advantages of hierarchies.)**

# Autonomous On-Road Driving
# Vehicle Echelon and Down

**Estimated numbers:**

**~ 200 tasks**
**~ 100 parameters**
**~ 1000 transition conditions**
**~ 10,000 objects or events**

**Other skills may require similar numbers**

# Summary

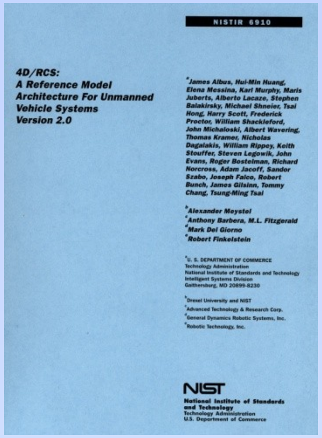
**4D/RCS Reference Model Architecture has a proven success record for intelligent control**

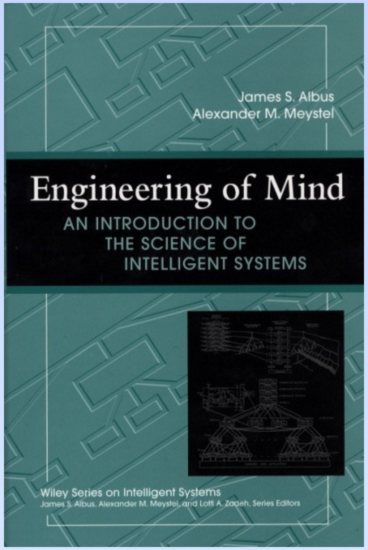
**4D/RCS Methodology provides a systematic approach to software engineering for tactical behaviors**

# 4D/RCS Documentation

**4D/RCS Version 2.0**

- NIST Report, 2002



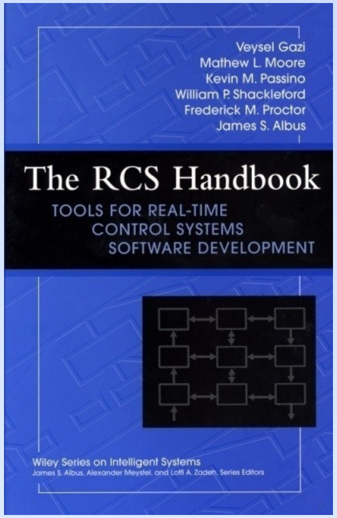




**Engineering of Mind**

- Wiley, 2001

**RCS Handbook**

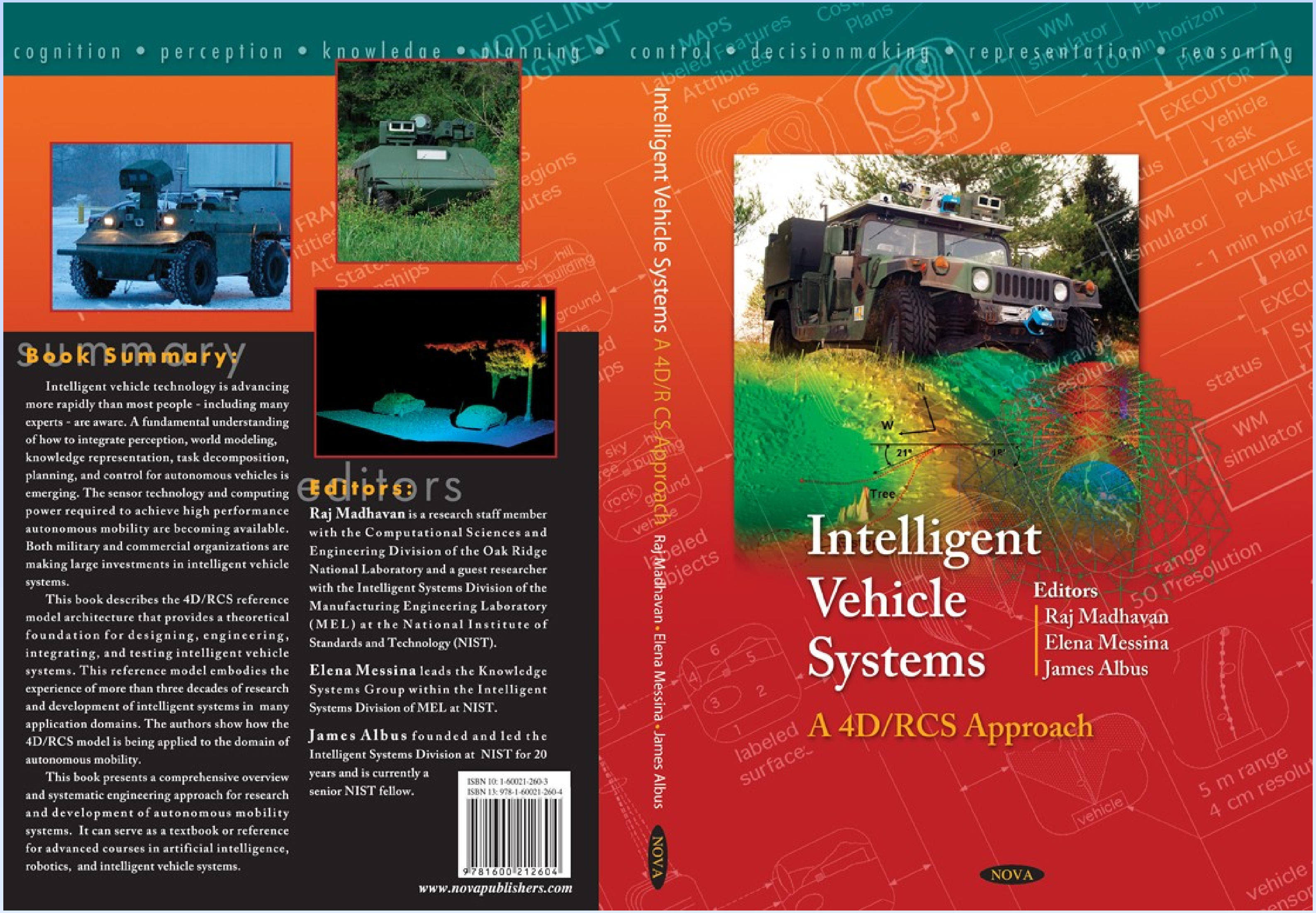
– Wiley, 2001




**Numerous journal articles, reports, and conference papers**

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

# Most Recent Publication 2007



# Conclusions

1. Useful autonomous on-road and off-road driving will be feasible by 2010

2. Human level performance in autonomous on-road and off-road driving will be feasible by 2020

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

# Questions?