Behaving in Space

How Robots Play Soccer

Thomas Röfer

Bremen Institute of Safe Systems

Center for Computing Technology (TZI)

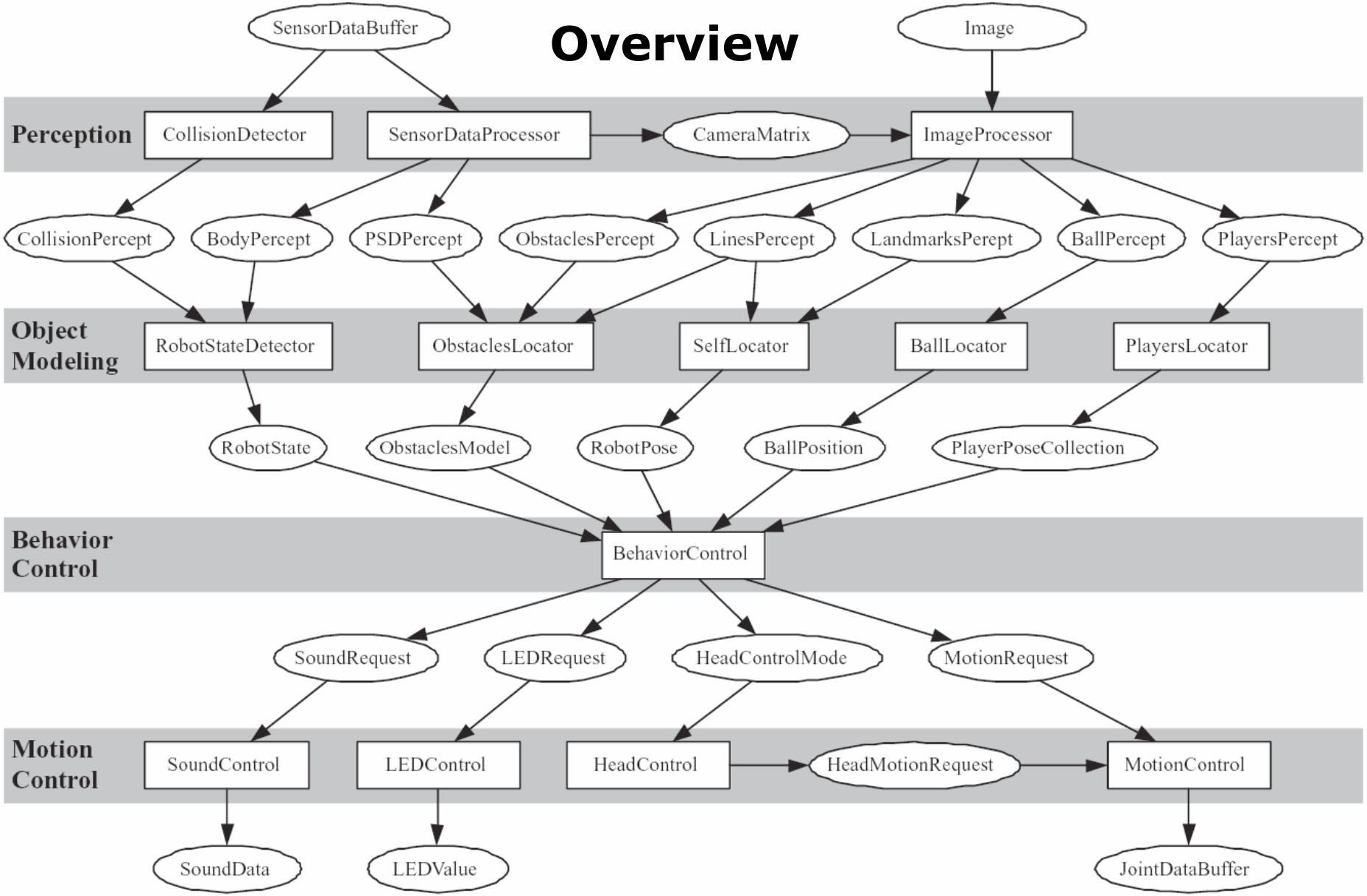
Universität Bremen

RoboCup – The Goal

- By the year 2050,
develop a team of fully autonomous humanoid robots
that can win against the human world soccer champion team. -

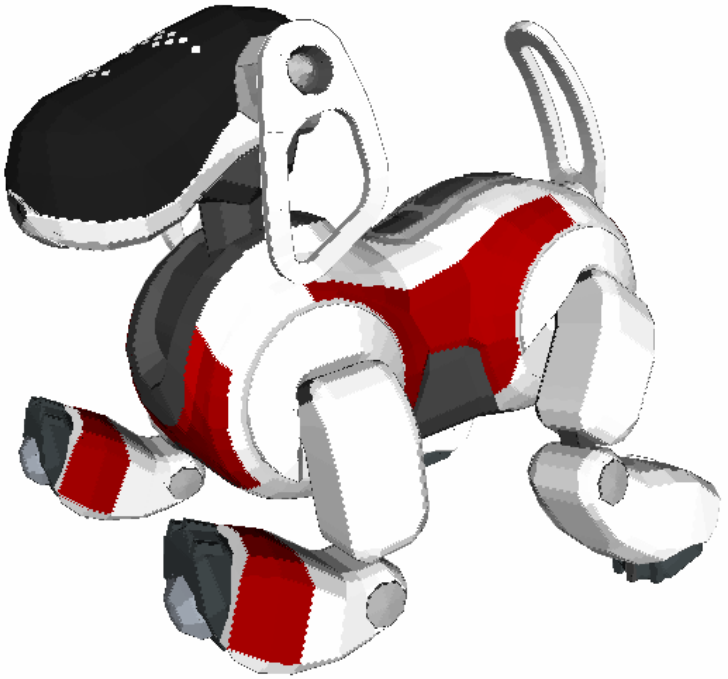
Goals in the Final

Overview


SensorDataBuffer
Image
Perception
CollisionDetector
SensorDataProcessor
CameraMatrix
ImageProcessor
CollisionPercept
BodyPercept
PSDPercept
ObstaclesPercept
LinesPercept
LandmarksPerept
BallPercept
PlayersPercept
Object Modeling
RobotStateDetector
ObstaclesLocator
SelfLocator
BallLocator
PlayersLocator
RobotState
ObstaclesModel
RobotPose
BallPosition
PlayerPoseCollection
Behavior Control
BehaviorControl
SoundRequest
LEDRequest
HeadControlMode
MotionRequest
Motion Control
SoundControl
LEDControl
HeadControl
HeadMotionRequest
MotionControl
SoundData
LEDValue
JointDataBuffer

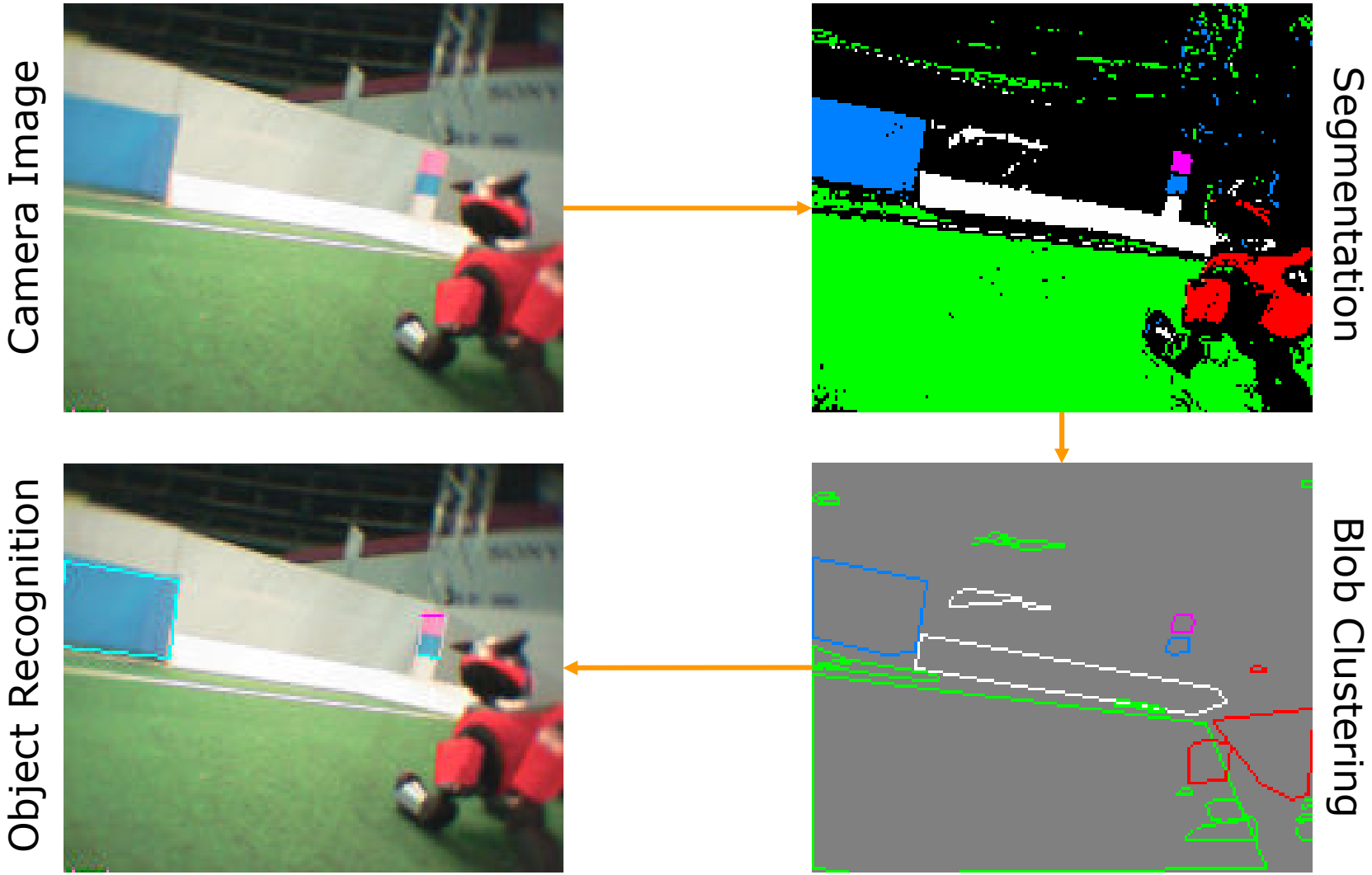
Sony Aibo ERS-7

- 20 degrees of freedom (can also be measured)
  - 3 per leg
  - pan and 2x tilt in the neck
  - chin, 2 ears, roll and tilt tail
- CCD camera
  - 416x320 pixels (brightness)
  - 208x160 pixels (color)
- Further sensors
  - 4 ground contact sensors
  - 3 PSD infrared sensors
  - 3 acceleration sensors
- 585 MHz Mips processor
- 802.11b wireless network



