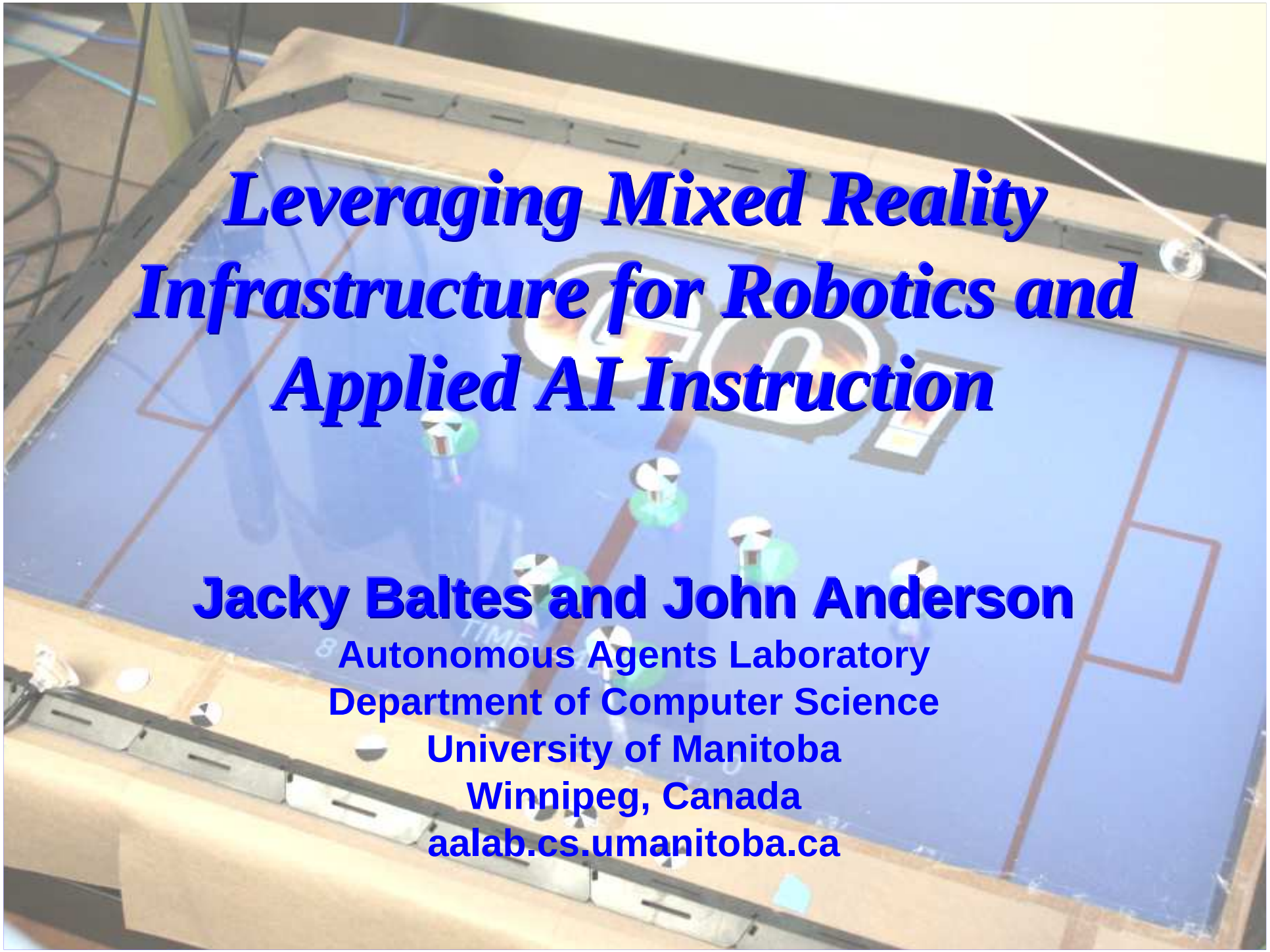


Leveraging Mixed Reality Infrastructure for Robotics and Applied AI Instruction

Jacky Baltes and John Anderson

Autonomous Agents Laboratory
Department of Computer Science
University of Manitoba
Winnipeg, Canada
aalab.cs.umanitoba.ca

