Computer Vision

Robotics and Vision

27 August 2008

j.van.de.loosdrecht@tech.nhl.nl, jaap@vdlmv.nl

Overview robotics and vision

- Introduction to Robots
- Introduction to Robotics
- VisionAria
- Overview of Visionlab and VisionAria
- VisionAria Commands
- Connecting the robot
- General commands
- Moving the robot
- Sonar system
- Arm general commands
- Arm moving
- RoboChallenge

Introduction to Robots

The word robot is used to refer to a wide range of machines, the common feature of which is that they are all capable of movement and can be used to perform physical tasks.

Definition of Robot:
"An automatically controlled, reprogrammable, multipurpose, manipulator, which may be either fixed in place or mobile for use in automation applications."

Definition of Robotics:
"Robotics is the science and technology of robots, their design, manufacture, and application."

28-8-2008 Computer Vision 3

Mobile Robotics

The robots used in this course are from Mobile Robots.

All robots use Arcos as their embedded software (Firmware)

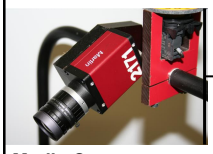
With the Mobile Robots 'Advanced Robot Interface for Applications' (ARIA) you are able to dynamically control every robot which uses Arcos.

28-8-2008 Computer Vision 4

Mobile Robotics

MOBILEROBOTS
ActivMedia Robotics

Information about the robot



Marlin Camera

Emergency Stop

Pioneer Arm

Sonar

5

Things to take in account ⚠

When you are working on the robot you have to take a few precautions.

- **Place the robot on the holder when the robot is on a desk.**
- **After working with the robot connect him to the charger.**
- **Always use a slow speed for the robot and pioneer arm.**
- **Standby, hit the emergency button when needed.**

- **Be careful with this expensive equipment.**

28-8-2008 Computer Vision 6

VisionAria

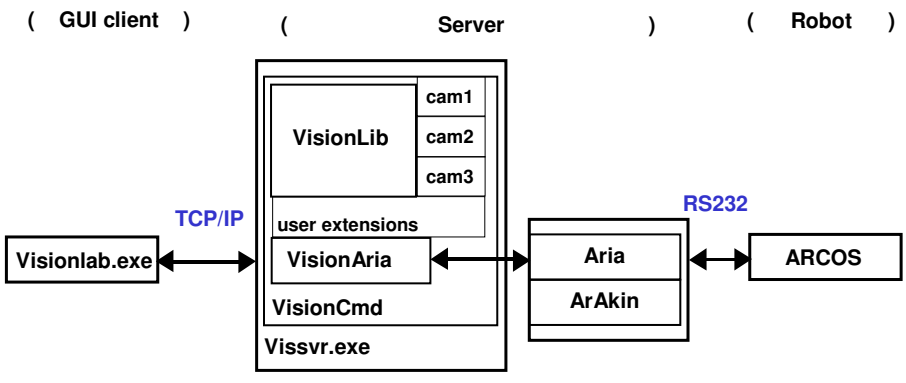
VisionAria is a library to extend the functionality of Visionlab with commands to use a Aria based robot.