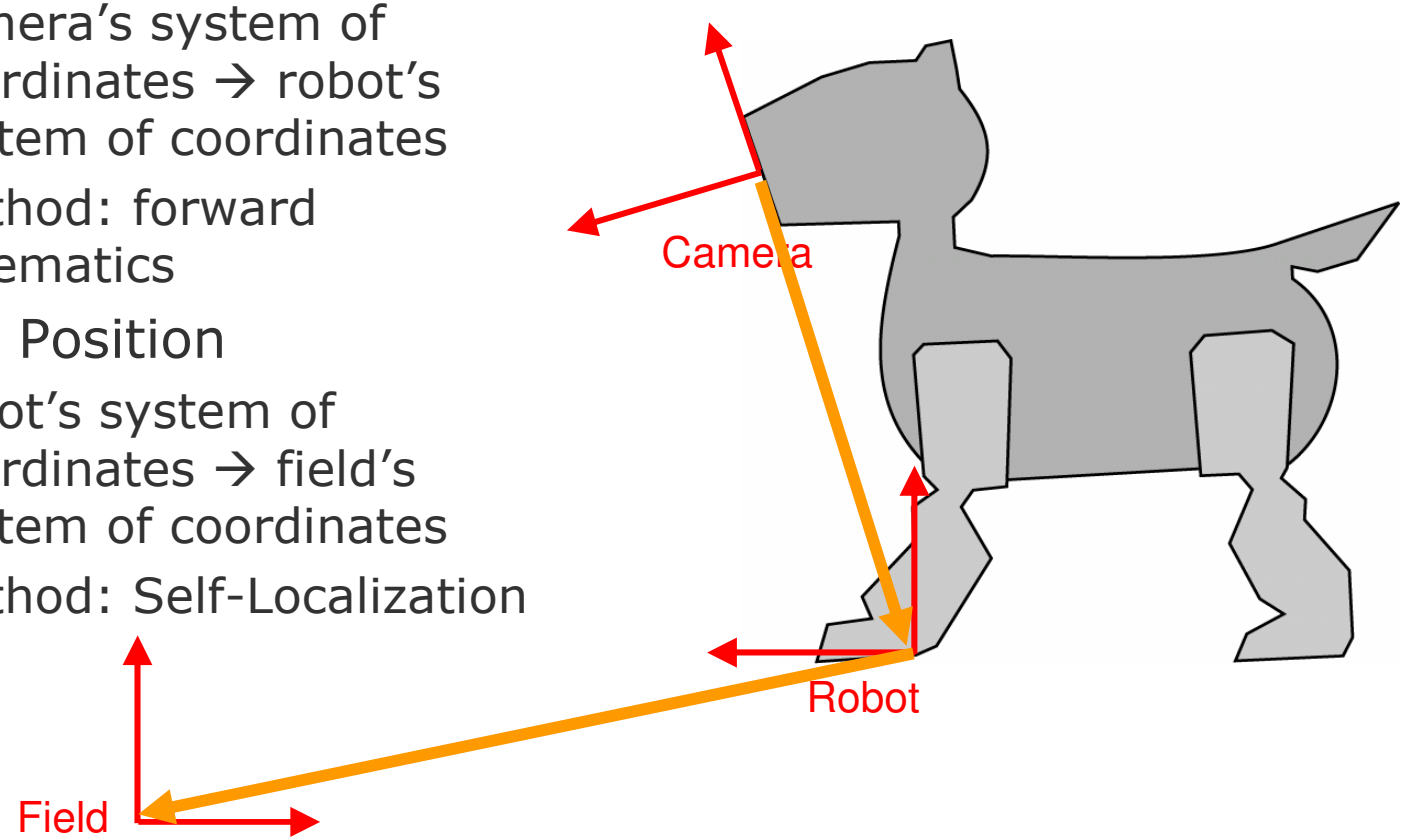
30 Seconds in the Life of an Aibo

Typical Image Processing Pipeline

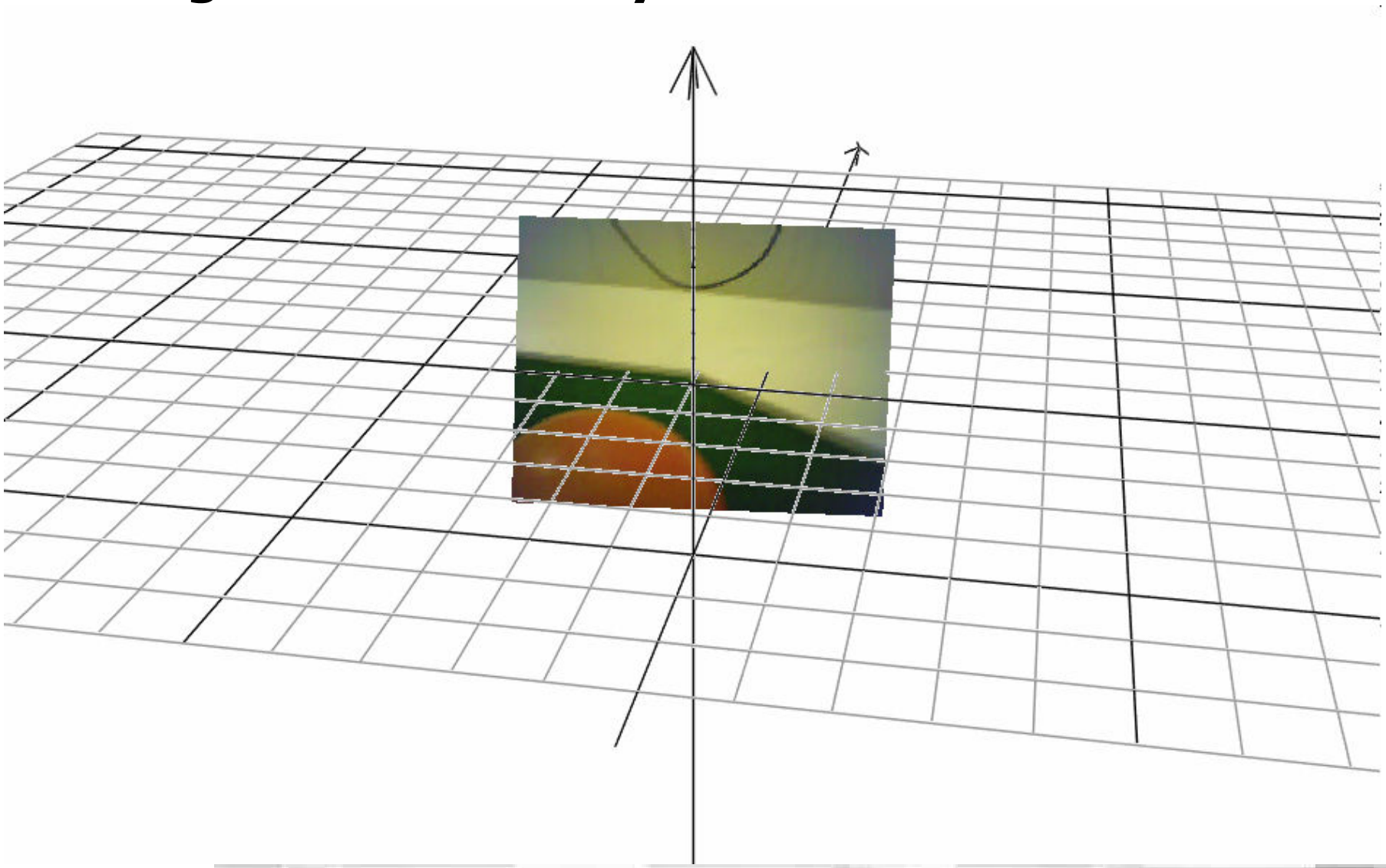


Systems of Coordinates

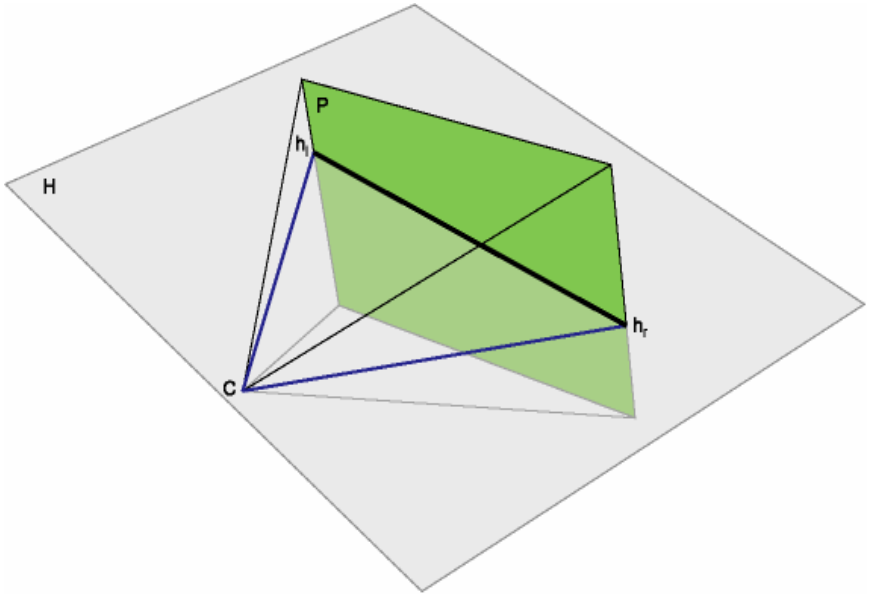
- Camera Position
  - camera's system of coordinates → robot's system of coordinates
  - Method: forward kinematics
- Robot Position
  - robot's system of coordinates → field's system of coordinates
  - Method: Self-Localization



