UNIVERSITÀ DEGLI STUDI DI PARMA

Dottorato di Ricerca in Tecnologie dell'Informazione

XXVIII Ciclo

Designing a Computer Vision System for Underwater Robotic Interventions

Coordinatore:

Chiar.mo Prof. Marco Locatelli

Tutor:

Chiar.mo Prof. Stefano Caselli

Dottorando: *Fabio Oleari*

Gennaio 2016

If we knew what we were doing,
it wouldn't be called research, would it?
A. Einstein

Contents

Introduction

This thesis deals with the challenging problem of designing systems able to perceive objects in unstructured and difficult environments. In the last decades research activities in robotics have advanced the state of art regarding intervention capabilities of autonomous systems. This fascinating step forward is based on advances in several research subfields: localization and navigation, real time perception and cognition, safe action and manipulation capabilities. State of art in these fields, applied to environments like ground (both indoor and outdoor) and space, has now reached such a readiness level that it allows high level autonomous operations. On the opposite side, the underwater environment remains a very difficult one for autonomous robots. Key technologies like localization and perception must tackle still unsolved problems in the underwater context. Water influences the mechanical and electrical design of systems, interferes with sensors by limiting their capabilities, heavily impacts on data transmissions, and generally requires systems with low power consumption in order to enable reasonable mission duration. Interest in underwater applications is driven by needs of exploring and intervening in environments in which human capabilities are very limited. Indeed, comparing with ground robotics, in which autonomous robots mostly substitute human activities, underwater robotics is a matter of reaching new frontiers that would not be reachable in other ways.

Nowadays, most underwater field operations are carried out by manned or remotely operated vehicles, deployed for explorations and limited intervention missions. Manned vehicles are directly on-board controlled by human operators that stay in the field of the mission, with advantages in terms of environment perception and intervention capabilities. The risk of human presence in a hostile environment is the most significant drawback of these solutions, together with reduced mission time. Remotely Operated Vehicles (ROV) currently represent the most advanced technology for underwater intervention services available on the market. Some specialized companies offer ROV-based maintenance services, especially to Oil and Gas offshore industries. These vehicles can be remotely operated for long time but they need support from an oceanographic vessel with multiple teams of highly specialized pilots. Moreover, the umbilical cable, needed for remote control, is often a problem for deep or far-away missions. Diffusion of ROV interventions is largely limited by impressive mission costs that generally reach dozens of thousands dollars per day, mainly due to vessels, vehicle remote control technologies and equipped rooms, and humans costs.

Increasing needs of long term missions for long range underwater operations imply the unsuitability of manned vehicles as well as ROVs, for both costs and risks, and have pushed research activities in autonomous underwater robotics. These activities so far have been mainly focused on exploration and survey tasks with important applications in fields of oceanographic, geological, biological and archaeological sciences. Vehicles equipped with multiple state-of-art sensors and capable to autonomously plan missions have been deployed in the last ten years and exploited as observers for underwater fauna, seabed, ship wrecks, and so on [1–6].

On the other hand, underwater operations like object recovery and equipment maintenance are still challenging tasks to be conducted without human supervision since they require object perception and localization with much higher accuracy and robustness, to a degree seldom available in Autonomous Underwater Vehicles (AUV). Indeed, the importance of underwater robots able to intervene in critical situations is ever more evident nowadays. Several fields would benefit from underwater vehicles with autonomous manipulation capabilities: the science community could exploit robots for sample collection and excavation and coring, offshore industries

could perform maintenance activities more easily and at a fraction of current costs, and moreover operations for disasters recovery and rescue would greatly benefit from autonomous interventions in hostile environments.

This thesis reports the study, from design to deployment and evaluation, of a general purpose and configurable platform dedicated to stereo-vision perception in underwater environments. Several aspects and challenging topics related to the peculiar environment characteristics have been taken into account during all stages of system design and evaluation: depth of operation and light conditions, together with water turbidity and external weather, heavily impact on perception capabilities. The platform proposed in this work is a modular system comprising off-the-shelf components for both the imaging sensors and the computational unit, linked by a high performance ethernet network bus. This design philosophy aims at achieving high flexibility in terms of potential perception applications, that should not be as limited as in case of a special-purpose and dedicated hardware. Flexibility is required by the variability of underwater environments, with water conditions ranging from clear to turbid, light backscattering varying with daylight and depth, strong color distortion, and other environmental factors. Furthermore, the proposed modular design ensures an easier maintenance and update of the system over time.

Performance of the proposed system, in terms of perception capabilities, has been evaluated in several underwater contexts taking advantage of the opportunity offered by the MARIS project, described in the next chapter. Design issues like energy power consumption, heat dissipation and network capabilities have been evaluated in different scenarios. Finally, real-world experiments, conducted in multiple and variable underwater contexts, including open sea waters, have led to the collection of several datasets that have been publicly released to the scientific community.

MARIS, Marine Autonomous Robotics for InterventionS

The testbed of the technologies developed in this thesis is the MARIS (Marine Autonomous Robotics for InterventionS) research project. Supported by MIUR (Ministero dell'Istruzione, dell'Università e della Ricerca) as a PRIN (Major National Relevance) project, MARIS aims at studying and then developing and integrating technologies and methodologies for deployment of underwater robots with interventions capabilities and able to cooperate. In particular, the project final objective is the realization of a proof-of-concept experimental demonstration in which two autonomous underwater vehicles, equipped with developed technologies like sensors, arms and grippers, detect a submerged pipe and recover it to the surface by performing a cooperative manipulation (fig.1) [7].

In order to pursue such ambitious objective, advances in a number of enabling technologies must be achieved. In particular, control and guidance of the floating base must be reliable and will exploit inertial sensors, doppler velocity meters, external acoustic positioning systems and new sensor fusion techniques. Control of object manipulation and grasping is integrated in the reactive control system of the entire vehicle for optimal exploitation of the available degrees of freedom, based on extant constraints. The cooperation between robots must rely, at "vehicle-level", on advanced underwater communication techniques and, at higher level, on a mission planning able to decompose tasks and distribute them to each agent [8]. Finally, each vehicle must be able to perceive the underwater environment to reliably detect the target object and to precisely estimate its position with enough accuracy to perform grasping operations. Underwater perception is committed to stereo-vision techniques mainly because of the lower costs and higher versatility of imaging systems with respect to laser- or sonar-based systems. In the underwater environment computer vision has both advantages and disadvantages, compared to other sensing modalities, as will be discussed in the next section. Moreover, given appropriate water and light conditions, computer vision becomes a key enabling technology for object localization and manipulation. The philosophy behind the idea of cooperative manipulation assumes the availability in the future of multiple underwater agents; therefore the

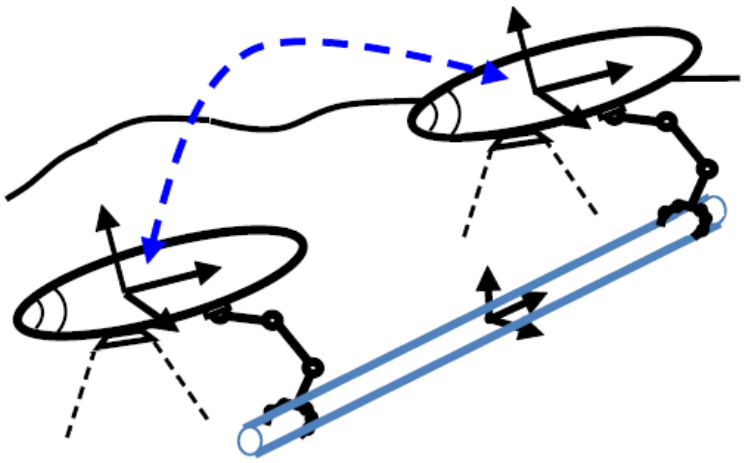


Figure 1: The MARIS concept: a team of floating manipulators performing an underwater cooperative intervention based on real-time perception of a target object.

overall cost of the perception system must be kept within reasonable values. This requirement is satisfied by the design of the perception system developed in this thesis, which is based on integration of off-the-shelf components and does not depend on expensive custom-designed hardware.

Required perception accuracies, challenging manipulation targets and real-time processing constraints of the MARIS project represent a test bench for the system and techniques proposed in this thesis. Furthermore, the underwater experiments conducted within the project have provided an opportunity to both collect datasets, whose availability is very scarce if compared with out-of-water robotics, and perform an on-the-field evaluation of the system. The design of the computer vision system has been carried out according to a spiral development model, involving construction of a preliminary prototype to gather early feedbacks from the field and to refine requirements. The final system, fully integrated in the MARIS vehicle, incorporates the improvements and suggestions resulting from the early prototype.

This thesis is organized as follows. Chapter 1 describes further the problem addressed by this work together with the state of art, with a particular focus on underwater systems for interventions. Preliminary studies, the deployment of a low-cost system, and its evolution in the final developed platform are described in Chapter 2. Chapter 3 evaluates the vision system in perception tasks. The evaluation of the developed system in real-world testing campaigns is discussed in Chapter 4. Chapter 5 describes the integration of the vision platform into the AUV developed by the MARIS consortium and the experiments that exploit perception in the manipulation control loop. Finally, some conclusions and recommendations for future works are proposed in the last chapter.

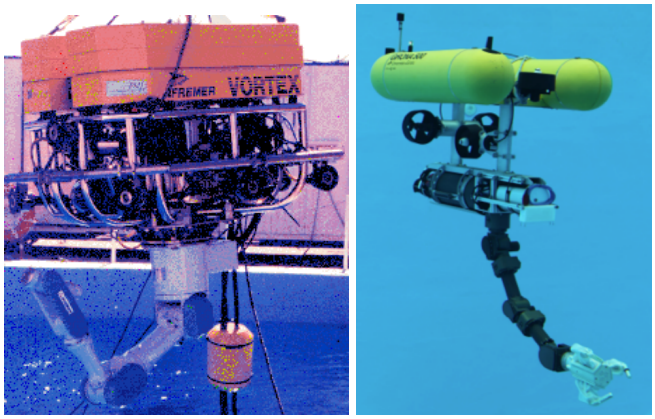
1

Problem Description and State of the Art

The increasing demand for underwater autonomous systems with interventions capabilities has driven the research in several fields. Floating control, localization and navigation approaches, mission planning, cooperative strategies, manipulation and grasping techniques are only some of topics involved in the step forward which is needed to improve the readiness level of AUV systems. Perception of the scene and environment understanding capabilities are among the key technologies on which autonomous intervention applications rely. Main issues are generally related to the capability of retrieving robust and reliable perceptual data while providing adequate real-time data processing and coping with all constraints arising in the underwater environment. The limited perception capability is one of the main obstacles preventing a larger progress of technologies for autonomous underwater interventions.

This research field was born at the beginning of the '90s with first experiments of autonomous floating vehicles with manipulation capabilities. An extend survey on the -rather slow- evolution of these systems is presented in [9], and main experiments, from pioneering endeavors like UNION [10] to modern vehicles like TRIDENT [11], are there summarized (fig. 1.1). Worth mentioning are also other early approaches to underwater autonomous manipulation like OTTER [12], ODIN [13] and AMADEUS [14, 15] projects. AMADEUS, in particular, represented the first attempt in developing a dexterous gripper suitable for underwater applications.

According to Pérez et al. [9], the reasons behind the slow progress of this quite new technology would be "*complexities on required mechatronics (e.g. the vehicle, hand-arm, all kind of sensors, etc.); very hard communication problems; intelligent control architectures needed; letting apart the hostile environment inherent to underwater (e.g. poor visibility, currents, increasing pressure with depth, etc.)*". Nevertheless, according to the "Long-term AUV vision" by Gilmour et al. [16] "*the technology for light intervention systems is still immature, but very promising. Intervention AUVs are currently in level 3 out of 9 (9 meaning routinely used) of the development cycle necessary to adopt this technology in the oil and gas industry, being expected to achieve up to level 7 by the end of 2018.*" This important motivation for pushing research on all enabling key technologies is agreed also by Ridao at al. in their survey on next challenges for AUVs [17].



(a) (b)

Figure 1.1: UNION, the first experiment of underwater manipulation in mid 90s (a). TRIDENT, the state of art of intervention AUVs (b).

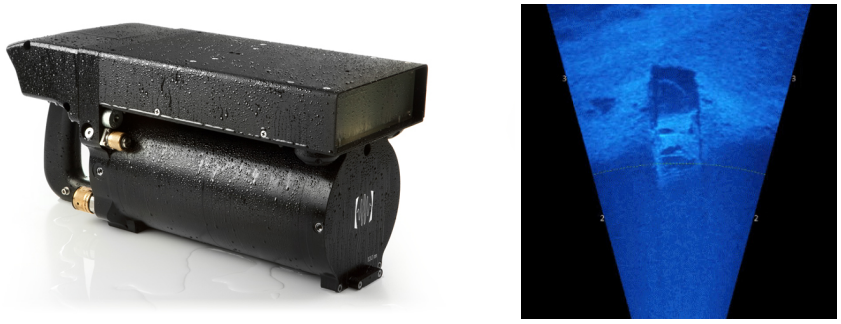
This thesis is focused on the perception in underwater environments based on computer vision approaches. Although computer vision is a major sensing modality in robotics, in underwater environments it is still a marginally investigated approach, especially in applications in which perception should support manipulation tasks for autonomous interventions. A confirmation of this fact comes by searching in the IEEE

Xplore digital library all scientific papers with keywords “underwater vision” and focusing on main pertinent conferences (OCEANS - the flagship conference for marine robotics, ICRA, IROS) of the last three years. The search returns a total of 103 articles and, among these, only 10 are related to object detection, interventions and 3D reconstruction. Other works deal with image calibration and enhancement, navigation and SLAM and, finally, seabed inspection for biological or geological purposes. The reasons that explain why computer vision is largely adopted in “out-of-water” applications and only marginally used in underwater environments can be summarized in two aspects: impact of water in light transmission and limited computational power of embedded systems.

1.1 Alternative sensing modalities

Main alternatives to computer vision for underwater perception are represented by acoustic and laser sensors. Ultrasonic sensing is a commonly used and robust underwater perception modality, in particular for localization, seabed monitoring and long-range detection of vessels wrecks, ruins or, in general, human artifacts. However, acoustic sensing is not suitable when an accurate and detailed reconstruction of the object shape is required. Sonar array cameras have been developed which exploit the emission of multi-frequency acoustic signals for detection and recognition of objects (fig.1.2). These systems allow 3D sonar imaging [18] and extend their application to recognition tasks, but their high cost, limited resolution, and operational complexity restrict their application domain. Furthermore, in scenarios of cooperative robotic interventions, in which multiple to many autonomous robots work in the same environment, problems of interferences between several active acoustic sensors may arise and their coexistence should be studied and verified.

A rather extensive survey and comparison of state-of-art ultrasonic technologies with vision in underwater scenarios is presented in [19]. Although some techniques have been proposed, object detection with acoustic techniques is difficult and authors point out that next challenges for this technology are related to terrain mapping and



(a) (b)

Figure 1.2: DIDSON Camera developed by Sound Metrics Corporation (a); an example image returned by the acoustic camera (b).

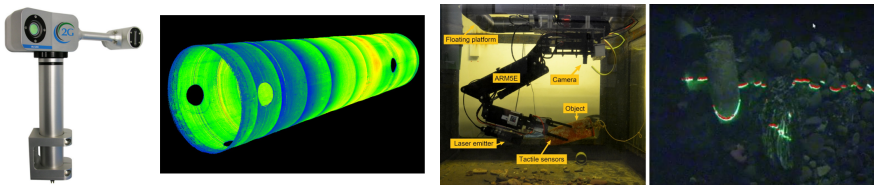
terrain based navigation. Furthermore, the authors also suggest that fields of seabed reconstruction and fauna monitoring would take advantages from the deployment of vision systems on underwater autonomous vehicles. The authors conclude that "A major challenge for the underwater instrument designers and engineers is therefore to be able to create and market modern equipment which is affordable not only to the specialists and high-tech companies, and that is easy to use and deploy, and that yields reliable results". As it will be later discussed, the easiness of deployment and maintenance and the reliability with multiple and different applications will be among main guidelines for the vision system design, addressed in this thesis.

Other active sensors for underwater perception are represented by lasers. Underwater laser scanners (fig.1.3.a) exploit an accurate modeling of light propagation in water means [20]. Such sensors can provide high performance in term of resolution and accuracy of acquired 3D images. However, underwater laser scanners are very expensive (hundreds of thousands Euro) and affected by the same operating problems of vision systems, since they also rely on light transmission in water.

Finally, very short range perception could rely on structured light techniques. This approach to object detection and pose estimation is described by [21]. This work describes a system for underwater object manipulation where object perception is achieved using a structured light laser attached to the forearm of the manipulator

(fig.1.3.b). Unknown objects are successfully grasped in a water tank environment. However, structured light systems are only available as research prototypes, suffer the same problems of light propagation in water, and require long and slow movements of the light emitter in order to reconstruct the scene.

Summarizing the survey on acoustic, laser, and mixed systems, it can be concluded that huge costs and task specificity are the main issues that prevent a larger adoption of these systems, making underwater computer vision an open and interesting research field.



(a) (b)

Figure 1.3: An example of a short range underwater laser scanner with a point cloud 3D reconstruction of a pipe (a). The structured light system proposed in [21]: system within the water tank testbed and image retrieved with laser peak detection (b).

1.2 Underwater vision

Artificial vision applications in underwater environments started to be adopted in tasks that do not require online processing. Tasks of seabed mapping with image mosaicing [22] can be planned with a preliminary environment exploration for images and data collection followed by an offline data processing task. This approach permits both to keep the underwater vision system as simple as possible and to exploit the performance of conventional computational units. Furthermore, the algorithmic approach to the problem does not face time-related constraints and several techniques of image restoration, enhancement and deblurring can be performed. Applications of seabed mapping are generally based on feature detection and association between multiple images with techniques translated from standard, out-of-water computer vi-

sion methodologies.

Feature-based techniques are also exploited in underwater navigation methods based on SLAM approaches. In underwater environments, navigation is indeed a challenging problem due to generally poor vehicle odometry and absence of GPS signal. The readiness level of SLAM methods in standard robotics has made them attractive also for underwater applications and some feature-based approaches have been studied so far [23]. Nevertheless, water environment peculiarities impose further investigations. The underwater environment indeed brings difficulties that prevent an easy re-use of technologies and methodologies already deployed and validated in other robotic fields like ground, air and even space. These difficulties mainly come from water and its response to signal transmission which results limited and distorted.

Water turbidity, color aberrations, light reflections and back-scattering phenomena represent the major problems with underwater computer vision applications. Furthermore these problems depend on several aspects like working depth, weather conditions, water surface movements, sandy or rocky seabed and, in general, the natural environment in which the application is deployed. Several aspects have been studied so far and some specific algorithmic solutions have been proposed. However, the design of a reliable underwater computer vision system, able to operate in-field and in such different conditions, must take care of those aspects and should be powerful enough to support multiple processing pipelines.