53 Aria Commands
For controlling a Aria Based Robot

28 ArAkin Arm Commands
For controlling a ArAkin Based Arm

28-8-2008 Computer Vision 7

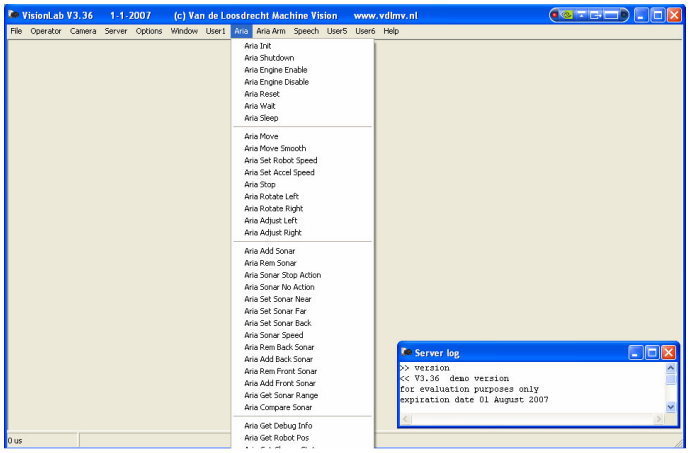
Overview of Visionlab and VisionAria



28-8-2008 Computer Vision 8

VisionAria Commands

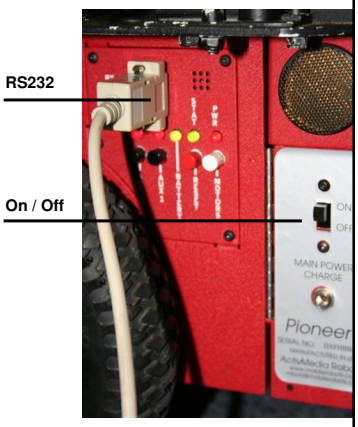
All VisionAria commands can be found under Aria and Aria Arm.



Connecting the robot

To communicate between the robot and VisionAria you need to connect them. The connection can be established with a serial RS232 connector.

- **Connect the robot with the computer**
- **Turn the Robot on**



Now you are ready to use the robot

Controlling the robot

Now it's time to control the robot. The most important VisionAria commands will be explained. Controlling the robot is devided in the following parts.

- **General commands**
- **Moving the robot**
- **Sonar system**
- **Arm general commands**
- **Arm moving**

28-8-2008 Computer Vision 11

General Commands

First things first. We start with some basic robot commands.

- **aria_init** *(Initializes the robot)*
- **aria_shutdown** *(shutdown the robot)*
- **aria_stop** *(stops the robot)*

- **aria_setrobotspeed <speed>** (sets the speed of the robot in mm/sec 800 is fast 400 normal 200 slow)
- **aria_wait** *(wait until the robot stops moving)*
- **aria_sleep <msec>** (waits <msec> after the robot moving)

28-8-2008 Computer Vision 12

Moving the Robot

Now we now the general commands we can start with letting the robot move. The robot can either move (forward or backward) or rotate (clockwise or counter clockwise).

- **aria_move <distance>** *(moves the robot. Distance is in cm. When distance is below zero the robot moves backwards)*
- **aria_rotateleft <degrees>** *(rotate's the robot counter clockwise)*
- **aria_rotateright <degrees>** *(rotate's the robot clockwise)*

28-8-2008 Computer Vision 13

One Small step for man, one Giant roll for a robot

- **Create new script**
- **Insert the next lines in the script**

aria_init
aria_setrobotspeed 200
aria_move 200
aria_wait
aria_rotateleft 90
aria_wait
aria_shutdown

- **Execute script**

make sure there is enough space for the robot to move.

28-8-2008 Computer Vision 14

One Small step for man, one Giant roll for a robot

The robot will move 20 cm forward and the rotate 90 degrees. The speed of the robot has been set to 200 so the robot will not move to fast.

Moving shape

28-8-2008 Computer Vision 15

Exercise moving the robot

Create new script which:

- **Lets the robot drive a square shape**
- **The robot has to finish on the same point it started**
- **Shutdown the robot at the end of the script**

Be sure that you don't hit any object!

Moving shape

Answer: move.jls

28-8-2008 Computer Vision 16

Moving the Robot Smoothly

The following commands don't wait until the last move is done and are there for easy for making smooth moves.

- **aria_movesmooth <distance>** *(moves the robot. Distance in cm. When distance is below zero the robot moves backwards)*
- **aria_adjustleft <degrees>** *(rotate's the robot counter clockwise)*
- **aria_adjustright <degrees>** *(rotate's the robot clockwise)*

28-8-2008 Computer Vision 17

Moving the Robot Smoothly

- **Create new script**
- **Insert the next lines in the script**

aria_init
aria_setrobotspeed 200
aria_movesmooth 500
aria_adjustleft 90
aria_shutdown

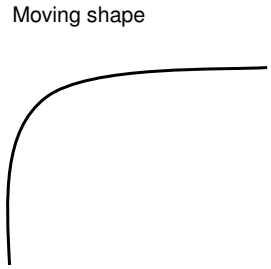
- **Execute script**

make sure there is enough space for the robot to move.

