Graphical User Interface in JDE robot applications

José M. Cañas & Vicente Matellán

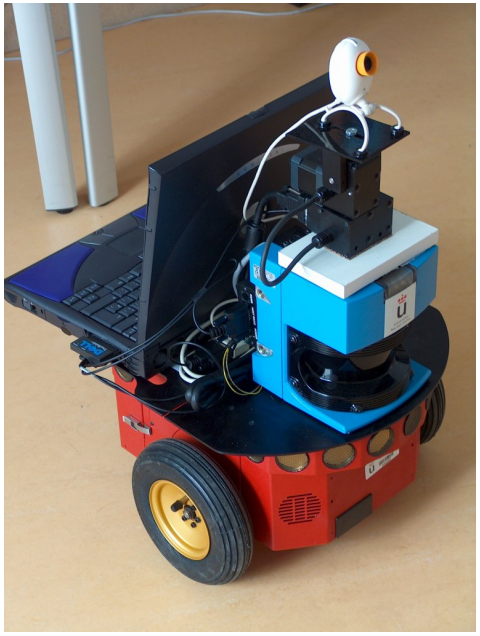
First Int. Workshop on TELEROBOTICS and Augmented Reality for TELEOPERATION

Contents

- Introduction
- Design of visualization in JDE
- Examples
- Conclusions

Introduction

Why do we need GUI in robot applications?

