



Tech United 2015/2016

Workshop – November 2015, Aveiro, Portugal



Planning toward RoboCup 2016

- Kinect 2
- Shooting at goal
- Faster gameplay



Planning toward RoboCup 2016

- Kinect 2
- Shooting at goal
- Faster gameplay

Kinect v1 has shortcomings



- Currently using Kinect v1

- CCD has low sensitivity:

Much light

Low exposure

30 Hz image stream

Low-light

High exposure

15 Hz image stream

- CCD has bad quality colors

- Stability problems

- Depth range limited to 6 m

- Full speed ball (10 m/s) arrives in 0.6 seconds after first possible detection

Solution: Kinect v2



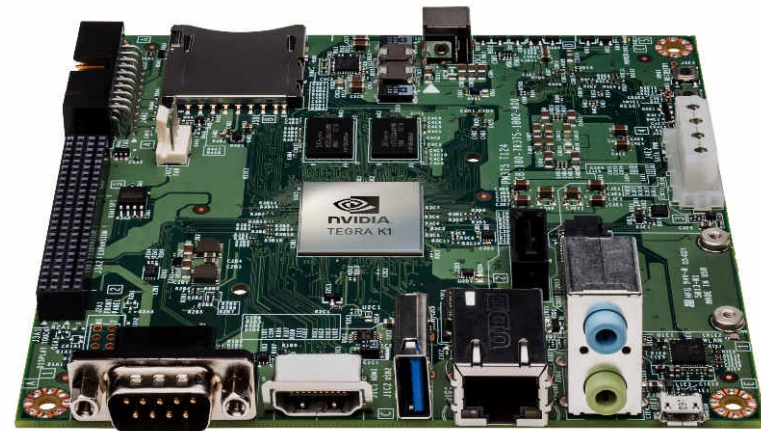
- Use the Kinect v2
 - Higher quality CCD
 - More sensitive: always 30 Hz
 - Better color quality: easier to find the ball
 - Less stability issues (hopefully 😊)
 - Increased depth range to 9 m
 - Time-of-flight i.s.o structured light
 - Increased view angle



Implementation: GPU dev board



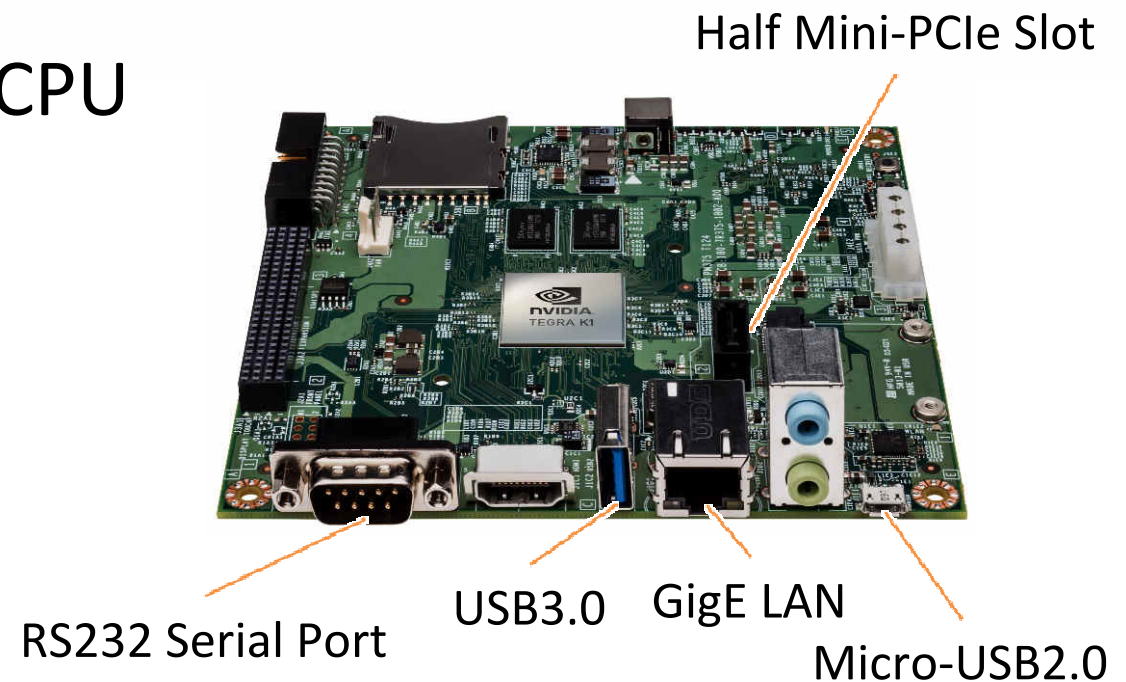
- Kinect v2 requires GPU
 - Robot pc's: Beckhoff industrial PC
 - No GPU
 - No space
- Solution:
 - Jetson TK1 dev board



