RobEx: an open-hardware robotics platform

J. Mateos	A. Sánchez	L. J. Manso	P. Bachiller	P. Bustos
IADex	Escuela Politécnica	Escuela Politécnica	Escuela Politécnica	Escuela Politécnica
	Univ. de Extremadura	Univ. de Extremadura	Univ. de Extremadura	Univ. de Extremadura
josems@iadex.es	agustinsd@unex.es	lmanso@unex.es	pilarb@unex.es	pbustos@unex.es

Abstract

Autonomous robots are usually given a considerable sensing capacity. However, they are seldom equipped with the necessary actuators to perform active perception techniques or human-like tasks. This paper presents two robot actuators and RobEx, the robot platform where they have been tested. The first of the actuators is a forklift. The other presented actuator is a four DOF stereo head. Both, the mechanical and electronic designs are released as open-hardware. The software that controls all the presented robotic hardware is also available under the GPL license.

1 Introduction

This paper presents a robot platform (RobEx) and two accessories that form a complete equipped robot. There are multiple scenarios in which a standard platform with a static camera or laser sensor is not enough. With such configuration it is extremely difficult to perform active perception, or visual attention techniques. The presented hardware intends to meet some of the autonomous robots needs both in academic and research fields.

Nowadays, most commercial mobile robot platforms have a considerable high price. Moreover, its design is usually not available to its customers. Thus, there is little space for customization and researchers have to accept the few options they are given. Our aim when designing RobEx and its accessories was the opposite. The goal was to design a low-cost open robotic platform with the similar features and quality of commercial platforms. This leads to other end-user advantages: **a)** because of the open-access design, users can freely modify it to meet their needs; **b)** as well as in software development, open designs tend to have less bugs than closed ones; and **c)** not only bugs but every single improvement can be shared with the rest of the users regardless of who proposed it.

Stereo heads allow robots to move its cameras without moving the whole robot. This makes robots much more agile and human-like, what is an important feature in social robotics. The presented stereo head have four degrees of freedom: a neck movement followed by a common tilt and two camera-specific pan movements. The neck allows cameras to point to objects on the sides of the robots without moving the robot platform. The tilt allows the robot to point to low or high positions. The pan movements can be used for vergence fixation of objects lying approximately in front of the camera pair. This is particularly useful in order to increase the binocular space and to reduce the 3D triangulation error.

The presented forklift is similar to its industrial counterparts. A robot forklift makes possible the development of many different types of experiments in robots. If it is sensitive, it can be used for active perception (e.g. the robot may infer if a box is full or empty, or even if there is no box). It can also be used to model industrial warehouse robots. The presented forklift is capable of supporting loads of up to 5kg safely, and has a span of 150mm. The gap between the forks separation can be manually adjusted.

The three hardware components are connected to an on-board laptop through a USB interface. The mechanical and electronic designs of the described hardware are freely available [1]. The software interface is released under the GPL license.

2 Mechanical and electronic design

There are several desirable characteristics that accessories and platforms for autonomous robots should have. Low cost, robustness, lightness, low energy consumption or ease of use are some of these characteristics. Hardware complexity should also be taken into account due to its impact on accuracy and durability. In addition, robot parts should be easy to (un)mount on different robot platforms. These goals have been considered in the design process of the RobEx platform and its accessories (see figure 1), as it will be detailed in the following subsections.

The communication interface of the presented robot parts is the well-known Universal Serial Bus (USB). While it is not appropriate for devices with heavy latency or bandwith needs, it provides the advantage of being present in almost any computer. Besides, several USB devices can be connected to the on-board computer through an USB-hub. Since ease of use is thought as one of the key factors, these topics should be taken into account when designing robotic hardware.

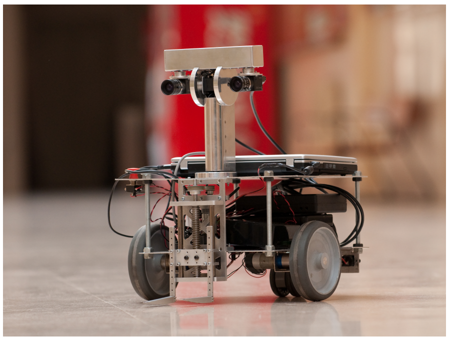

Figure 1: The RobEx platform and its accessories

2.1 RobEx

The main design goal for RobEx was to have a quality robot platform easy to build and use. It was also necessary for RobEx to be robust and cheap. Since aluminum provides a good tradeoff between lightness, robustness and price, most of the parts are built with aluminum.