As stated above, in underwater environments different scenarios lead to different and multiple problems to be faced. Rahman et al. [24] state that in underwater environment light refraction through multiple media (air, water, lens, etc.) leads to an increment of the radial distortion. In the literature, different approaches for camera calibration correction have been exploited. In particular, starting from the Brown lens distortion model [25] several automatic correction methods have been proposed. Gonzalez-Aguilera et al. [26] presented an iterative numerical approach for the automatic estimation and correction of radial lens distortion using a single image. The proposed method used several geometric constraints such as rectilinear lines and vanishing points of a single image acquired in outdoor environment.

Water turbidity affects the performance of feature descriptors as presented by

Garcia et al. [27] in an extensive comparison between different methods, although not used for object detection purposes. Since underwater imaging suffers from short range, low contrast and non-uniform illumination, simple color segmentation is one of the few viable approaches. Several object detection algorithms [28–30] exploit color segmentation to find one or more areas and perform a more accurate assessment on the found region of interest. Aulinas et al. [31] search salient color regions of interest in order to select stable SURF features as landmarks in SLAM applications. Without color segmentation, the data association is unreliable even for scene description purposes. When searching for human made objects, regions of interest could be detected by using criteria based on contour and texture information. This problem has been addressed by Olmos et al. [32], however the proposed method does not return a region in the image containing the object, but simply a binary decision about the object presence.

1.3 Underwater stereo vision

Stereo vision systems have been only recently introduced in underwater applications due to the difficulty of calibration and the computational performance required by stereo processing. To improve homologous point matching performance, Queiroz-Neto et al. [33] introduce a stereo matching system specific for underwater environments. Disparity of stereo images can be exploited to generate 3D models, as shown in [34,35] and some further investigations have also been conducted on asynchronous stereo vision system by Leone et al. [36]. Although the 3D reconstruction achieved by underwater stereo vision may be satisfactory to represent the main elements of a scene, its accuracy is generally not sufficient for the detailed perception required in object detection and recognition, not to mention object manipulation.

In [30] the underwater stereo vision system used in the TRIDENT European project is described. Object detection is performed by constructing a color histogram in HSV space of the target object. In the performed experiments, there is an intermediate step between inspection and intervention where the real images of the underwater site to manipulate are available and used for acquiring the target object appearance

and for offline training of the recognition system [37].

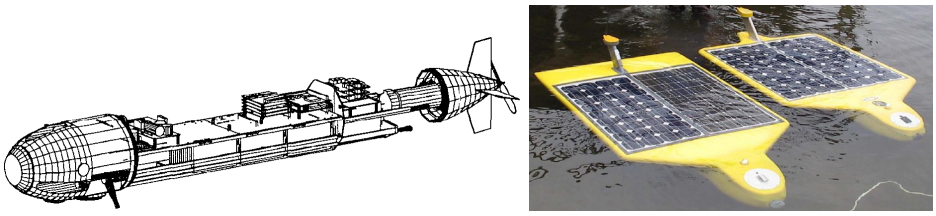
To our knowledge, TRIDENT represents the state of art of published research of vision for underwater autonomous manipulation. In TRIDENT manipulation experiments have been successfully carried out in clear waters in laboratory tanks [38, 39] and in the sea [11, 30], targeting an object with contrasting color and well defined features. Object detection could take advantage from previously acquired underwater images of the target object. A commercial stereo camera has been used (Bumblebee), but apparently no stereo processing has been performed.

It is worth mentioning two recent endeavors regarding underwater 3D reconstruction, presented at OCEANS'15. Massot-Campos at al. [40] describe a sensor fusion method for 3D reconstruction exploiting stereo vision and structured light, performed with off-the-shelf imaging sensors. Results are promising and the point cloud density and completeness are increased with respect to purely stereo vision approaches. However, tests have been performed in lab using a simulated environment and no details about the processing system are given. It is reasonable to assume that standard computers have been used, still leaving open the question about the fully underwater feasibility of the proposed approach. Bonin-Font at al. [41] present a stereo SLAM approach for robust and dense 3D reconstruction. Results are also based on test sessions performed in sea waters and authors state that images are grabbed at 10 fps and algorithms "run online and simultaneously at the same rate". It is unclear, however, whether processing has been performed onboard or exploiting an auxiliary processing unit in surface whose details, in any case, are not described.

1.4 Embedded systems for underwater vision

On the basis of the state of the art, it is evident that an underwater computer vision system could be effective in real-world context only if it is able to cope with the variability of underwater environment conditions, exploiting multiple algorithmic approaches. So far, however, the processing units deployed with underwater autonomous vehicles have fairly limited computational power, enough only to perform simple and predefined perceptual tasks.

Due to their very small form factor (90 × 96 mm), their modularity, and the definition as a standard in the early nineties, PC/104 boards are widely used in underwater applications. The first documented AUV exploiting computational units based on PC/104 boards is REMUS [42], a semi autonomous vehicle developed in 1997 for scientific and military missions such as coastal ocean surveys, pollution identification and source tracking (fig. 1.4.a). The computational unit exploits an Intel 486SX CPU which was released to the market seven years earlier, in 1990. The authors also provide an interesting analysis about components power consumptions and how they impact on batteries capacity and thus on reachable mission time. The performance gap between components exploited in underwater ECUs and mainstream CPUs is even more evident looking at SAUV vehicles [43]. These solar-powered autonomous underwater vehicles (fig. 1.4.b) have been deployed in 2006 and the main SAUV computer is a PC-104 stack with a 100 MHz x86 CPU, likely a Pentium or Pentium-like processor. In the same period, the first Intel Core 2 Duo CPU was released to the market, with a base operating frequency of 1.2 GHz. SAUV vehicles have been designed for very long term missions so they clearly take advantages from systems with low power consumption. On the other hand, an increased onboard computational power accompanied with solar cells for battery recharge would have opened further possibilities in terms of advanced sensors and autonomous operations for these systems.



(a) (b)

Figure 1.4: Illustrations of REMUS vehicle (a) and solar-powered autonomous vehicles SAUV (b).

As a matter of comparison between the CPU used in the SAUV underwater system and the one released to the market in the same period, some benchmarks can be

analyzed. The website cpu-world.com [44] provides several benchmarking results for both CPUs which are summarized in figure 1.5. The scale of "score" axis is logarithmic in order to better appreciate differences between tested CPUs. The modern CPU performs better than the older one by about two orders of magnitude in most of the tests.

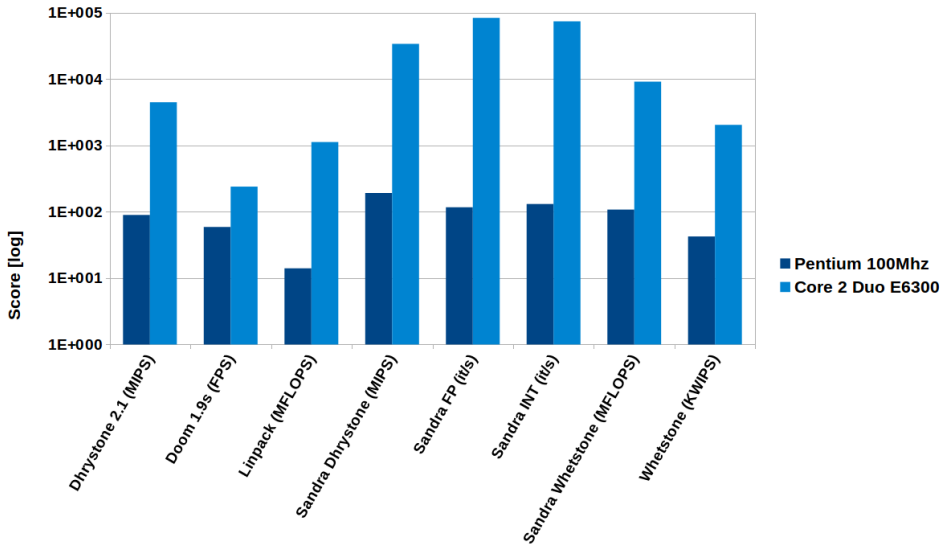


Figure 1.5: Available benchmark results for 100Mhz Pentium and 1.2GHz Core 2 Duo E6300. Source: cpu-world.com [44].

Recently, Albiez et al. [45] have proposed a novel sensor head for autonomous inspections of underwater structures [46]. The system exploits a high resolution camera (AVT Prosilica GE1900C) and a structured light source based on a class IIIb laser emitter. The computational unit relies on an Intel Core 2 Duo CPU, confirming the trend of using processors of past generations.

In last years, with the broad diffusion of mobile devices, marine and underwater hardware architectures take advantage from the availability of low power consumption CPUs, like the Intel ATOM microprocessor family. Several papers published in the last OCEANS'15 conference show that PC/104 systems are still a "de facto"

standard for underwater computational units and they now exploit mobile, ultra-low-power CPUs [47–50]. While Novi et al. [47] and Marouchos et al. [50] use x86 ATOM CPUs for their vehicles, the development of simpler applications like autonomous buoy systems, as presented by Nishida et al. [49], could exploit ARM family microprocessors. These CPUs, however, still offer limited computational power with respect to other available CPUs for desktop or notebook systems.

This thesis aims at overcoming the bottleneck in underwater perception for autonomous manipulation determined by dedicated processing systems with limited performance. The thesis thus investigates the trade-off between computational power and power consumption of a computer vision systems for underwater object detection tasks. The goal is pursued by developing a high performance hardware architecture whose deployment in an underwater sealed canister is feasible. This is not a trivial issue because safe working conditions, including operation within thermal limits, must be ensured to the system when it works both in air and in water. The vision system, indeed, is designed to be integrated into an autonomous underwater vehicle which is typically maneuvered in and out water and the various robot subsystems must run safely across all operations.

2

Design of Underwater Vision Systems

Computer vision has been considered a promising sensing modality for several tasks since early research studies regarding underwater autonomous applications. In 1994, Santos-Victor et al. [51] pointed out that "*in the context of autonomous underwater vehicles, tasks, such as object avoidance or recognition, grasping, docking, seabed reconstruction, underwater surveillance, inspection, cable maintenance, are among the set of those where computer vision may have an important role*". Although many interesting algorithmic approaches have been proposed to address issues rising in underwater environments, thorough discussion at the system-level is often omitted. However, the deployment of computer vision systems in real underwater environments requires that system-related issues are investigated and addressed.

For a deeper, better understanding of specific issues characterizing underwater perception for intervention missions, the design of the underwater vision system pursued in this thesis has followed a multi-step approach. In the early stage of investigation, an embedded prototype has been designed and developed. This system has led to a rapid deployment in water and its outcomes have been exploited in the design of the final high-end system.

This chapter will go through the description of the embedded prototype, of the final vision system, and of the tests carried out to assess the performance of both systems.

2.1 Embedded multi stereo vision system

The design of the embedded system has taken into account the constraints and requirements of the underwater application in terms of computational capacity, power consumption and thermal dissipation, waterproofing, and implementation time. The system has been conceived as a low-cost prototype to be assembled in a short time and to be eventually adapted during the development. Low cost vision systems have been successfully exploited in several out-of-water robotic applications [52–56] so that a similar approach could be adopted in underwater tasks, especially given the prototype nature of the system.

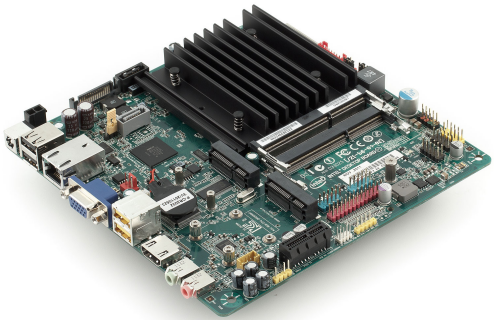
The main motivation for the development of a prototype system was the need of a first working vision system, ready to be deployed in short-time, to investigate the main issues of underwater vision and to collect an initial dataset to evaluate object detection algorithms. The prototype has been designed as an autonomous perceptual system, able to be deployed in water and record image sequences of underwater scenarios, even without any remote control. Logistics of marine and underwater experimental sessions is, indeed, difficult due to the working environment and the equipment requirements (waterproof instruments, supporting vessels for offshore applications, etc.). Thus, the development of a "system-in-a-box" was one of the requirement guidelines that led to the architecture design and the component selection. In order to speed-up the prototype development, only off-the-shelf components have been exploited.

For these reasons, a general purpose plastic box has been chosen as the container. This canister (260x330x92mm) has a flat transparent plate and a certified protection rating IP-68, achieved through a silicon O-Ring gasket placed on the box closure. Due to plastic box characteristics, water proofing reaches only few meters depth. The prototype has been thought for uses right below the water surface, so the plastic box characteristics met the requirements. Furthermore, the box small size is enough to contain a whole system, while keeping easy its maneuverability in water.

The embedded prototype comprises an Electronic Computational Unit (ECU), the imaging sensors, a low level layer for self monitoring and remote control, and a

power supply unit for autonomous operations. The imaging sensors are driven by the ECU, which is responsible for frame grabbing and storing. Since the whole system is sealed inside a plastic box, a remote control device has been deployed for common operations like powering on and shutting down. Furthermore, the hostile environment and the testing aim of the system suggest a wide range monitoring of temperature and humidity inside the canister, tasks that are assigned to a dedicated microcontroller.

The key design problem for such an enclosed processing system is the achievement of a trade-off between computational requirements, electrical power consumption and heat dissipation through the plastic enclosure. High resolution and high frequency frame acquisition requires adequate computational capabilities and compatibility of the platform with common software frameworks, libraries and hardware drivers. On the other hand, the CPU power consumption and *thermal design power* (TDP) should be as low as possible, since the battery storage is limited and, above all, the cooling down inside a waterproof sealed canister is performed through conduction.



Processor Number	N2800
L2 Cache	1 MB
Instruction Set	64-bit
Lithography	32 nm
# of Cores	2
# of Threads	4
Processor Base Frequency	1.86 GHz
TDP	6.5 W

(a) (b)

Figure 2.1: Mini-ITX Intel Desktop Board DN2800MT (a) and technical details of Atom N2800 CPU from Intel website (b).

The embedded system mounts a Mini-ITX Intel Desktop Board DN2800MT (fig. 2.1.a) with an Intel Atom processor N2800 (TDP 6.5 W), which is a trade-off between the power-efficient ARM architecture processors (TDP 5.0 W for ARM Cortex

A9) and the powerful commodity processors in the x86 architecture (low consumption embedded Intel Core i7-3517UE has TDP 17 W). The Atom N2800 is a 64bit, 1.86 GHz CPU in the x86 architecture with one of the lower TDP among available CPUs in the same architecture. Figure 2.1.b summarizes the technical details of this processor. The system also features a 60 Gb Kingston SSD hard drive and 2 GB RAM.

The imaging sensors consist of consumer webcams. In details, the vision sensors are three Logitech C270 webcams, capable to record images in high definition (1280x960) at $7.5 fps$. Once unmounted from the plastic enclosure, these low-cost cameras consist in a small electronic board with a fixed lens. Due to the cost and size constraints and the easiness of integration, webcams have been preferred with respect to professional devices. The system consists of 3 cameras to test three different configurations of stereo camera pairs (fig. 2.2), each with a different baseline and range coverage.

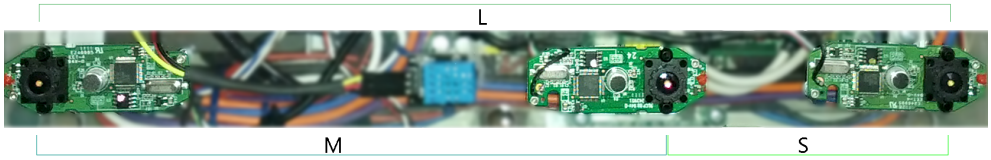
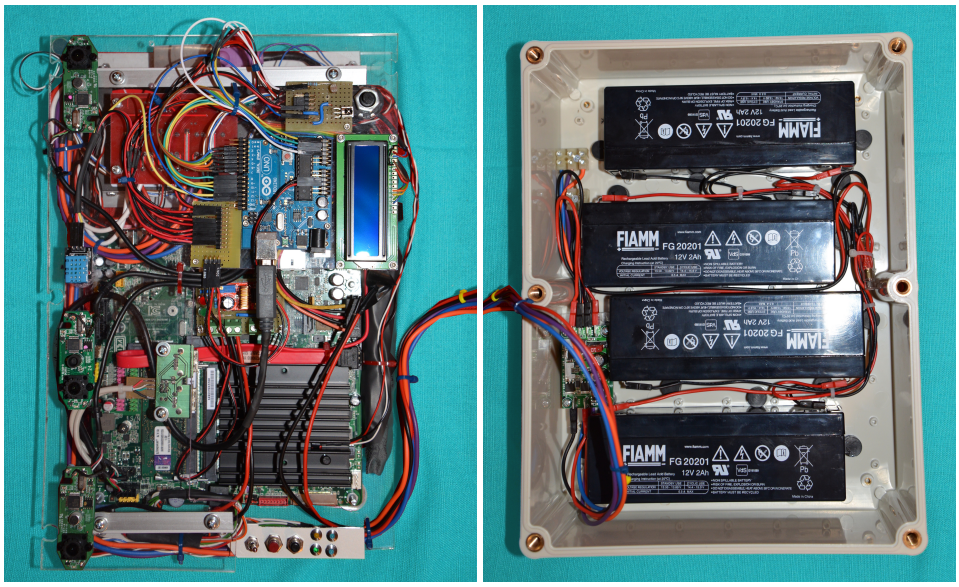


Figure 2.2: Multi-baseline configuration of imaging sensors (Logitech C270 webcams).

Logitech C270 webcams have been chosen also due to their successful exploitation in other projects [55, 57]. Each webcam is detected by the operating system (Linux Ubuntu 12.04) with a unique device identifier which can be customized using appropriate UDEV rules. This is not a trivial issue because in the application context, with calibration data associated to each specific camera, it is mandatory that webcams are not swapped. Webcams from other producers have proven unreliable in past activities for this kind of applications.

The internal structure of the canister has been organized in three vertical layers. The bottom layer contains 4 lead acid batteries ($12V$, $2Ah$ each, $8Ah$ whole battery pack) and a DC UPS board (10A) which is responsible for charging batteries, when

the external power cable is plugged, and switching back to batteries if the external power is lost. The middle layer contains the processing hardware described above and a $12V$ to $5V$ DC-DC step-down unit to supply the sensors. The top layer contains the cameras, mounted as close as possible to the canister glass, and an Arduino UNO board to control the system. Middle and top layers are removable from the canister for maintenance purposes, as shown in figure 2.3 that illustrates the realized system.



(a) (b)

Figure 2.3: Embedded prototype canister opened: ECU, cameras and monitoring electronics (a), batteries and UPS board (b).

The Arduino UNO microcontroller is responsible for low-level control of the system (internal temperature and humidity monitoring, ECU power on/off using a remote controller, LCD display to log information). Thanks to its ultra low power consumption (below 500mW), the Arduino board is always powered on. On the contrary, the main ECU must be powered only during the underwater experiments both to prevent system overheating and to avoid discharging the battery pack.