Mixed Reality Environments

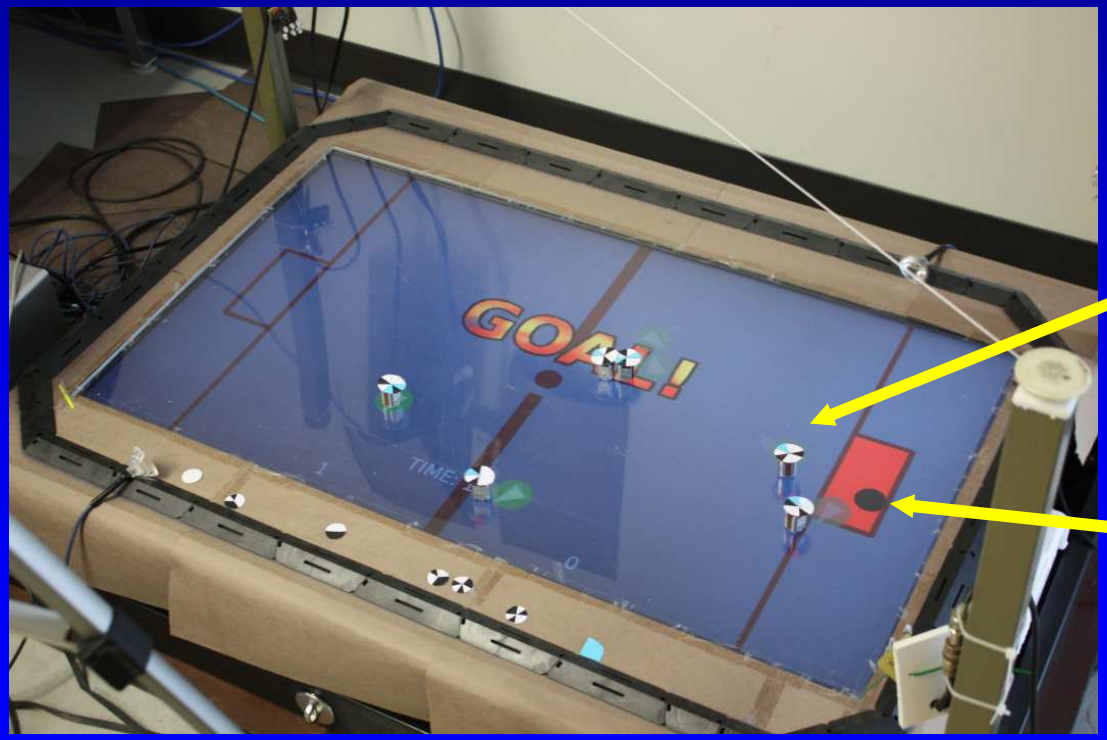
- The use of a mixture of real and artificial elements to provide an environment, where those elements are intended to be undifferentiated by an agent inhabiting it
- Reality is *Augmented* in many ways using current technology (e.g. geotagging)

Mixed Reality Environments

- A *Mixed* reality environment goes further than this: the artificial elements are intended to be perceived as part of the world, rather than supplements to it

Mixed Reality Robotics

- Creating part of a robot's reality through artificial means, without the intent of the robot knowing this



