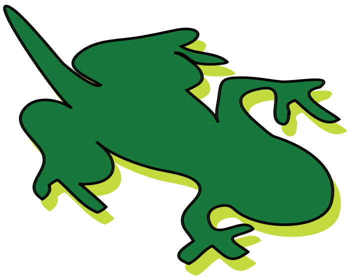
TETARD: What a Nice dive !



Laure Amate, Louis-Kenzo Cahier, Bertrand Copigneaux, Nicolas Dalmasso, Sylvain Héraud, Alexandre Hubert, Angelica Lim, Franck Lourme, Nelly Pustelnik, Fabien Valoit

University of Nice-Sophia Antipolis
Laboratoire I3S
2000 route des lucioles
Les Algorithmes, Bât. Euclide B, BP 121
06903 Sophia Antipolis cedex
FRANCE

Abstract. Already participating to SAUC-E contest in 2006, we come back this year with another AUV. It has the same global structure: 2 watertight compartments, a central structure, 4 propellers and some sensors. Nevertheless major improvements have been done: compartments are more reliable, better quality webcams have been purchased ans we add a Micron Sonar by Tritech. We also ameliorate our embedded code by using Synccharts formalism. Highly supported by our sponsors detailed in the last page of the report and researchers from I3S laboratory, our team is composed by 10 students from University of Nice-Sophia Antipolis listed in the first page. Working all together on this project has been a great challenge for people with various competence.

1 Introduction

Involved for many years in the community of underwater roboticians, some researchers from I3S [1] laboratory decided in 2005 to participate to the SAUC-E contest. Situated at the heart of a technological park and as part of the university of Nice-Sophia Antipolis (UNSA) they aimed to gather students, engineers and researchers to build an autonoumous underwater vehicle (AUV) that would compete with all-over Europe teams gathered in the same place for the competition days.

The first edition of SAUC-E in 2006 was the starting point of the participation of our team. Unfortunately we could not compete because of a major failure of our PC board the day before the contest. Nevertheless the work performed during the year is not useless and it places the basis of our participation in 2007.

Our team is composed of 10 students. Issued of different formations of the university, the abilities of each of us are complementary: computer science, signal and image processing, electronics, mechanics, ... We called our robot ”Têtard” that is the first step of development of a frog, in French. It has 4 propellers, 2 watertight compartments, various sensors and a PC board. It is developped in the next sections.

Even if we re-used some part of last year's work, we improve our robot with adding some innovations such as the Aluminium watertight tubes instead of PVC and the purchase of a Sonar. Added to that major changes, we improved the embedded code and the repartition of electronic components in the tube.

---

[1] Informatique Signaux et Systèmes de Sophia antipolis

2 Design and constraints

Têtard's mechanical structure (Figure 1) have been inspired from several AUV structures that have been developed by american university teams. The design was meant to be simple and as modular as possible. That is why we decided to build an AUV with 2 waterproof compartments, 4 thrusters and a skeleton that linked all parts. This choice has been done in 2006 and tests confirmed our choice:

- The 2 compartments were large enough to put all electronics,
- The bottom compartment filled with batteries stabilizes the robot,
- The robot was positively buoyant,
- The Aluminium structure was modular and we could add sensors or change their positions.