Temperature and humidity inside the canister are crucial environmental values to be monitored in order to avoid over heating and prevent severe breakages due to water seepages. In order to monitor the temperature and humidity these sensors have been integrated in the system: three analog temperature sensors (LM35) for CPU, hard drive and batteries, and a digital sensor for temperature and humidity (DHT11) located in the top layer. Both sensors provide fully calibrated outputs. The LM35 sensor maintains an accuracy of $\pm 0.8^oC$ over a range from 0^oC to 100^oC. The LM35 sensor draws $60\mu A$ and possesses a low self-heating capability. DHT11 can measure temperature from 0^oC to 50^oC with an accuracy of $\pm 2^oC$, and relative humidity ranging from 20% to 95% with an accuracy of $\pm 5\%$. The Arduino UNO board checks the temperature and humidity measurements and is in charge of the shutdown of the system, if the measurements exceed their respective safety thresholds. All values measured by the sensors are sent to the main ECU by means of a serial-over-USB connection and a specifically designed communication protocol. The ECU collects the received data for experimental evaluation purposes.

Feedbacks about the system status are returned by means of a monochrome, 16x2 characters LCD display which represents the system HMI. The user can browse multiple pages by acting on the infrared remote controller.

Finally, a 4-relay board is used for sending the ATX power command to the mainboard and for powering some external high-power LED illuminators. Figure 2.4 illustrates the embedded system components and their connections.

The operating system of the embedded prototype is Ubuntu Linux 12.04 LTS, server release. The system exploits the state-of-art robotic framework ROS [58] (Groovy release) for the image acquisition and data storing. Due to the particular camera configuration, leading to three different stereo couples, the image acquisition driver, available in ROS, has been slightly adapted in order to grab frames from the three sensors and publish the camera calibration data for the three feasible baseline configurations: large, medium and small. In this way it is possible to evaluate the stereo performance of the different configurations by simply remap the ROS topic names, without stopping the image acquisition.

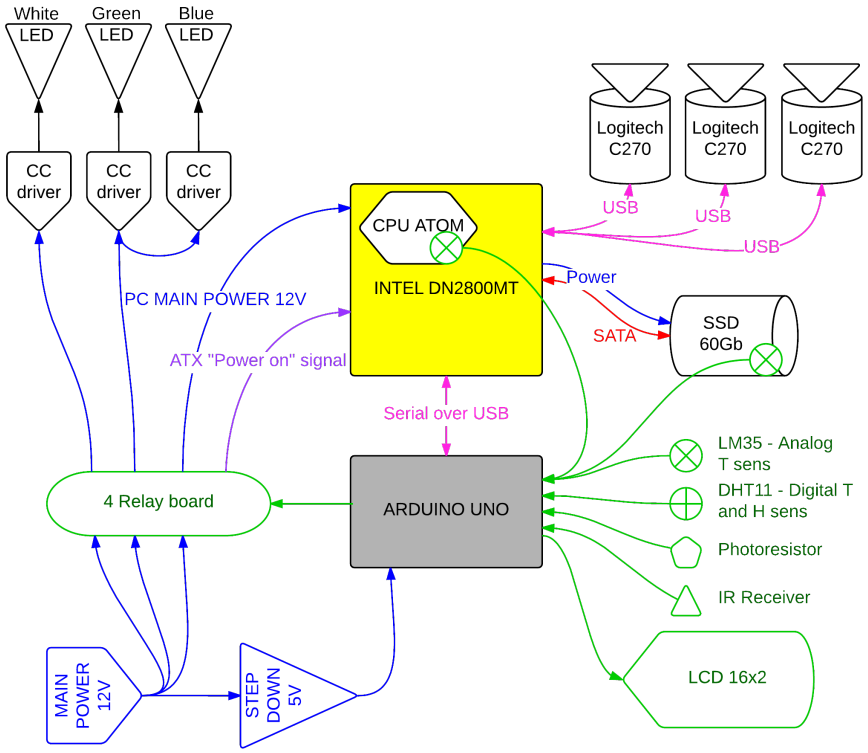


Figure 2.4: Embedded prototype: block diagram of internal components and connections.

The developed prototype has been tested in a laboratory setup before its exploitation in real-world experimental sessions. Canister water-proofing has been assessed by submerging the system in a barrel at roughly 1 m depth for 24 hours. Logged humidity values were approximately constant over the whole experiment validating the watertight seal. The approximate power consumption for the whole system in stressed conditions (CPU loaded, webcams running, SSD reading and writing) is about 17 W. Temperature monitoring and safe self-shutdown feature provided by the low level controller have been successfully verified by simply lowering the warning temperature threshold. The control system has safely arrested the ECU.

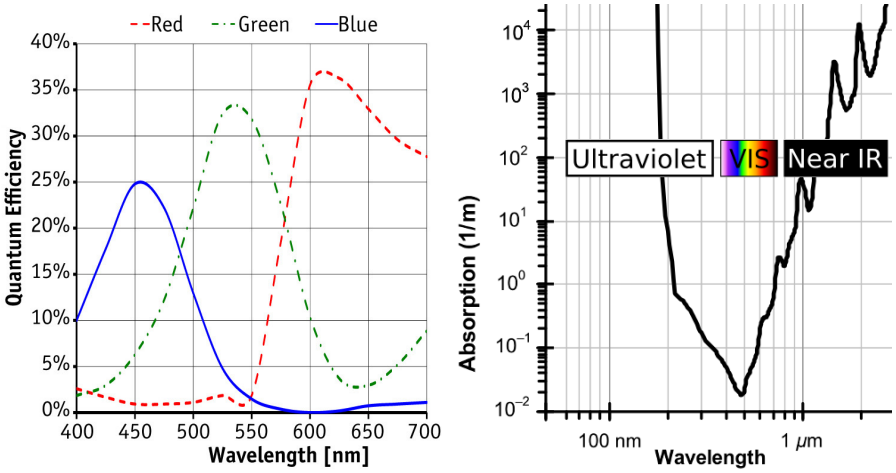
Real-world evaluation of the system (described in chapter 4) and early algorithmic approaches to the problem of pipe detection in underwater contexts provided valuable insights for the design of the final vision system. Lessons learned comprise the variability of the underwater environment in terms of visibility, light reflections and colors aberrations, and the required flexibility of the vision system to cope with these problems. In particular, the experiments performed with the embedded prototype highlighted the importance of camera synchronization and optics configuration. Furthermore, the aforementioned flexibility must be supported by a powerful computational unit in order to allow deployment of complex processing pipelines optimized for the specific underwater context. The next section shows how these issues have been faced in the design of the final vision system.

2.2 High end vision system for deep water

The high end vision system has been conceived as a versatile hardware platform for underwater computationally-demanding applications. Lessons learned in the prototype development and testing stages have been exploited for the requirement definition of the final vision system. In particular, regarding the imaging subsystem, synchronism is mandatory as well as the possibility to intervene on the optical configuration in order to adapt the cameras to the specific working context. Experimental trials of underwater object detection, performed with the embedded prototype, have also shown the importance of color as a distinctive feature for submerged targets.

The imaging subsystem has been designed to fulfill requirements of advanced underwater perception tasks. Typical requirements like precision and fast data update rate lead to high resolution images and high frame-per-second grabbing capabilities. These considerations, together with size constraints and reliability requirements, led to the choice of industrial cameras. The imaging subsystem is based on two AVT Mako G125C GigE cameras, an ultra compact device whose dimensions are 60.5x29x29mm. The connection bus is a standard ethernet link with support to PoE (Power over Ethernet). Cameras mount a Sony ICX445, 1/3”, high resolution color sensor, capable of acquiring frames at 30 fps in full resolution of 1292x964 pixels.

The sensor spectral response is shown in figure 2.5.a suggesting a good camera sensitivity to red wavelengths. This is an interesting feature because water acts as a filter and absorbs, in particular, wavelengths in the red spectrum (fig. 2.5.b).



(a) (b)

Figure 2.5: Spectral response of the Sony ICX445 image sensor (a) and absorption spectrum of liquid water centered in visible light wavelengths (b).

The chosen cameras support standard C/CS-Mount optics. For a higher grade of flexibility, two varifocal lenses have been chosen. Kowa LMVZ4411 lenses have a focal length range between 4.4 mm and 11.0 mm, a manual focus and iris controls and a minimum focusing range of $0.3m$. Iris maximum aperture of $f/1.60$ suggests a very luminous optic. Lenses fit requirements of mega pixel cameras, up to 1/1.8" sensor size. By using these lenses with smaller imaging sensors, like the ones in Mako cameras, a reduced Field of View (FoV) is compensated by less distorted images.

AVT Mako firmware exposes approximately the same features of AVT flagship series "Prosilica", which is widely used in robotic tasks [59, 60] as well as in few underwater applications [61–63]. Among available functionalities, the most interesting ones are the possibilities to manually tune the exposure time, white balance, and gain, to trigger the frame acquisition with a digital signal, and to perform an on-board color correction by acting on hue and saturation channels. Furthermore, if network

bandwidth limitations and computational power constraints require an image processing at a lower resolution with respect to the sensor native one, Mako cameras can perform the image binning directly onboard. Image binning, when performed at a sensor-level, allows charges from adjacent pixels to be combined. The effects are benefits in readout speeds and improved signal to noise ratios albeit at the expense of reduced spatial resolution.

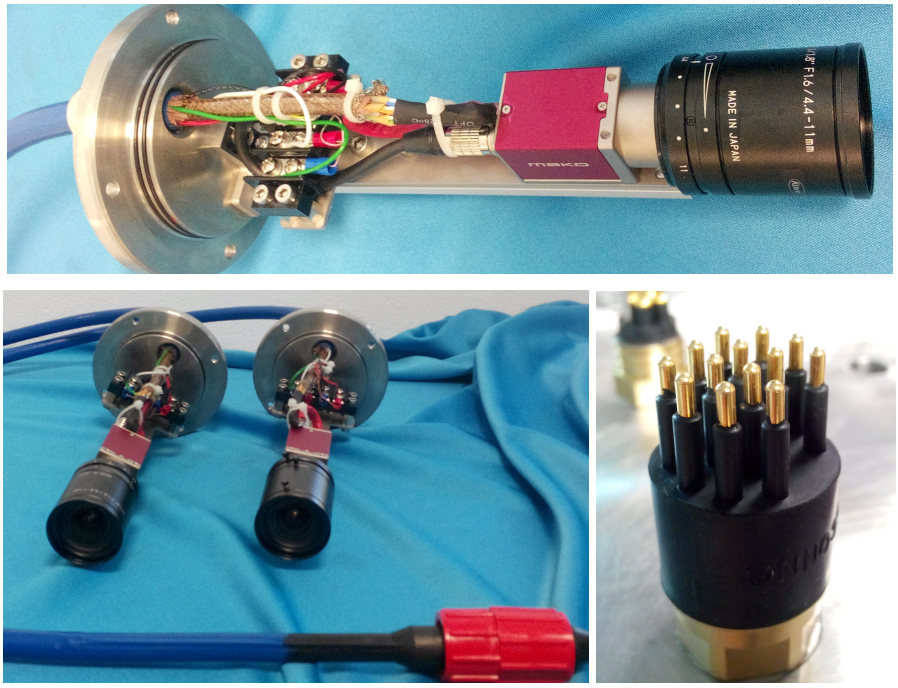


Figure 2.6: Imaging subsystem components and cabling. AVT Mako mounted on sliding bracket (top); cameras and connection cable (left) and 13-pin cable socket on the vision canister (right).

The cameras are housed in two separate small canisters made of black PVC, with a stainless steel back cover and a plexiglass transparent glass. Power supply, camera triggering and data cables are joined into a single 13pin submarine cable that enters the canister from the back steel cover (fig. 2.6). The cable plug gender is different for left and right cameras, for an error-free mounting. Each part of the

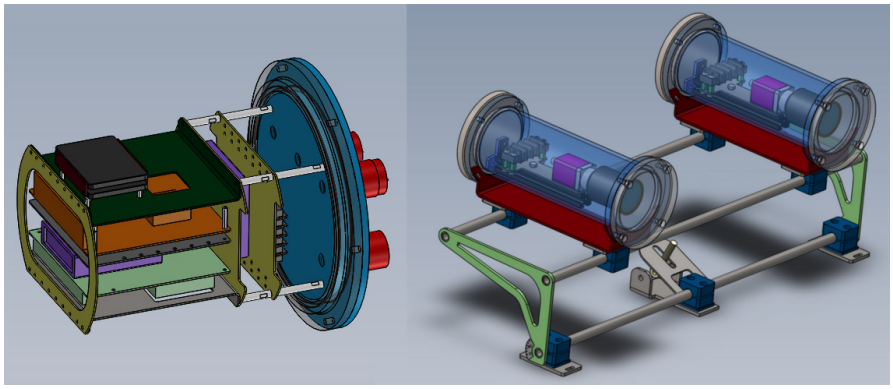


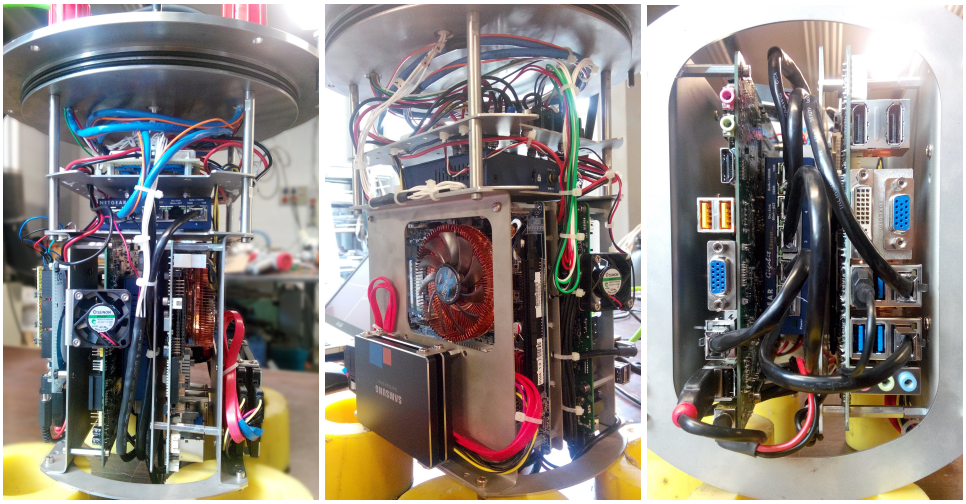
Figure 2.7: 3D CAD model of the vision system: vision canister (left) and stereo rig (right).

camera canisters has been worked out from solid blocks to avoid soldering or gluing, and redundant O-Rings have been used for junctions thus guaranteeing watertight protection up to 50 m depth. Inside the canisters, cameras are mounted on sliding brackets that allow a precise positioning of the lenses next to the canister glass, in order to avoid unwanted reflections or ghost-effects on images. Camera canisters are then mounted on a pitchable pipe slide that allows a fine positioning of the stereo rig and permits to change the camera baseline, depending on the expected operational distance for each carried application (Figures 2.7 and 2.12).

The underwater computational unit was designed balancing power consumption, thermal dissipation and system performance. The ECU (Electronic Control Unit) inside the vision canister was designed as a modular system (Figure 2.8), including two x86 CPUs, one ARM-based board and a microcontroller. The microcontroller (Arduino MEGA 2560) is responsible for camera triggering and temperature monitoring inside the canister. In contrast to the embedded prototype, the vision system does not need to be self powered because it is conceived to be powered by the hosting AUV. Due to this reason, the system starts automatically when powered, removing the need of a remote controller.

The ARM-based board consists of a Raspberry-Pi, chosen because of its small size, the onboard H.264 video encoder, and the minimal impact on heat generation and power consumption. Raspberry-Pi features a BCM2835, ARMv6 700MHz CPU,

512Mb RAM and is equipped with a 16 Gb Class 10 SD-CARD running Raspbian Linux operating system. Performance of this board is not enough for high-resolution and high-frequency computer vision tasks. However, the embedded H.264 hardware encoder can easily encode a video stream with minimal load on the CPU. Thus, the Raspberry-Pi board is encharged of remote streaming an encoded video for monitoring purposes.



(a) (b) (c)

Figure 2.8: Components of the vision system: side view with additional cooling fan (a), Intel Core i7 copper heat sink and Solid State Drives (b), and bottom view of motherboard connections (c).

The main system relies on an Intel Core i7-4770TE @3.33GHz with 4 physical cores and Hyper-Threading technology, mounted on an industrial Mini-ITX board. The mainboard (BCM MX87QD) features two independent network controllers which are exploited for splitting the system network from the camera network. This PC is equipped with 8Gb of 1600MHz DDR3 RAM and two 120Gb Samsung 840 Pro series SSDs. The CPU heat sink is a low profile Zalman, pure copper unit (CNPS2X) with heatpipe technology, capable of cooling up to 120W (fig. 2.8-b). The heatpipe is placed in direct contact with the CPU dye and ensures an efficient heat transfer from

the CPU to the heat sink fins.

The i7 processor mounted on the main ECU belongs to the 4th generation of Intel Core CPUs, which was the last Intel architecture released on market at the time of system development. The CPU details are summarized in figure 2.9.

Launch Date	Q2'13
Processor Number	i7-4770TE
Intel® Smart Cache	8 MB
DMI	5 GT/s
Instruction Set	64-bit
# of Cores	4
# of Threads	8
Processor Base Frequency	2.3 GHz
Max Turbo Frequency	3.3 GHz
TDP	45 W
Intel® Hyper-Threading Technology	Yes

Figure 2.9: Technical details of Core i7-4770TE CPU.

The second PC is based on an Intel DN2800MT Mini-ITX mainboard with an ATOM 1.86GHz processor, whose characteristics have been already described in section 2.1.

Network connection is managed by two independent gigabit networks using Netgear GS-105 switches. Due to the high traffic rate generated by the vision system, the two cameras and the i7 ECU are connected by a dedicated network to avoid traffic jams. The second ethernet interface is used to enable communication between the vision system and the AUV.

The electronic components are mounted on an aluminium trellis, expressly designed in 3D taking care of all components and connectors sizes (fig. 2.7), feasible mounting and easy maintenance constraints, and heat dissipation flows. The top layer hosts the power supply conversion and distribution boards. Right beneath them, the camera network switch is horizontally fixed. Then, three plates hold all the other components mounted vertically. Cable connections of the two x86 ECUs are on the bottom for easy access and maintenance (fig. 2.8.c). Two additional small fans are

placed on sides of the trellis (see figure 2.8.a and b), slightly oriented and blowing toward the canister metal inside. The resulting air circulation facilitates the heat exchange with the canister and then with the outside medium. The aluminium trellis can be unmounted from the steel cover by simply unscrewing in 4 points.

Each part of the camera canisters and the cover of the main canister have been worked out to avoid soldering or gluing and redundant O-rings have been used at each junction thus guaranteeing watertight protection up to 50 m depth. The main canister is made of a portion of a stainless steel pipe and the bottom closure is soldered. The internal diameter is 244mm and whole available volume for internal components is approximately 16dm^3. The external diameter ranges from 300mm of the closure to 264mm of the canister outside surface. Additional external fins have been placed to increase the canister surface and therefore the heat dissipation. Figure 2.10 illustrates the mechanical design of the canister.