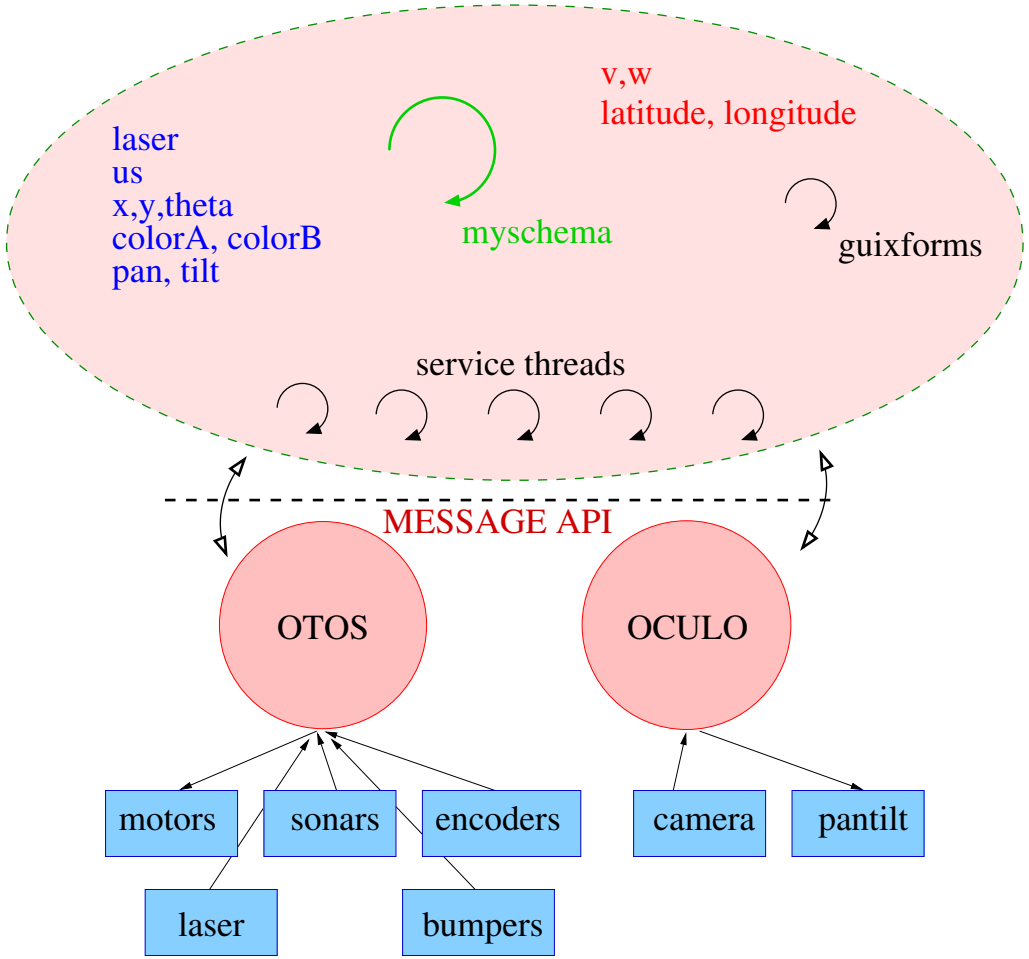


- Debugging
  - See sensor readings
  - See internal data
  - Command actuators
  - Change internal values
- Behavior is the main issue, not the GUI
- It should not disturb autonomous operation

Design of visualization in JDE

- Robot application is a set of schemas
- Each schema is implemented as a thread with iterative execution
- There is one schema for GUI
  - refresh display with new data
  - check for user interaction (buttons, sliders...)
- Its iteration frequency is under control
- Remote GUI: remote servers oculo and otos
- X-Window system on Linux: Qt, GTK+, XForms,...