In order to fulfill with all the requisites, the best option was to choose a differential steering system (figure 2). It is one of the most simple designs, and it makes possible turning radius of very small lengths, even zero [2]. This configuration is also widely used in other models, as in the Pioneer 3Dx and the PowerBot of MobileRobots [3]. The wheels have a radius of 7.5cm, which is enough to absorb the effect of floor irregularities while providing good motor-to-wheel power transfer and speed.

RobEx provides the necessary space and load support to carry up to four on-board laptops. In order to provide this space, it has a two-storey design. All the electronics are located in the lower part, both power and control circuits. This way, the upper level is available to place computers or any kind of robot sensor or actuator.

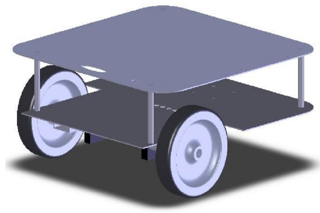

Figure 2: Design of the RobEx platform

Two Maxon A-Max AC motors [4] provide torque to the wheels. Each motor is equipped with gearhead and an optic incremental encoder HEDS-5540 with a resolution of 500 ppr. This resolution can be extended to 2000 ppr when using the squaring codification, providing a final resolution of 66000 steps per turn of the motor axle. Both motors are controlled by two opto-isolated LMD18200 (H-Bridge). Thus, the motor control circuit is totally isolated.

The main part of the control system is an 8 bits Atmel ATMega32 microcontroler [5]. Its main task is to keep track of the PID control of both motors at a frequency of 1 Khz. The IC used in order to read the position of the motors is the LSI/CSI LS7266R1 [6]. It reads the position of both motors through the encoders, and provides the microcontroller with a 24 bit counter for each motor. Besides, the microcontroller also reads the platform commands through a FTDI USB-TTL interface [7]. All commands for the platform control are specified in ASCII code using a simple syntax.

2.2 Forklift

Forklifts are not too widely used in the field of autonomous robotics. Only few examples can be found in the robotics field. One of them is offered by the Lego Mindstorms company [8]. Although its use is more educational than professional, they are equipped with a small computer and communication systems to allow its control. A more sophisticated model is the one used in the "Robotics Research Lab." of the "Technical University of Kaiserslautern". However, it is not thought as an accessory that can be coupled to any existing robot.

The forklift presented in this paper was designed in order to improve the existing models, recalling its industrial counterparts. Figure 3 shows a CAD model.

The chassis is all aluminum made: 5 mm thick in the back side and 8 mm thick for the rest. Each part is mechanized using CNC (Computer Numerical Control). Therefore, they are highly accurate, providing a perfect final adjustment. At the back of the chassis, there have been placed two plates in order to couple the forklift to the chassis of the RobEx platform.

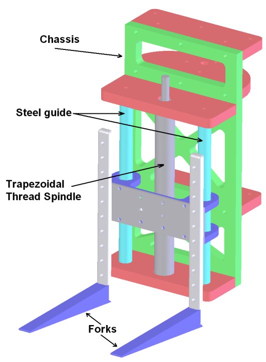

Figure 3: Design of the forklift

The movement system of the forklift is based on a trapezoid thread-spindle which is assembled over bearings. There are many advantages in the use of spindles in comparison to the hydraulic systems used in industrial forklifts. The main advantage is the absence of pumps, valves and deposits. Moreover, the position of the forks can be easily computed using an encoder situated in the spindle or even in the motor that moves the spindle. Another important advantage is that, in absence of movement, the platform stands still. This is a very important fact, since if there is no movement, there will not be energy consumption which allows saving battery charge.