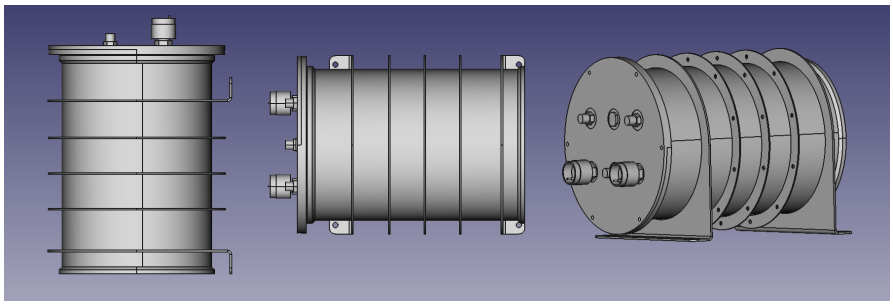


Figure 2.10: Mechanical design of the canister: orthogonal and perspective views.

Connection between the canister and external devices like cameras, power supply units and other networks are possible due to appropriate underwater connectors. Five connectors are available on the canister enclosure: power supply (24V, 5A), external ethernet network, camera network and USB external device. The connection to the camera exploits a marine cable and dedicated connectors which carry a certified gigabit network, power supply and additional digital signals (triggers). With this solution, connections to the camera have been condensed in one cable instead of two.

Connection scheme of the vision system components is shown in figure 2.11 and the finished system is displayed in figure 2.12.

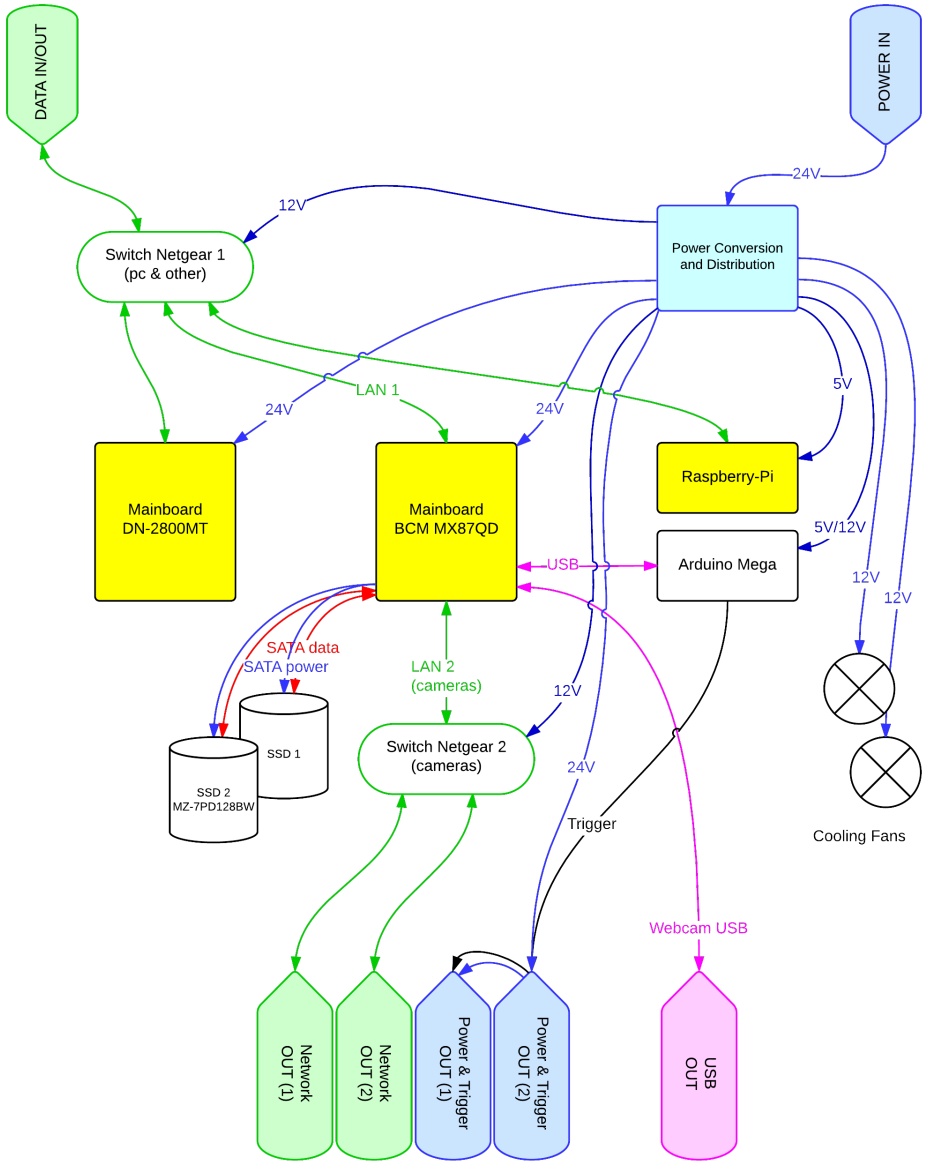


Figure 2.11: High end vision system: block diagram of internal components and connections.

Figure 2.12: Underwater vision system: the vision canister (left) and the stereo rig (right) with the two camera canisters.

Both x86 architecture boards (Atom and i7) run an Ubuntu Linux 12.04 (Long Term Support) operating system. A "server release" has been preferred because the desktop environment is not needed. The systems rely on the ROS robotic framework for camera image acquisition and message sharing. The ROS infrastructure manager (`roscore`) can run arbitrarily on either ECU as well as on an external system. The Mako cameras driver has been developed from the released AVT Software Development Kit for Prosilica series and slightly adapted taking care of differences between the two camera series. Furthermore, the developed driver handles both cameras: frames are grabbed from left and right sensors with a multi-thread routine, and configuration parameters, which are dynamically adjustable, are sent to both cameras at the same time.

Due to the hardware triggering, frames are always shot at the same time. Nevertheless, delays in network transmission and in data acquisition produce slightly unsynchronized image timestamps at driver-level. The developed software reconstructs the correct frame pairing and sets grabbed images with a unique timestamp. Images are then published as a `sensor_msgs\image` standard ROS topic. Cameras also expose a programmable memory area which is exploited for storing the calibration data. The driver handles both the writing and reading of calibration data, which are then published as a `sensor_msgs\CameraInfo` topic. The developed driver complies with ROS camera driver guidelines, thus enabling the use of the image pipeline stack.

2.2.1 Hardware performance evaluation

To evaluate system performance, experiments have been conducted both in a laboratory setup and in a real underwater environment. In particular, once the assembly stage of the hardware was completed water resistance experiments were performed for the vision and the camera canisters. In the laboratory setup canisters have been inflated with air at 6 atm pressure to simulate underwater conditions at 50 m depth. Then, the canisters have been placed for 24 hours inside a water container to identify possible leaks. A pressure gauge was used to keep pressure monitored. Underwater tests have been conducted with the canisters sinked at 50 m depth in the see in front of La Spezia (Italy). No water leakages occurred inside the canisters.

Long term experiments have been conducted in the laboratory setup to evaluate power consumption, heat dissipation, working temperature and network performance. Power consumption and heat dissipation have been tested by stressing the CPUs at a high load using the Linux `stress` command. Results are reported in Table 2.1 and show that the power consumption is well below 100 W, thus achieving a good performance for autonomous robot applications.

The maximum temperature registered in the vision canister in water with all the CPUs stressed was 60^oC, far below a dangerous value. In the laboratory setup, with the vision canister in air and external temperature at 25^oC, temperature inside the canister after 15 minutes of CPU stressing reached 72^oC. Since the internal temper-

Test	Power (W)
Idle	43.2
CPU stress - 1 core - i7	65.0
CPU stress - 2 cores - i7	78.0
CPU stress - 4 cores - i7	79.2 (peak 86.4)
CPU stress - 4 cores/8 threads - i7	79.2 (peak 86.4)
CPU stress - 2 cores/4 threads - Atom	44.9
Stereo Vision frames acquisition	46.6
CPU stress on all cores/CPUs	**81.6 (peak 91.9)**

Table 2.1: Power consumption tests at different CPU loading levels

ature was still increasing, the test suggested that the system cannot support heavy computational loads in air with external mild temperature for long time. It should be remarked that with no stress of the CPUs the temperature remained stable at a lower, safe value. Hence, the vision canister can also be operated in air at standard CPU load level for a long time.

Bandwidth throughput of both networks has been tested exploiting `iperf` [64, 65], a state of the art tool for network benchmarking. Experiments have been conducted using both TCP and UDP protocols, half and full duplex communication and different packet size. Despite cable soldering and usage of marine connectors, data transfer rate was about 850 Mbps, full duplex, for both system-to-cameras communication and system-to-vehicle communication. Results were computed from about 1TB of transferred data.

The performance of the vision system to reconstruct a disparity map was tested using the standard Absolute Differences (SAD) correlation method. A throughput of 12.5 frames per second (1292x964 resolution) was achieved.

The developed system represents and advancement in the state of art regarding underwater computer vision platforms for autonomous robots. The system modularity allows an efficient allocation of processing tasks and a grade of failure awareness. The CPU exploited in the main ECU was from the latest released series, ensuring the best available performance within the power consumption and heat dissipation constraints. Finally, the imaging system allows a fine tuning of lens configurations and camera parameters, thus enabling specific optimized setups for different perceptual applications and underwater environments.

3

Testing Computer Vision Algorithms on the Developed Underwater Systems

The design of the underwater vision system was based upon the requirements of intervention applications. These requirements are often expressed in terms of accurate target detection, precise measurements, and high frequency data processing. These tasks are also challenging in out-of-water robotics, even though, nowadays, the availability of precise sensors and powerful computational units have made these applications feasible.

As stated in the thesis introduction, the test bench of the developed underwater vision system is represented by the MARIS project which is aimed at performing a proof-of-concept autonomous intervention on common submerged artifacts, and specifically on pipes. Interest in this kind of underwater objects is confirmed by several published works about pipe inspection [66–68] and by a recent survey [69] on robotics applied to oil-and-gas industries. Validation of the proposed vision system therefore included the evaluation of algorithms supporting adequate image processing, object detection and pose estimation, combined in different processing pipelines.

Vision-based object detection may be addressed by different approaches according to the input data and the available a-priori knowledge of the target object. Potential approaches comprise image processing of mono-camera data or more complex shape matching algorithms based on stereo processing. The set of algorithms consid-

ered in this thesis spans several phases of the vision pipeline (fig. 3.1), whose flow depends on the available knowledge about the object to be detected.

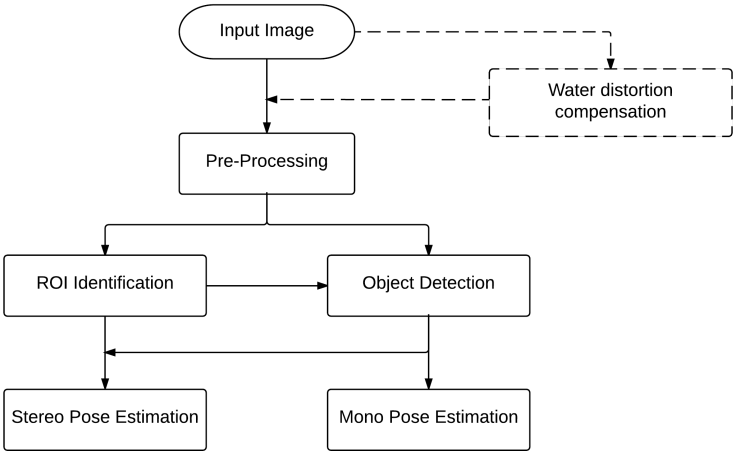


Figure 3.1: Schema of vision-based object detection and pose estimation algorithms.

An offline calibration stage aims at reducing the distortion of underwater images caused by the different light refraction behavior through water. The preprocessing step addresses issues related to image aberrations introduced by water which depend on lighting, turbidity, depth and other environmental factors. The initial steps aim at detecting salient regions representing candidate objects, possibly with no prior knowledge about the target. The output of such steps may be a broad candidate ROI (Region Of Interest) to narrow the searching area in the successive phases or the classification of the regions as target object or not (respectively the blocks *ROI Identification* and *Object Detection* in Figure 3.1).

The point cloud is used only to estimate the pose of an already detected object (*Stereo Pose Estimation* block in Figure 3.1). Alternatively, if the geometric model and the dimension of the object are known, the pose can be estimated from the shape of object projection in a single frame.

In the following sections, the main algorithms investigated by the MARIS research team are summarized with their achieved results.

3.1 Water distortion compensation

Camera calibration is typically performed with state of art methods based on a-priori known patterns like checkerboards [70]. However, due to difference between light refraction in air and in water, "*any trial to localize or reconstruct an object observed by an underwater camera has to go through a calibration phase*" as pointed out by Lavest et. al [71]. Recently, it has been observed that underwater cameras are correctly modeled as "axial camera" [72]. In practice, the classic pin-hole model is used, after calibrating the intrinsic parameters in water medium. However, due to the difficulty of in-situ underwater camera calibration, a method for image distortion compensation has been developed and tested with the underwater vision system [73].

This calibration approach assumes that cameras have been already calibrated in air and relies on the possibility of observing underwater cylindrical objects like pipes or other known regular shape objects. The automatic method performs a radial distortion correction based on rectification of reconstructed surfaces exploiting a set of stereo images to improve robustness

The considered lens distortion model is called *plumb bob* [25] and describes the correction applied to the image by means of a set of scalar values. Since the main effects of radial distortion are determined by the scalar values K_1 and K_2, the aim of the proposed approach is to achieve an automatic correction of these two parameters. The method exploits a brute-force approach: several combinations of K_1 and K_2, sampled in range of values

$$\Lambda = [K_1^{air},\ K_1^{air} + \Delta K_1] \times [K_2^{air},\ K_2^{air} + \Delta K_2]$$

(where K_1^{air} and K_2^{air} are the initial values obtained after in-air calibration) are applied to a set of N stereo image pairs. The method aims at finding the best value combination that minimizes images distortion over the set of N image pairs.

For each sampled pair $[K_{1i}, K_{2j}]$ the disparity map is computed and a 3D point cloud is obtained. The point cloud is then masked in order to filter out the background and maintain only the cylindrical object. Each pipe point cloud, obtained by a differently distorted image pair, contains a differently distorted object. The object is then approximated with a plane and distances between the object 3D points and the plane are accumulated. The greater the sum, the greater the distortion of the pipe. The accumulated value, associated with the pair $[K_{1i}, K_{2j}]$, represents the image distortion, so that finding its minimum over the domain Λ corresponds to finding the best distortion correction parameters. Figure 3.2 shows an example of the sampled distortion function.

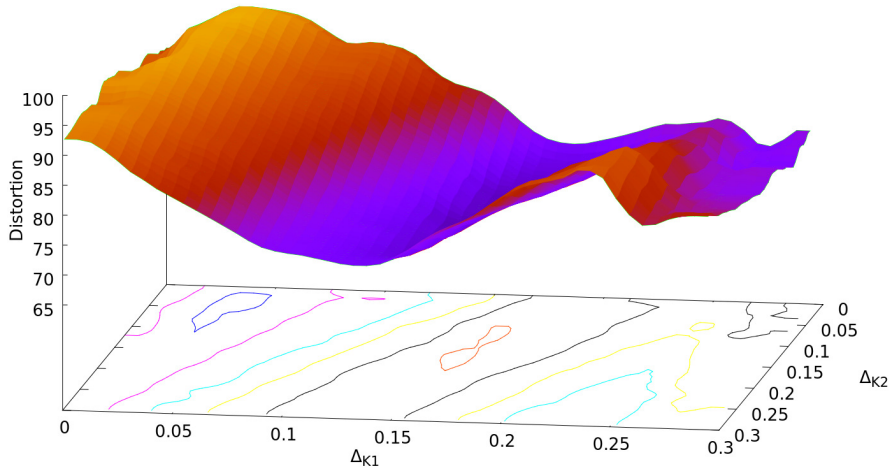


Figure 3.2: Example of a sampled distortion function.

For reliability, the method is performed on a set of N image pairs. Experimental tests on sea waters images (see section 4.2) have shown that 5 stereo image pairs are enough to achieve a good estimation of the distortion parameters. The computational time is about 28 minutes on the vision system Intel Core i7 ECU for each image pair, with Λ sampled in 2500 value pairs (which leads to 2500 disparity maps to be computed). Figure 3.3 shows results of the calibration procedure. Additional details and results are presented in [73].

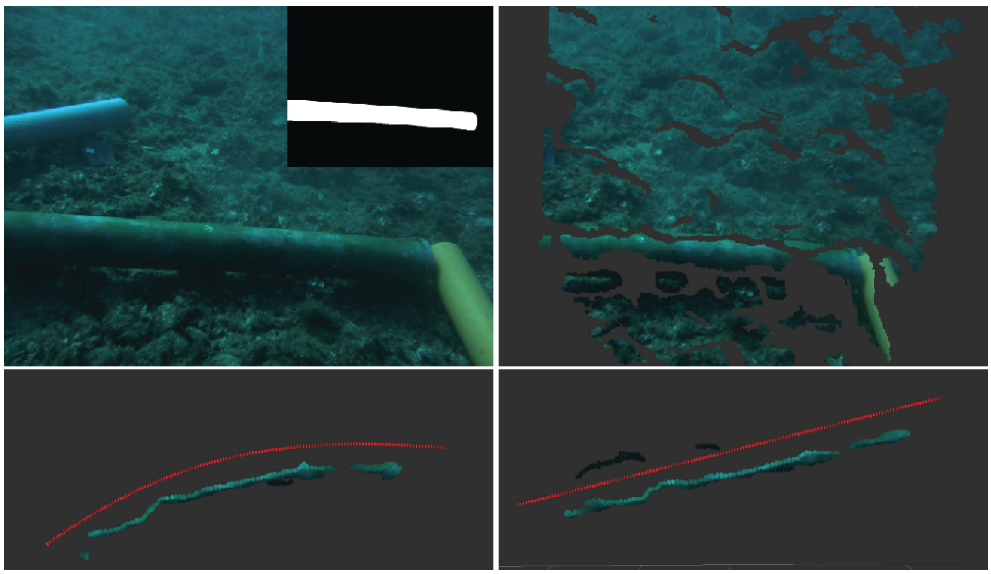


Figure 3.3: Example of the automatic calibration approach. Image and segmented pipe (top left). 3D point cloud from disparity map (top right). Distorted pipe using in-air calibration parameters (bottom left). Pipe rectification using optimized camera parameters (bottom right).

3.2 Image pre-processing

The image distortion is only one of the side effects introduced by water. Light attenuation and back-scattering produce blurred images with limited contrast. Furthermore, the water peculiar absorption spectrum of visible light modifies colors of underwater images with a prevalent green-blue component. Improving image quality by attenuating the water filtering effects is the first step to be performed in an image processing pipeline, so that two methods with effects on contrast and colors have been evaluated.

The first method aims at reducing image blurring. A *contrast mask* method based on component L of CIELAB color space [74] is applied to the input image. A Contrast-Limited Adaptive Histogram Equalization (CLAHE) [75] is then performed in order to re-distribute luminance. The combined application of contrast mask and CLAHE lessens light attenuation and reduces the effect of light artifacts on the objects, as shown in Figure 3.4 and extensively described in [76].