Integration of indistinguishable

Physical

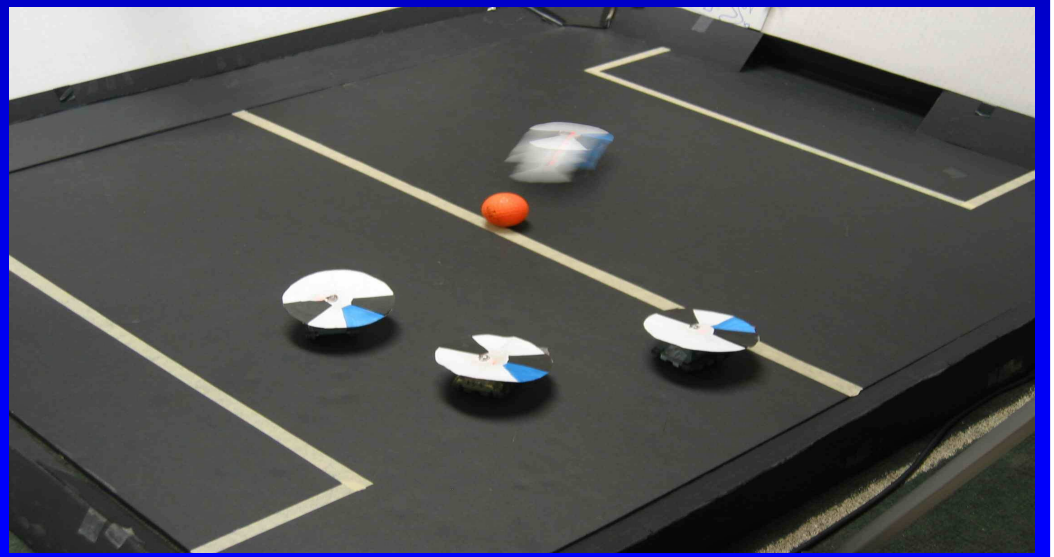
and

Virtual

elements

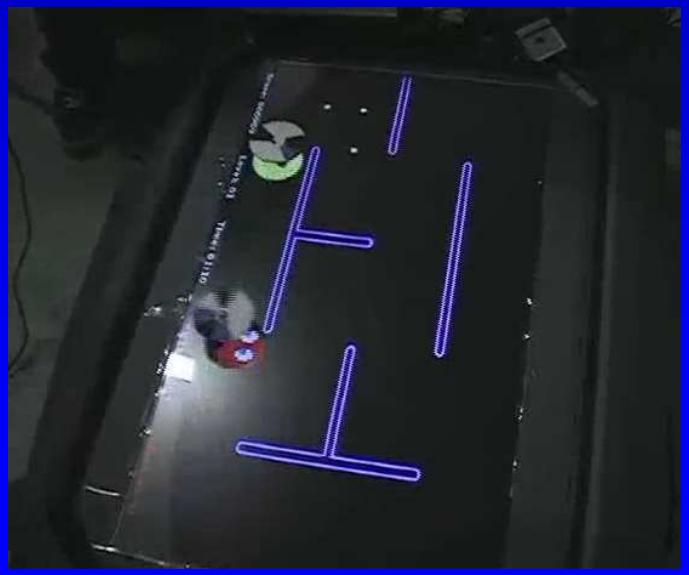
Educational Rationale

- Scaffolds for complexity: support rich environments without large amounts of peripheral material (line-trackers vs. full applications); allow removal of scaffolds
- Reduce frustration from elements of software student don't (need to) understand (e.g. robust color tracking)