Jetson TK1



- Tegra K1 SOC
 - NVIDIA Kepler GPU with 192 CUDA Cores
 - NVIDIA 4-Plus-1™ Quad-Core
 - ARM® Cortex™-A15 CPU



Ball detection



- Registration of depth and color image
- Color segmentation using Google annotated database
- Floodfill algorithm for blob detection
- Blob selection based on:
 - Size:distance ratio
 - Width:height ratio
- Transformation to robot coordinates

Jetson implementation

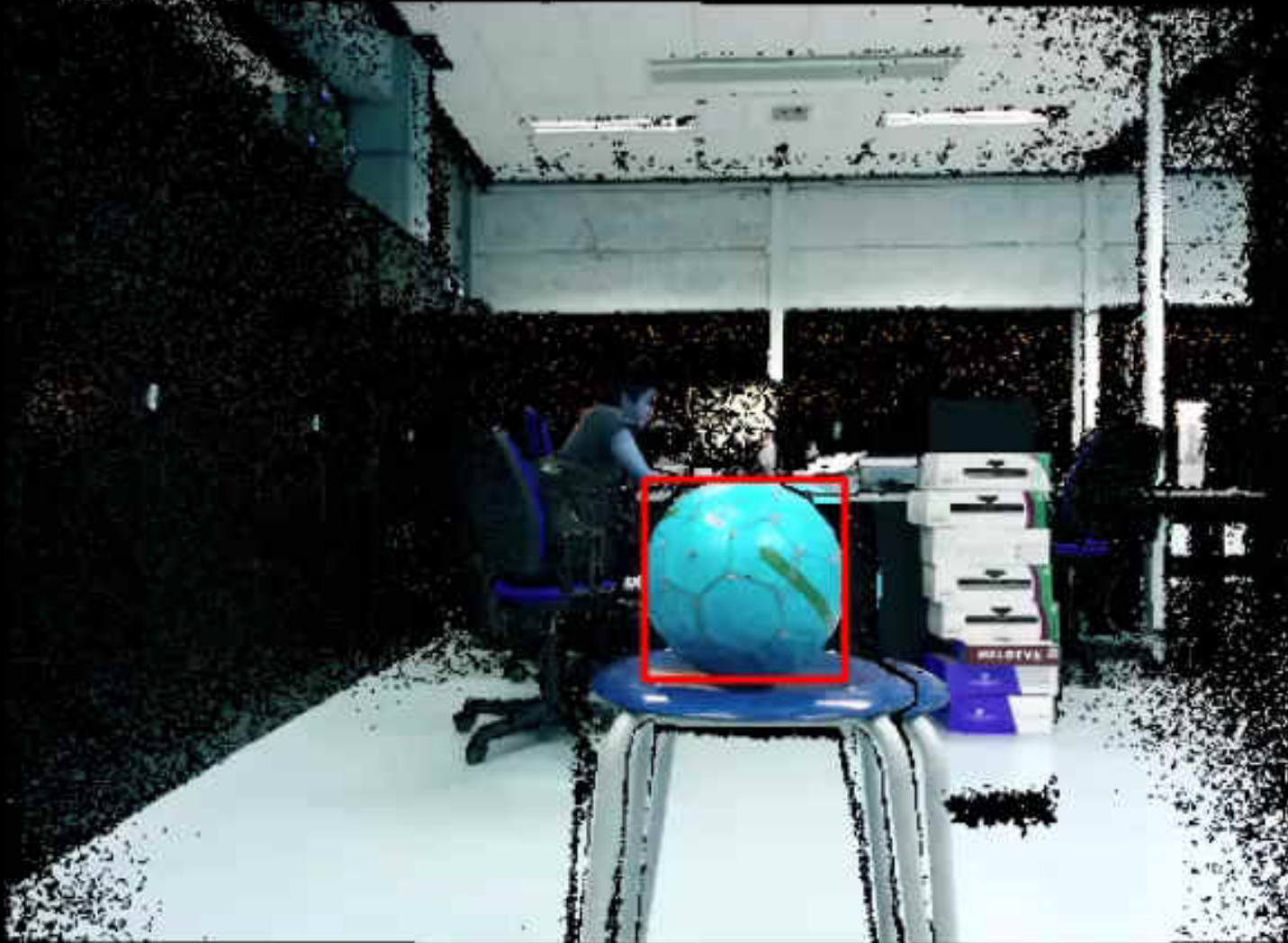


- GPU
 - JPEG decompression of color stream
 - Depth reconstruction from IR phase images
 - Floodfill
- CPU
 - Registration (near future: GPU)
 - Color segmentation (fast enough)
 - Blob selection