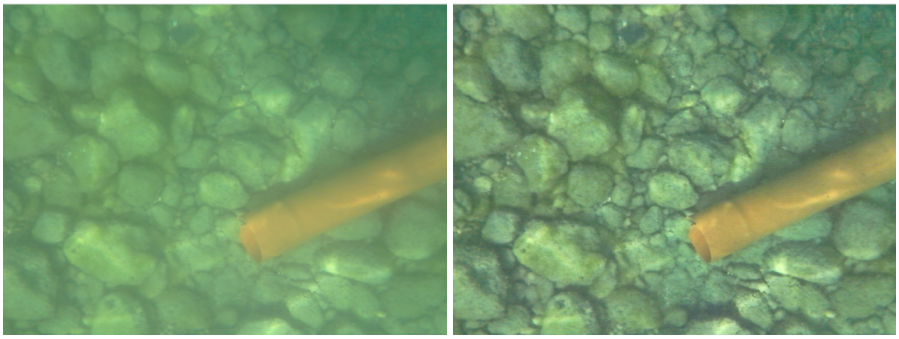


Figure 3.4: An underwater image before (left) and after (right) the application of contrast mask and CLAHE.

In deep water, the water filtering effect is more evident on color components which are shifted toward blue tonalities. In order to facilitate detection methods that exploit color as an object distinctive feature, it would be convenient to restore the color shades to their original color in air. Several approaches have been proposed for color restoration [77]. Performed experiments showed that a color constancy method based on *grey-world hypothesis* [78], which assumes the average edge difference in the scene to be achromatic, sufficed. The results are illustrated by the example in Figure 3.5.

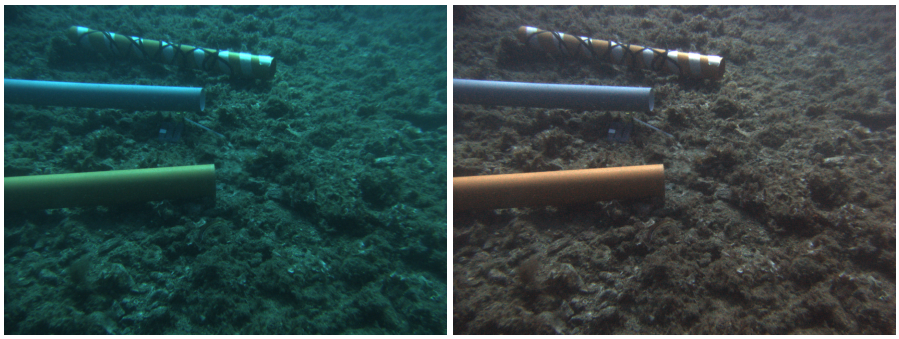


Figure 3.5: An example of color restoration: originally acquired image (left) and image after color restoration according to grey-world hypothesis (right).

The presented preprocessing methods can be applied stand-alone to raw images or in a combined configuration, depending on water conditions and application context. Both methods are not computationally demanding and have been successfully performed on both Core-i7 and Atom ECUs. This result suggests that image preprocessing can be assigned to an auxiliary embedded ECU with the drawback of an increased network load due to the image exchange.

3.3 ROI Identification

Region of Interest identification algorithms aim at selecting an area in images that roughly includes or corresponds to the target object. This step facilitates further approaches to object detection and pose estimation by filtering out the scene background. The ROI may be searched according to different criteria based on specific features of the target object, like color and dimension constraints. Both these criteria have been exploited in evaluating methods for ROI detection, leading to results which are discussed in [76].

The first method, labeled as ROI_{area}, is based on the assumption that the unknown object never occupies more than a given portion of image pixels and has a uniform color. The H channel of HSV (Hue Saturation Value) color space is quantized in 16 levels and the input image is partitioned into subsets of (possibly not connected) pixels with the same hue value. A convex hull of pixel is performed to obtain connected areas and only regions whose area is less than a given percentage of the image are selected as part of the ROI_{area} (fig. 3.6).

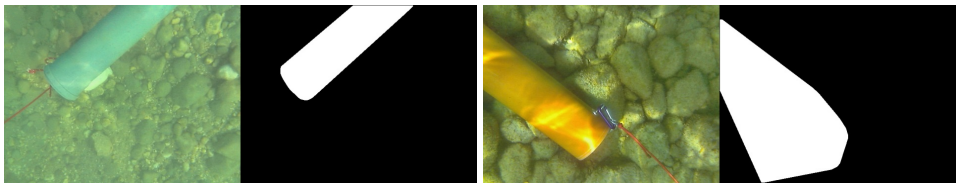


Figure 3.6: Mask generation based on ROI_{area} approach. On right image, an over estimation effect is evident.

Although color is used to partition the image in subsets, no information regarding the exact object color is given to the algorithm. When this distinctive feature is known, it can be exploited for a more reliable identification of the region of interest, as done in ROI_{color} method. Hence, the ROI_{color} is obtained composing the regions where color is close (up to a threshold) to the expected target color (fig. 3.7).

(a) (b) (c)

Figure 3.7: Mask generation based on ROI_{color} method: original image (a), hue channel after color reduction (b) and resulting masks (c).

The region computed either by ROI_{area} or ROI_{color} is made available for further processing. These ROI estimation techniques only exploit the relative color uniformity of a texture-less object, but they do not identify a specific object. Both ROI_{area} and ROI_{color} are lightweight methods and their impact on computational time is in the order of few milliseconds on the Core-i7-based vision system proposed in this thesis. Thus, ROI identification could also be assigned to an auxiliary ECU.

3.4 Object detection

With respect to ROI identification approaches, object detection methods rely on a thorough knowledge of the target object and its characteristics.

The proposed ROI_{shape} algorithm performs object detection in two steps: *image segmentation* and *contour shape validation*. The goal is the identification of a connected region with straight and sharp contours like typical human-made artificial objects. In contrast with the previously described coarse segmentation approaches, ROI_{shape} is able to detect whether the target object belongs to the image.

The *image segmentation* step classifies each pixel of the image according to its corresponding vector of following local features: color channels of HSV space, respectively hue h_i, saturation s_i and value v_i, and gradient response to a *Sobel* filter. Image is then clustered according to k-means algorithm [79]. Figure 3.8.a-b illustrates a typical output of k-mean clustering: the artificial objects are classified as belonging to the same cluster whereas the background is split into a lighted region and a dark region. The image is further refined by computing the connected components of the cluster (figure 3.8.c).

The *shape validation* step is applied to each cluster obtained from image segmentation and it is based on a contour extraction algorithm (figure 3.8.d). The contour of human-made regular-shape objects, in particular when their projection in the image plane is approximately a rectangle, often consists of parallel edges. Under this assumption, the target region is recognized by detecting parallel lines-segments from the contour, e.g. using the *Hough Transform* and angle histograms (figure 3.8.e). Finally, a cluster is classified as an object with regular shape if the histogram is "peaked", i.e. it is distributed along few principal directions.

Based on achieved results, fully reported in [76], ROI_{shape} has proven robust to color similarities between objects and background. The precision (above 90%) and recall (above 80%) outperform methods of ROI identification that do not exploit a thorough knowledge of the target object.

As a drawback, the processing of ROI_{shape} requires about 160 ms for full-resolution (1292x964) images, on the Intel i7-4770TE @3.33GHz CPU, thus leading to a low frame rate of about $6 fps$. This approach should therefore be used only when more naive methods are not adequate for the specific application context, and is likely applied to lower resolution images.

3.5 Object pose estimation

Object pose estimation can be performed exploiting either single images or stereo pairs. Single-image processing must rely on an a-priori knowledge of object shape and size in order to estimate its 3D position. On the contrary stereo-image process-

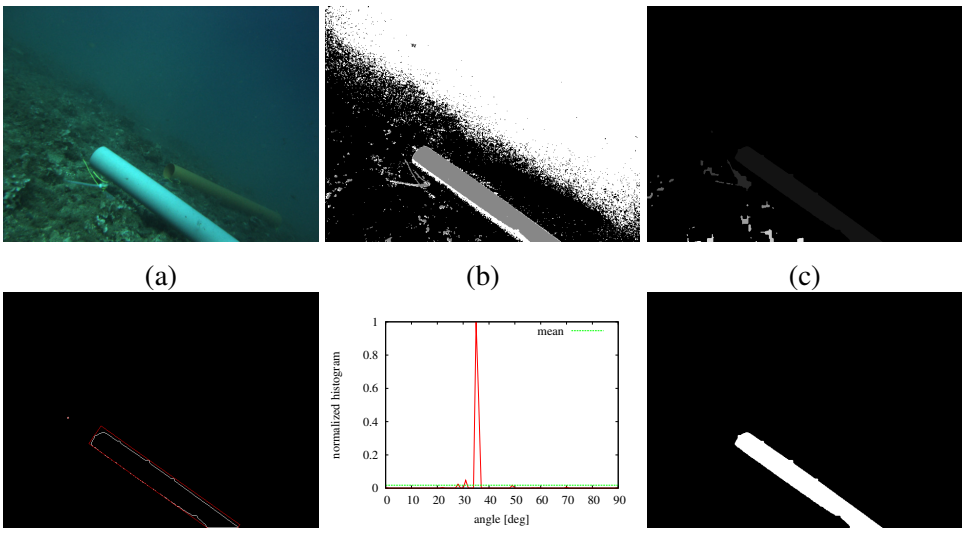


Figure 3.8: Steps of ROI_{shape} algorithm: (a) the input image; (b) the clusters obtained from k-means; (c) the connected components extracted from one of the clusters; (d) the contour image of one component (within its bounding box); (e) the corresponding angle histogram; (f) the output mask.

ing consists in a "full-3D" approach that only exploits the geometrical setup of the sensing system. Despite 3D stereo reconstruction is a widely used approach in out-of-water perceptual tasks, in underwater environment issues like image quality, light reflections and calibration are not trivial and impose an accurate method assessment.

3.5.1 Full-3D approach

Object detection and pose estimation are performed on the 3D point cloud computed using stereo vision techniques. The ROI obtained from single camera processing is used to restrict the region where the object is searched. Since the 3D object recognition step requires computationally expensive operations on point clouds, ROI identification is a mandatory step in the processing pipeline.

The developed approach to object pose estimation is focused on objects that have

a cylindrical shape and is based on model matching between the object parametric representation and the filtered point cloud. Three different algorithms have been implemented and evaluated: *RANSAC* model fitting, *Particle Swarm Optimization* (PSO), *Differential Evolution* (DE) [80]. An extensive discussion of achieved results is reported in [76, 81] together with even more sophisticated solutions which have been evaluated.

The effectiveness of these methods is strictly related to accuracy and completeness of the 3D point cloud. Field experiments, discussed in chapter 4, have shown the difficulty of obtaining accurate 3D reconstructions of underwater texture-less objects. Thus, alternative methods based on object geometrical features have been investigated.

3.5.2 Geometry-based approach

In general, object pose estimation cannot be performed on a single image unless an a-priori knowledge of its shape and size is available. This paragraph is focused on pose estimation in single (or multiple) images of cylinder-like objects, although a similar approach could be developed for box-like objects and other regular 3D shapes.

This method is based on the detection of pipe borders and terminals and relies on the prior knowledge of object size. Lines in image plane built upon borders and terminals are projected as planes in the 3D space. Exploiting information about cylinder radius and searching for appropriate plane intersections, the 3D pose of the target object can be estimated [82].

The accuracy of such estimation depends on the image resolution and on the extraction of the two lines. Robustness of this method can be improved by performing the same processing on stereo images, if available. Obtained results can be compared and refined in order to converge toward an optimal solution.

In actual experiments, this method has proven reliable and robust to different colors, points of view, distances and light conditions. Despite the pipe natural shadows near the seabed, that puts a strain even on the human sight in detecting borders, the algorithm is able to estimate the pipe position (fig. 3.9). Due to the absence of ground truth regarding the real pipe pose, a qualitative evaluation has been performed by

visually comparing the pipe with the re-projection of its 3D pose (blue line in figure 3.9) on the image plane. Additional details about this geometry-based approach to pipe detection are presented in [82]

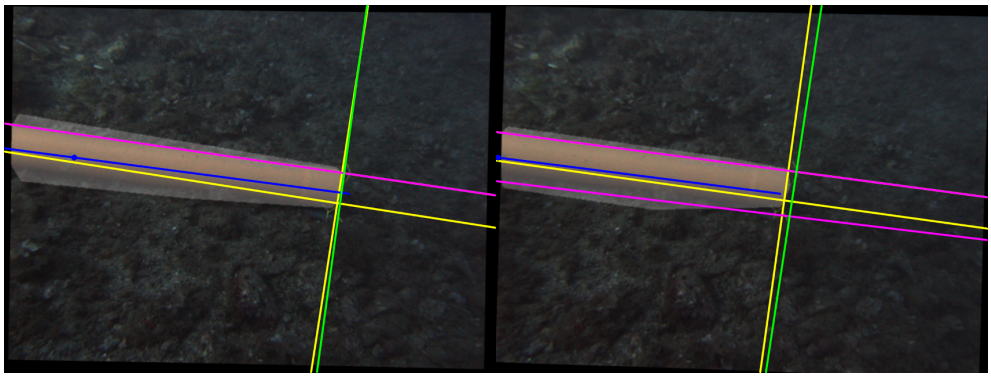


Figure 3.9: Geometry-based pipe pose estimation. Yellow lines represent the detected border directions. Refined borders are highlighted in purple (long borders) and green (pipe terminal). The blue line represents the re projection of the detected pipe in the image plane as a qualitative feedback of the pose estimation accuracy.

The main outcomes stemming from the evaluation of object detection and pose estimation algorithms for underwater environments, discussed in this chapter, are the importance of providing a customized processing pipeline, tailored upon the specific application context. Wide variability of underwater conditions, together with limited computational resources, prevent the development of a "silver bullet" vision processing pipeline for object detection and pose estimation tasks that does not exploit a prior knowledge about target object and perceptual environment.

Next chapters will show how the evolution of the processing pipeline has been supported by field experiments, performed since early stages of the project, and, vice versa, how results of the developed algorithms defined the system requirements.

4 Field Evaluation

The design of the underwater computer vision system has aimed at developing a platform suitable for real world applications. Increasing the Technical Readiness Level (TRL) of underwater applications is the first step to make them attractive for industrial deployment and exploitation. As described in chapter 2, both the initial prototype and the final vision system have been developed for a robust deployment in underwater applications. For these reasons, both systems have been evaluated in real world scenarios and the following sections will report achieved results.

4.1 Embedded prototype

The embedded prototype has been developed in order to have, in short time and at low costs, an imaging platform able to collect image sequences of submerged objects. The evaluation of the system has pursued both the objectives of validating the system in real world and of gathering underwater images.

The experiment location has been chosen at the Lake of Garda (Italy) and two separate sessions have been conducted in Bardolino and near the city of Malcesine, as shown in figure 4.1. In these experiments no boating or underwater support has been available, and therefore all activities have been conducted from the shore. Lake of Garda has been chosen because of its clear water, the rather deep seabed next to

Figure 4.1: Testing locations at Lake of Garda: Bardolino and Malcesine.

shores and the availability of wharfs, useful for logistics support.

The prototype, due to systematic use of plastic materials, is positively buoyant and tends to float. However, cameras needs to be submerged at least some tens of centimeters to avoid light reflections of water surface and air bubbles between the canister glass and the water. Canister has then been ballasted with iron weights, fixed to an appropriate frame on the back side of the box. Once reached a desirable balance, which means a slightly negative buoyancy, the canister was fixed with chains to a floating unit to avoid its sinking and ease maneuvering (fig. 4.2).

The canister was connected to the base control station with a power supply cable and an ethernet cable, coupled and made floating with small buoys. Since the canister was self-powered by onboard batteries, the power supply cable was often unplugged and back-up batteries available at base stations were used to power additional control devices (computers, ethernet devices, etc.). The canister was remotely controlled and visual feedbacks were available. The system was moved around with the help of ropes handled manually from shore. Despite the trivial method, in each session an

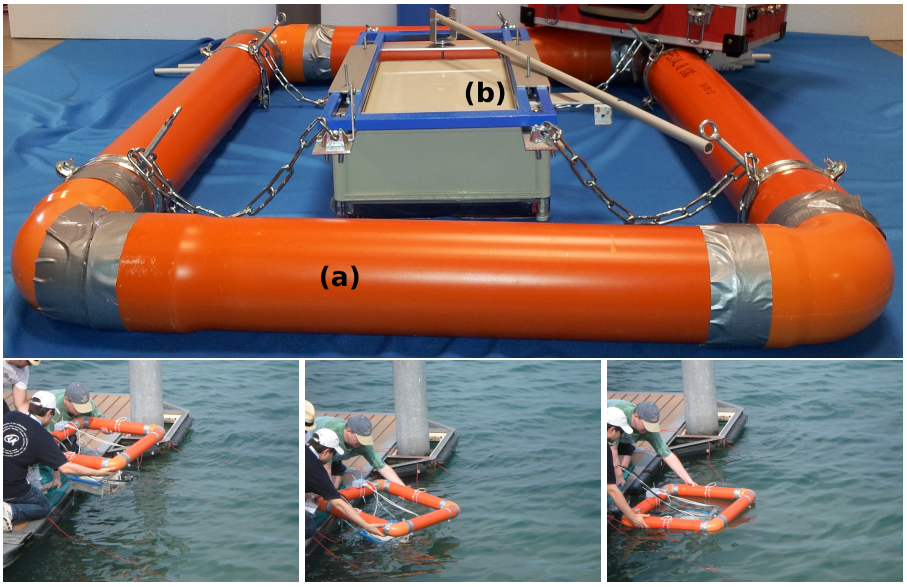


Figure 4.2: Top: embedded prototype with sinking weights (b) and floating buoyancy (a). Bottom: system launch in water at Bardolino testing site.

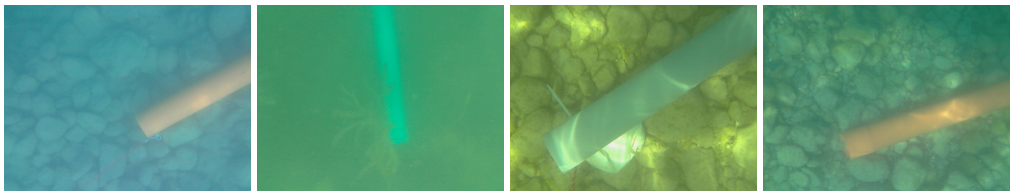
approximate area of $30m^2$ has been covered.

The chosen testing sites mainly differed for the water turbidity, lake floor depth and actual weather conditions. Water resulted clearer near Malcesine, whose seabed is slightly deeper with respect to Bardolino. On the first day of experiments, in Bardolino, weather was cloudy, leading to lower direct illumination and less surface reflections. The next day in Malcesine sky was sunny so that direct illumination combined with clearer water produced typical reflections of water surface on the seabed and objects. Details of both experimental sessions are shown in table 4.1.

Cylindrical PVC tubes of different colors and same size ($10cm$ diameter, $1m$ length), were submerged and laid down on the seabed, at a depth that ranged from $1.8m$ to $3m$. Examples of images collected are shown in figure 4.3. The figure points out the differences between the two testing sites, especially regarding water clearness (frames (b) and (d) refer to objects at approximately the same depth) and weather conditions effects on underwater light reflections.