Educational Rationale

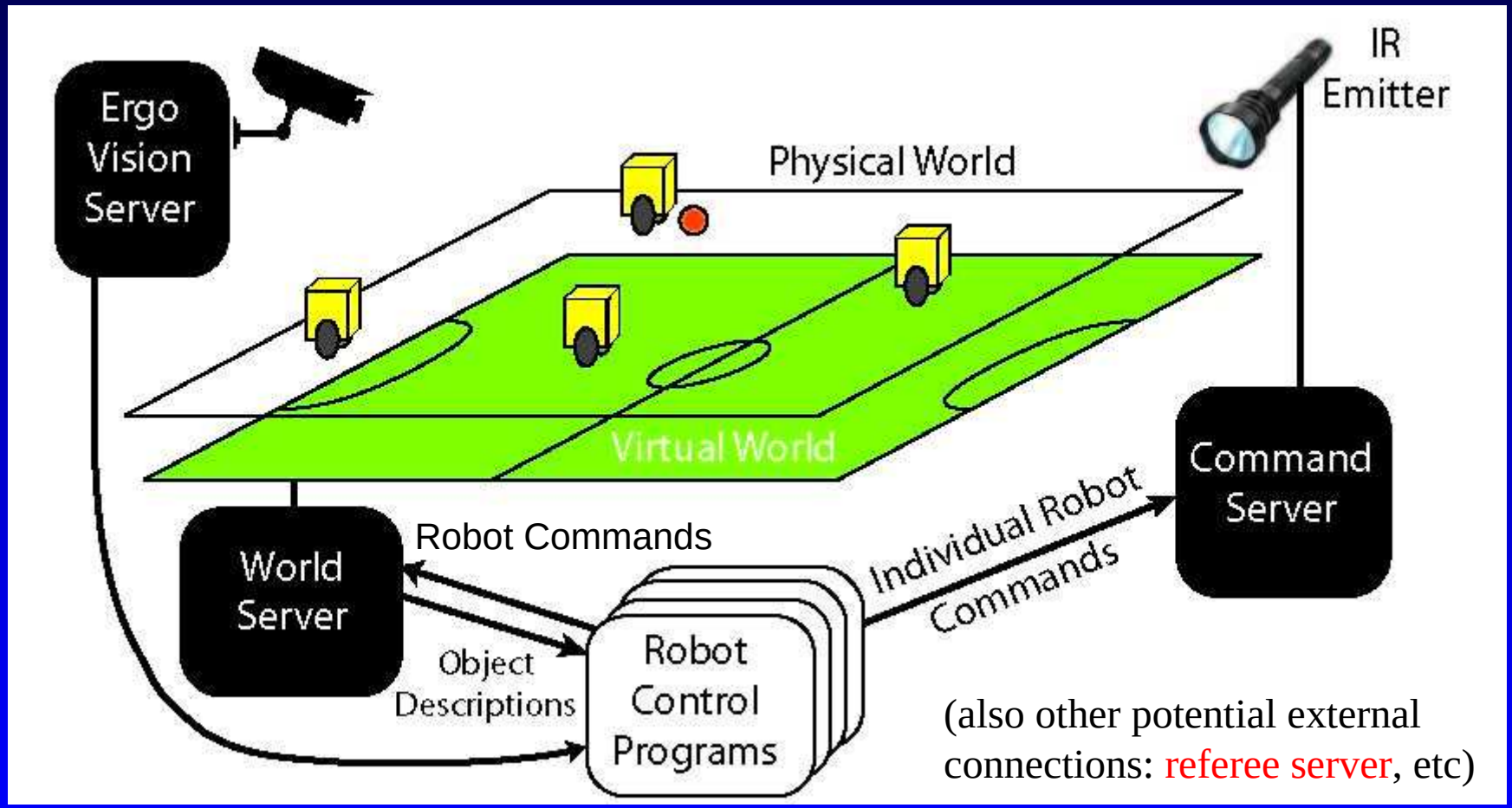
- Support rich environments while reducing brittleness/setup & configuration/maintenance (student-maintainable & portable)
- Repeatability/automation in judging for grading
- Allow a greater focus on core material



This Work

- Tightly integrate the mixed reality approach by moving beyond using it for scaffolding: including work on improving MR itself as part of coursework
- Gentle intro to elements computer vision gives them the basics needed to get started
- A peek under the hood results in broader understanding, generates interest in elements for which MR is acting as a scaffold
- Also emphasis on software engineering benefits

Mixed Reality Framework



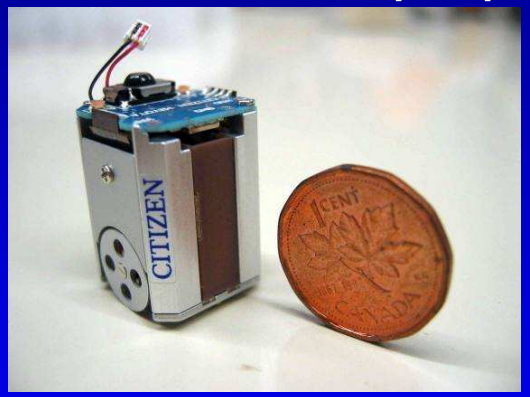
- Based on earlier E-League Soccer; students can work with completely abstracted perception