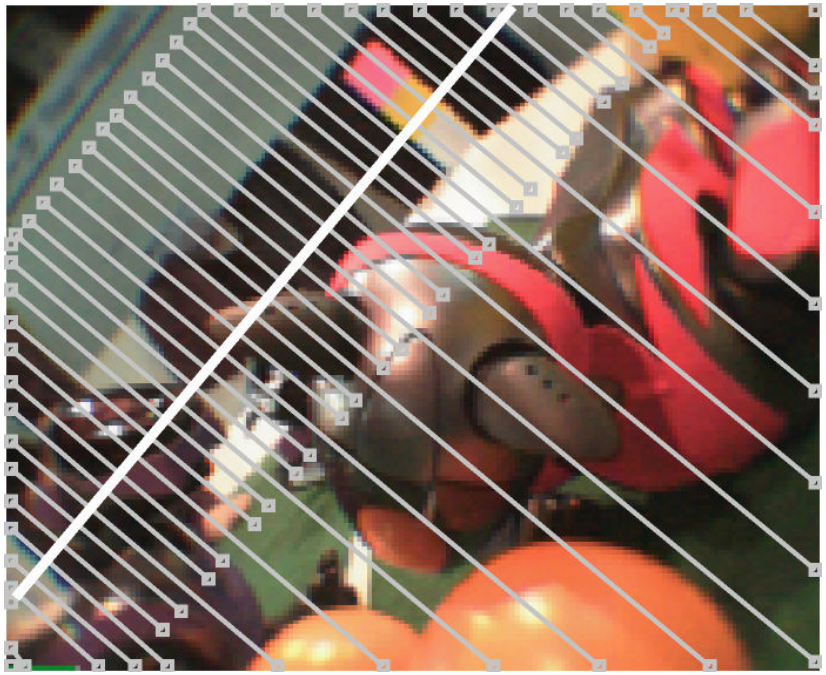
Images in Robot's System of Coordinates



Prespective-Based Image Processing

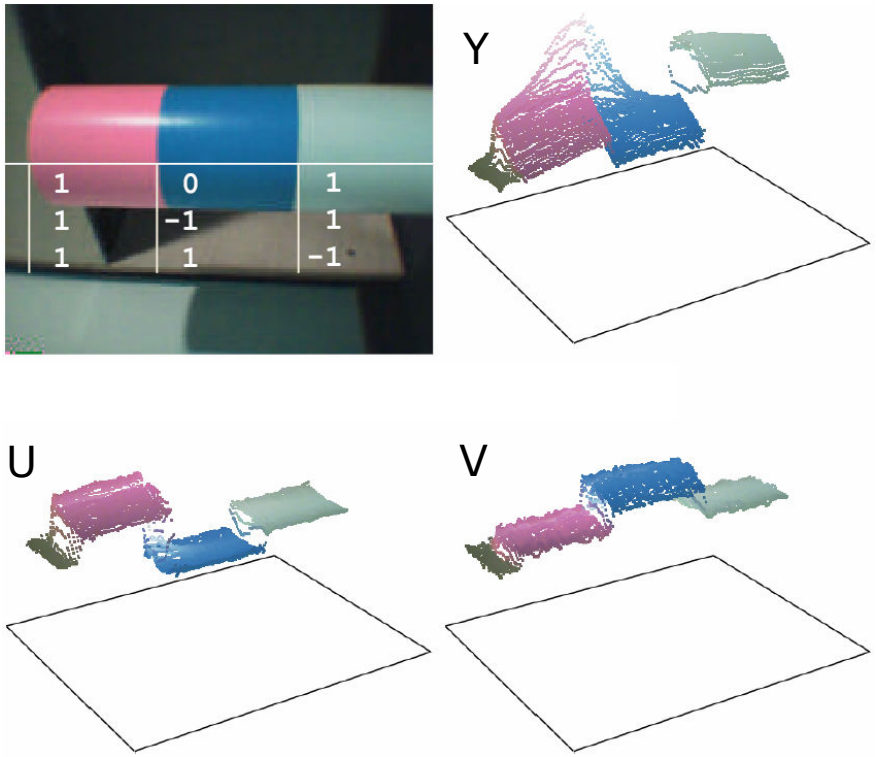


Horizon

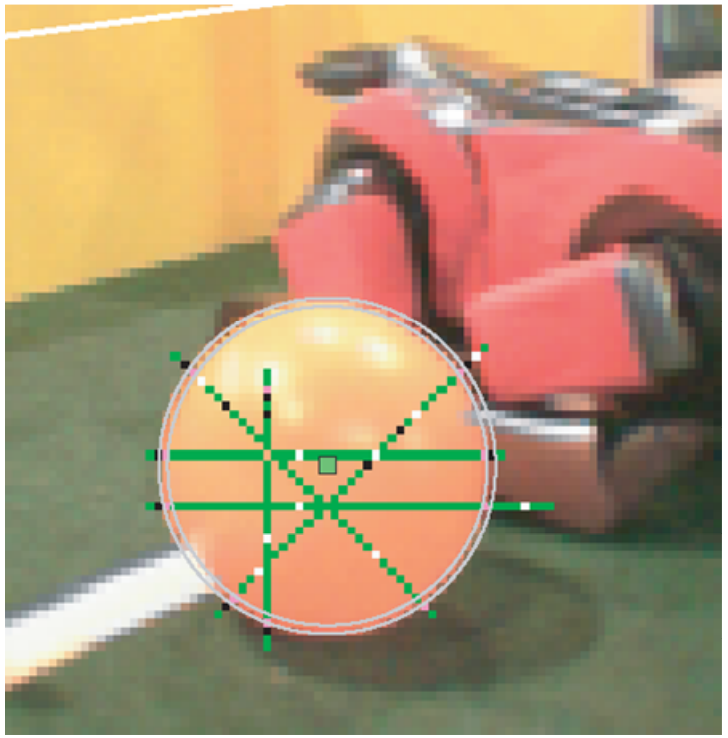


Horizon-aligned grid

Anchor Points and Specialists

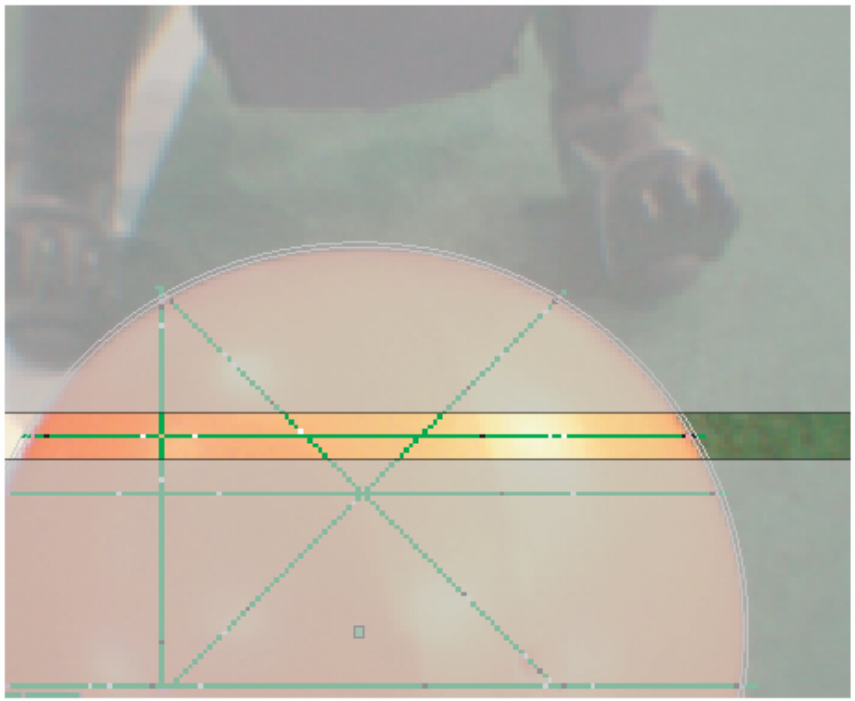


Pattern of a beacon

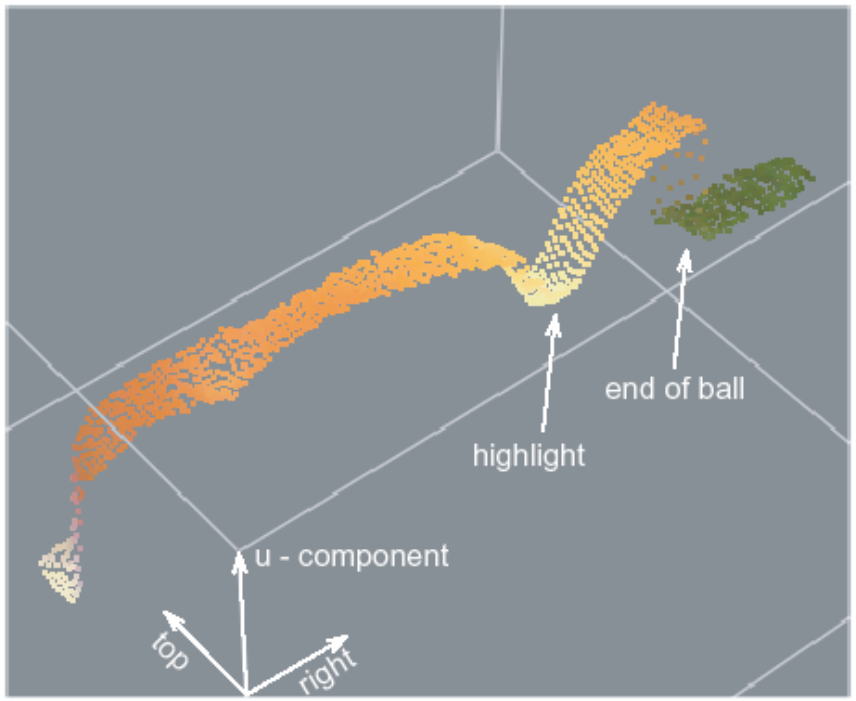


Ball specialist

Using Gradients



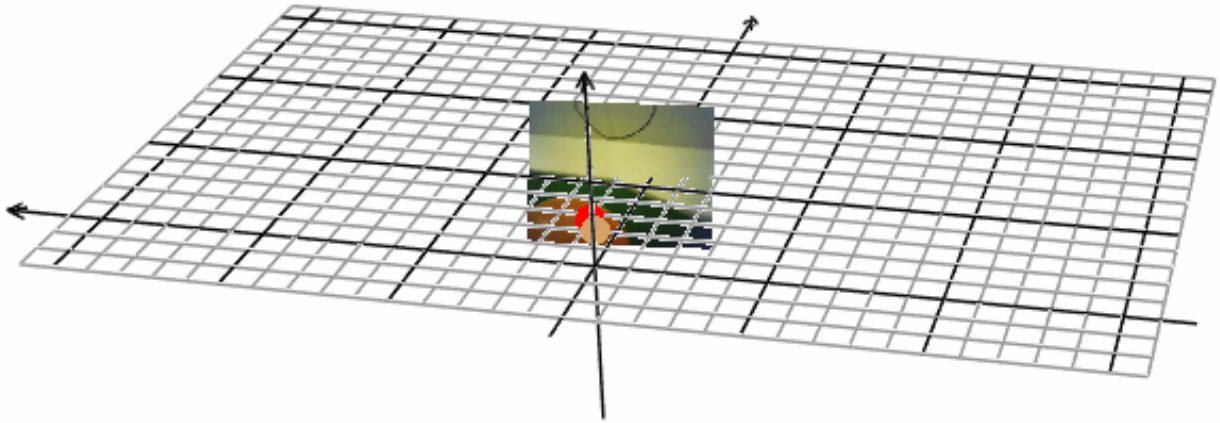
One scanline of the specialist



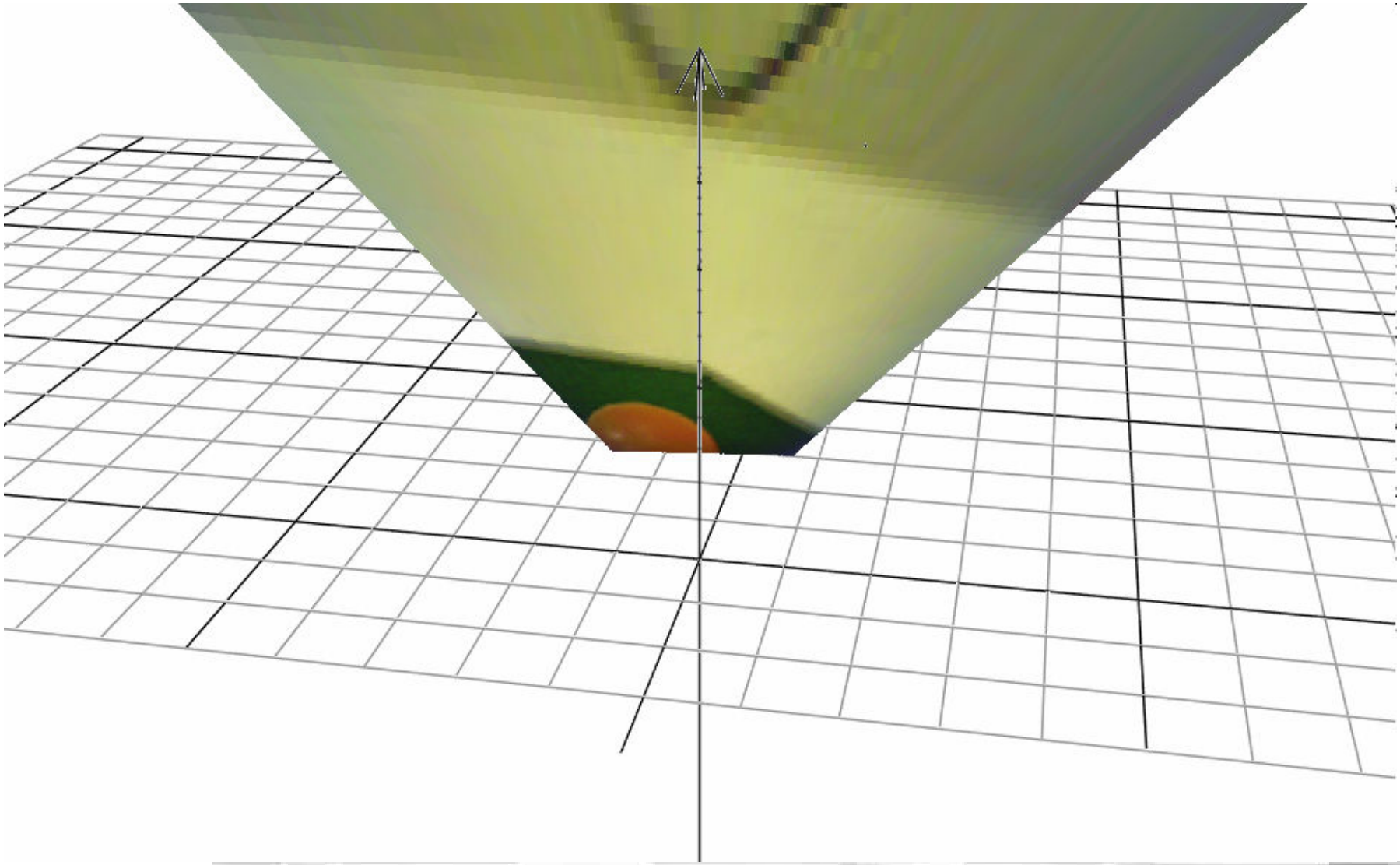
U-channel on scanline

Size-Based Positions of Percepts

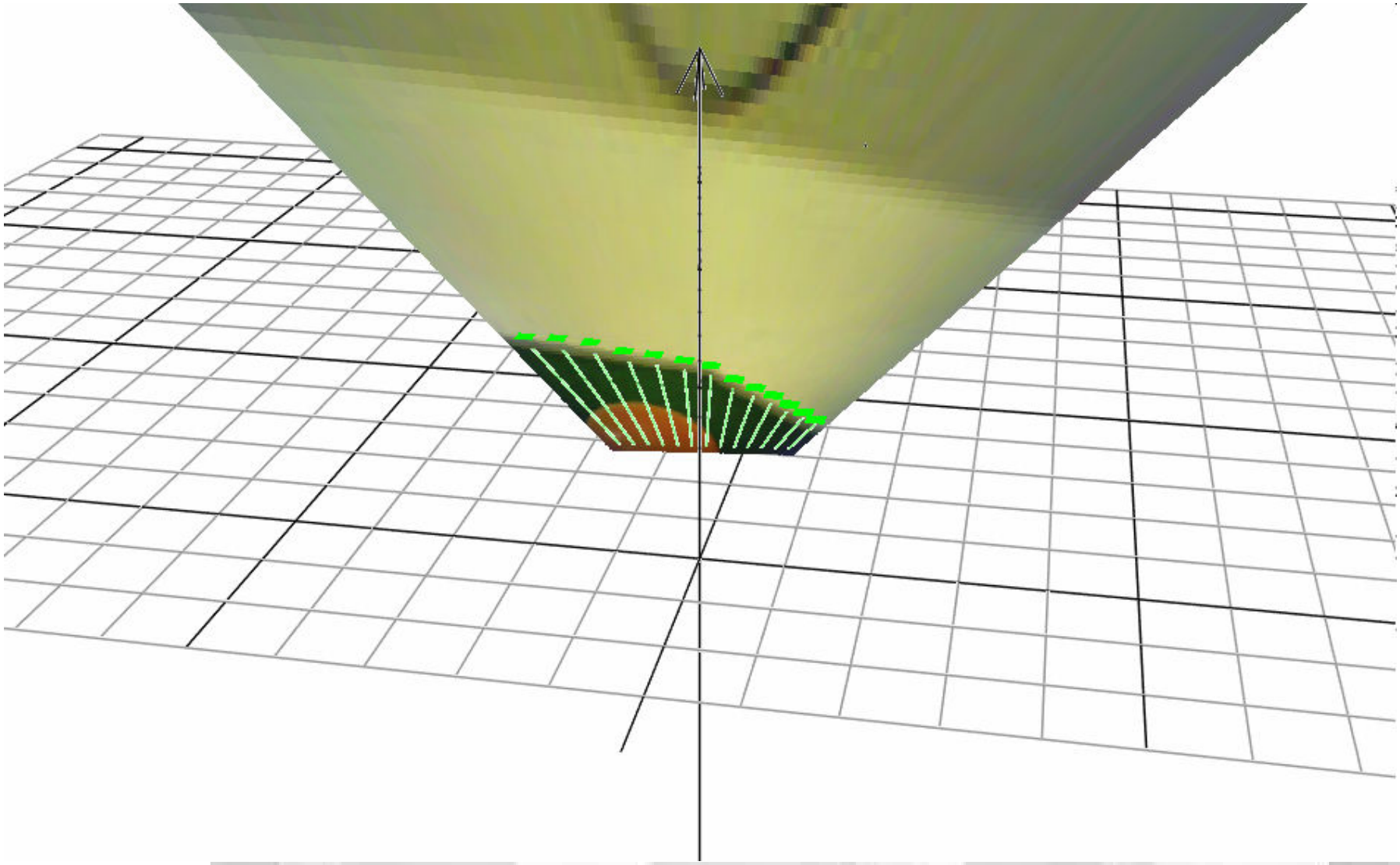
- Egocentric Positions of Percepts
  - direction (from images and camera position)
  - distance (from object size)