The choice of the best spindle for our purpose requires taking into account several concerns: low cost, low weight, availability, high advance per turn and close fit. A great variety of spindles can be found which can be used for several applications: CNC systems, coordinates tables, positioning systems, etc. Since we are not working with large axial loads (the forklift is designed to work with loads of up to 50 Newtons), it is not necessary to use too sophisticated or expensive spindles. This is the reason why we choose a basic spindle, with a diameter of 15 mm and a pitch of 4 mm. However, it is very important to calculate the torque needed to move the forks. The following expression must be used:

$$torque = \frac{F \times P}{2000 \times p \times ha} + 7.7 \times d^4 \times i \times 10^{-13} \quad (1)$$

where F is the axial load measured in Newtons, p is the pitch measured in millimeters (mm), ha is the system performance, d is the diameter of the spindle (mm) and i the length of the spindle (mm). Using this equation for the selected spindle with a maximal load of 50 N, it is obtained:

$$torque = 0.39Nm = 39Ncm \quad (2)$$

Once the necessary torque has been calculated, the most suitable motor must be chosen. One of the key aspects in this task is the motor speed. Since the system specifications are very common, there can be found geared-motors with high features and low cost which are very suitable for our application. Specifically, the selected motor presents the following features: torque = 160 Ncm, gear = 100:1, speed = 300 rpm. Since the motor torque is much higher than the required one, it was decided to transform such power into speed. For this purpose, toothed pulleys were added providing a multiplication factor of 1:3. With this configuration, there can be reached speeds of 300 rpm, which is a linear speed of 5 revolutions per second. With a spindle of 4 mm per turn, we get a final speed of 20 mm per second, which is a very suitable speed for our system.

The guidance system where the forks are attached to has been built using 10 mm rectified steel guides with nylon bushings. In order to get a good sliding, there have been placed two bushings per guide with a separation of 30 mm.

The electronic system controlling the forklift is based on the Atmel ATMEGA32 microcontroller. It also includes a LSI counter which is used for reading the encoder position. Using such a microcontroller allows extending easily the performance of the forklift adding different electronic devices devoted to data collection: a pressure system, for estimating the weight of the object lifted, or ultrasound / infrared sensors, to measure the distance between the base of the object and the forklift.

The communication protocol with the device is based on the serial protocol through an USB-serial converter. All commands are specified in ASCII code following the same philosophy as the one used for controlling the RobEx platform.

2.3 Stereo head

Currently, most stereo vision systems are designed as motionless systems devoted to obtain 3D visual information from a pair of fixed cameras. The motion of a robotic head is an important aspect to be considered in many robotics problems. With an static head, the robot needs to move its whole body to get visual access to the different regions of interest of its environment. An articulated vision head provides a means of exploring a local environment without any displacement of the robot. It allows perceiving the outside world without significantly increasing the uncertainty of measurements caused by issues such as odometry errors. This is a very interesting feature for several problems in robotics such as visual SLAM.

The stereo head we have built has four degrees of freedom: a neck-alike movement, a tilt movement affecting both cameras, and two separate camera-specific pan movements. The head chassis is entirely made of aluminum, giving a good consistency to the final model. Figure 4 illustrates the stereo head CAD model.

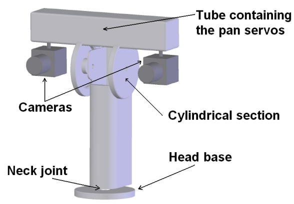

Figure 4: Design of the stereo head

One of the most important decision in this design is the location of the motor that moves the neck. It can be placed whether inside the head or outside, in the chassis of the robot. The last option is easier but leads to a complex transmission system prone to bigger looseness. Placing the motor inside the head results in a more compact and practice design. This way, mounting or unmounting the head on the robot platform is an easier procedure. For this reason, in our design, the neck motor has been placed inside the head. Thus, the chassis of the motor that provides the neck movement is fixed to the neck chassis and its axle is attached to the head base. Such base is used to couple the head to the RobEx platform. An extension of the motor axle is attached to the base using two bearings. This eliminates looseness in the movement of the neck. Moreover, with this design, the weight of the head rests on the bearings, achieving much smoother movements.

The pan movement is independent for each camera. This allows controlling the vergence movement that is necessary to maximize the common visual field of the two cameras. The servo-motor that controls the pan movement is directly attached to the camera though an adjustable base. With this base it is possible to align the motor axle with the optical center of the camera. This alignment allows maintaining a constant distance between the center of projection of both cameras, which simplifies the 3D computation problem. Both servos are placed within a 40x40 mm square-shaped aluminum tube.