Result



0.917319



Future work

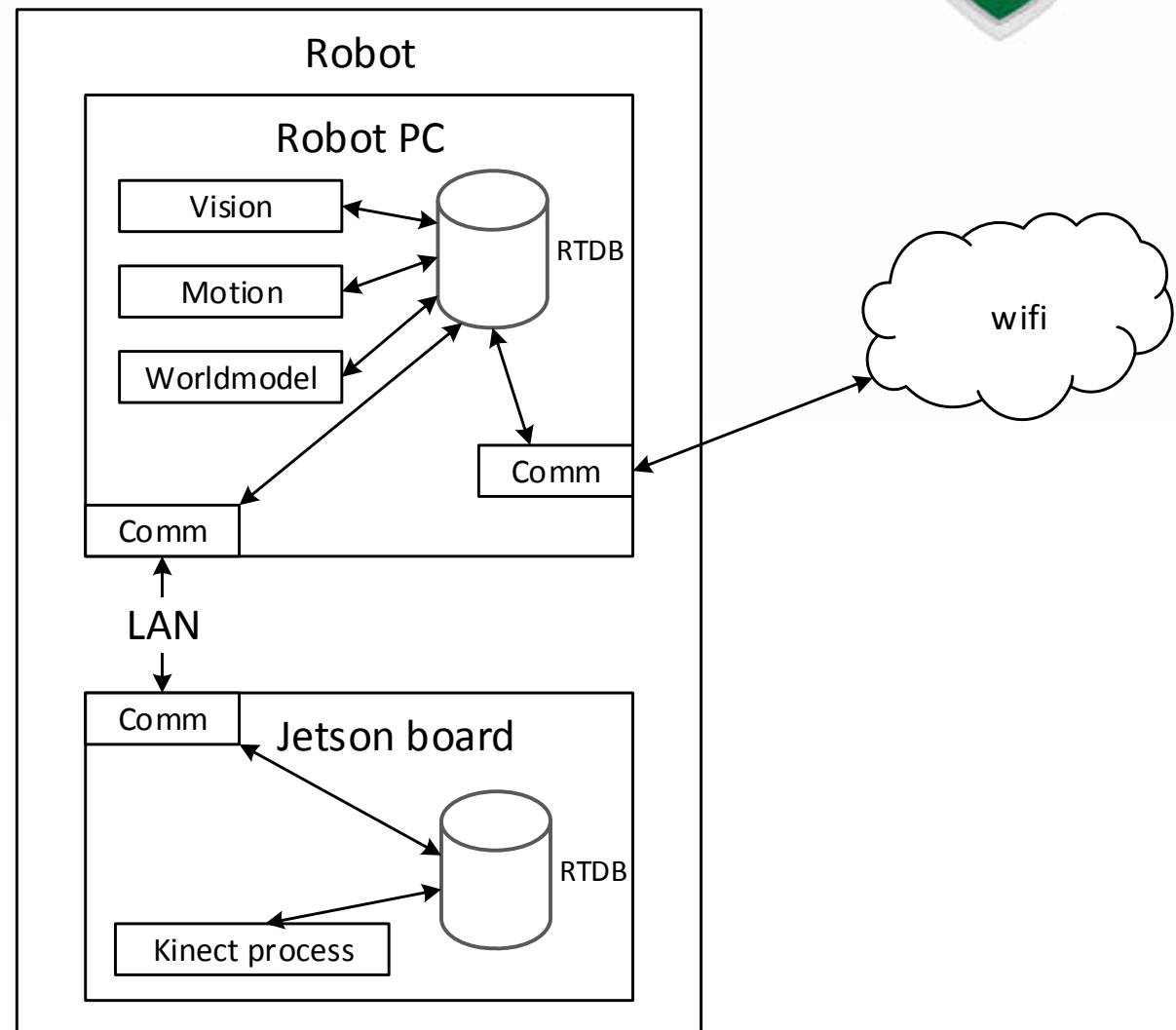


- Use field to compensate for mechanical tilt
 - Adopt from Kinect v1 implementation
 - Iteration with SVD and removal of outliers
 - Lowpass filter on tilt angles
- Detect obstacles
 - Remove field from registered image
 - Everything left is ball or obstacle
- Detect lines + goal
 - Improves sensor fusion + shot accuracy

Communication with Jetson



- Re-use of RTDB/comm
 - Second comm process
 - Flexible interprocess communication
- Possible performance / reliability improvements
 - Remote records
 - TCPIP i.s.o UDP
 - Triggered i.s.o fixed rate





Planning toward RoboCup 2016

- Kinect 2
- Shooting at goal
- Faster gameplay

Shooting at goal



Efficiency semi-final and final RoboCup 2015

- Semi-final 29% successful goal attempts
- Final 7% successful goal attempts



Shooting at goal



Solution for improvements

- Improve lob shots at goal
- Implement shooting with effect at goal



Improve lob shots



Problems with current lob shots

- Inaccurate
- Shooting angle fixed at 45 degrees
- Low velocity in xy -direction

Goal

- Accurate lob shot with adjustable shooting angle

Improve lob shots



Objectives

- Validate ball model for a bouncing ball
- Determination of relation ball trajectory and lever settings
- Inverse problem; from desired target xyz to initial inputs K and L

Improve lob shots



Ball model validation

- Parabolic ball trajectory dependent on gravity, ball velocity v_0 shooting angle α_0 .