28-8-2008 Computer Vision 18

One Small step for man, one Giant roll for a robot

The robot will move 500 cm forward. But because there is a aria_adjustleft 90 after the move operation the robot will not wait for the move to finish.

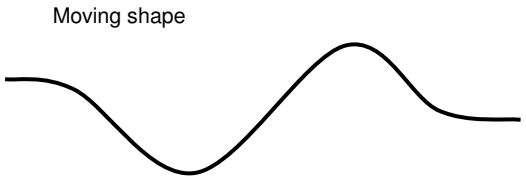


Exercise smoothly moving the robot

Create new script which:

- **Sets the speed of the robot to 200**
- **Lets the robot drive a sinus shape**
- **Shutdown the robot at the end of the script**

Be sure that you don't hit any object!



Answer: movesmooth.jls

Sonar system

The following commands concern the sonar system. Sonar works with sounds which are transmitted. The time it takes to get return to the sonar indicates a distance.

- **aria_addsonar** *(Adds the sonar function to the robot. Standard collision detecting is enabled by using this function.)*
- **aria_sonarspeed <speed>** *(Changes the speed of the sonar. Sets the time it takes to make a full sonar cycle)*
- **aria_sonarstopaction** *(Action with the sonar. If an object is getting closer to the robot than the <stopdistance> variable, the robot will cancel its movement forward and continue with the next command.)*
- **aria_sonarnoaction** *(Default action with the sonar. If an object is getting closer to the robot than the <stopdistance> variable, the robot will halt until the object moves away. After that the robot continues his way.)*
- **aria_setsonarnear<stopdistance><slowdistance><slowspeed>**
  (Changes the parameters for the sonar collision detection in the near surrounding. Standard is: <300> <600> <250>. <stopdistance> is the distance to stop, slowdistance is the distance from where to go at max <slowspeed>.)
- ***aria_setsonarback<stopdistance><slowdistance><slowspeed>****(see above)*
- ***aria_getsonarrange <sonar>*** (Gets the sonar reading from a single sonar sensor.)

28-8-2008 Computer Vision 21

Stopping on Sonar

- **Create new script**
- **Insert the next lines in the script**

```
aria_init
aria_addsonar
aria_sonarspeed 10
aria_setsonarnear 200 10 400
aria_sonarstopaction

while true do
  aria_move 200
  aria_wait
endwhile

aria_shutdown
```

- Execute script

make sure there is enough space for the robot to move.

28-8-2008 Computer Vision 22

Stopping on Sonar

In a while loop the robot is moved 20 cm. When there is an object in de range of 20 cm the robot will stop. When the object is removed the robot will continue.

To act on a sonar you need some more commands.

- **aria_frontsonar_washalted** *(returns true when the robot has been halted on a object in front of him.)*
- **aria_backsonar_washalted** *(returns true when the robot has been halted on a object behind him.)*

28-8-2008 Computer Vision 23

Acting on Sonar

- **Create new script**
- **Insert the next lines in the script**

```
aria_init
aria_addsonar
aria_sonarspeed 10
aria_setsonarnear 200 10 400
aria_sonarstopaction
while true do
  $halted = aria_frontsonar_washalted
  syncvars
  if $halted === false then
    aria_move 200
    aria_wait
  endif
  if $halted === true then
    aria_remfrontsonar
    aria_rotateright 30
    aria_wait
    aria_addfrontsonar
  endif
endwhile
aria_shutdown
```

Execute script

Make sure there is enough space for the robot to move.

28-8-2008 Computer Vision 24

Acting on sonar

With aria_frontsonar_washalted you check if there is an object in front of the robot. It there isn't an object the robot will move 20 cm forward.

```
if $halted === false then
   aria_move 200
   aria_wait
 endif
```

If there isn't the sonar is stopped so the robot can rotate. After the rotation the sonar is turned on again.

```
if $halted === true then
   aria_remfrontsonar
   aria_rotateright 30
   aria_wait
   aria_addfrontsonar
 endif
```

! When there is an object the aria_sonarstopaction will stop the last command. You need to turn of the sonar inorder to let him rotate.

Pioneer Arm general commands

The following general commands concern the Pioneer Arm.