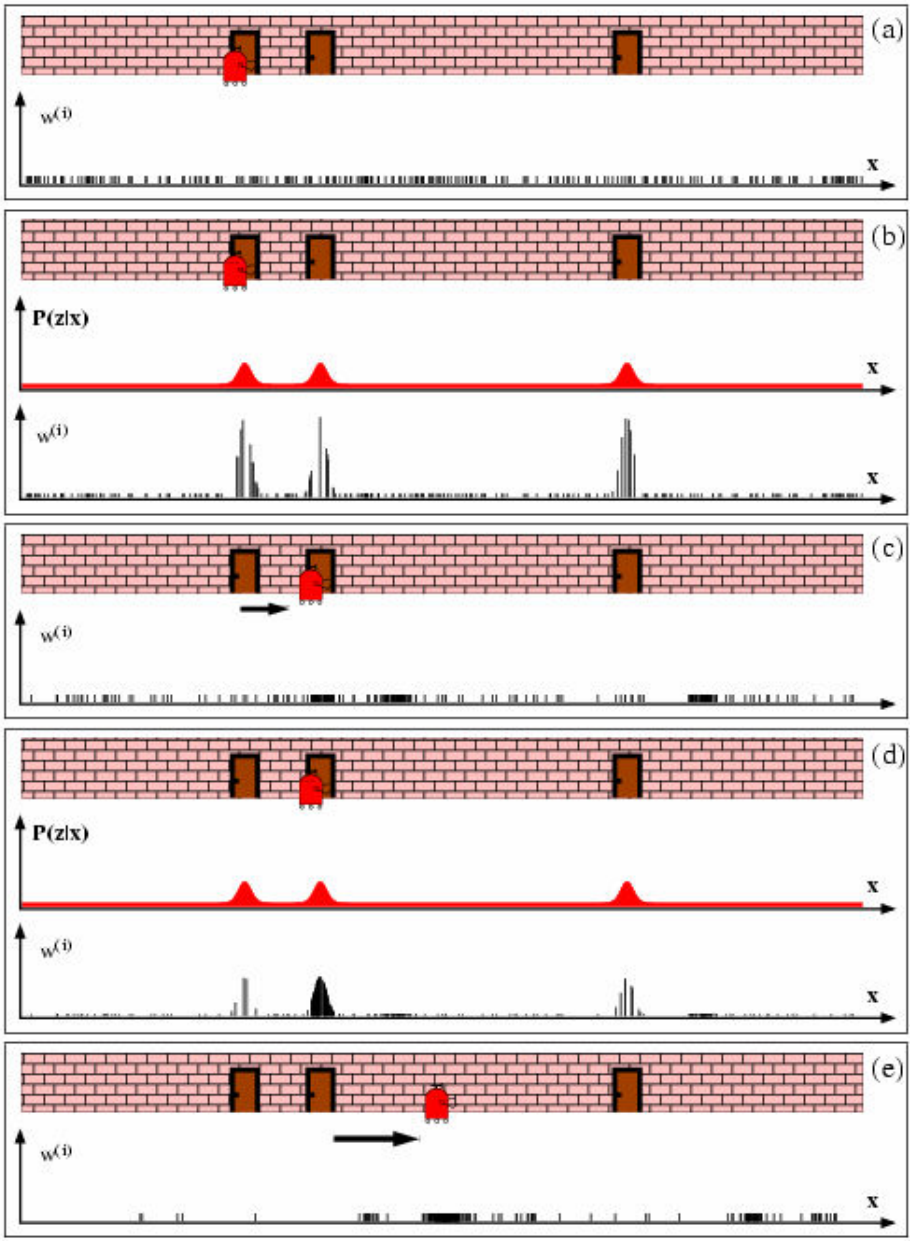
Projection-Based Positions of Percepts

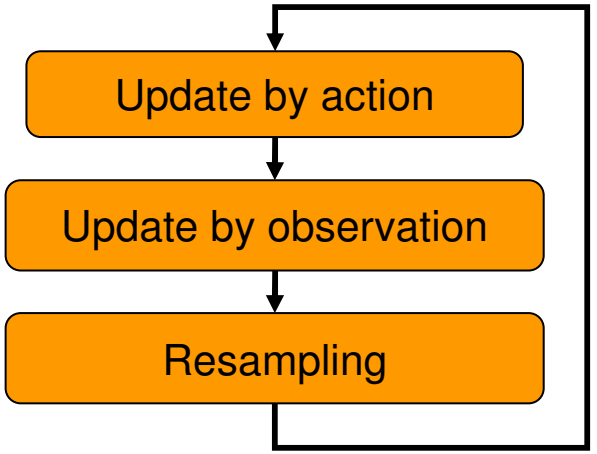


Projection-Based Positions of Percepts

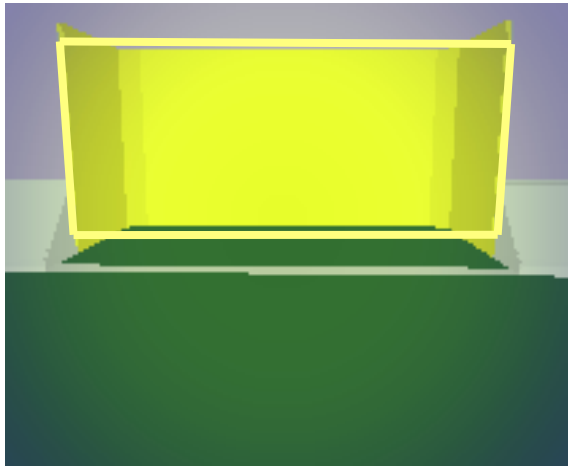


Self-Localization – Particle Filter

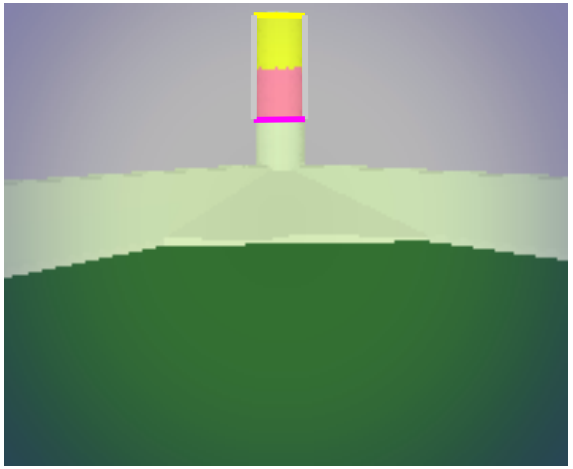




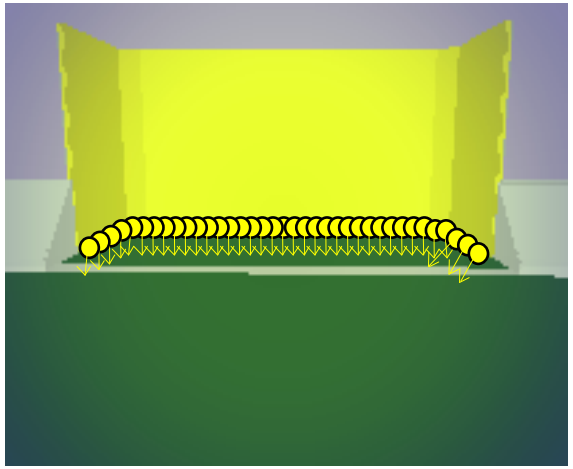
Self-Localization – Percepts Used



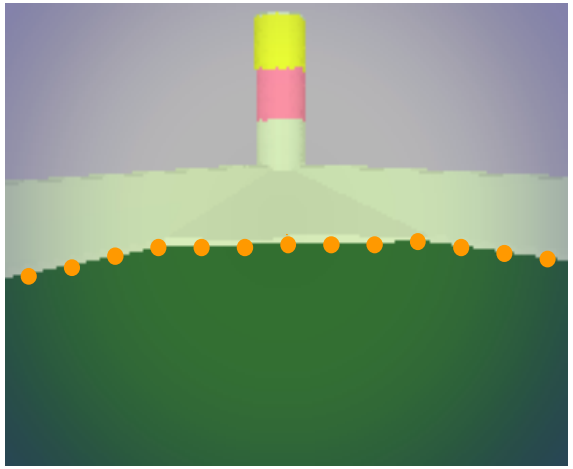
Goals



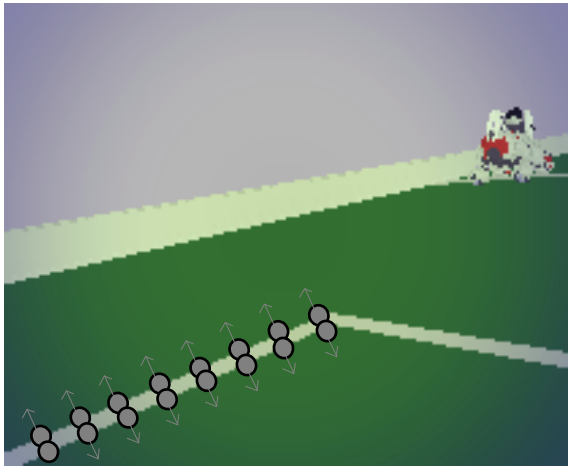
Beacons



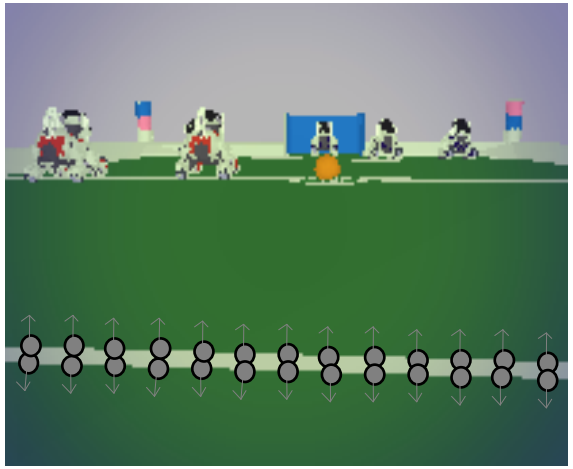
Goal points



Edges field/wall



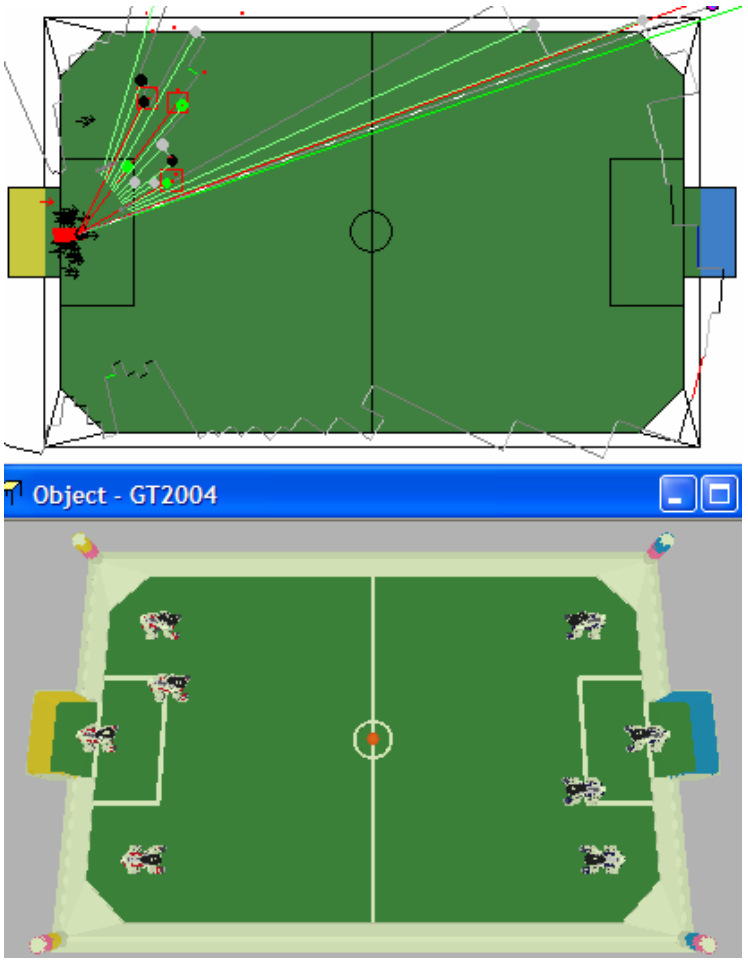
Field lines (vert.)



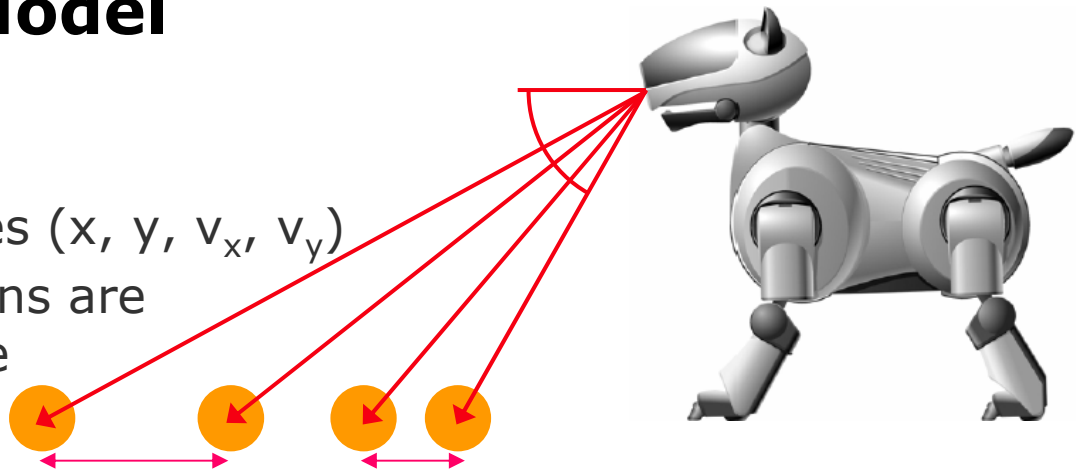
Field lines (horiz.)