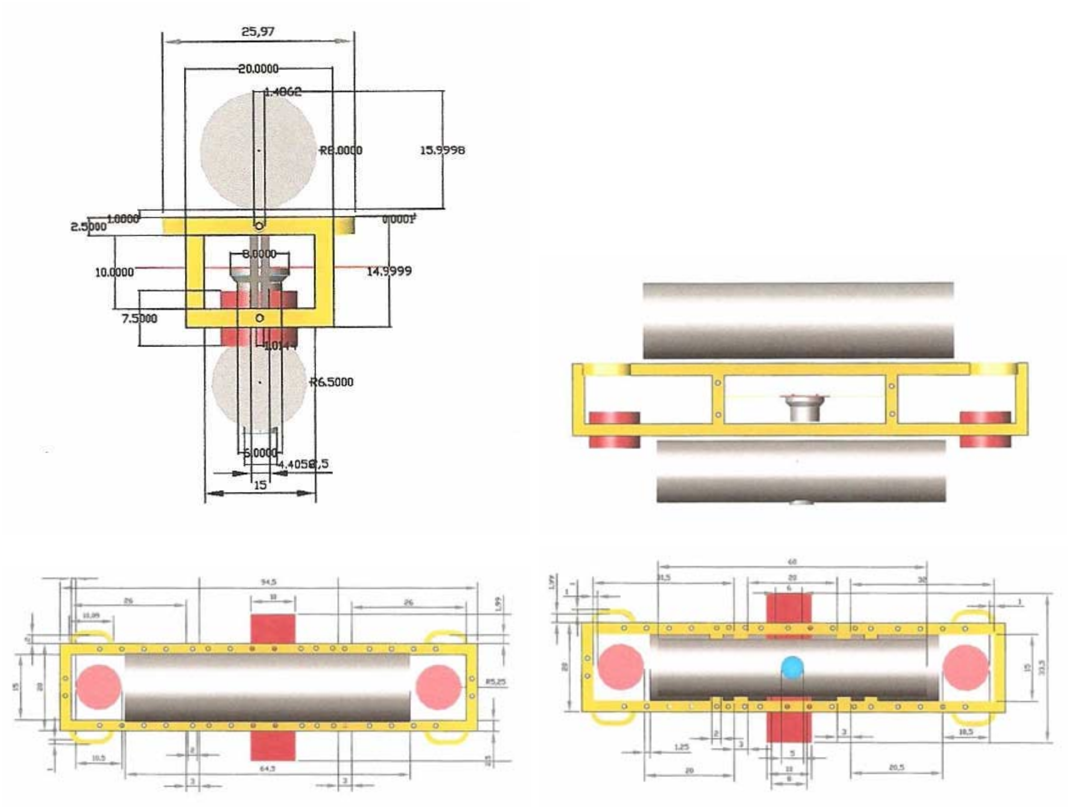


Fig. 1. Drawings of Têtard: front, lateral, top and bottom.

Last year robot was build with PVC pipes and we have had problem to maintain their watertightness. So we decided to build them using Aluminium. Added to the reliability of this material, it is heavier than PVC so that we obtain a less positive buoyancy.
The existence of four motors provides 4DOF (surge,heave,pitch and yaw) which can be actively controlled. The two horizontal motors control surge and yaw while the two vertical motors control heave and pitch.

The combination of a large and light upper hull with a small and heavy lower hull keep the center of buoyancy high and the center of mass low. This geometry avoids the need for roll thrusters because of its passive roll stability.

An hydrodynamic study has been done to know the hydrodynamic coefficients of our AUV. With fitted devices we determined forces applied on our robot and with theoretical tools we find some coefficients like added mass, gravity center, inertia, ...
To drive our autonomous vehicle we design a hydrodynamic model based on these determined coefficients. This model is useful if we want to know the underwater behavior of Têtard. However, between theory and practice there are some differences and our hydrodynamic model was not exactly the same in reality, so we need to improve it by taking into account some parameters like the real repartition of weight, the motor rotation speed and accuracy of sensors.