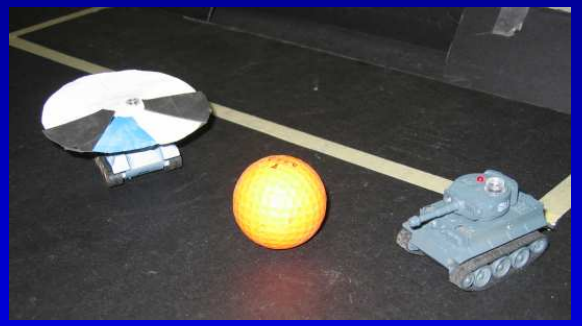
Connected Servers

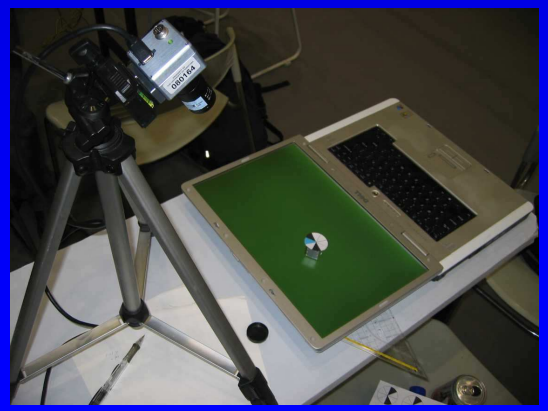
- Each describe objects of interest (likelihood match), their location, and velocity via packets over Ethernet
- e.g. vision server reports locations of robots; world server reports location of moving obstacles
- Noise means that every object is not visually identified in every frame (also network delays – effects of imperfect perception still there!)
- Since agent programs must assume this, it is easy to have multiple inputs from multiple servers on different ports treated transparently

Physical Implementation

- Depends on space, budget, resources, desired portability: IR robot may be used, from expensive microrobots to cheap toys
- Minimum of 2 laptops