Location	**Bardolino**	**Malcesine**
GPS coord.	45°32'40"N 10°43'11"E	45°43'57"N 10°47'13"E
Time	$10:00-12:00$	$10:00-12:00$
Weather	Cloudy	Sunny
Floor	Stones and algae	Stones and algae
Object depth	$[1.8m, 2.3m]$	$[2m, 3m]$
Camera depth	$\sim 40cm$	$\sim 40cm$
Max canister temp	68^oC	63^oC
Max fps	$3.3Hz$	$6.6Hz$
Collected data	$26.1Gb$	$23.1Gb$

Table 4.1: Lake of Garda experimental session data.



(a) (b) (c) (d)

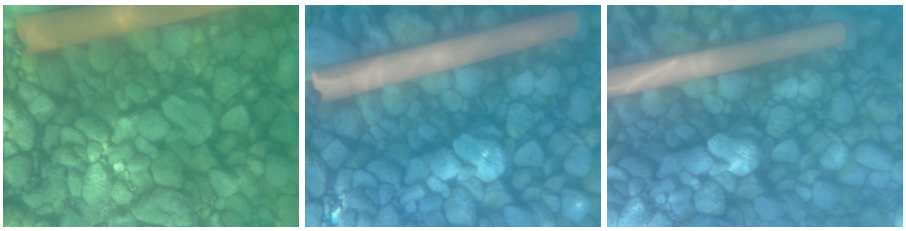
Figure 4.3: Examples frames from Bardolino (a,b) and Malcesine (c,d) datasets. Bardolino: orange pipe in shallow water (a) and gray pipe with algae in deeper seabed (b). Malcesine: gray (a) and orange (b) pipes in rocky seabed with evident surface reflections.

Despite the webcams in this prototype were able to grab high resolution frames (1280x960) at 7.5 fps in a stand-alone configuration, and VGA frames (640x480) at 15 fps when combined with other cameras, during the experiments the actual frame rate was lower. In particular, in the Bardolino session, cameras were driven at the maximum achievable frequency of 15 fps but the acquired dataset showed an actual rate of only 3.3 fps. In the following experimental session, near Malcesine, the grabbing frequency of the webcam driver was set to 10 fps and the effective framerate increased to 6.6 fps. The reason behind this unexpected behavior is still unexplained. However, it is possible that problems in ROS topics synchronization occurred due to

the increased complexity of the distributed network, comprising the embedded system and 2 remote PCs. Furthermore, data collection has been performed onboard with the `rosbag` tool. Data rate on ROS topics has been measured in $8.75 Mb/s$, far lower than SATA bus or SSD physical limits.

In both sessions the average depth of the camera was about $40cm$ below water level. However, the whole structure of the embedded system and buoyancy swung rather fast due to the continuous wave movements. Webcams were not synchronized by a hardware trigger and the shooting jitter between two cameras was on average about 65 *ms*. Rapid fluctuations combined with the absence of hardware triggering and slow framerate led to occasional episodes of evident loss of synchronism between cameras, as shown in figure 4.4. The left frame is clearly unsynchronized with the central and right ones. Due to typical slow movements of systems deployed in underwater tasks, perceptual applications generally do not require high frequency data processing. Nevertheless, camera synchronization has proven to be a mandatory requirement which has been addressed in the design of the final vision system.

Figure 4.4 also highlights another issue related to low-cost imaging sensors. The white balance was automatic adjusted and, consequently, the image color temperature dynamically changed. This unwanted behavior produced color aberrations different in each image and prevented the adoption of an effective color restoration policy on the whole image sequence. This issue is especially restricting in case of online image processing.



(left camera) (central camera) (right camera)

Figure 4.4: Absence of synchronization effects: image from left camera is not synchronized with central and right frames. A side effect of auto-whitebalance is also shown: left image vs. central and right images.

Another side effect of synchronization jitter on image shooting time was produced by surface light reflections. In sunlight weather conditions, the water surface projects characteristic reflections on the seabed and submerged objects. These artifacts act like an image texture with altered brightness and color, contrast gradients and fictional borders. This fake texture changes with high frequency due to waves, so that, in performed experiments, small differences in frames acquisition led to substantial image dissimilarity, as shown in figure 4.5.

(a) (b)

Figure 4.5: Differences on left (a) and right (b) images, caused by surface light reflections and synchronization jitter in shooting time.

One of the goals of the experiments was to test the physical properties of the embedded system, such as the water-proof endurance of the low-cost canister and the thermal balance of the electronic devices inside it. During the experimental sessions, the onboard monitoring system has collected the temperature and humidity values through three analog thermometers, a digital thermometer and a hygrometer. Figure 4.6 illustrates the temperature values measured during the two sessions, each lasting more than one hour. Sensors *temp analog 1* and *DHT11* measure the environmental temperature inside the canister, *temp analog 2* is placed on the heat sink of the CPU and *temp analog 3* is placed on the SSD hard drive. The maximum temperature value of 68°C has been measured by the sensor on the CPU, as it might be expected.

The temperature measured by ambient sensors decreases at the beginning of each session, as soon as the canister is submerged. This result highlights that the thermal balance of the system facilitates the heat exchange with the external environment.

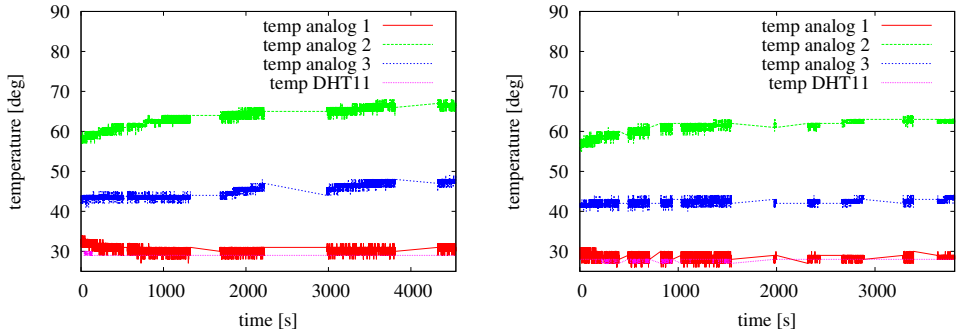


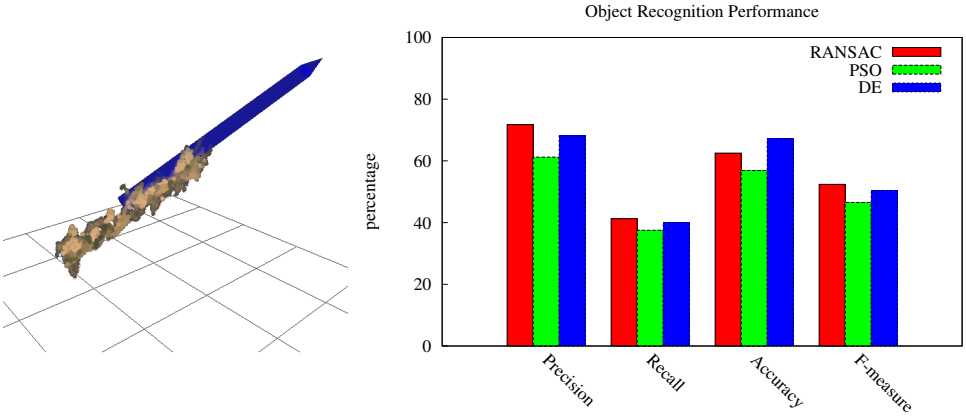
Figure 4.6: Temperature trends over time for the experimental sessions in Bardolino (a) and Malcesine (b).

All measurements seem to converge to stable and safe values. In particular, the canister environmental temperature, measured around 30°C, is approximately 6°C higher than the average temperature of lake water, which is 24°C in July. Thus, the thermal dissipation behavior of the embedded system has proven adequate for its correct operation.

Collected image sequences were aimed at assessing the object detection and pose estimation performance on the point cloud acquired in the stereo camera configuration. Unfortunately, the point clouds obtained from the underwater dataset are rather sparse and noisy. As mentioned above, in water the embedded system was attached to a floating support, and the camera baseline swung due to waves. Since the webcams are not synchronized by a hardware trigger, the computed disparity image becomes noisy and inaccurate. Thus, the 3D point clouds obtained from stereo processing have not allowed a reliable object detection and localization.

The promising results obtained by low-cost stereo systems [55] in recognizing challenging 3D objects, motivated a deeper analysis the impact of underwater scenarios on the algorithms effectiveness. As a matter of comparison, an alternative dataset has been collected out-of-water. In this alternative setting, the target pipes laid in a dry river bed among sand and stones and the canister with the stereo camera was man-

ually moved. Figure 4.7.a shows the recognition of a cylinder from 3D point cloud data, whereas figure 4.7.b summarizes the object recognition results for RANSAC, PSO, and DE algorithms. The three algorithms obtain comparatively similar recognition results, overall satisfactory, showing better performance than when applied to underwater images. It can be concluded that the accuracy, resolution, and acquisition rate afforded by inexpensive webcams have proven inadequate for challenging underwater perception tasks.



(a) (b)

Figure 4.7: An example of pose estimation by matching the raw point cloud and a cylinder model (a). Object recognition results on the point cloud (b). Results obtained on an out-of-water dataset.

Although the experimental evaluation showed its strong limitations, the initial vision system prototype provided insights for the development of more the advanced underwater vision systems. Furthermore, low-cost stereo vision technologies seem adequate for evaluation of early-stage image processing algorithms in underwater environments.

4.2 High end vision system

The high end vision system has been designed to work on an underwater autonomous vehicle. Before the system deployment and integration on the underwater robot, some intermediate tests were required. Furthermore, advances in pipe detection algorithms needed to be tested on image sequences acquired with the final system which incorporates several improvements with respect to the initial low-cost prototype.

An underwater acquisition campaign took place in deep sea waters, near Portofino (Italy) on September 6th, 2014 (fig. 4.8). The testing campaign required a complex logistics support, kindly offered by the divers of Federazione Italiana Attività Subacquee of Parma*.

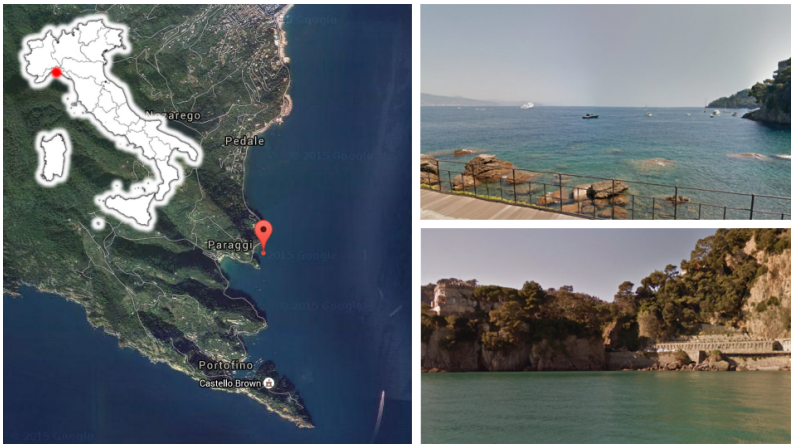


Figure 4.8: Testing location in the Ligurian Sea, near Portofino (Italy).

The expedition was made of three technicians and eight divers, responsible for both surface and underwater operations. Two rubber dinghies were deployed for people and equipment transportation. The identified site for testing (GPS coordinates: `44°18′41"N 9°12′49"E`) is a small inlet in front of Paraggi (Genova) and it is 0.7 NM† distant from Portofino and 2.6 NM from the harbor of Rapallo. The seabed

*FIAS, http://www.fiasparma.it/
†Nautical miles. 1 NM = 1.85 km

was roughly flat, with submerged cliffs and a water depth of approximately 10 m. Despite the presence of sand on the bottom, water was clear and visibility perfect.

The underwater stereo vision system was submerged and maneuvered by two divers (fig. 4.9). The main canister buoyancy is slightly negative, so it tends to sink. For this reason it was also tied to a rope and held from the surface at mid-water, in order to ease divers maneuvering. The main canister was connected to the control station on the support dinghy with a power supply cable and an ethernet link. The required 24V power was supplied by the boat. The ethernet link was arranged with non-marine cables, due to their prohibitive costs over long distances. This led to a loss of performance of the LAN connection that autonomously switched from Gigabit to Fast Ethernet, preventing a smooth remote monitoring. However, the network link bandwidth was adequate enough for remote operating the image acquisition. A set of cylindrical pipes with different colors, patterns and radii ranging from 5 to 6 cm was submerged and laid on the seabed. Cameras were slowly moved around in order to collect images of both the environment and pipes, from different distances and angles.

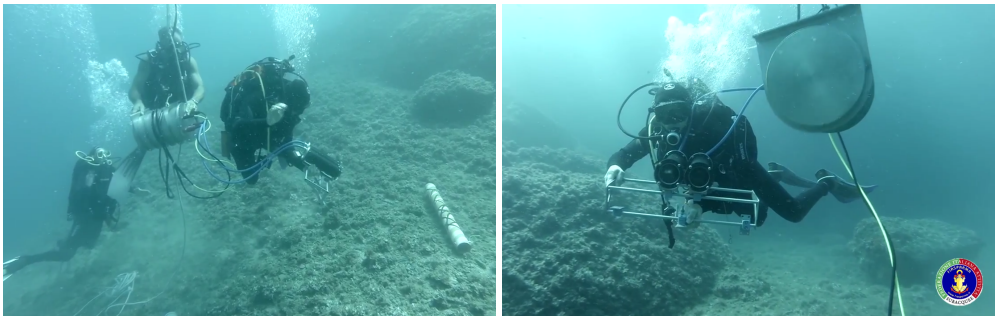


Figure 4.9: Vision system submerged at 10 m depth by a team of divers.

Cameras have been configured with a small baseline due to rather proximity of target objects in the workspace. Images have been acquired at the maximum feasible resolution of 1292x964 pixels, in Bayer encoded format. Examples of grabbed frames are shown in figure 4.10. Although cameras would be able to grab images up to 30 fps, they were configured at 15 fps mainly because in underwater percep-

tion applications images must be processed online, and a target frequency of 10Hz is already challenging. Furthermore, acquiring images at higher frequency would have led to bigger amount of data to be stored, with the unwanted risk of dropping some frame.

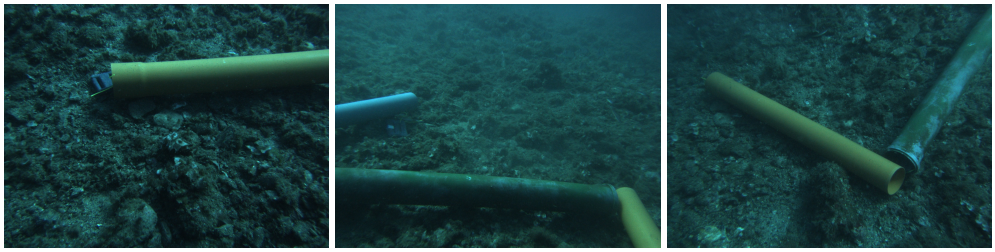


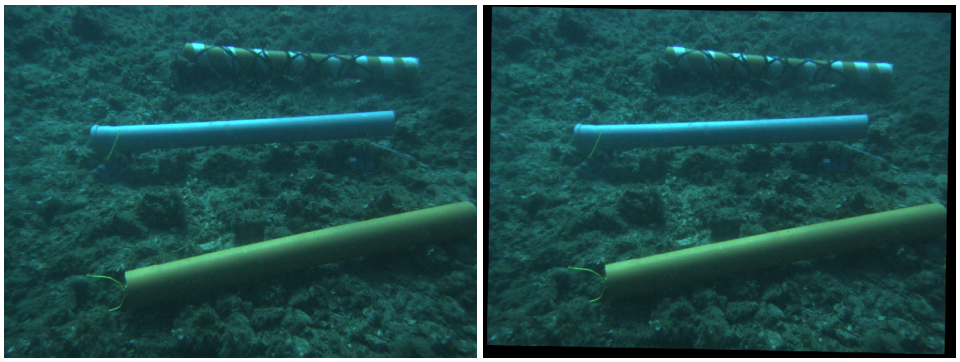
Figure 4.10: Examples of collected frames with different types of submerged pipes.

The underwater testing session lasted approximately 22 minutes. The dataset is made of roughly 20,000 image pairs, whose total size is about 44 Gb. The sequence, cleared from useless frames of the initial sinking and final raising of the equipment, is made of 10,123 stereo images. As stated in chapter 1, public datasets of underwater objects are very scarce. For this reason, the collected dataset has been made available for download at http://rimlab.ce.unipr.it/Maris.html, accompanied with camera calibration parameters in `yaml` file format [73].

During this campaign of data collection and in-field validation, the developed system did not show any notable problem. Images have been acquired in high resolution and at the desired frame rate. Camera triggering, driven by an external microcontroller, produced fully synchronized image pairs. Several camera parameters could be configured at a driver level, in particular exposure time, gain and white balancing. The appropriate combination of these parameters led to rather good and quality-constant image sequences, without variable color aberrations.

High quality lenses, even if used in a wide angle configuration, produced low distorted images, as shown in figure 4.11. The rectified image (b) is rather similar to the original image (a). This result has been also achieved by exploiting lenses designed for larger image sensors. Camera sensors type was 1/3“ which means an

active diagonal of 6.00 mm while lenses could be combined with larger sensors, up to 1/1.8", corresponding to a diagonal of 8.93 mm. Consequently, only the central part of the image projected by lenses is collected by the sensor. Border areas, typically characterized by greater distortion, are thus discarded.



(a) (b)

Figure 4.11: Lens distortion: comparison between original image (a) and rectified image (b).

With the described combination of image sensor dimensions and used focal length of 4.7 mm, the resulting camera field of view was roughly 2000 x 1500 mm at 2.0 m distance. Moreover, the high resolution sensor in which each pixel corresponds to 1.58 mm^2 at the same working distance of 2.0 m, led to a precise scene reconstruction. These achieved results satisfy the accuracy requirements for object detection and pose estimation tasks, in the application context.

Iris range starts from $f/1.6$, suggesting luminous lenses, although they are of varifocal type. In the experiments, a large diaphragm aperture has been used in order to increase the quantity of light impressing the sensor. This configuration aided to keep as short as possible the exposure time, although at 10 m depth the light radiation was strongly reduced. A short exposure time led to still image frames, as shown in figure 4.12. The top row shows two consecutive frames, taken by a camera moving fast. The detail in the bottom row is taken from the top right image: pipe borders are rather sharp with no typical motion blur effects.