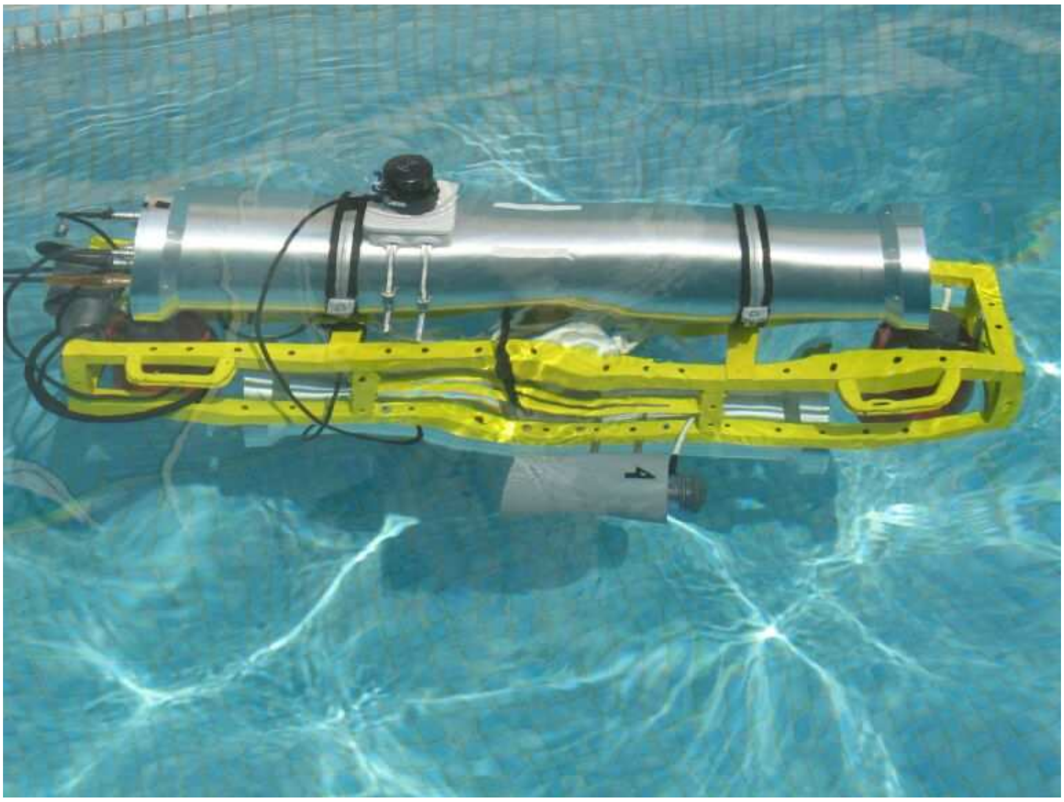


Fig. 2. Picture of Têtard.

3 Electronic part

Our robot is controlled by using propriocetive and exteriocetive information that are produced by different sensors. The entire electronic system allows that raw signals from sensors are correctly conditionned so that the intelligent part of the robot can use them.

3.1 Power supply

The execution of the mission by our AUV requires 3 lead batteries connected in series of 12V and 7Ah each. They are placed in the bottom compartment to move down the mass center of the robot.

We add a security so that the circuit is closed as soon as a magnet is placed on a magnetic interuptor. It is re-open when the magnet is placed a second time. It allows a diver to totally stop the robot in case of emergency.

3.2 Sensors

Figure 3 shows the global electronic architecture onboard and details below. Most of sensors are linked with the Beck board with an I2C bus composed of 2 lines: one for the clock signal and on efor the data signal. The voltage signals from sensors are transformed into data in the Beck and then they are transmitted to the PC board via an Ethernet link. The mission's tasks are managed onboard the PC. Then the motors are also command via the I2C bus. Only the 2 webcam are directly linked to the PC board. The Sonar data are extracted using the Beck board.

The utility of the 2 boards is that the Beck acquires and conditions the raw signals and only the useful data is sent to the PC.
The power supply board provides all voltages nedded by the system: 5V, +/-12V, 3.3V.