Remote GUI

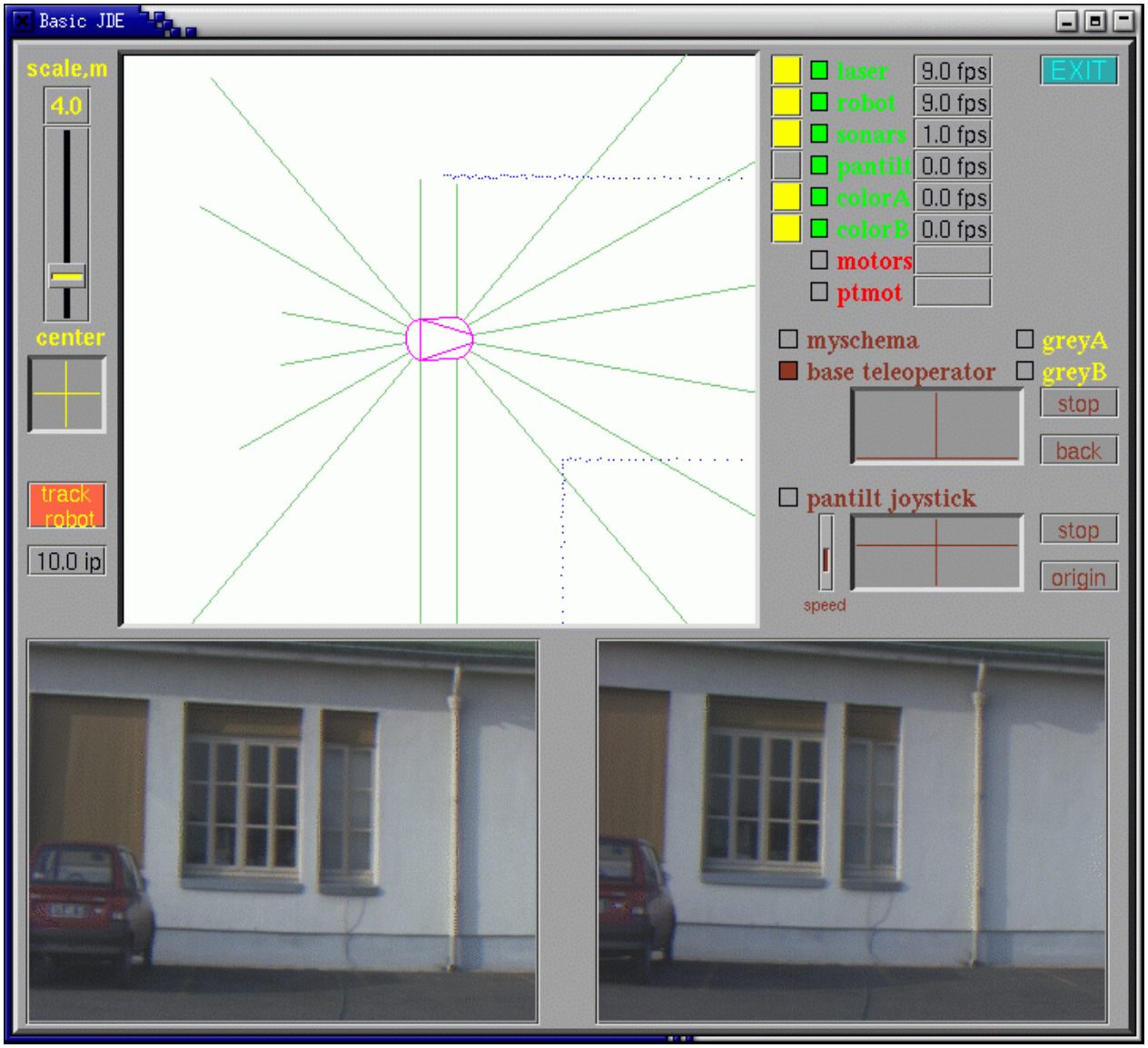

v,w
latitude, longitude
laser
us
x,y,theta
colorA, colorB
pan, tilt
myschema
guixforms
service threads
MESSAGE API
OTOS
OCULO
motors
sonars
encoders
camera
pantilt
laser
bumpers

XForms library

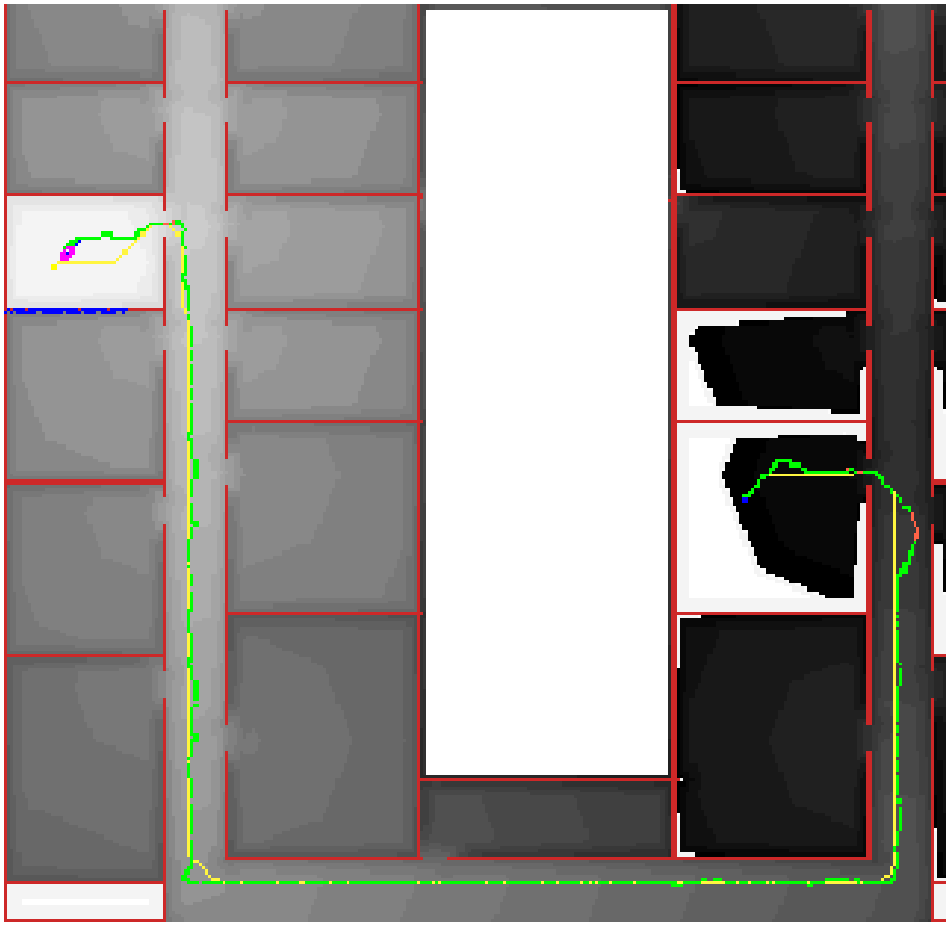
- Non-blocking mode of operation
- Pool of graphic objects for GUI
  - canvas
  - images
  - sliders
  - positioner
  - text input/output
  - ...