The tilt movement is also driven by a servo. This motor provides a rotational joint between the main chassis part and the aluminum tube where the pan servos are mounted. To connect the servo to the upper part, two aluminum cylindrical sections provide a rigid connection with the servo's axle and with a fake axle attached to the servo frame in the opposite side. The aluminum tube containing the pan servos is located in the superior part of these cylindrical pieces.

A Dynamixel RX-servo [9] was chosen for each DOF. Its price and its properties make this servo an ideal option for this application. The case of the servo includes, besides the motor, a gearhead and the RS485 control electronics. This last feature is extremely interesting in practice, because it allows controlling the joints with just four wires: two for power and two for data. The servo bus scales up to 32 nodes. Moreover, because the same servos were used in previous designs, the software controlling the stereo head can be reused.

The only drawback of using the RS485 standard is that computers does not usually support it directly. However, there exist affordable RS485 - USB converters. Another advantage of using RS485 is that it is still alive industrial quality standard (i.e. it will probably be forthcoming improvements). It also supports long distance communication and noisy channels.

3 Software interface

The hardware interface has been built using component-oriented software engineering technology [10]. Thus, for every presented hardware, a separate software component has been implemented. All this components are part of RoboComp [11], an open-source robotics framework.

RoboComp was created in 2005 by the Robotics and Artificial Vision Laboratory of the University of Extremadura. Since then, it has been widely used by many students and researchers of the laboratory. Now, it can be considered a mature project which integrates many components with different functionalities: hardware interfacing (e.g. cameraComp, differentialRobotComp, laserComp, forkliftComp), data processing (e.g. visionComp and roimantComp, for visual features detection, and cubafeaturesComp, for laser features detection), robot behaviors (e.g. gotopointComp, wanderComp) and many others. RoboComp is also equipped with different tools that facilitate component creation, management and debugging.

Regarding the Hardware Abstraction Layer (HAL) components, those related to the hardware elements described in this papers are presented in this section. Listings 1, 2 and 3 shows the implemented interfaces of the RobEx platform, the forklift and the stereo head. Due to space limits, these listings correspond to simplified versions of the actual ones.

3.1 Robot platform interface

The listing 1 shows a simplified version of the RobEx platform interface. It includes methods for controlling and monitoring a robotic mobile differential platform. For instance, the method of line 5 makes the component to return the state of the robot platform (i.e pose, and speed). Those in lines 6-7 and 8-9 are provided in order to control the base speed and odometer respectively.

Listing 1: *DifferentialRobot interface*

```
1  module RoboCompDifferentialRobot
2  {
3   interface DifferentialRobot
4   {
5    void getBaseState(out TBaseState state);
6    void setSpeedBase(float adv, float rot);
7    void stopBase();
8    void resetOdometer();
9    void setOdometer(TBaseState state);
10  };
11 };
```

3.2 Forklift interface

The developed component for the forklift accessory implements the Forklift RoboComp interface. Its simplified version is shown in list 2. Line 5 corresponds to a method that provides the forklift state (i.e. its position and the force it is making). The method in line 6 is used for sending new positioning commands (i.e. the goal position and the maximum allowed force).

Listing 2: *Forklift interface*

```
1  module RobolabModForklift
2  {
3   interface Forklift
4   {
5    void getState(out TForkliftState state);
6    void move(int position, int max_force);
7   };
8  };
```

3.3 Stereo head interface

For the stereo head control, it has been developed a RoboComp component named HeadNTP which implements the interface shown in listing 3. As in the other two listings, it is shown a simplified version of the actual interface. In this listing, methods in lines 5-6 are used to stop the head whether in its *zero* position or in the current one. Those in lines 7-14 are used to move any of the most commonly used combinations of the head motors. In line 15, it is defined a method to make the head to look with both cameras to a specific 3D point. Methods in lines 16 and 17 return the head state and its configuration, respectively. The one in line 18 returns *true* if the head is moving, *false* otherwise.