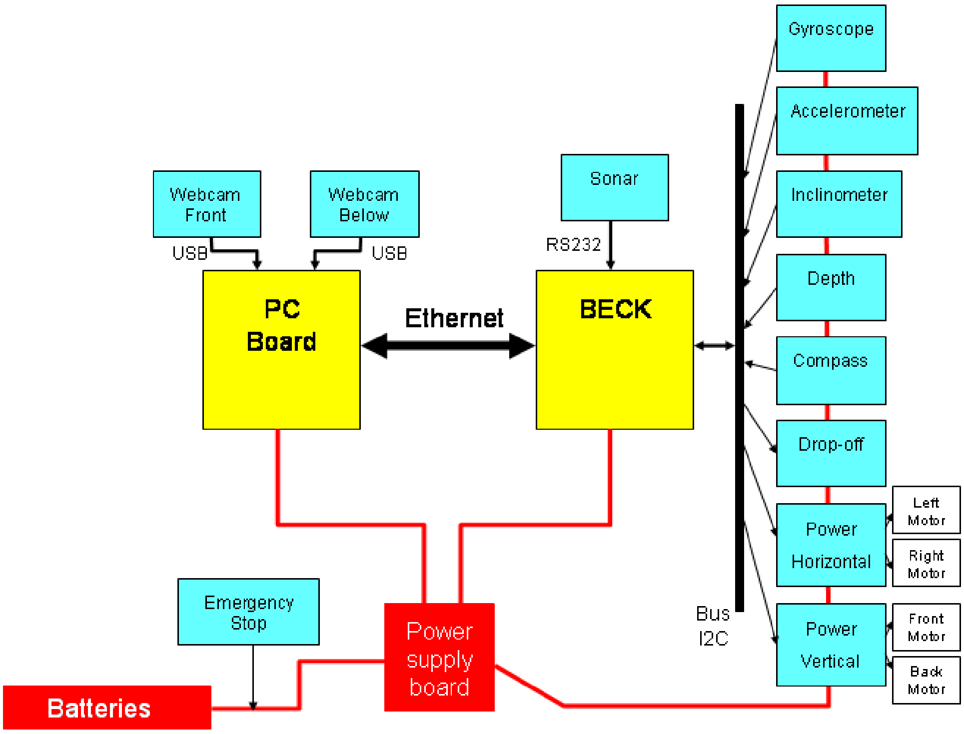


Fig. 3. Electronic schema.

Sensor	Description	Link	ref. manufacturer
Gyroscope	This integrated dual-axis angular rate sensor provides voltages proportional to the angular rate.	I2C	DG-300 by InvenSense
Accelerometer	This 3-axis accelerometer delivers voltages proportional to accelerations.	I2C	MAA7260Q by Freescale
Inclinometer	This Dual axis sensor provides the 2 inclinations with respect to the plane with a resolution of 0.02 degrees.	I2C	Spectrotilt SSY0090
Depth	This pression sensor provides a linear voltage output proportional to the depth of the robot.	I2C	MPX2200 by Motorola
Compass	It provides the orientation of the robot with respect to the North with a resolution of 0.5 degrees.	I2C	KVH C100
WebCams	It provides images of the environment of the robot so that, with an appropriate processing, we can extract information to perform the mission. It can produce 30 images per second.	USB	Logitech QuickCam Fusion
Sonar	It is used to detect object for obstacle avoidance and to measure the wall of the pool. After extracting line from the measures we can reconstruct the pool and estimate the position of the robot.	RS232	Micron by Tritech

3.3 Embedded PC

"Têtard" works with a *Versalogic Cheetah*. It is a PC/104 embedded PC with a Pentium M Processor (600 Mhz to 1.6 Ghz fully adjustable), 1 GB of SODIMM DDR RAM, a 400Mhz processor-side bus, a Compact Flash Socket and an on-board I/O (with 2 USB 2.0 ports, 2 COM Ports, an Ethernet Port, an IDE 2.5Ť interface and a LPT Port).

As operating system, we choose Windows XP Pro for its compatibility with almost all existing webcams, with the Sonar and with wifi key. We plug onto it a 40GB hard drive for logs and image saves. We choose Cheetah for its tiny size (3.55" x 3.775" on two boards) and its performances with our image processing algorithms: the analysis of one image takes 140ms.