The developed stereo imaging system exploited discrete high-end components for

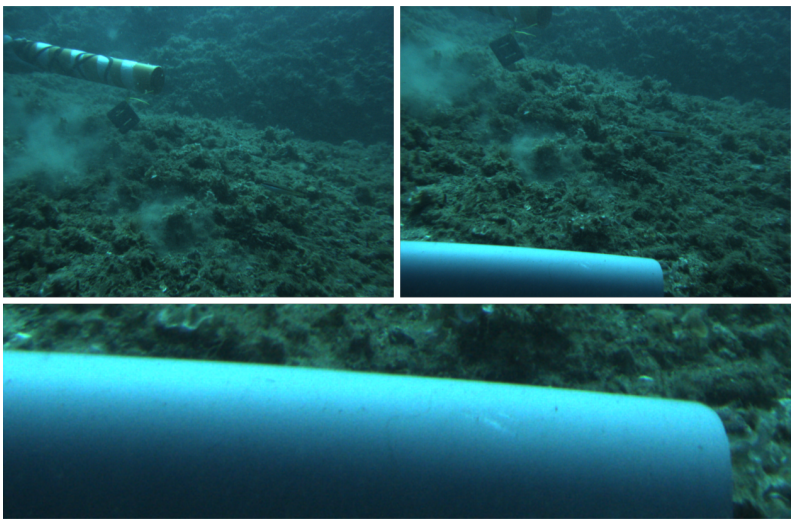
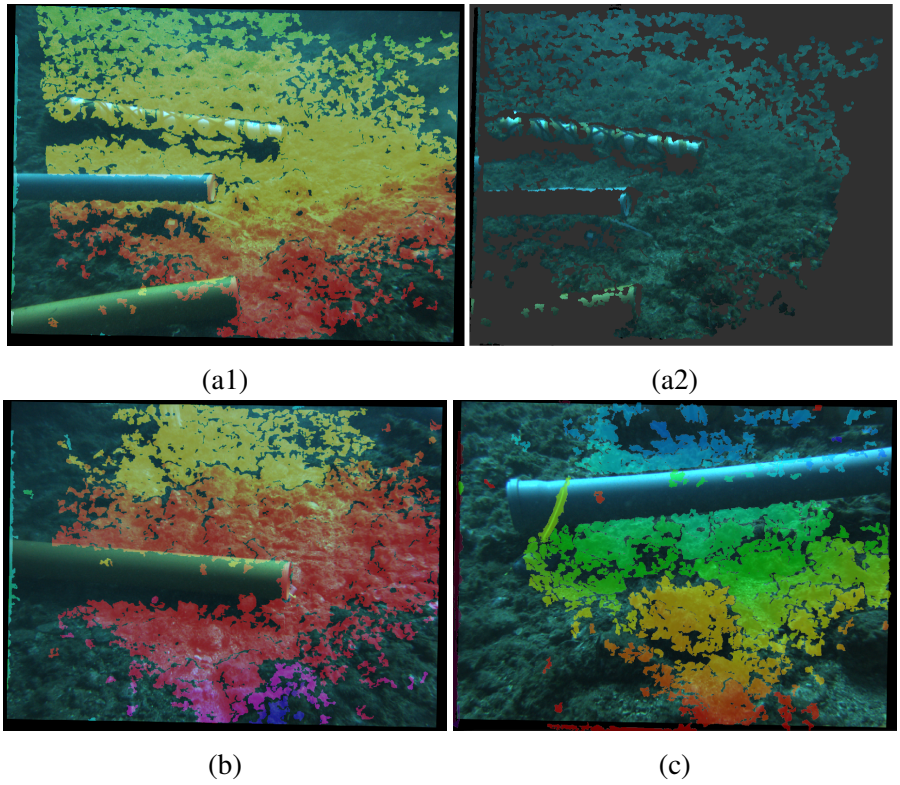


Figure 4.12: Absence of motion blur effects in images taken with moving camera. In the top row, two consecutive frames. In the bottom, a detail taken from last frame.

both cameras and lenses, if compared with alternative out of the box stereo cameras like Bumblebee. Lenses allowed to act on focal distance and iris to obtain a desired FoV with optimal response to light. On the other hand, cameras permitted modification of several parameters with impact on image stillness, brightness, contrast and colors with a wide range of feasible configurations. The possibility of fine tuning the image acquisition stage gives relevant benefits to the following processing steps.

The high end vision system has then been tested in 3D scene reconstruction tasks. Although the disparity map, computed with standard OpenCV methods, reconstructs seabed, fishes, divers, and other textured objects, it generally fails with pipes, as shown in figure 4.13. Pipes have a flat texture which negatively impacts on block matching algorithms [83] and pixel similarity methods [84], exploited in disparity map processing.

Slightly blurred images and a bit of particulate in water suspension may cause a decreased success rate in block matching. Thus, relaxing some parameters of disparity computation algorithm, like `uniqueness_ratio` and `texture_threshold`, a more dense depth map could be obtained, even in correspondence of flat textured



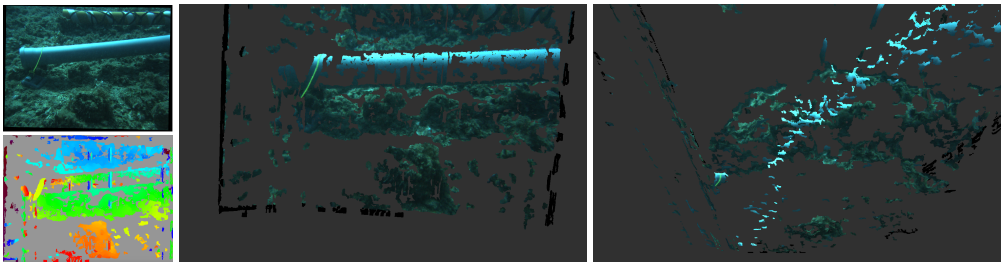
(a1) (a2)

(b) (c)

Figure 4.13: Disparity issues with untextured objects. Image with superimposed disparity map (a1) and 3D point cloud representation (a2). Examples of pipe reconstruction failures (b, c)

objects. However, the trade-off is in reconstruction precision. As shown in figure 4.14, some distortions and rough errors occur, so that approaches exploiting the 3D reconstruction may prove unreliable.

This qualitative result has been quantitatively confirmed by testing cylinder recognition methods based on model fitting. The algorithm operates only on the points corresponding to a ROI obtained, for example, with color-based methods. The success rate of this approach is generally low with precision value about 60% and recall about 40%. Experimental results [76], based on a small subset of available image sequences, showed that when there are enough 3D points lying in the ROI, the cylinder



(a) (b) (c)

Figure 4.14: Gross errors in disparity computation. Original image and disparity map (a). Resulting 3D view of the pipe from two points of view (b,c) with evident reconstruction errors.

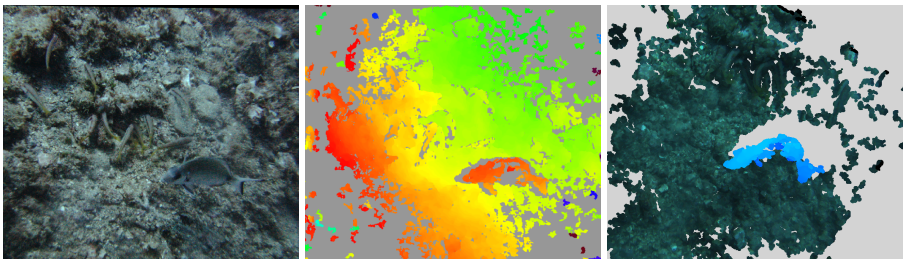
	Grey	Orange
Num. Frames	107	111
True Radius [mm]	45.0	50.0
Avg. [mm]	55.6	85.7
Std. Dev. [mm]	36.9	32.2
Max [mm]	249.5	191.4

Table 4.2: Cylinder radius estimated in Portofino dataset for grey and orange pipes.

axis is computed with acceptable accuracy. Otherwise, the resulting pose is rather inaccurate. The assessment of the object dimensions and, in particular, of its radius in Table 4.2, showed an average estimated value rather close to the ground truth. However, a high value of standard deviation (about 3 *cm*) for both the grey and the orange pipes suggested a noisy estimation, as it could be expected due to unreliability of initial data.

Despite a full 3D reconstruction approach proved unreliable for underwater pipe pose estimation, the seabed and underwater fauna were correctly reconstructed, suggesting alternative valuable uses of the proposed stereo vision system. Stereo images acquired by synchronized and high-quality cameras lead to dense depth maps of the underwater sea-life and backgrounds, which are often characterized by highly dis-

tinctive patterns. Thus, a 3D point cloud reconstruction of submarine environments can be performed with the developed high-end vision system and exploited in applications like seabed mapping and biological targets monitoring (fig. 4.15).



(a) (b) (c)

Figure 4.15: Sea-life monitoring vision application. Original underwater image (a), disparity map (b), and 3D point cloud reconstruction with highlighted fish (c).

The hardware architecture was able to compute the disparity map on full resolution images (1292x964), at a resulting frequency range between 3 and 6 Hz, depending on processing parameters, and at even higher frequencies on resized images. The achieved 3D processing performance matches the real-time processing requirements of most underwater applications, thanks also to typical lower speed motion of submarine vehicles, compared with land robots.

A further analysis has been performed on popular approaches to object detection like the feature constellation methods. The standard SIFT keypoint feature [85] has been tested on sample images with different light conditions. Figure 4.16 shows few examples of the resulting feature association between the model to be found and an image containing the same object. The results are clearly unreliable. Although features are in general less stable with texture-less objects like the orange pipe, the associations are strongly affected by a different luminance of the target (e.g. in the leftmost example of Figure 4.16 the model features are matched with another pipe). Thus, a feature constellation method, which depends on the association between the object model features and the extracted ones, is not recommended for underwater recognition of texture-less objects.

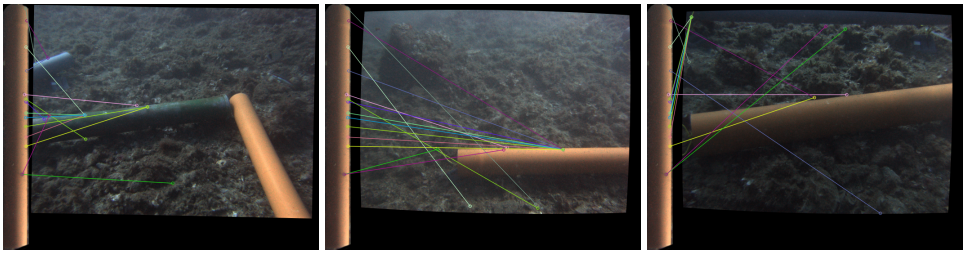


Figure 4.16: Results of constellation method with SIFT feature extraction and matching applied to images at different light conditions. The matches are represented by colored segments between the features extracted from object model image (on the left) and those extracted from each image. The resulting associations are rather unreliable.

Lessons learned in the performed field validation of the vision system suggested a different approach to the specific pipe detection application within the MARIS project, exploiting geometric features of the target object and an a-priori knowledge of its characteristics.

5

MARIS Experimental Setup

As introduced in the preface of this thesis, the MARIS research project represents a perfect test bench for the developed underwater vision system. The MARIS project pursues the general objective of realizing a proof-of-concept experimental demonstration of the achievable capabilities in terms of submerged object detection and cooperative grasping [86]. The first objective of the project comprises the realization of an autonomous underwater vehicle, equipped with a robotic arm, an end-effector and all required technologies to detect and grasp a submerged pipe. Once achieved this result and verified the feasibility of grasping tasks by a single AUV, a second vehicle will be equipped with all MARIS subsystems, and distributed and collaborative control policies will be deployed and tested.

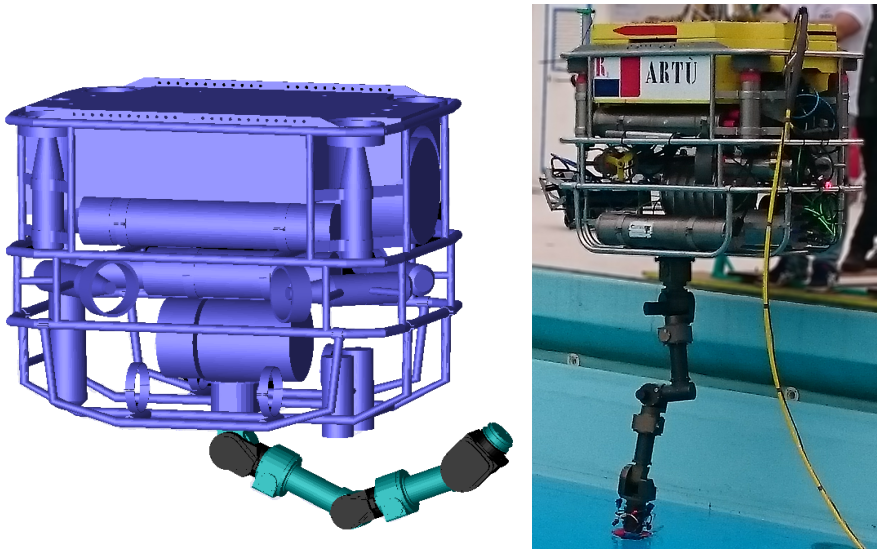
The goal of detecting and estimating the pose of a submerged pipe, with enough accuracy to grasp and recover it, is "per-se" challenging and a step forward with respect to current state of art. The most advanced experiment of underwater manipulation described in literature has been performed within the TRIDENT research project [11]. In TRIDENT, the proof-of-concept experiment was the grasping of a textured target object whose exact underwater representation was a-priori known, in the form of pre-acquired images. Before TRIDENT, it is worth mentioning the ALIVE [87] project in which the autonomous docking of an underwater vehicle was performed exploiting a vision system and markers placed on the station.

The MARIS project involves several italian research institutes, each one responsible of a specific task. In particular, CNR-ISSIA is charged with the task of vehicle development, University of Genova is responsible for the underwater robotic arm and the free-floating control of the combined system vehicle-arm. The grasping hand is developed by University of Bologna and integrated with the robotic arm, and, finally, the underwater environment perception task is assigned to the University of Parma, which is responsible for the development of the stereo-vision module. The Genova CNR team was also responsible for system integration and testing, therefore activities described in the following sections took place with their support and under their supervision.

The expected results of MARIS project research and experiments are, first of all, the availability of small, agile and configurable autonomous underwater manipulation platforms that should per se represent a step-forward with respect to current state of art. As a matter of comparison, among few examples of developed AUVs for interventions, it is worth mentioning the SAUVIM vehicle which weights 6 tons and is equipped with a 60 Kg arm [88]. The MARIS project philosophy of small, cooperative vehicles lead the way toward real-world future applications, especially in the field of offshore industries, which reasonably cannot be performed by a single robot. Long pipe installations and maintenance, underwater manufacturing, large objects retrieval and disaster recovery are examples of tasks in which a cooperative intervention is required. Furthermore, MARIS project has pushed the research on underwater computer vision systems. Results achieved in this thesis and in the development of vision based techniques for underwater 3D reconstruction lead to a sensory subsystem with promising applications in different marine fields.

5.1 System integration

The underwater vehicle designed and developed by CNR-ISSIA, named “Artù”, is made of a stainless steel frame which holds the buoyancy, the eight thrusters, batteries and the main canister including vehicle low-level control units, power distribution, sensors and network connections (fig.5.1). Beneath the robot, an additional



(a) (b)

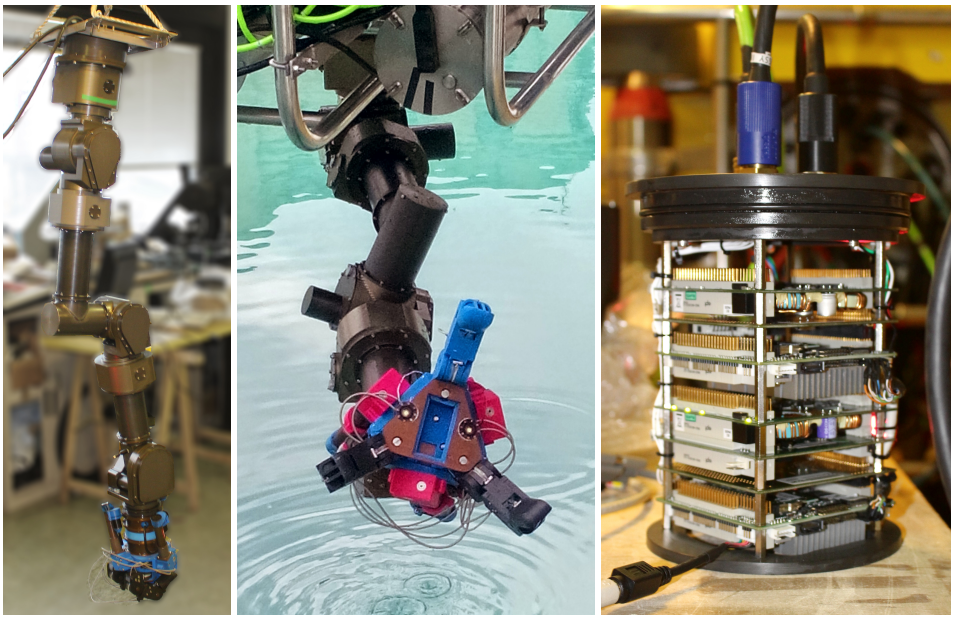
Figure 5.1: CAD model of AUV “Artù” and manipulation arm (a); the real vehicle before its first launch (b).

frame is attached to carry the payload. This modular design simplifies the integration of subsystems by splitting the vehicle itself from additional items whose presence depend on the particular experiment the vehicle is involved in. Furthermore, logistics and maintenance activities are simplified too due to the possibility of decoupling the vehicle from its payload.

The vehicle is about 300Kg in air, while it results neutral in water. Artù can work as an autonomous underwater vehicle (AUV), powered by batteries, or as a remotely operated vehicle (ROV), connected to the base control station by means of an umbilical cable. The cable has been designed for uses in deep water and for long range missions, so its length exceeds 500 meters. It carries high voltage power supply to the vehicle, analog video channels for environment monitoring cameras and a low bandwidth LAN network. High bandwidth links (over 10 Mbps) cannot be established due to the length of the umbilical cable. Hardware and software interfaces between the vehicle and the MARIS system are handled by the “MARIS main canister”. This

canister distributes 24V power supply to all subsystems and represents the center star of the MARIS local area network. The low level vehicle control is uncoupled with the MARIS control architecture for safety reasons: this design permits to reliably switch to robot manual control in case of malfunctioning of the MARIS system.

The robotic arm provided by University of Genova has 7 degrees of freedom and is mounted on the bottom of the payload frame. It weights 30 Kg in air and about 10 Kg in water (fig.5.2.a). A gripper (fig.5.2.b) designed and assembled by University of Bologna as an improvement of the one developed for the TRIDENT project [89] is attached to the manipulator wrist. The robotic hand has three fingers and its kinematic configuration allows to execute both parallel and precision grasps on objects with diameter up to 200 mm. The gripper has 8 degrees of freedom actuated by only three motors by means of a suitable coupling of the joints obtained through

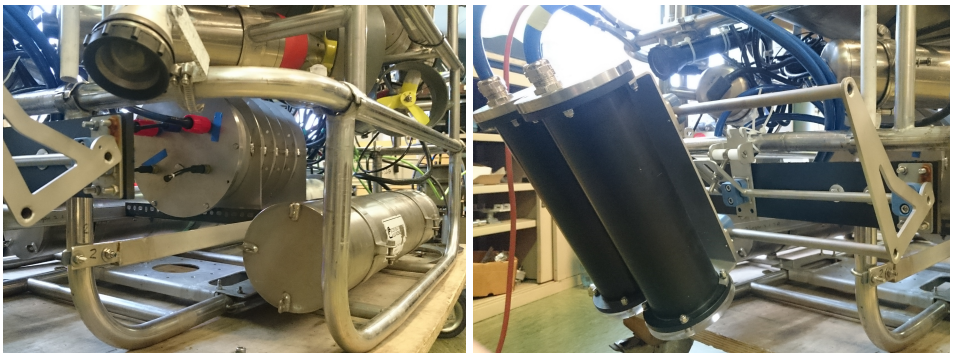


(a) (b) (c)

Figure 5.2: Robotic arm and gripper in a stand-alone setup during integration tests (a); gripper detail (b); ECU for both arm and gripper (c).

the cable transmission. The gripper and the robotic arm share the same control unit held in a very small underwater canister (fig.5.2.c). Communication between ECU and actuators exploits a standard CAN bus.