Listing 3: *HeadNTP interface*

```
1  module RoboCompHeadNTP
2  {
3   interface HeadNTP
4   {
5    void resetHead();
6    void stopHead();
7    void setPanLeft(float pan);
8    void setPanRight(float pan);
9    void setTilt(float tilt);
10   void setNeck(float neck) ;
11   void saccadic2DLeft(float lPan, float tilt);
12   void saccadic2DRight(float rPan, float tilt);
13   void saccadic3D(float lPan, float rPan, float
         tilt);
14   void saccadic4D(float lPan, float rPan, float
         tilt, float neck);
15   void setGaze3D(float x, float y, float z);
16   HeadState getHeadState();
17   HeadParams getHeadParams();
18   bool isMovingHead();
19  };
20 };
```

4 Experimental results

Since the development of the RobEx platform in early 2006 it has been extensively used in both, research and academic environments. The incorporation of new accessories to this platform has given the possibility of testing increasingly complex robot behaviors.

In past editions of this workshop some results of different experiments conducted using the RobEx platform has been presented. They were mainly focused on active perception [12] and autonomous navigation using vision [13]. In this paper a new experiment with this platform is shown. It is about an experiment of modeling of the environment using both, visual information and odometry. Figures from 5 to 12 show the results of this experiment. Since the main goal of this section is to illustrate the potential of the robotic platform and its accessories, no details are given about the employed methods for solving this modeling task. However, it must be stressed that no SLAM techniques have been applied in order to show the proprioceptive capability of the robot.

In this experiment, the robot has to explore and model an environment ($4.9 * 3.8\,m^2$) composed by two rectangular rooms. The robot start the exploration at some point of the first room and computes 3D points of the environment using the visual information provided by its two cameras. Both cameras move driven by a vergence control system that allows maximizing the common visual field to provide a more accurate computation of 3D points. After the exploration of the first room (figure 5), the robot is capable of extracting a model of the explored space (figure 6). The correspondence of this model with the real room is very accurate, as it can be observed in figure 7. This is due not only to the 3D estimation, but also to the precision of the platform odometer.

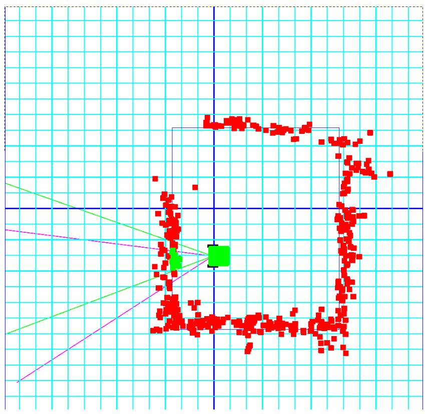

Figure 5: 3D representation of the environment after exploring the first room.

Once this initial model is obtained, the robot carries on with the exploration of the environment, until it reaches the second room. An exploration of

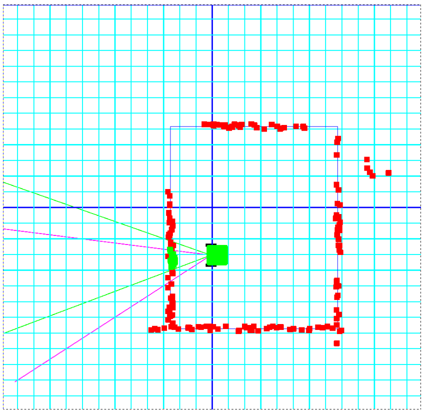

Figure 6: Model of the environment after exploring the first room.

Figure 7: Real situation at instant of the images 5 and 6.

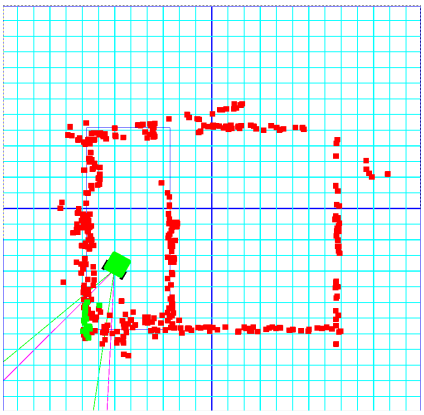

Figure 8: 3D representation of the environment after exploring the second room.

this new space is then performed, producing the 3D estimation result of figure 8. This new estimation allows the robot to model the second room, obtaining the map of the whole environment that is shown in figure 9. The accuracy of this model can be appreciate comparing it with the real image of figure 10.