Examples

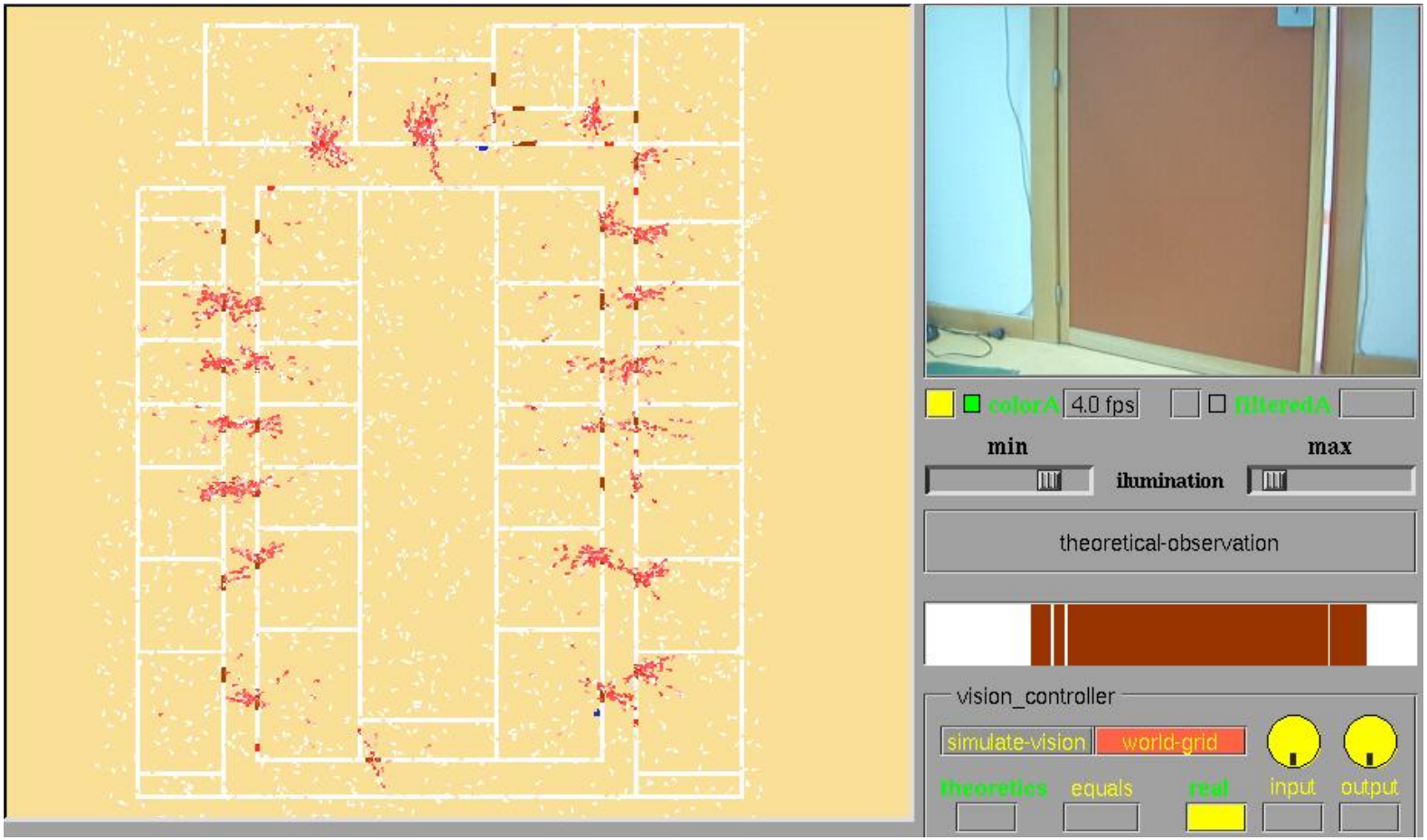
- Teleoperation and sensor readings
- Maps, grids and route planning
- 2D and 3D particles
- Several cameras and filtered colors
- Composed scene and temporal evolution