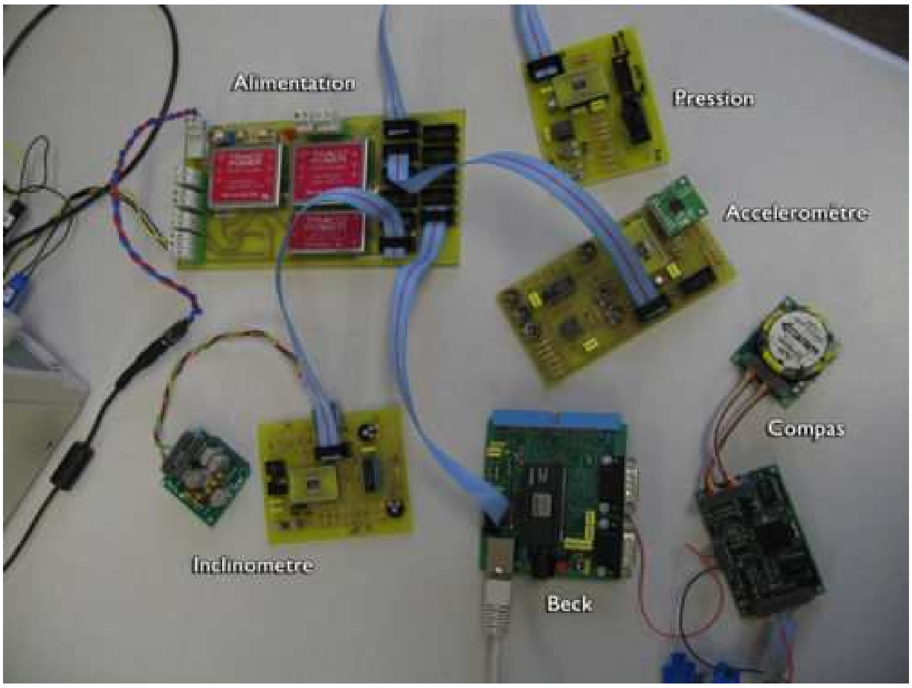


Fig. 4. View of some electronic components.

4 Mission planning and simulation

Now that we have developped the electronic and the mechanical parts of our robot, we are interested here in the intelligence onboard the robot. The planning of the mission is performed by an automaton coded using Esterel Studio software. C++ Algorithms are included in this automaton to process images from webcams and to process Sonar data. Added to these embedded softwares, we developped a simulator with the help of Robosoft company. This simulator is very important to first test our codes in the lab, before testing into water.

4.1 Automaton

The control automaton is the module which decides what the robot should do at any given moment, and coordinates the actions of other modules as a result. We used Esterel Technologies Esterel Studio and their Safe State Machines formalism to develop the control automaton of our AUV. This technology was chosen for several main reasons: reactive programming support; the reliability and proven formalism of their tools; the possibility of interactive validation/simulation with the automaton.

The automaton is made up for 4 parallel components (see Figure 5), corresponding to the following concurrent tasks:

Control	The *control component* is in charge of orchestrating the different parts of the system and is the only module with the power to make decisions. In general, it takes into account information sent by the other automata to make its decisions, and it sends them instructions to execute.
Guidance	The *guidance component* is responsible for putting the robot into one of the specified guidance modes. Each guidance mode corresponds to a common goal for the set of motors, such as *pass through door*. This automaton ensures that the motors are activated for only one given task at a time.
Observation	The *observation component* is used to prioritize object searches based on Têtard's observed environment. The automaton keeps a list of dynamically-allocated priorities, which are calculated depending on, for example, the objects already found, or objects which are more important because they are used in more than one mission (eg. cone).
Security	The *security component* verifies that the robot is safe at all times. If it detects electrical problems, sensor glitches , or an imminent collision, it sends a signal alerting the other components of the system.

4.2 Image processing

We use images from our forward-facing and downward-facing webcams for two purposes: to guide our robot around or toward objects, and to recognize the cone, tire, buoy, dummy and target. The raw images are stored under OpenCV image