Self-Localization – Details

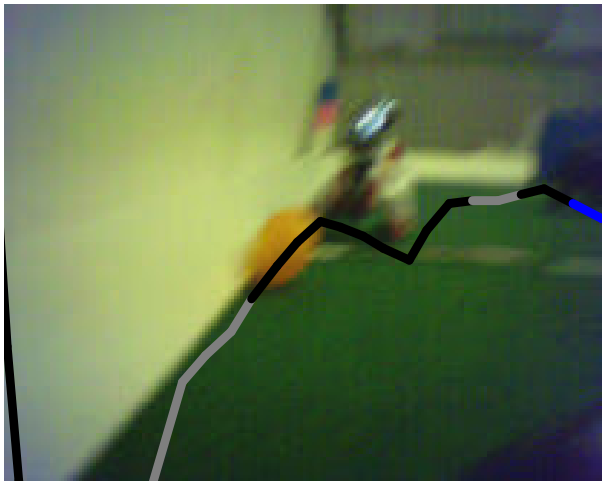
- Probability of samples
  - Probability is adapted slowly
  - Separate probabilities for different edge types
  - Samples are randomly moved, weighted by their probabilities
- Sensor resetting
  - Draw samples based on the ratio of their probability and the average probability
  - Replace them by candidate postures that can be derived from observations
- Calculating candidates in advance
  - A large number of random postures is generated
  - Their distance to the edge they are pointing to is determined
  - The postures are indexed by their distance and edge type

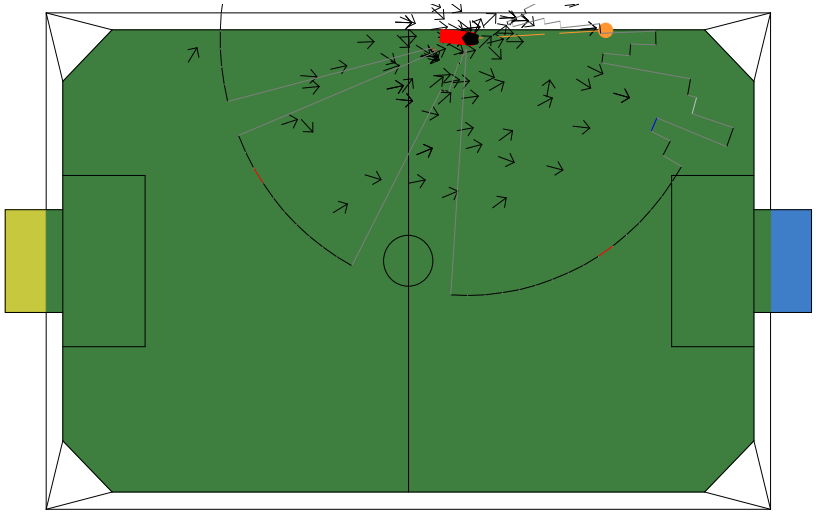


Obstacle and Ball Model

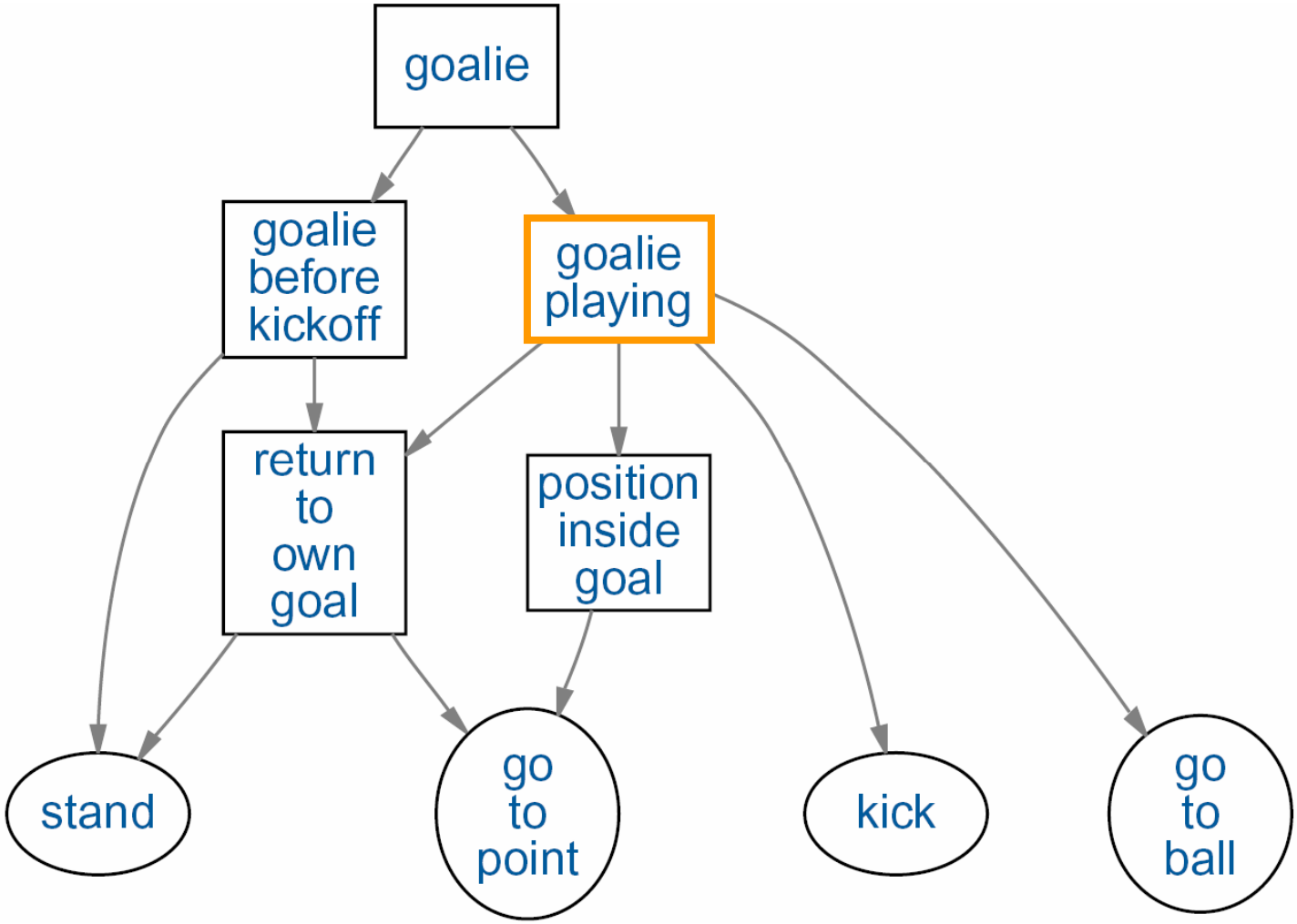


- Ball Model
  - Kalman Filter estimates (x, y, v_x, v_y)
  - Communicated positions are only used after a while
- Obstacle Model
  - Polar representation of free space
  - Border of free space is labeled by obstacle types

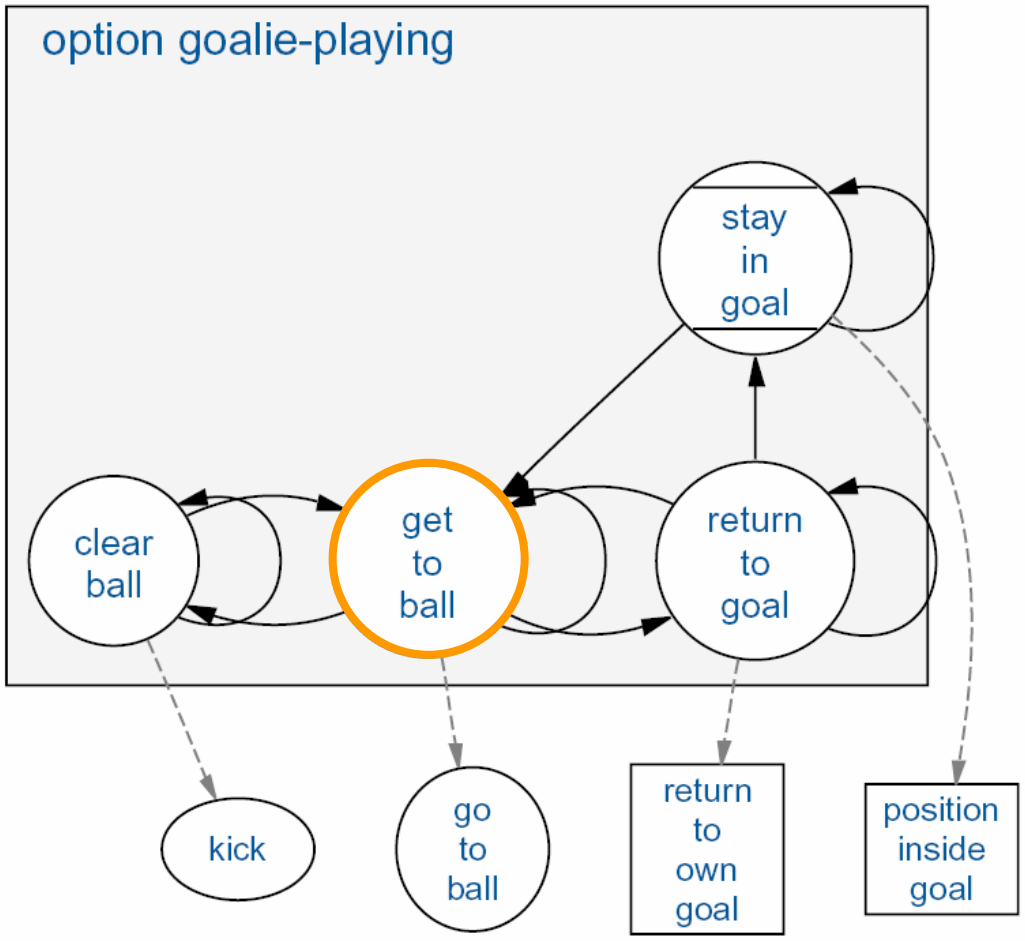


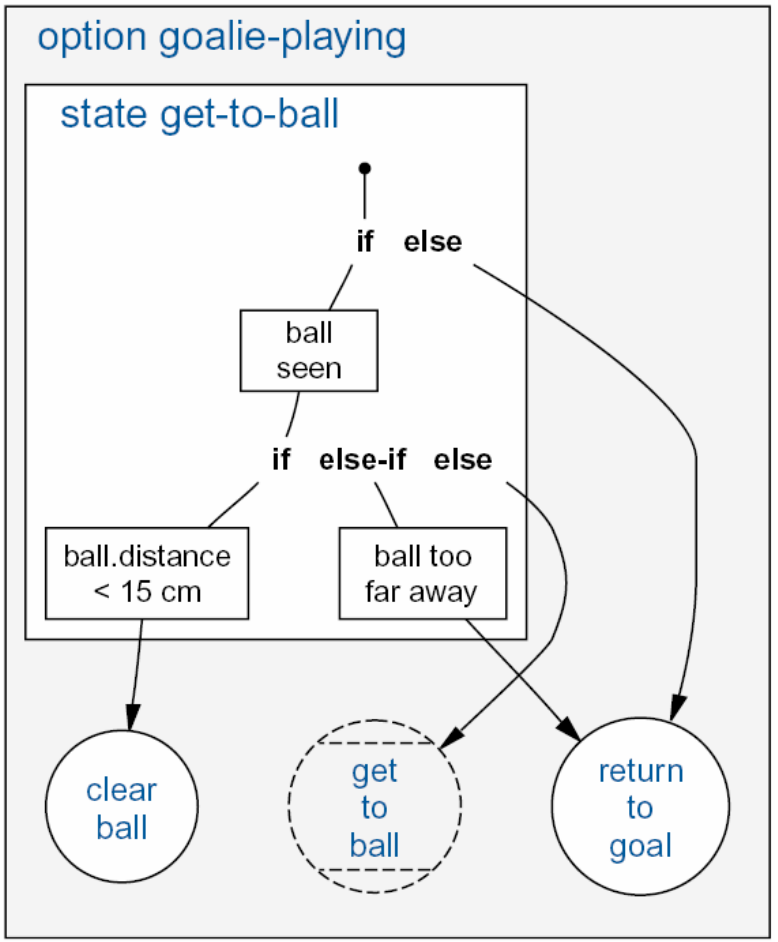


Extensible Agent Behavior Specification Language (XABSL)