Basic JDE
scale,m
4.0
center
track robot
10.0 ip
laser 9.0 fps
robot 9.0 fps
sonars 1.0 fps
pantilt 0.0 fps
colorA 0.0 fps
colorB 0.0 fps
motors
ptmot
EXIT
myschema
base teleoperator
greyA
greyB
stop
back
pantilt joystick
speed
stop
origin

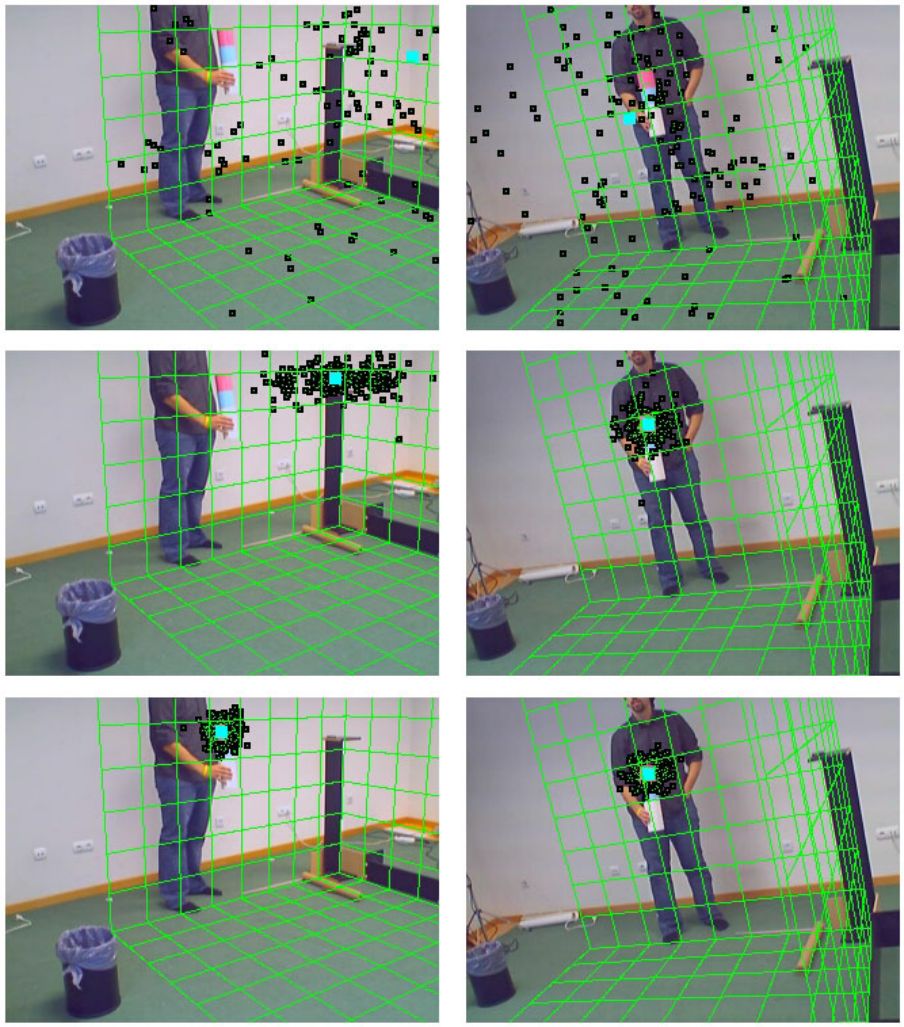
Maps, grids and route planning



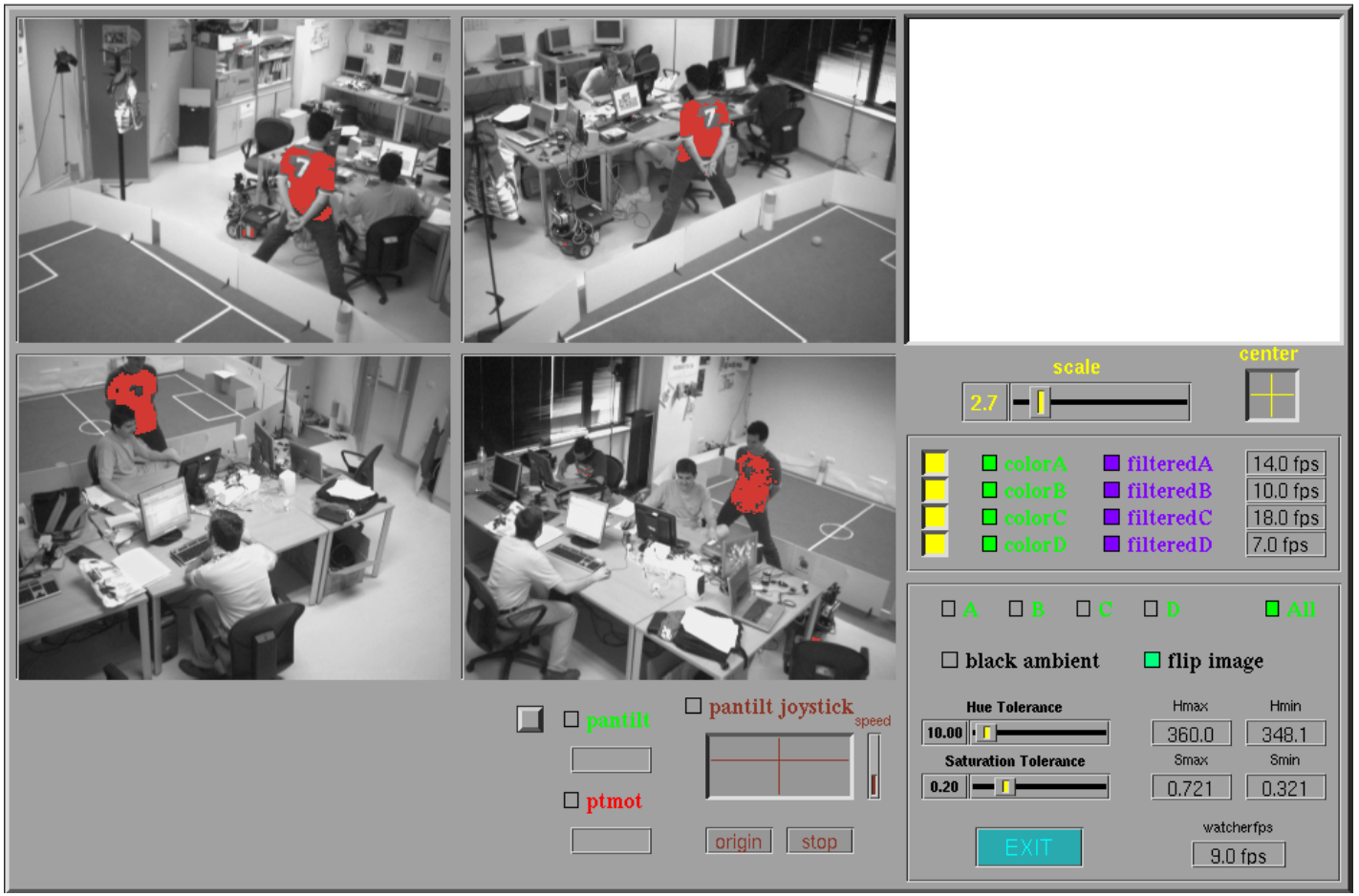
2D particles



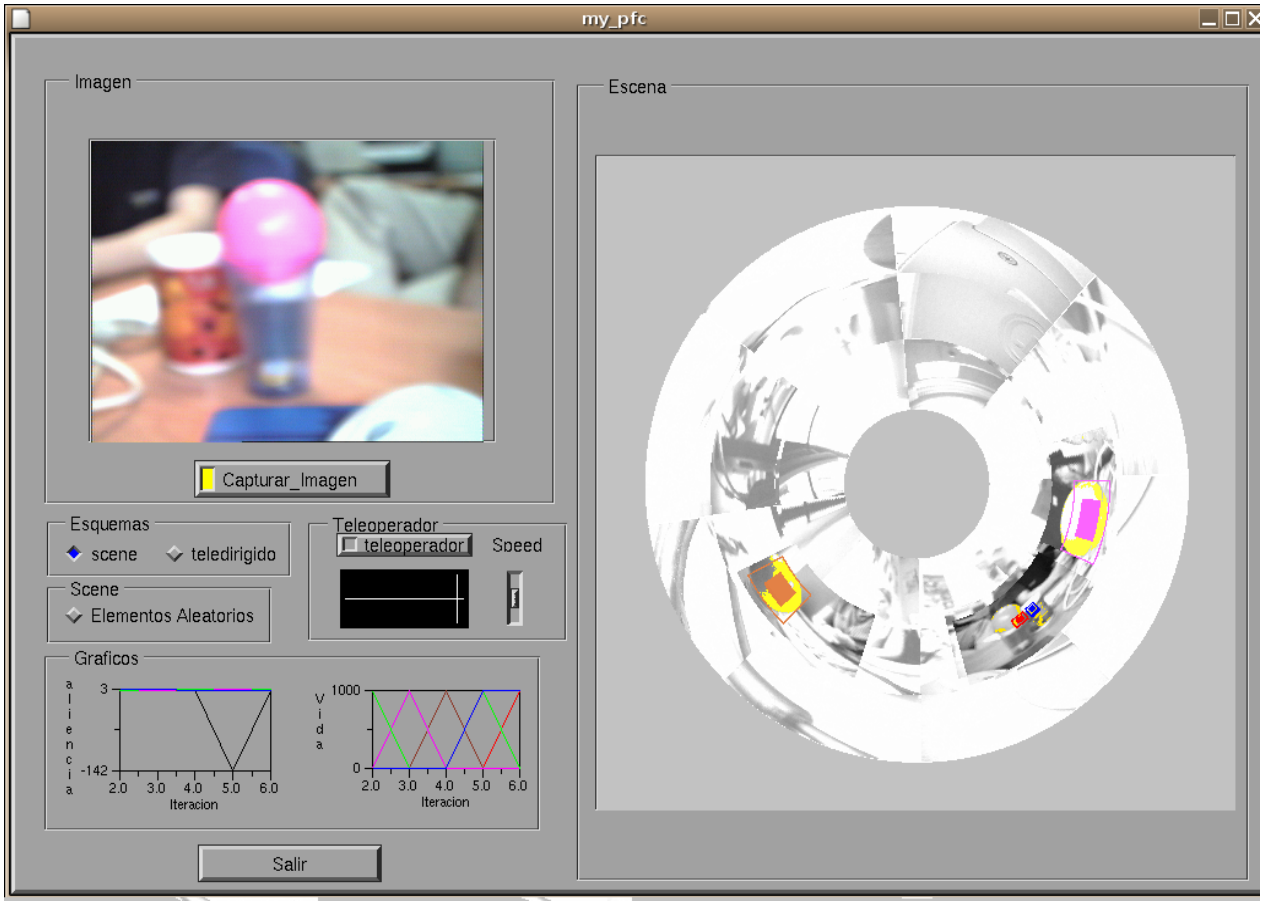
3D particles



Several cameras and filtered color



Composed scene and temporal evolution



Conclusions

- Processing time devoted to GUI is under control in JDE robot applications
- GUI can be disabled without recompiling
- Debugging is easy with JDE programmed GUIs
- Future: visualization of 3D data using virtual camera, OpenGL