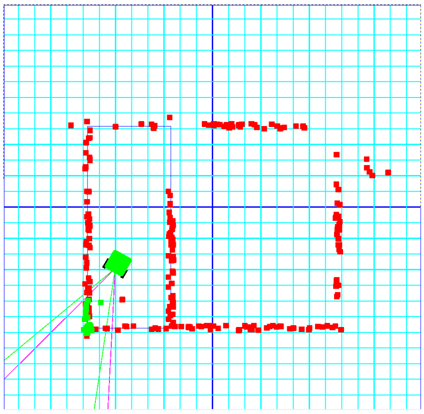

Figure 9: Model of the environment after exploring the second room.

Figure 10: Real situation at instant of the images 8 and 9.

To finish this experiment, figures 11 and 12 depict the robot situation after returning to the first room. Again, results show a high correspondence between the robot estimation and the real world.

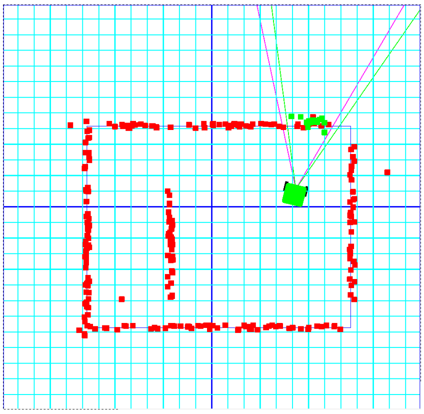

Figure 11: Model of the environment after returning to the first room.

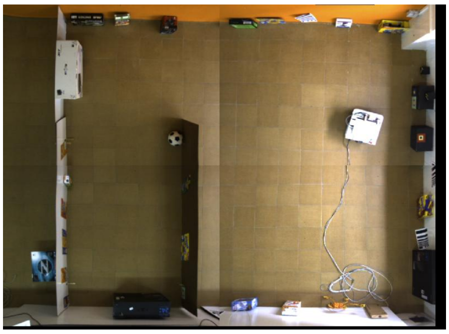

Figure 12: Real situation at instant of the image 11.

5 Conclusion

This paper has described the hardware of a robot platform and some interesting accessories that may be useful for both, research and academic purposes. The mechanical and electronic designs of this hardware are freely available. Also, the software that complements the described robotic system is released under the GPL license.

Since its initial design, many improvements have been made on RobEx. These improvements have given as result a totally equipped robot that can perform many different experiments. Its accessories can be attached to any other robotic platform, allowing researchers to build their own robots in a customize way.

Acknowledgment

This work has been supported by grant PRI09A037 and GRU09064, from the Ministry of Economy, Trade and Innovation of the Extremaduran Government, and by grant TSI-020301-2009-27, from the Spanish Government and the FEDER funds.

References

[1] RobExArena, *http://www.robexarena.com*, 2010.

[2] R. Siegwart and I.R. Nourbakhsk, *Autonomous Mobile Robots*, Massachusetts Institute of Technology, 2004.

[3] Mobile Robots Inc., *http://www.mobilerobots.com*, 2010.

[4] Maxon Motors, *http://www.maxonmotor.com*, 2010.

[5] Atmel Corporation, *http://www.atmel.com*, 2010.

[6] LSI Computer Systems, Inc., *http://www.lsicsi.com*, 2010.

[7] FTDI Chip, *http://www.ftdichip.com*, 2010.

[8] Lego Mindstorms, *http://mindstorms.lego.com*, 2010.

[9] Dynamixel, *http://www.tribotix.com*, 2010.

[10] J. He, X. Li and Z. Liu, *Component-based software engineering: the need to link methods and their theories*, Proceedings of ICTAC 2005, Lecture Notes n Computer Science 3722, pp. 70-95, 2005.

[11] RoboComp, *http://robocomp.sourceforge.net*, 2010.

[12] P. Bachiller and P.Bustos, *Manteniendo el foco: control de seguimiento y vergencia en un sistema de atención visual*, WAF'2008, Vigo (Spain), 2008

[13] L. J. Manso, P. Bustos, P. Bachiller and J. Moreno, *Obstacle Detection on Heterogenous Surfaces Using Color and Geometric Cues*, WAF'2009, Cáceres (Spain), 2009