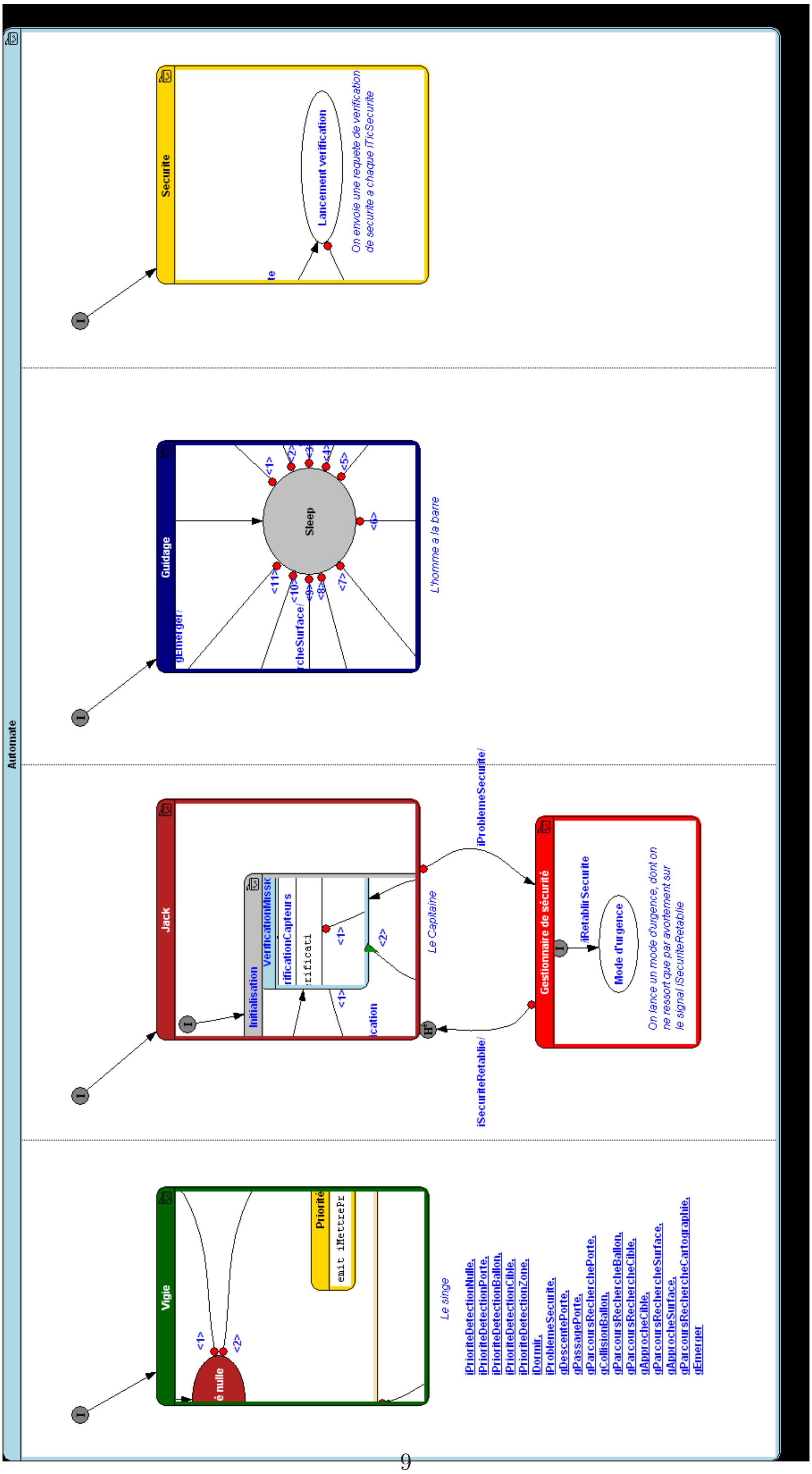


Fig. 5. Automaton.

format for processing using our own ”home-made” colour filters and shape recognition functions based on geometric properties.

We found that the most efficient way to detect objects is to filter images first by colour, and then by shape. Because the color filter would be used most frequently, we implemented the fast color detection algorithm described by Bruce and al. [2]. With this method, we first convert our images into the HLS colour space, where each pixel is represented by Hue, Lightness, and Saturation components. In real-time robotics, this color space is the most useful because the ”lightness” component accounts for shadows and variations in lighting. To find a blob in the image of particular colour, we check to see if a pixel's colour values fall within specified bounds in 3D space (see figure 6). However, this check requires 6 comparisons on integer values, and proved inefficient when performed on all pixels in an image. We were able to increase our image processing speed by 30% by representing colours in binary array form and performing bit-wise AND operations, as described in Bruce et al.

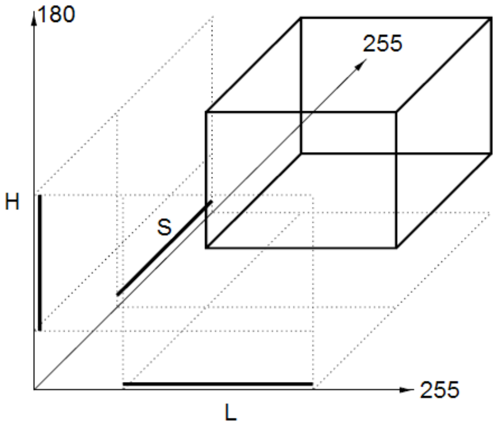


Fig. 6. HLS colorspace.