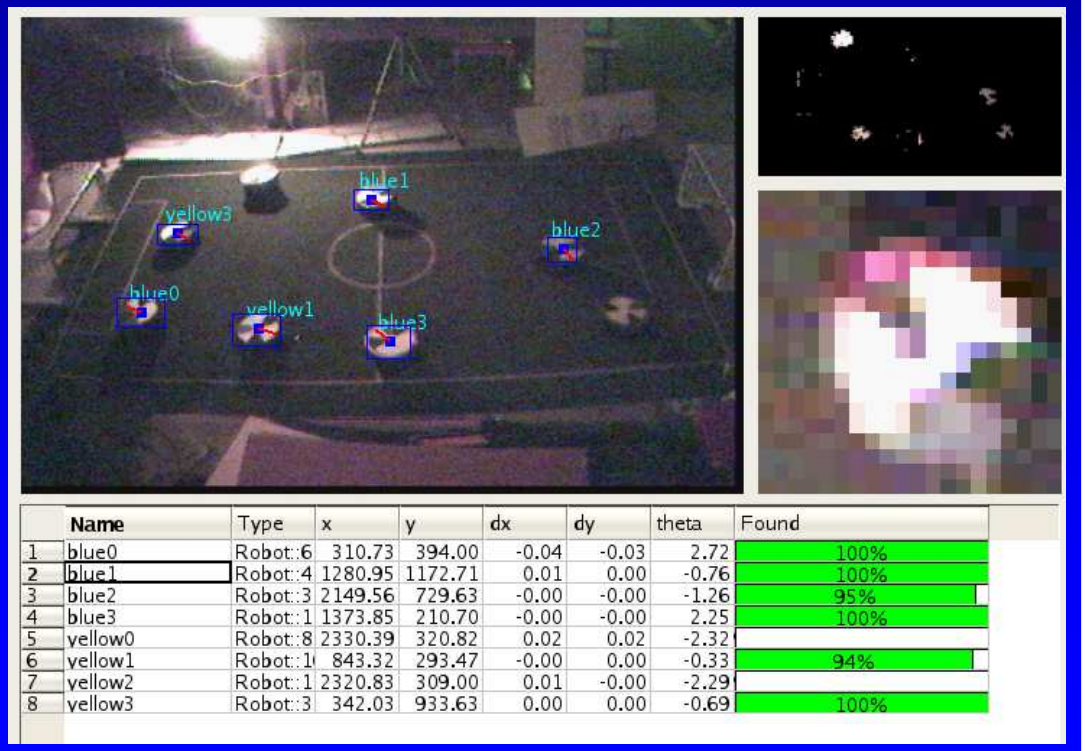


Necessary Software

- Infrared communication
- Agent control programs: may be skeletally provided to students

Necessary Software

- World server, to manage physics of virtual world – example configuration files easily alterable
- Vision Server (*Ergo*): open-source, robust, color-indifferent, reduced calibration



	Name	Type	x	y	dx	dy	theta	Found
1	blue0	Robot::6	310.73	394.00	-0.04	-0.03	2.72	100%
2	blue1	Robot::4	1280.95	1172.71	0.01	0.00	-0.76	100%
3	blue2	Robot::3	2149.56	729.63	-0.00	-0.00	-1.26	95%
4	blue3	Robot::1	1373.85	210.70	-0.00	-0.00	2.25	100%
5	yellow0	Robot::8	2330.39	320.82	0.02	0.02	-2.32	
6	yellow1	Robot::1	843.32	293.47	-0.00	0.00	-0.33	94%
7	yellow2	Robot::1	2320.83	309.00	0.01	-0.00	-2.29	
8	yellow3	Robot::3	342.03	933.63	0.00	0.00	-0.69	100%

Integrating Mixed Reality

- Course: 4th year introduction to mobile robotics
- Prerequisite is 3rd year intro to AI, but students also generally have a broad base of CS experience
- Typically 15-20 students, all work done in groups of 3-4, publicly demonstrated
- Skills (and code) from each assignment build toward next
- Initial course material consists of rudiments of computer vision: camera calibration, convolution, edge detection, integral image, Haar features

A1: Ergo

- Add new features to the existing computer vision system, in this case to improve tracking
- Focus on simple applied computer vision and on practical code maintenance
- Part 1: implement Viola and Jones' [2001] Haar-based feature tracker
- Part 2: automated camera calibration. Use world server to project calibration points with known world coordinates on the screen and record position via video server – one-click recalibration

A1: Ergo

- Side benefits of learning about code maintenance: Ergo = 20k lines C/++
- Students must put themselves in position of prior developers, ask about their choices
  - and see things generationally when they become grad students
- Added motivation of doing a good job and becoming contributors not only to the class, but to a large open-source software project
- While doing this, class covers basics of robot motion, path tracking control (PID, Sliding mode, lookahead, fuzzy)

A2: Racing/Treasure Hunt

- Racing involves path tracking in real time around a curving track (multiple laps)
- Introduces sensor/reasoning/action cycle in a real-world environment
- Even in MR: noise/robot loss in vision server; network delays; infrared coverage
  - Practical considerations not seen in theoretical papers (e.g. assuming known tire torsion coeff., ignoring velocity as well as steering [toppling]; dealing with effects of battery drain)
  - SE: implementing visualizers for debugging; coordinating access with other teams

A2: Racing/Treasure Hunt

- Part 2, cooperative treasure hunt: 2 robots start at opposite field ends, 5 treasures randomly placed. Robot digs (consumes) a treasure by sitting on top of it
- Treasure is perceived through vision server, so noisy
- Must modify world server to represent treasures, monitor time a robot spends on top of them (sensor feedback from video server - note occlusion, keep track of time for >1 robot)

A2: Racing/Treasure Hunt

- Point stabilization, but still not trivial: must brake early, account for turning radius (simple controller will get close and possibly circle forever)
- Strategy: must compute plan for order in which to dig up treasures – greedy vs. effort for searching, effects of other robot
- Shortest path may not be fastest due to time needed to turn; different groups have different control strengths from previous assignment; amount of effort to put into a better planner vs. better control
  - Good choice is a complex combination of factors: Systems engineering