XABSL – Options and States

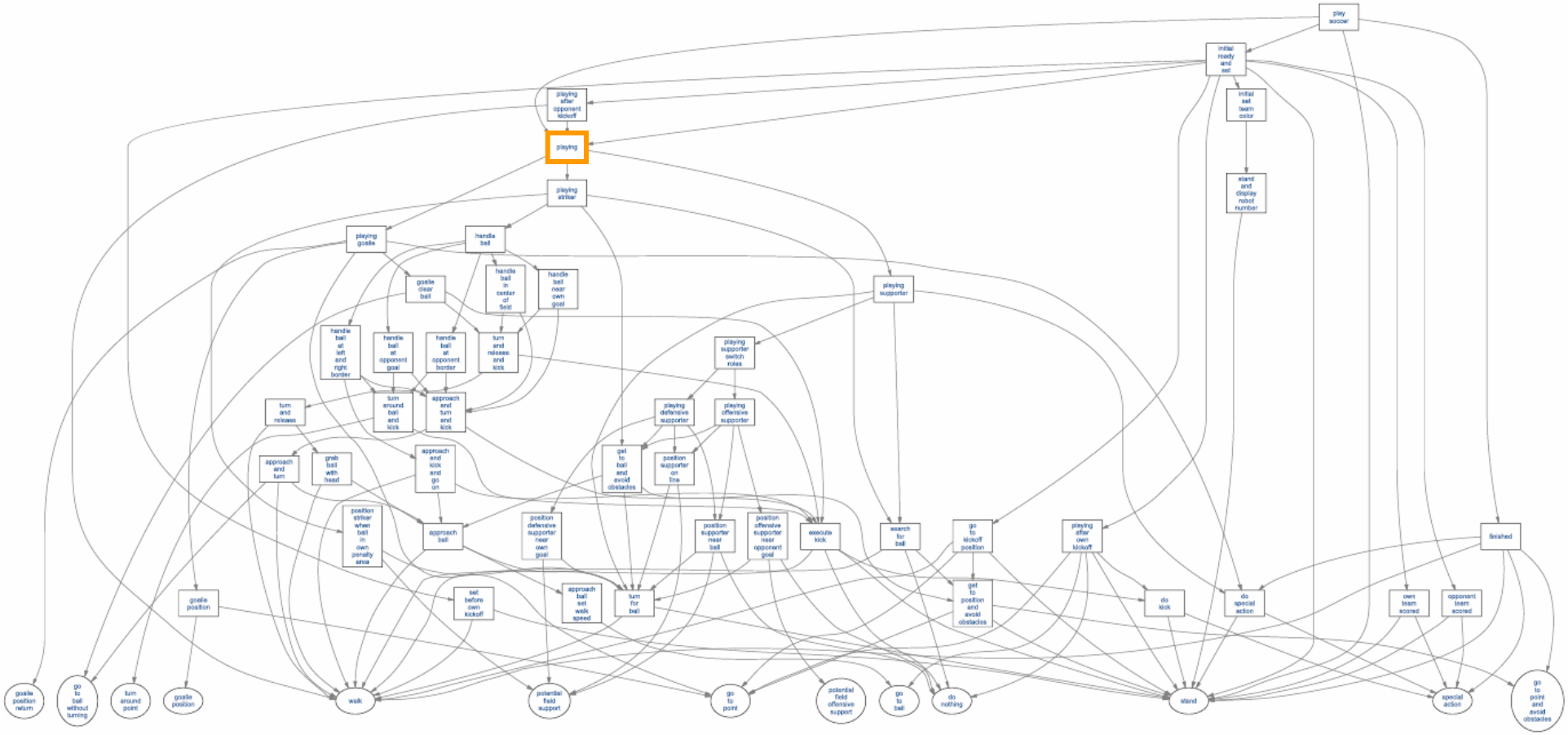




XML Source

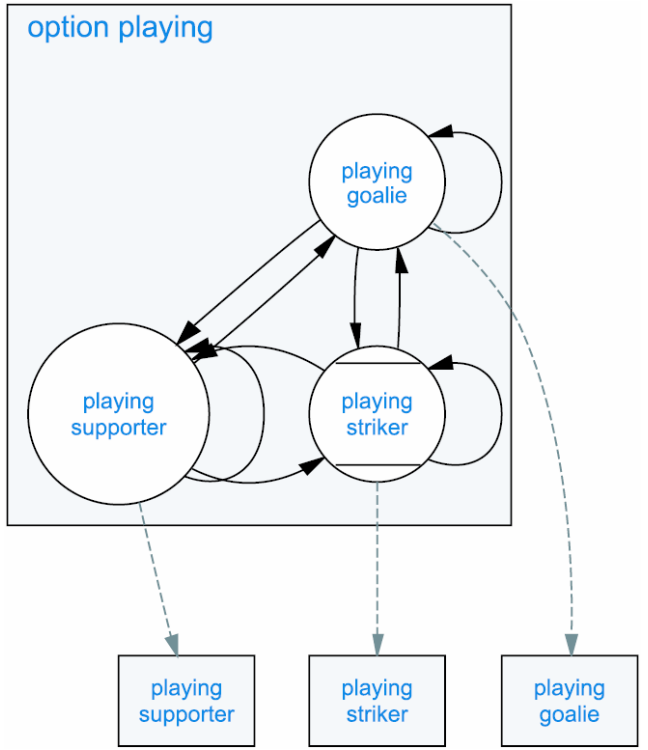
```
<option name="goalie-playing" initial-state="stay-in-goal"
        description="goalie playing behavior">
 <state name="get-to-ball">
  <subsequent-basic-behavior ref="go-to-ball"/>
  <set-output-symbol ref="head-control-mode"
              value="search-for-ball"/>
  <decision-tree>
   <if>
    <condition description="ball seen">
     <less-than>
      <decimal-input-symbol-ref
       ref="ball.time-since-last-seen"/>
      <decimal-value value="2000"/>
     </less-than>
    </condition>
    <if>
     <condition description="ball kickable">
      <less-than>
       <decimal-input-symbol-ref ref="ball.distance"/>
       <decimal-value value="150"/>
      </less-than>
     </condition>
      <transition-to-state ref="clear-ball"/>
    </if>
    <else-if>
     <condition description="ball too far away">
      <greater-than>
       <decimal-input-symbol-ref ref="ball.distance"/>
       <decimal-value value="900">
      </greater-than>
     </condition>
     <transition-to-state ref="return-to-goal"/>
    </else-if>
    <else>
     <transition-to-state ref="get-to-ball"/>
    </else>
   </if>
   <else>
    <transition-to-state ref="return-to-goal"/>
   </else>
  </decision-tree>
 </state>
 ...
</option>
```

XABSL-Graph of GermanTeam 2004

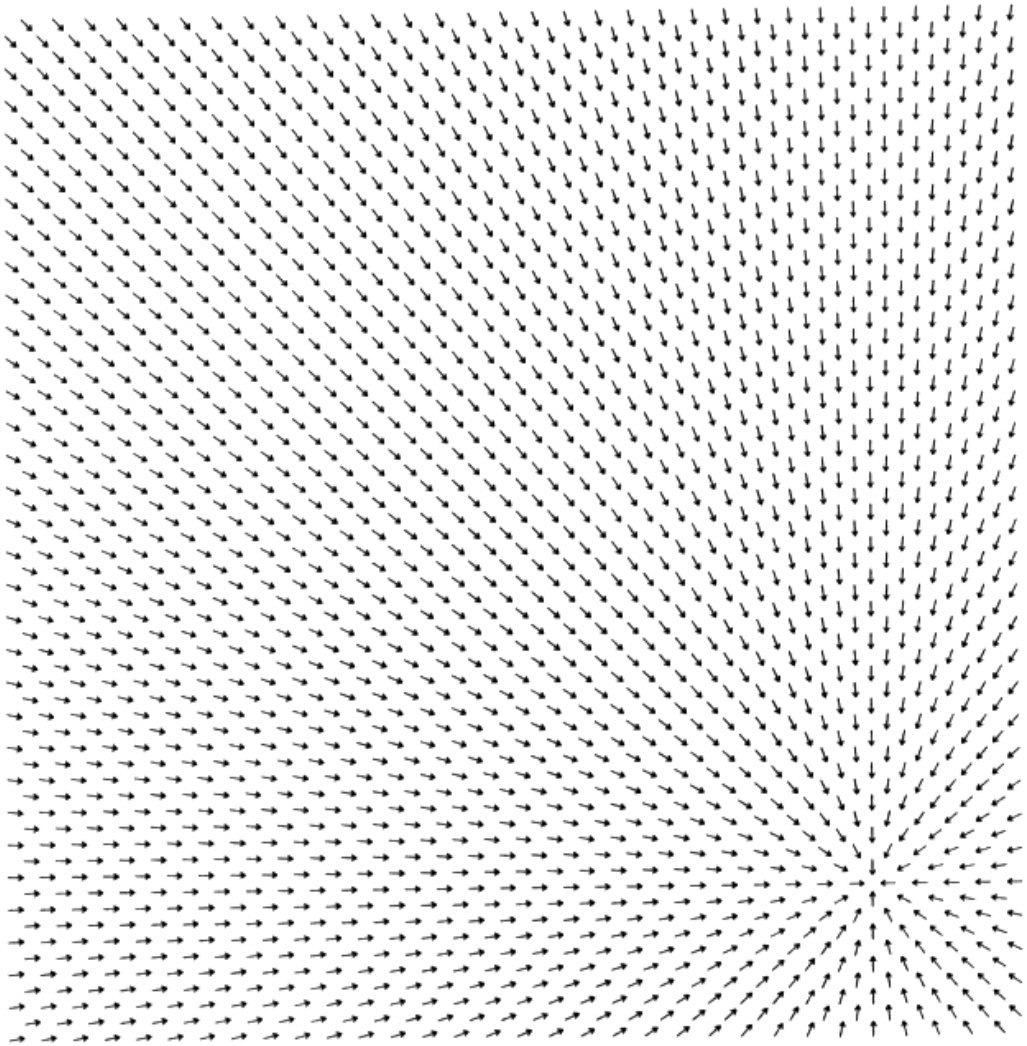


Dynamic Role Assignment

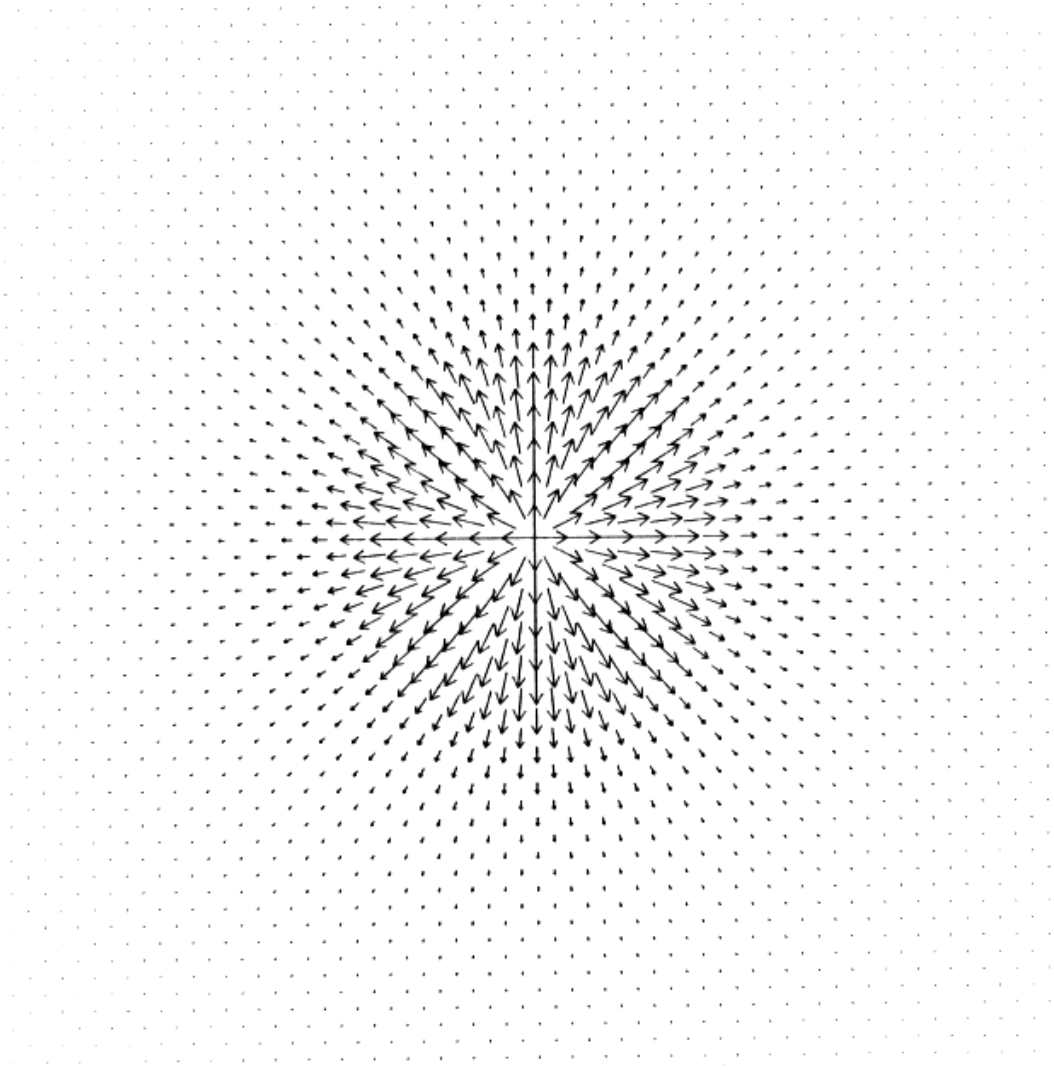
- Roles
  - Goalie (defined by rules)
  - Striker, Offensive Supporter, Defensive Supporter
- Assigning Role „Striker“
  - Computed by all robots and sent to teammates:
    estimatedTimeToReachBall =
    5 s/m × *distanceToBall*
    + 1.25 s × ∠(*ball, opponentGoal*) / π
  - Minimum of:
    estimatedTimeToReachBall
    - 0.5 s × *isAlreadyStriker*
    + 2 × *timeSinceBallWasSeenLast*
- Assigning Role „Offensive Supporter“
  - Minimum of:
    distanceToOpponentGoalLine
    - 0.3 m × *isAlreadyOffensiveSupporter*