Once a specified colour blob is found in an image, we run a dilation operation on the shape to reduce noise and irregularities in colour caused by reflections. We then perform the associated shape algorithm on the resultant pixels. In the case of the buoy, dummy, target and tire, we look for circles. Our circle detection algorithm is based on a simple ratio of a shape's area to its perimeter:

$$\frac{area}{perimeter} \simeq \frac{radius}{2} \tag{1}$$

---

[2] J. Bruce, T. Balch, M. Veloso, *Fast and inexpensive color image segmentation for interactive robots*, Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '00), vol 3, pages 2061-2066.

4.3 Simulator

We decided to use a 3D Simulator to test and validate the TETARD Automaton. This simulator is based on the Microsoft Robotics Studio simulation environment, mainly because itŠs a convenient and reliable tool for robotic simulation but also because we already had a good knowledge of this environment (2 students are currently placed by Robosoft for their internships).
The TETARD Simulator has been conceived as a distributed application, the Simulator runs on one machine, while the TETARD framework runs on another. The communication is done threw a UDP protocol. The simulation sends information of simulated sensors (orientation, depth, speed, angular speed) and of a simulated camera. And the TETARD framework sends order to the simulation, controlling directly the speed of the robots motors. (See Figure 7)

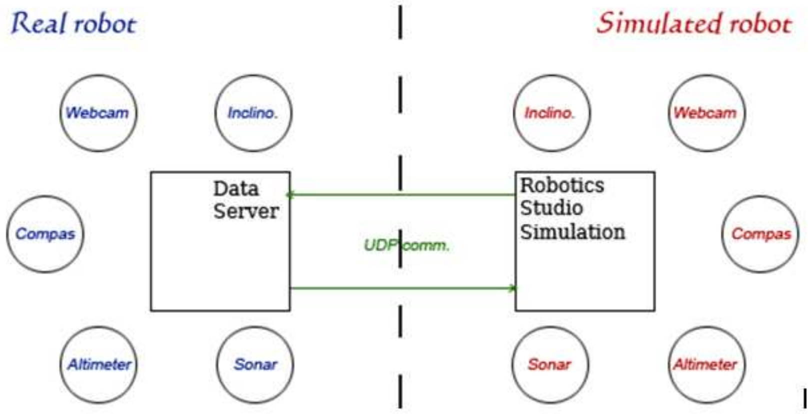


Fig. 7. Simulator Protocol.

The main interest of the simulation is to test the automaton and the TETARD AI, but thanks to the fidelity of the simulated environment we created, the images of the simulated camera can also be used to test and calibrate the image processing algorithms. We modelized a complete 3D environment representing the real challenge environment, with all the accessories. And we will also adjust the simulated environment during the competition to be able to test our software during the competition even if we do not have a constant access to the pool. Thanks to the Microsoft Robotics Studio environment we will also be able to apply post rendering effect on our simulated images, using pixel shaders, to render a more realistic water. In figures 8 we can see 2 snapshots of the Simulator.