MAS Issues

- Previously done competitively, but cooperatively gives them more exposure to MAS issues early
- Students know they are working toward team play, so begin to communicate over the network, and rapidly find common MAS problems (e.g. agent committing to a given treasure, then failing)
- SE: systems-level coordination issues: command server can handle one agent flooding messages, but not two
  - Discover the need to manage bandwidth (limiting/timing messages or a mux scheme)
  - Also code reuse/encapsulation/modularity in reusing racing code

A3: Obstacle Run/Obstacle Pong

- Global (Visibility Graphs, Voronoi, variants of quadtrees) and Local (potential fields) path planning covered in class
- Implement methods of their choice for two different environments
- The obstacle run has two robots at opposing ends; world server projects obstacles moving in straight lines across the playing field. One point per traversal, 5 mins max
  - Builds on treasure hunt, but with dynamic element

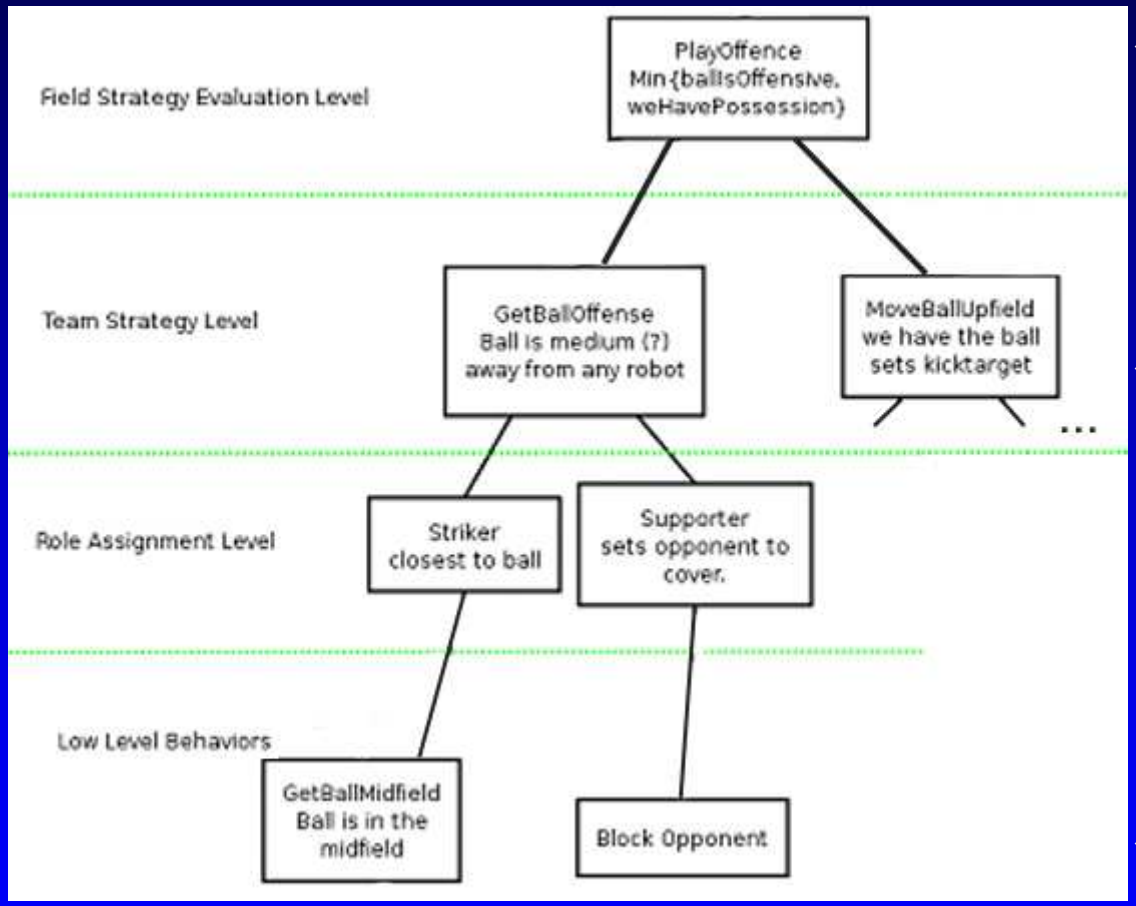
A3: Obstacle Run/Obstacle Pong

- Obstacle Pong is the first competitive domain: blocking the pong ball sends it back to the opponent (and increases speed)
- MR element: define a virtual paddle interacting with the world server, allowing the ball to be deflected in a chosen direction
  - Agent and world augmentation
- Touching fixed obstacle disables paddle (world server modification, real-time path planning, static domain)
- Simpler assignment (refactoring of previous code still necessary) to leave room for capstone

A4/Capstone: Multi-Robot Game

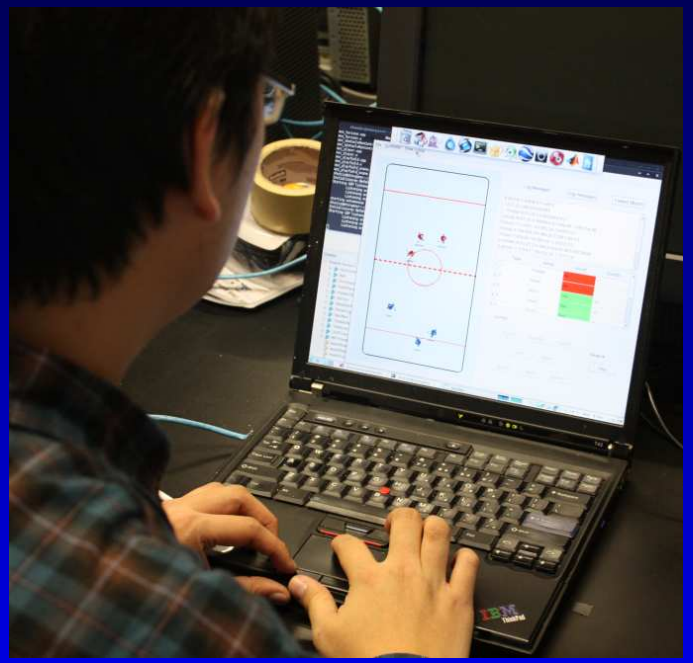
- Capstone assignment builds on (and uses) all the coding done thus far
- A competitive multirobot game is chosen requiring students to code multiple agent roles in a dynamic environment (known @ start of course)
- Coursework covers agent architectures: finite state machines, BDI, behaviour trees, subsumption