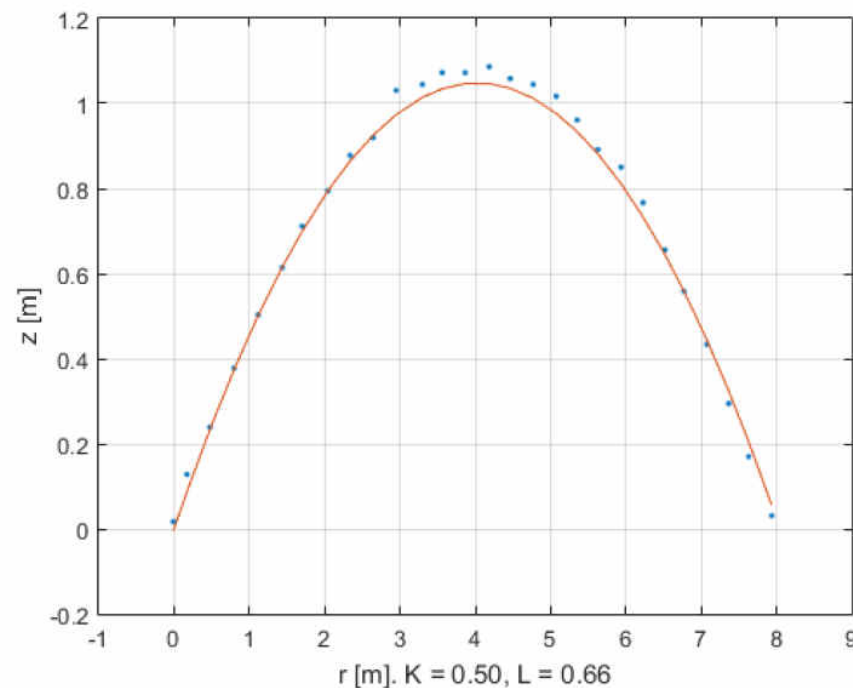


Improve lob shots



Ball trajectory for K and L

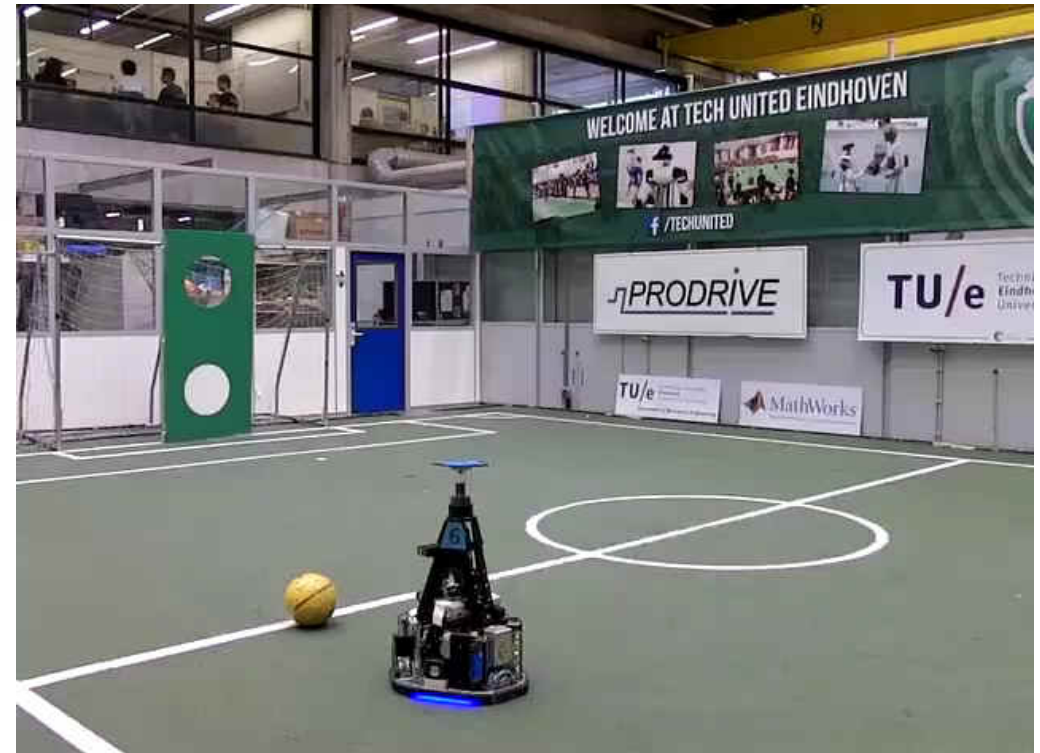
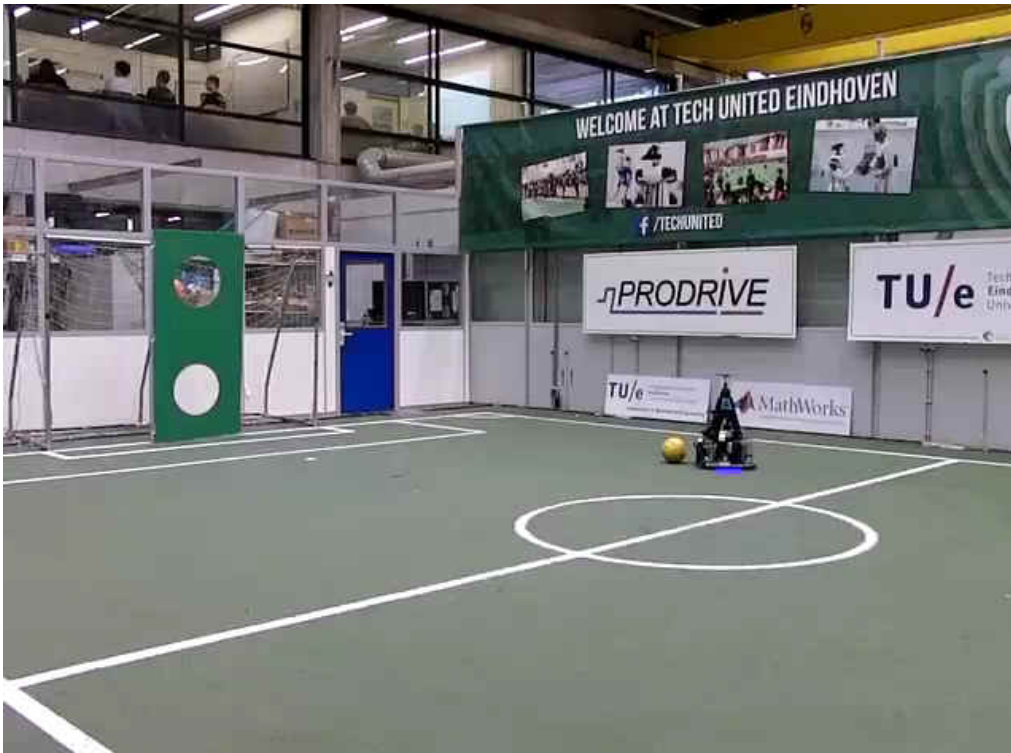
- Determination of relation ball trajectory and lever settings (K and L)



Results



Inverse model; desired target xyz



Implement shooting with effect



The screenshot displays the MATLAB R2013b environment. The main window shows the script 'AnimatieChina.m' with the following code:

```
314 t2G = t2G+Wm(f,1)/1000;
315
316 xnew = Xm(f,1)+R*sin(t1G).*sin(t2G);
317 ynew = Xm(f,2)+R*cos(t2G);
318 znew = Xm(f,3)+R*cos(t1G).*sin(t2G);
319
320 sh.XData = xnew;
321 sh.YData = ynew;
322 sh.ZData = znew;
323 refreshdata(sh)
324
325 surf(xnew,ynew,znew,cWhite);
326 hold off
327 pause(1/20)
328 end
329 end
330
331
332 end
```

The Command Window shows the following execution steps:

```
get(gcf).CurrentAxes
FH = get(gcf);
FH.CurrentAxes
FH.CurrentAxes
get(FH.CurrentAxes)
AxH = get(FH.CurrentAxes)
AxH.Box='on'
clc
getframe
size(ans.cdata)
AnimatieChina
zr
AnimatieChina
close all
AnimatieChina
```