- **aria_arm_init** *(Initializes the robot arm, prints out information about all the joints of the arm and then powers on the arm itself.)*
- **aria_arm_shutdown** *(Disconnect the robot arm from the robot.)*
- **aria_arm_wait** *(Waits until the arm has stopped moving.)*
- **aria_arm_home** (Home the arm back to its base)
- **aria_arm_park** *(Sends the arm to its home position and parks it by turning the power off.)*

- **aria_arm_setjoints_speed <speed>** *(Sets the speed of all joints. This value must be between 1 and 127. The value represents ms. Standard value is 1.)*
- ***aria_arm_setjoint_speed <joint> <speed>*** *(Sets the speed of a specified joint. The value must be between 1 – 127. The value represents ms. Standard value is 1.)*

Moving the Pioneer Arm

Now we now the general commands of the pioneer Arm we can start with letting the arm move.

- **aria_arm_setjoint <joint> <value>** *(Sets the position of the given joint to the given value, if it falls between the minimum and maximum range of the given joint.)*
- **aria_arm_setjoint_degrees <joint> <value>** *(Sets the position of the joint to a given degree.)*
- **aria_arm_setjoints<joint1><joint2><joint3><joint4><joint5><joint6>**(Set s the position of every joint, *if it falls between the minimum and maximum range of the given joints.)*
- **aria_arm_setjoints_degrees <joint1> <joint2> <joint3> <joint4> <joint5> <joint6>** *(Sets the position of every joint by a angle.)*
- **aria_arm_getjoint_min <joint>** *(Gets the minimal position of the given joint.)*
- **aria_arm_getjoint_max <joint>** *(Gets the maximum position of the given joint.)*
- **aria_arm_getjoint_center <joint>** *(Sets the position of every joint by a angle.)*

28-8-2008 Computer Vision 27

Moving the Pioneer Arm

- **Create new script**
- **Insert the next lines in the script**

```
aria_init
aria_arm_init

aria_arm_setjoints_speed 50
aria_arm_setjoints_degrees 120 130 120 90 90 90
aria_arm_wait
aria_arm_setjoints_degrees 120 50 50 120 120 50
aria_arm_wait

aria_arm_home

aria_arm_wait
aria_arm_shutdown
aria_shutdown
```

- **Execute script**

make sure there is enough space for the arm to move.

28-8-2008 Computer Vision 28

Moving the Pioneer Arm

The arm is moving to two divert positions. These positions are set with the aria_arm_setjoints_degrees command.

Before the arm is shutdown the aria_arm_home command is executed.

Exercise "A helping hand"

Create new script which:

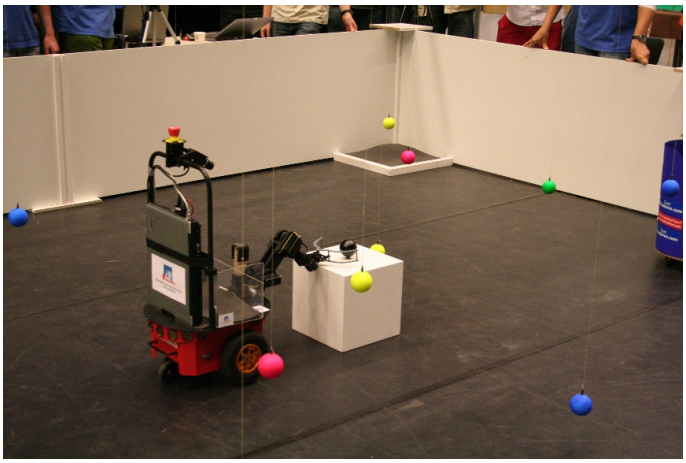
- **Sets the speed of the arm to 50 mm/sec**
- **Lets the arm pickup a ball**
- **And drop it on the robot**
- **Shutdown the robot at the end of the script**

Be sure that you don't hit any object with the arm!