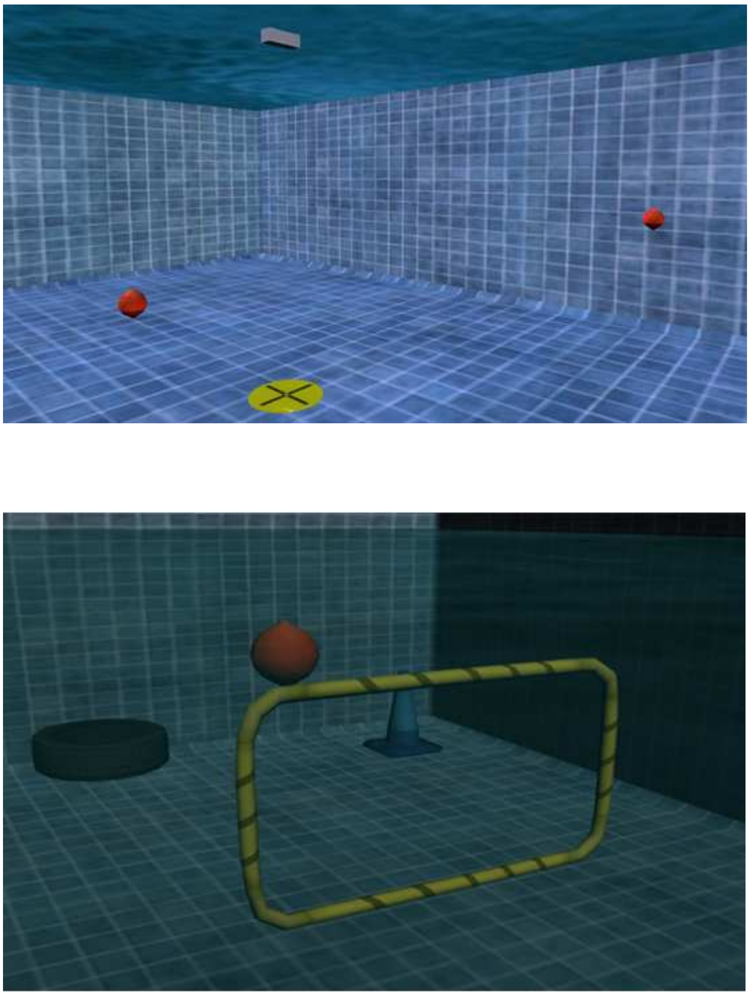


Fig. 8. Snapshots of the simulator.

5 Improvement

This year AUV ”Têtard” is equiped with more sensors and thus the processing of obtained data is much complex and reliable. Indeed we add a webcam with its image processing for positioning and navigation of the robot. The same aim is achieved by using the Micron Sonar even if, as a new sensor, the processing of scans is not as performant as we would have wanted. Work has still to be done in this area.

Adding to these sensors, our embedded PC is much more performant than the one we used last year. Because all the process of webcams images and Sonar scans must be done onboard, we need a fast processor and a big memory storage.

Moreover we change the material of the watertight compartments. We learned from last year experiments that watertightness is not an easy task to achieve. Especially because of that we choose to build them in aluminium because of its lightness. It proves to be a good choice as we now have a light AUV, very reliable.

6 Financial summary

Expenditure:

Materials	**Total**	**13600€**
	SONAR	6000€
	Cameras	250€
	Embedded PC	1800€
	Inertial components	200€
	Electronic components	600€
	Spare items	250€
	Structure	2000€
	Watertight connectors	1500€
	Laptop	1000€
Trip	**Total**	**6000€**
	Car location	600€
	Highway and oil	1000€
	Ferry	200€
	Plane and train/coach tickets	2700€
	Accomodation and food	1500€
Communication	**Total**	**500€**
	Posters	150€
	T-Shirts	350€
Total		**20100€**

Income:

Sponsors	**Total**	**19600€**
	DGA	10000€
	UNSA	3000€
	Fischer Connectors	3000€
	Siemens	3000€
	Air France	600€
Balance of 2006		**500€**
Total		**20100€**

7 Conclusion

Our participation to the SAUC-E challenge comes from the willingness of a group of students to invest their time in building an autonomous underwater vehicle. We were supported by Maria-João Rendas and Christian Barat, researchers from I3S, and Jean-Pierre Millet, founder of PEASM industry.

As part of this project, we involved oursleves in ROSSA[3] association that was created in 2006 to promote underwater robotics in the Sophia Antipolis area particularly with the SAUC-E contest and also with various actions directed toward public (Science festival, ...). As exposed in the financial summary we found different support to participate to SAUC-E. We particularly want to thank DGA, Fischer Connectors, UNSA, DCN, Siemens, Air France for their material, financial and advice help. We want also thank people from GESMA for their help. We want also thank Gilles Menez and Viviane Rosello for their support, and Jean-Marc Fédou for his pool.

---

[3] Robotique sous-marine de Sophia Antipolis

DGA

DCN
L'EXPERT DES
SYSTÈMES NAVALS

Université
Nice SOPHIA ANTIPOLIS

fischer
CONNECTORS

SIEMENS

i3S
sophia antipolis

iut
Nice Côte d'Azur

AIR FRANCE

ESTEREL
Technologies

PEASM
Industry