The Workspace window shows the following variables:

| Name | Value | Min | Max |
|-------|-------------------------|----------|-----|
| Dbr | 1.1764 | 1.17... | 1 |
| Dsb | 6.7082 | 6.70... | 6 |
| Dsr | 7.3937 | 7.39... | 7 |
| LOCs | [-2 -1 0; 1 5 0; 0.3... | -2 | 6 |
| R | 0.1196 | 0.11... | 0 |
| SUM | [0; 1.7000; 2.190... | 0 | 3 |
| Tm | 1318x1 double | 0 | 1 |
| V0 | 10x3 double | 2.56... | 1 |
| V0n | [3.8798; 7.7595; 3... | 3.79... | 7 |
| Vm | 1318x3 double | -5.72... | 9 |
| W0n | [-95.7739; -168.2... | -168... | 0 |
| Wm | 1318x3 double | -168... | 0 |
| Xball | 17x17 double | -0.11... | 0 |
| Xbm | [1.0000 5.0000 0... | -2.09... | 1 |
| Xm | 1318x3 double | -2.09... | 1 |

Shooting with effect



Objectives

- Estimating path of ball for n bounces as a function of initial velocity \bar{V}_0 and initial rotational velocity $\bar{\omega}_0$
- Inverse problem; from desired target to initial inputs
- Model the effects of shooting lever and ball handling on the ball
- Use model to achieve correct desired \bar{V}_0 and $\bar{\omega}_0$

Shooting with effect



Results so far

- Estimating path of ball for n bounces as a function of initial velocity \bar{V}_0 and initial rotational velocity $\bar{\omega}_0$
- Inverse problem; from desired target to initial inputs
- Model the effects of shooting lever and ball handling on the ball
- Use model to achieve correct desired \bar{V}_0 and $\bar{\omega}_0$



Planning toward RoboCup 2016

- Kinect 2
- Shooting at goal
- Faster gameplay

Faster gameplay



Current strategy implementation

- Role-based strategy
- Fixed behavior implemented in role
- Behavior change requires role switch

Limitations

- Tasks/action coupled to role
- “State” is not preserved during role switching

Faster gameplay

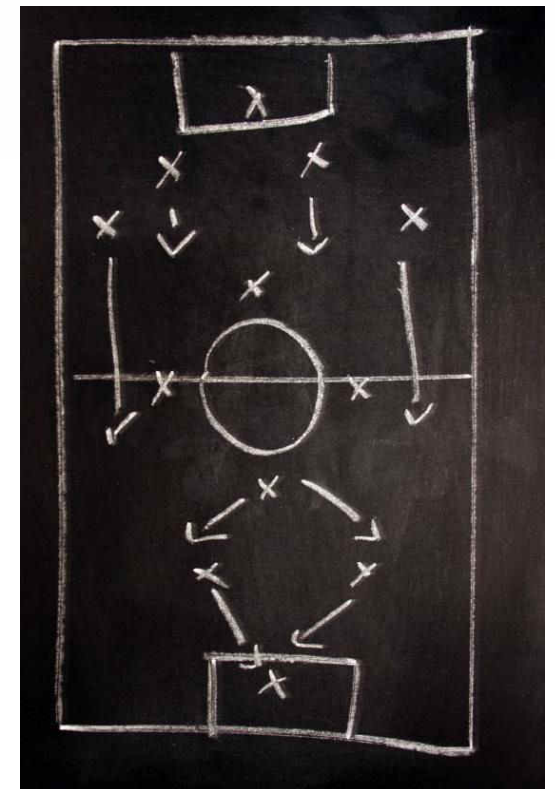


New implementation

- No behavior implemented in roles
- All actions can be shared between roles
- Sequences of actions

Improvement enables

- Faster gameplay
- Easier passing with multiple robots
- Longer planning horizon



www.robocup-europeanopen.org

RoboCup

European Open 2016

Evolution Eindhoven

March 30 - April 3





Thank you