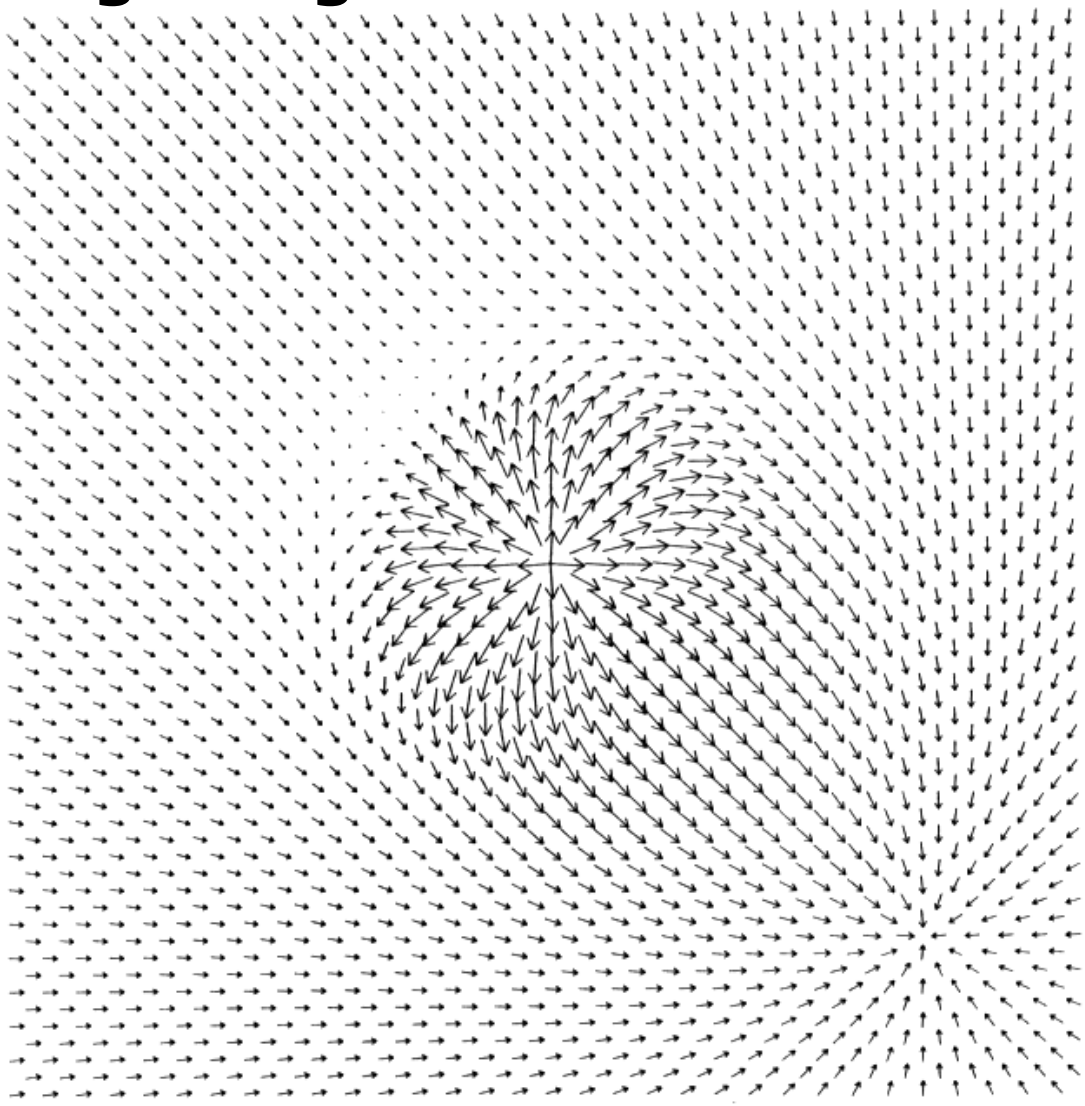
Positioning using Potential Fields



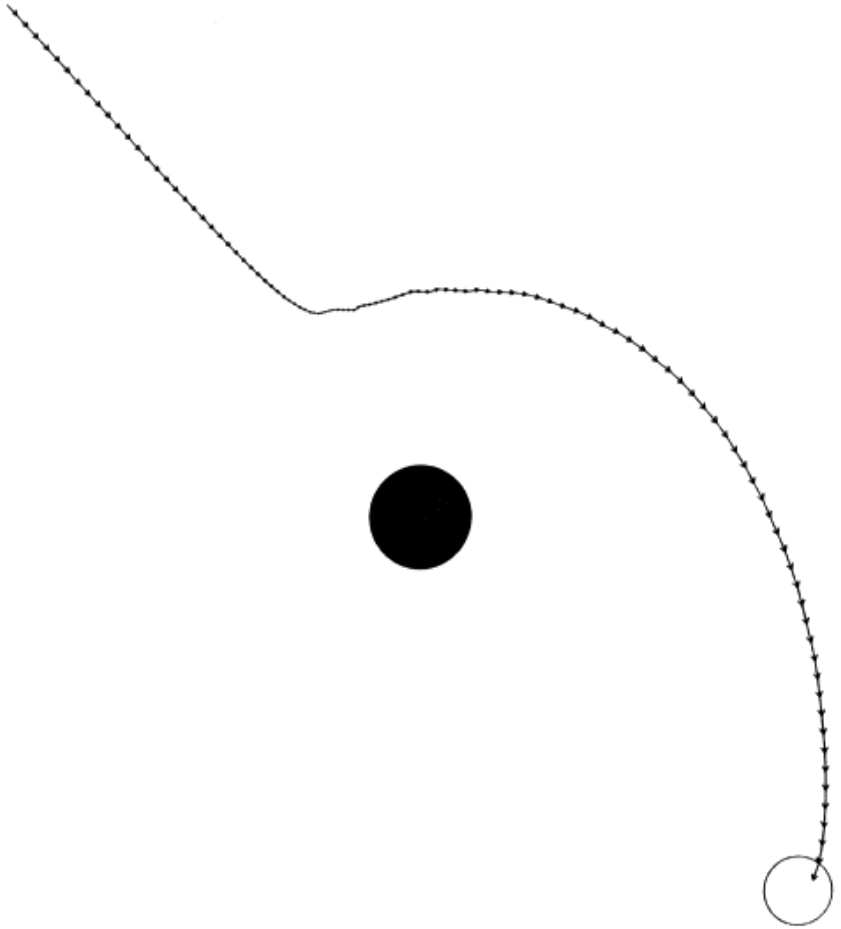
Positioning using Potential Fields



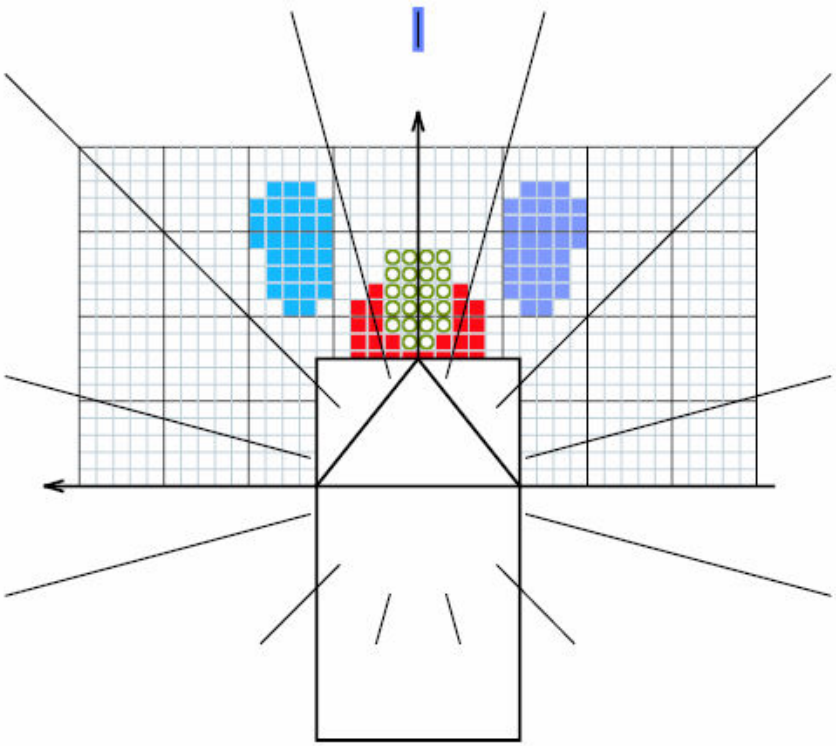
Positioning using Potential Fields



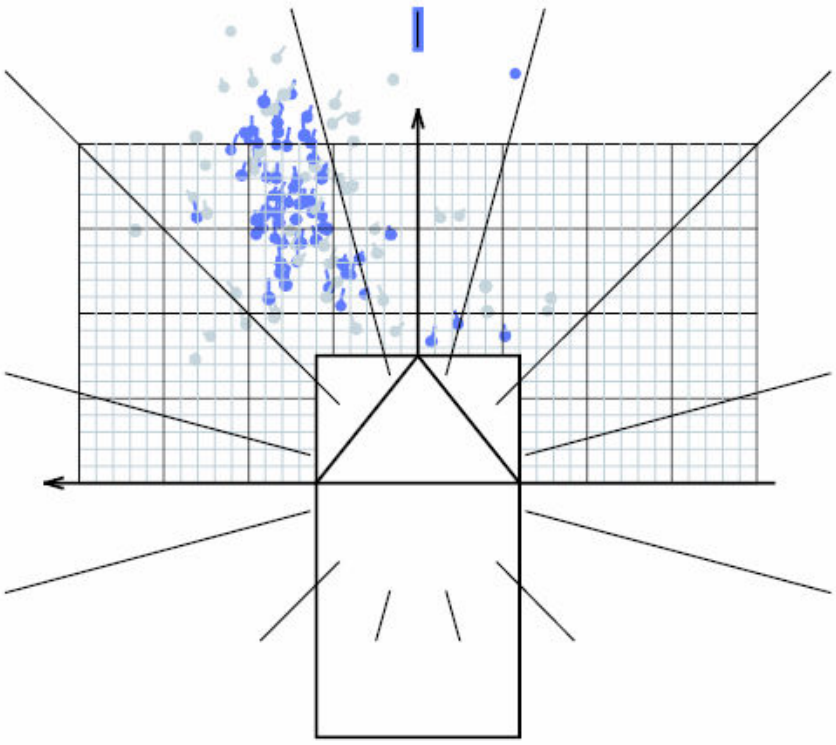
Positioning using Potential Fields



Reactive Kick Selection



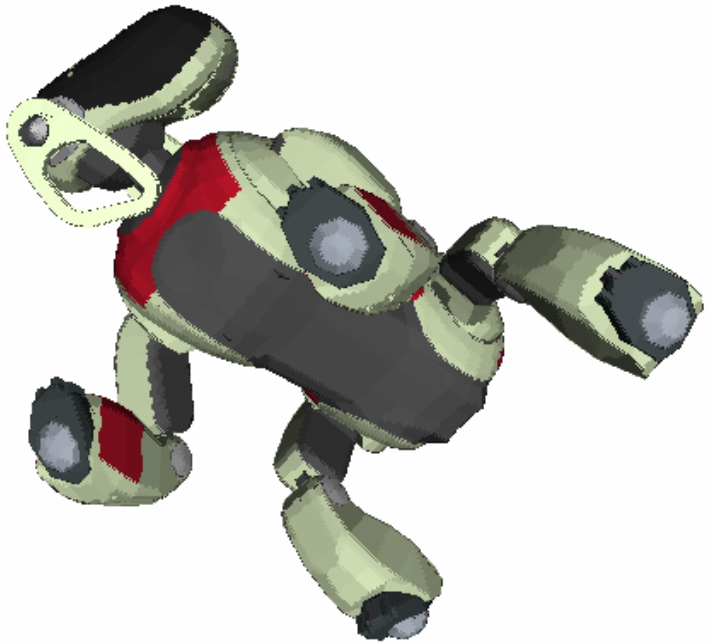
Kicking forward



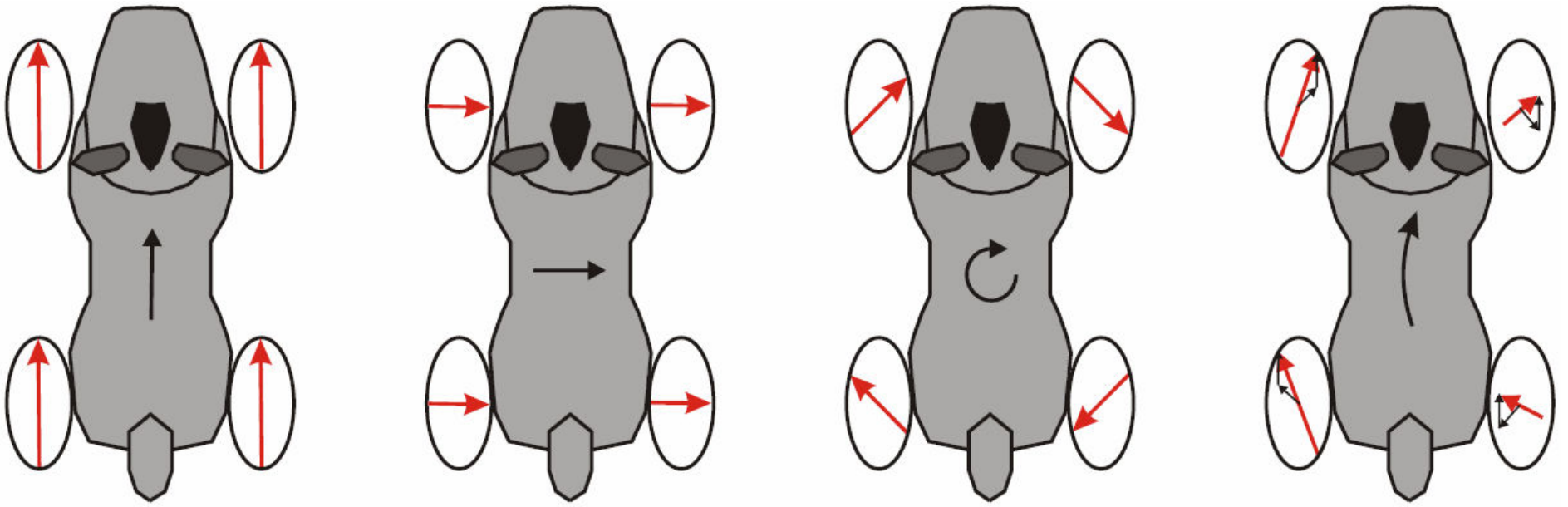
Data recorded for left paw kick

Motion Control

- Head Control
  - Symbolic modes
  - Separate XABSL engine
- Walking Engine
  - Egocentric speed vector: (x, y, θ)
- Getup Engine
  - Getting up after the robot fell down
  - Based on accelerator measurements
- Wakeup Engine
  - Getting up after the robot was started
- Special Actions
  - Kicks, cheering moves
  - Fixed joint angle sequences