- Students may work with any architecture, usually choose behaviour trees because of ease of use

Behaviour Tree (Soccer)



- Must deal with behaviour oscillation: timer on switching, hysteresis function
- Preliminary versions of lowest level components (Static and Dynamic path planning, path and point control) from prior assignments
- Sophisticated games can be done quickly at this point (partly because of MR)

Example: 3 on 3 (or more) Hockey



- Virtual stick on robots, allows puck to be raised off ice and over competitors (slap shot – made less accurate by adding noise in world server – virtual puck)
- Many other possibilities for MR extensions, e.g. ice physics in world server, disallowing sharp turns at speed; hits on boards disabling robots depending on velocity

Capstone

- This does more than give them an interesting/demonstrable end to the course and a way to put pieces together
- Students must learn to live with their choices: time-savers at one stage can be huge liabilities later
  - Real-world Software Engineering
  - Thinking ahead helps within practical limits (e.g. forward/reverse drive for path following helps a lot in hockey!)

Capstone

- Allows students to focus their interests (e.g. MAS elements)
- More opportunity for world server extensions, referee servers
- Other domains: Soccer (passing), Pac Man

Summary

- MR is already a great way to motivate as well as to use for scaffolding purposes
  - Richer domains for given student level
- Interest can be leveraged by making modifications to MR itself part of the assignments
- Greater understanding by experience of what would otherwise be hidden – Learn By Doing
- All of these exercises demonstrate principles of software engineering to students as well, and could be taught from that context
- Students claim to learn more practical SE from this course than prescribed SE curriculum