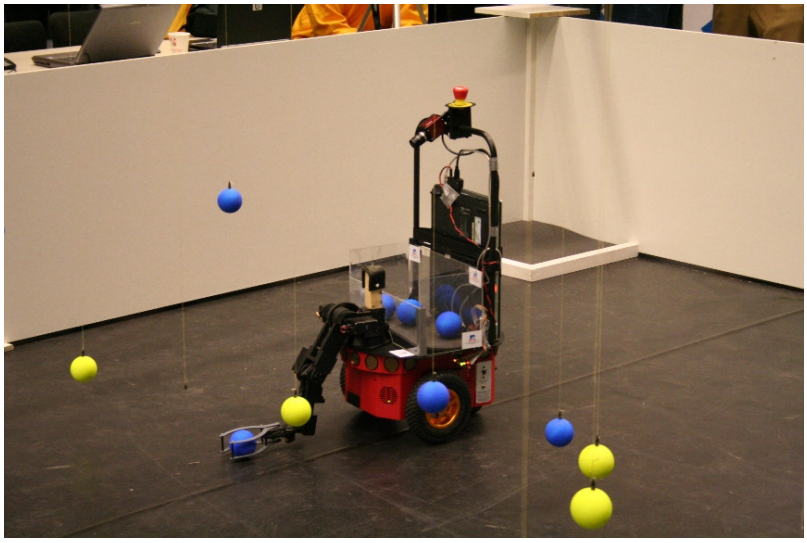
Answer: arm_helping.jls

Robochallenge

Robot in action at the Robochallenge.



Robochallenge



All Aria commands

aria_init
aria_enginedisable
aria_sleep<msec>
aria_setrobotspeed <speed>
aria_rotateright <degrees>
aria_addsonararia_remsonar
aria_comparesonar<sonar1><sonar2>
aria_rembacksonar
aria_addfrontsonar
aria_resetorigin
aria_getrobottype
aria_getbatteryvoltage
aria_robotfinddistanceto<x><y><th>
aria_comstring <command><arg1>
aria_logstart<filename>
aria_robot_finddeltaheadingto<x><y><th>
aria_pointfindangleto<x1><y1><th1><x2> <y2><th2>
aria_pointfinddistanceto <x1> <y1> <th1> <x2> <y2> <th2>
aria_setsonarnear<stopdistance><slowdistance><slowspeed>
aria_setsonarfar<stopdistance><slowdistance><slowspeed>
aria_setsonarback<stopdistance><slowdistance><slowspeed>
aria_shutdown
aria_reset
aria_move <distance>
aria_setaccelspeed <speed>
aria_adjustleft <degrees>
aria_sonarstopaction
aria_comparesonarsides
aria_addbacksonar
aria_sonarspeed <speed>
aria_getrobotpos
aria_getname
aria_play <tones>
aria_com <command>
aria_com2byte<command><arg1><arg2>
aria_logstoparia_logadd <text>
aria_frontsonar_washalted
aria_engineenable
aria_wait
aria_movesmooth<distance>
aria_stoparia_rotateleft<degrees>
aria_adjustright<degrees>
aria_sonarnoaction
aria_getsonarrange<sona
aria_remfrontsonar
aria_getdebuginfo<param
aria_getchargestate
aria_getversion
aria_robotfindangleto<x><y><
aria_cominteger <command><arg1>
aria_backsonar_washalted

28-8-2008 Computer Vision 33

All Aria Arm commands (ArAkin)

aria_arm_init
aria_arm_isactive
aria_arm_setjoint_speed <joint><speed>
aria_arm_setjoints<joint1><joint2><joint3><joint4><joint5><joint6>
aria_arm_setjoints_degrees<joint1><joint2><joint3><joint4><joint5><joint6>
aria_arm_getjoint <joint>
aria_arm_getjoint_max <joint>
aria_arm_getjoint_home <joint>
aria_arm_ispowered
aria_arm_wait
aria_arm_park
aria_arm_movearmtoposition_joint <x> <y> <z> <jointangle>
aria_arm_poweroff
aria_arm_opengripper
aria_arm_getversion
aria_arm_shutdown
aria_arm_setjoint <joint> <value>
aria_arm_setjoint_degrees <joint><value>
aria_arm_setjoints_speed<speed>
aria_arm_getjoint_min <joint>
aria_arm_getjoint_center <joint>
aria_arm_test
aria_arm_iscommunicating
aria_arm_home
aria_arm_movearmtoposition <x> <y> <z>
aria_arm_poweron
aria_arm_closegripper
aria_arm_carry

28-8-2008 Computer Vision 34