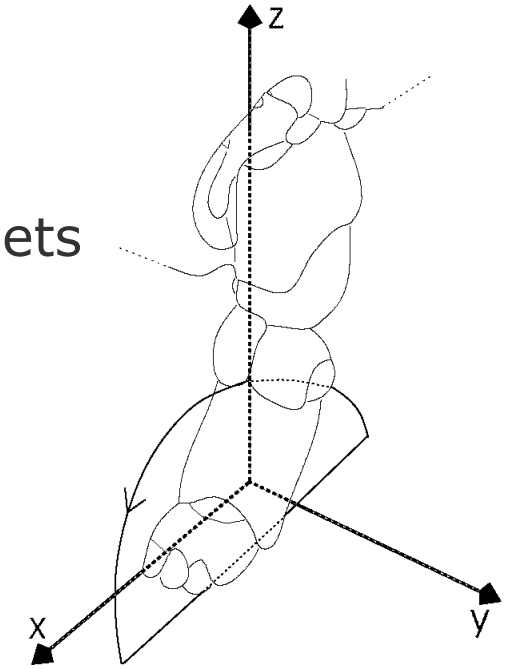
Omni-Directional Walking

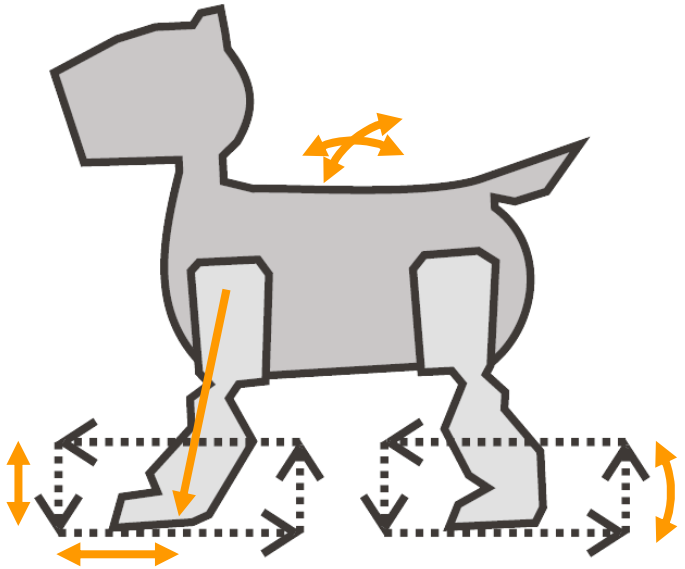


Forward | Sideward | Turning | Omni-directional

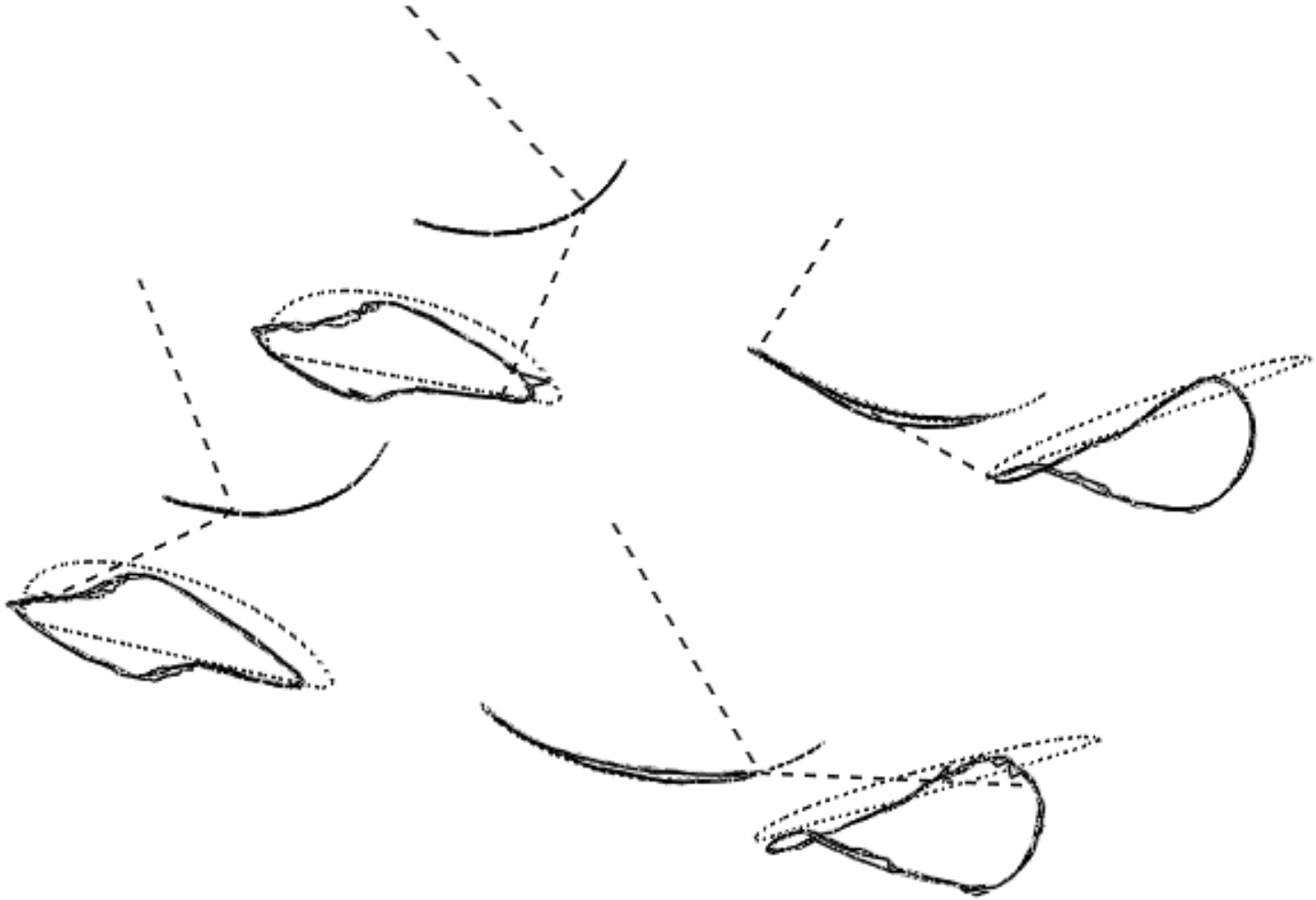
Gait Parameters

- Front/Rear locus
  - x, y, z offsets
  - step height
  - tilt
  - ground, lift, air, and lowering phases
- Step
  - size
  - duration
- Rear to Front offsets
  - x speed ratio
  - phase shift
- Body shift
  - x and y ratios
  - phase offset

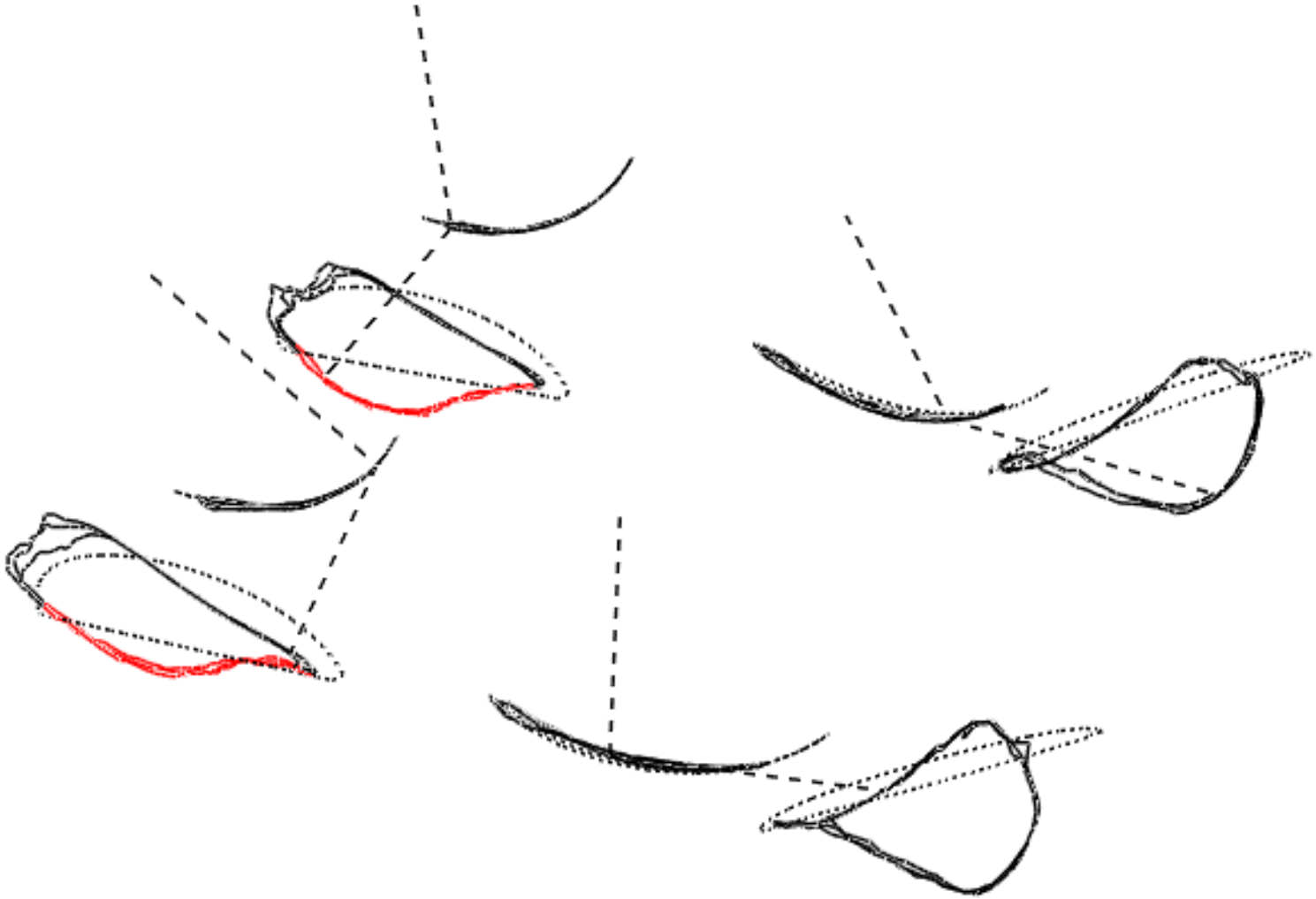




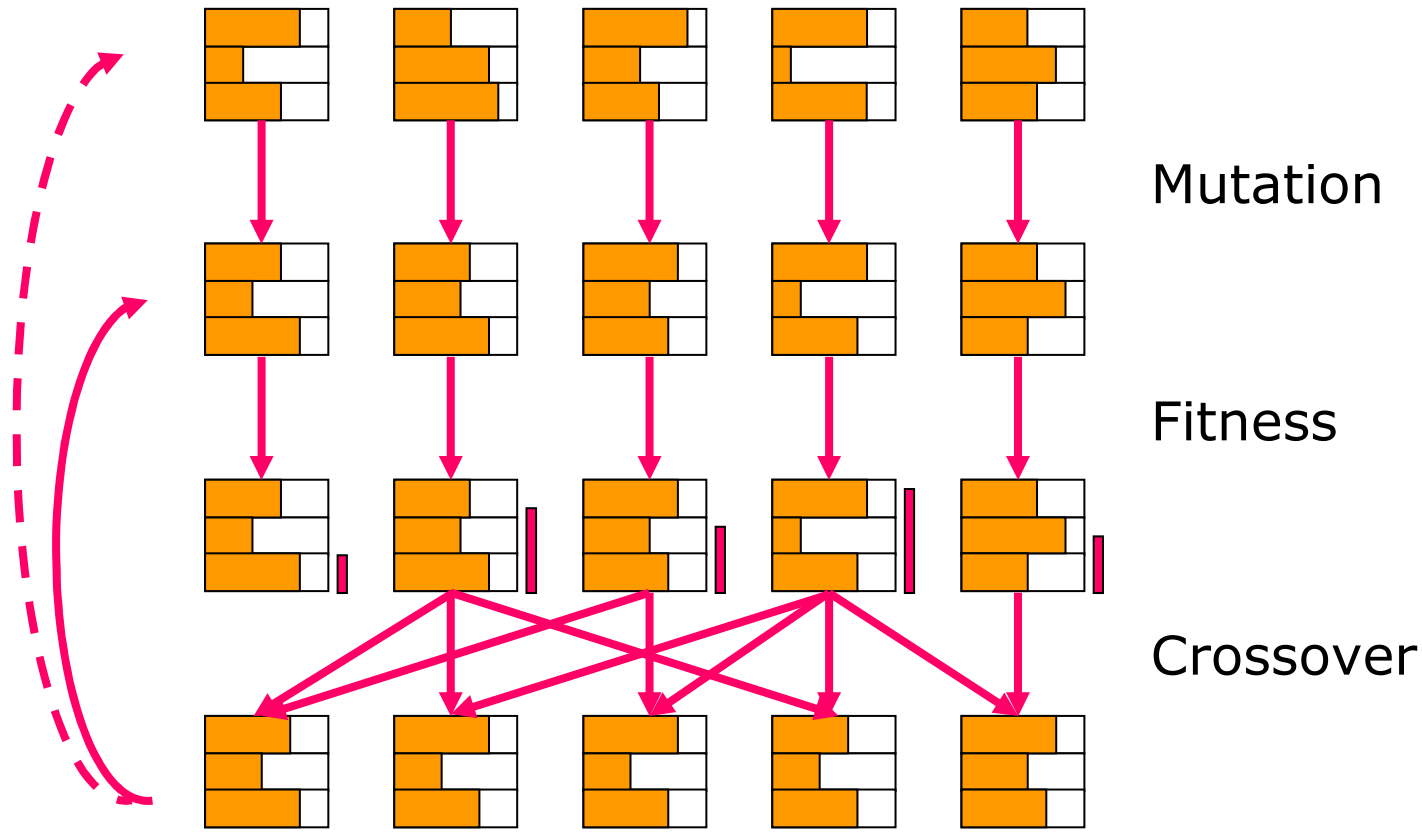
Target Joint Angles and Actual Joint Angles without Ground Contact



Target Joint Angles and Actual Joint Angles with Ground Contact



Evolutionary Algorithm



Resulting Gait – ERS-210

311 mm/s

Conclusions and Future Work

- Conclusions
  - RoboCup: compare complex robotics solutions
  - Perception
  - World modelling
  - Behavior control
  - Motion control
- Future Work
  - New field, new problems (larger field, no field walls)
  - Integrated, probabilistic world model
    - *per robot (Kwok & Fox, 2004)*
    - *for whole team*
  - Handling variable lighting…

Questions?