MARIS subsystems have been placed on the payload frame taking care of dimensions and weights. The vision system canister is the one with biggest diameter and it is heavier than the other two systems. Due to these reasons, it has been placed in the middle of the frame (fig.5.3.a). The robotic arm is fixed beneath the vision canister, slightly moved toward the bow of the vehicle. The MARIS canister and the arm controller canister are placed at the sides of the vision system, shifted toward the vehicle stern. For a better distribution of masses, two additional battery packs have been fixed on the far sides of the payload frame. This distribution of weights should prevent unwanted rotations of the vehicle when the arm moves around.



(a) (b)

Figure 5.3: Mounting position of vision canister (a) and cameras (b).

Cameras must be able to observe the workspace of the robotic arm, possibly from a point of view in which occlusions projected by the arm itself are reduced. The best found solution was to put cameras outside the shape of the vehicle, looking down with a small tilt angle. Cameras are mounted on a special bracket with rails that allow to modify the baseline. In planned manipulation experiments, the system is expected to work within distances of about 2 meters, so cameras have been kept as close as possible with a resulting baseline of about 15 cm (fig.5.3.b). For same

reasons, optics have been configured with a short focal length leading to a wide-angle lens configuration. The result is a wider field of view with a trade-off in terms of image distortion that can be anyhow reduced with camera calibration.

All the system are then connected with appropriate underwater cables. The complete scheme of the MARIS system is summarized in figure 5.4.

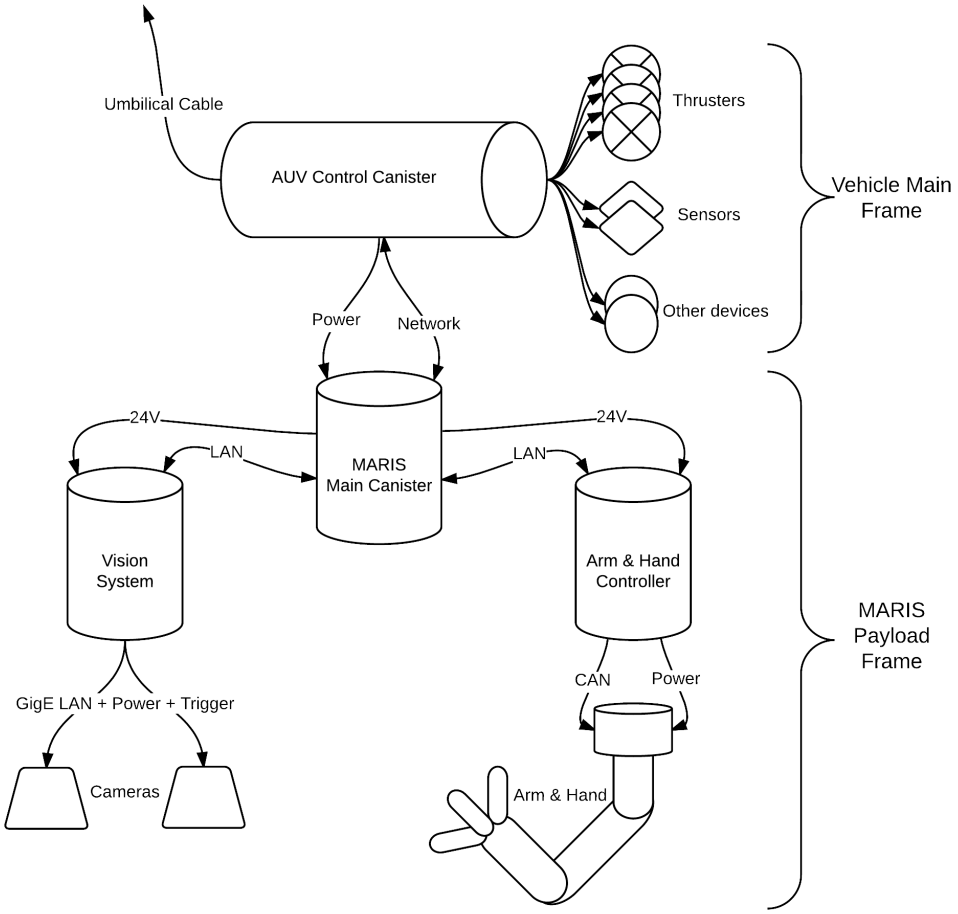


Figure 5.4: Simplified scheme of devices and connections of the MARIS AUV “Artù” and other subsystems.

MARIS systems relies on the state-of-art robotic framework ROS. ROS is widely used among the robotic scientific community especially because it simplifies the communication between multiple agents and provides diagnostic and monitoring tools useful during integration tasks and experiments. The definition of exchanged messages footprint was performed in initial project development stages so that each partner was able to develop every subsystem by perfectly knowing which kind of information would be available from other systems and which ones had to be, on the contrary, provided. Focusing on the vision system, it provides the grasping goal frame and, optionally, an additional frame for camera tracking. It also publishes some calibration matrices, which are statically advertised. The vision system does not need information from other systems, although it can eventually subscribe to the manipulating robot joints position in order to estimate arm self occlusions to camera images. The consistency of the ROS architecture has been verified by evaluating the communication between each subsystem and the key manager of the ROS network, named "`roscore`".

Once all subsystems and their mutual communication have been verified, the MARIS vehicle was ready to perform first tests in water.

5.2 Manipulation experiments

The preliminary expected outcome of the MARIS project is the deployment of an intelligent autonomous underwater vehicle, able to detect an interesting target and recover it to the surface. This result should necessarily be achieved before proceeding with cooperative manipulation experiments. The following section describes the activities that have been carried out within the in-field validation of the MARIS system. In particular each subsystem has been verified, adapted and fine-tuned in order to pursue the MARIS objective of performing a fully autonomous manipulation of a submerged pipe.

Some earlier tests, performed in laboratory and out-of-water, have been conceived, by the MARIS consortium, in order to verify the vision system and the floating arm within the control loop. In a laboratory setup at CNR-ISSIA in Genova,

available space constraints prevented the use of the AUV with the installed arm beneath it. As a feasible alternative, the arm was disconnected from the vehicle and fixed to the laboratory ceiling while keeping all the MARIS subsystems connected as they would be in the final setup. With this trick, the arm was able to move in an appropriate workspace and some manipulation experiments could be performed. In order to avoid error prone procedures like pipe detection and ensure a reliable 3D pose estimation for system calibration and integration purposes, the vision system was reconfigured to detect a checkerboard.

Goal of these tests was a preliminary qualitative validation of the system, especially concerning information exchange among all its subsystems. This goal was obtained by performing a manual guided grasping of a pipe. Technically, this experiment was performed by manually moving the checkerboard in front of the vision system, which estimated a pose, used as fake grasping target. The goal frame was then transmitted to the arm controller which consequently moved the manipulating robot. Finally, when the end-effector was correctly positioned on the target object, the grasp command was manually sent to the robotic hand.

The success of this experiment (fig. 5.5) allowed validation of the interaction between vision system and arm controller and excluded rough errors. The MARIS AUV was then considered ready for its launch in water.

Requirements of long testing sessions, together with the availability of a better logistics support, have driven the decision of performing experiments in a pool (fig. 5.6) and not directly in deep sea waters. The chosen location (Piscine di Albaro, Genova) offers a 3.5 m depth, outdoor pool with a working area of about 15 x 20 m. Logistics has been made easy by availability of power supply for all the devices, wifi internet access and the possibility to deploy a fully equipped base control station for experiments coordination and remote monitoring.

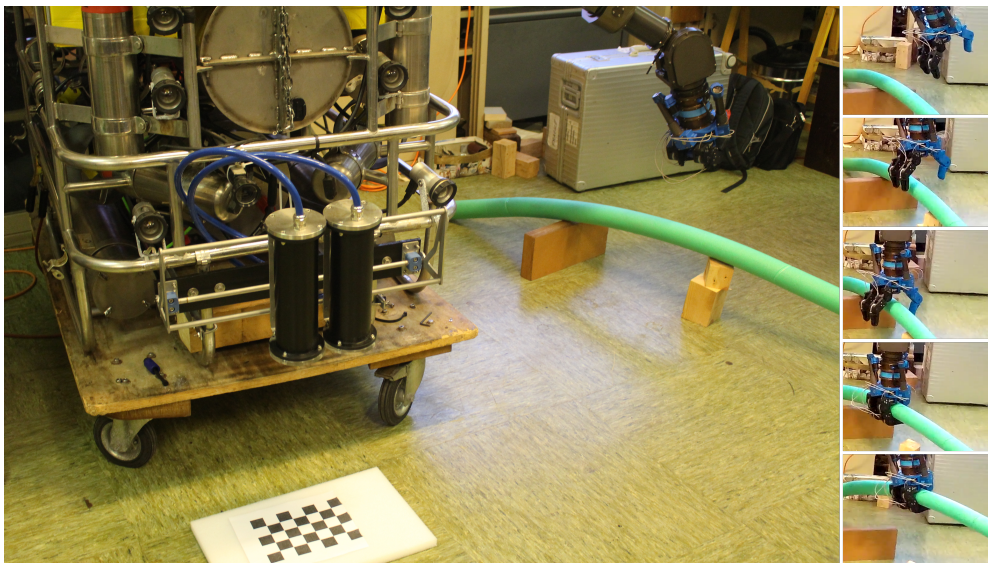


Figure 5.5: Preliminary grasping experiment for vision and arm interaction validation. Experiment setup (left) and image sequence of performed grasp (right).

Figure 5.6: Experiments location: “Piscine di Albaro” (Genova), an outdoor swimming pool with depth of about 3.5 m.

5.2.1 System calibration

The MARIS vehicle is made of multiple, heterogeneous systems working together: vehicle, arm and vision system. Systems refer to a common reference frame (v), chosen for convenience on the vehicle, and their configuration with respect to the reference frame must be known. For these reasons a system calibration is needed. The arm subsystem is made of base (b) and several joints (J_i), whose odometry with respect to the base is perfectly known. consequently the arm calibration consists in determining the configuration of the arm base with respect to vehicle frame (vTb). The plate for arm mounting on the vehicle is precise and not prone to tolerances, so the vTb transformation has been computed from technical drawings of the vehicle. Regarding cameras, once the intrinsic and extrinsic parameters are known, the 3D representation of the scene is returned with respect to the left camera frame (c). The transformation that represents the camera frame in the vehicle reference system (vTc) must be determined. Cameras mounting has been conceived to be flexible and adaptable to different scenarios and cameras inclination has been adjusted manually and cannot be inferred from technical drawings. Thus, the vTc transformation should be obtained by an appropriate calibration procedure. Frames and transformation matrices involved in the calibration procedure are shown in figure 5.7.

The idea behind the calibration is to exploit the precise arm odometry and obtain vTc determination through a composition of multiple transformations and the help of a checkerboard placed on an arm joint, as shown in figure 5.8. In details, a checkerboard is placed on the sixth joint, with the detected cross as precise as possible on the joint rotation axis. In this way, the resulting transformation between arm joint and checkerboard ($6Tk$) can be easily estimated. With an appropriate checkerboard detector, the vision system estimates the cTk transformation so that the resulting transformation bTc is computed as:

$$bTc = bT6 \times 6Tk \times (cTk)^{-1}$$

Once obtained bTc, the vTc transformation can be obtained by composition of bTc and vTb, which comes from arm base calibration.

The vision system also requires that intrinsics and extrinsics camera parameters

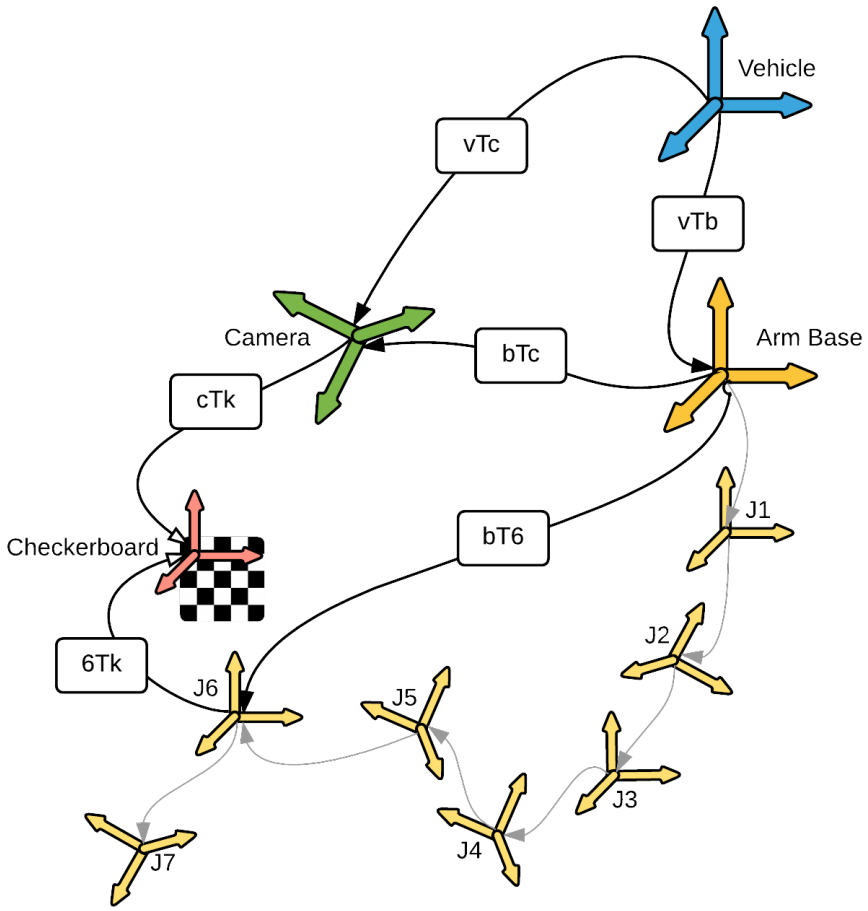


Figure 5.7: Scheme of camera calibration procedure by means of a checkerboard placed on arm joint 6 and arm odometry.

are precisely calibrated underwater. Light refraction through multiple media (water, canister glass, air, lenses) produces distortion that needs to be corrected. Manually performing underwater calibration is often difficult due to needs of specially equipped operators. A semi-automatic procedure has therefore been designed, exploiting again

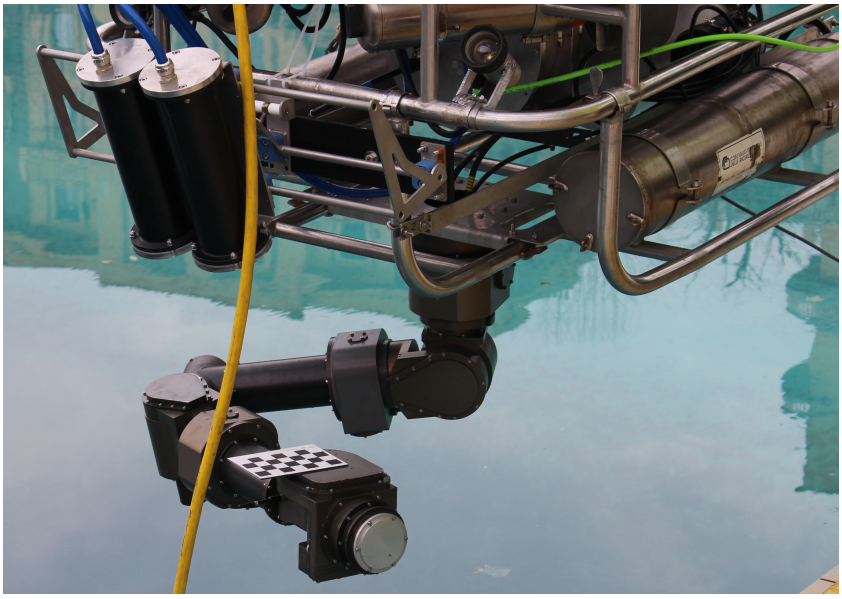


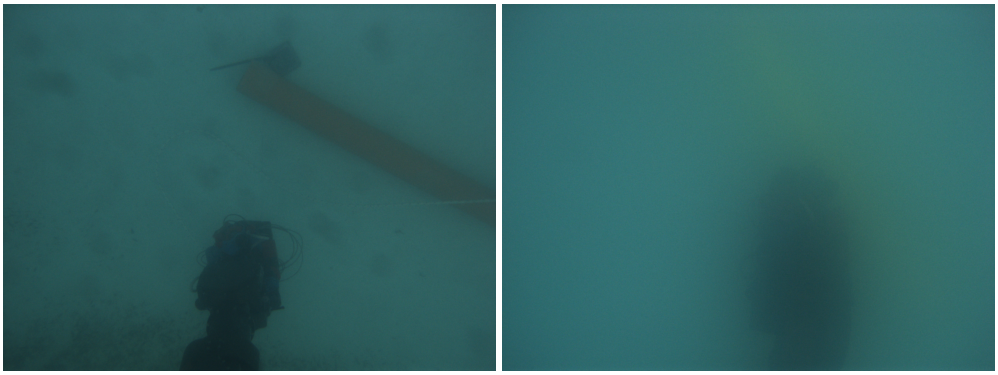
Figure 5.8: Calibration checkerboard on the arm sixth joint during the calibration procedure.

the robotic arm. With the checkerboard mounted on the sixth joint, as described before, the AUV has been submerged. The conventional camera calibration tool, based on Zhang et al. [70] method and available in ROS [90] has been launched on the onboard vision system. The arm has been manually controlled in order to move the checkerboard in several positions and orientations in front of the cameras. Resulting calibration was based on 60 image pairs and the computed matrices have been saved for use in underwater environments. Out-of-water calibration data have been saved too because the image acquisition driver is able to switch between different sets of camera parameters depending on the context of operation.

Once the vehicle had been calibrated and prepared for water, it was possible to proceed with preliminary tests of all components. In particular, the computer vision system required a fine tuning of parameters and an adaptation of the processing pipeline, depending on the working scenario in terms of water turbidity and light conditions.

5.2.2 Preliminary tests

The main objective of the first testing campaign, performed within the MARIS project in March 2015, was a general testing of all subsystems and the collection of experimental data for further improvements. Although the testing environment was a swimming pool, water conditions were expected to be better. Lots of dirt was present both at the bottom and on the water surface. Main problems, however, were caused by the particulate in water suspension. Water turbidity, as shown in figure 5.9.a, led to minimal visibility and blurred images. Furthermore, some dirt particles were collected by the glass of the cameras, resulting in blotchy and distorted images.



(a) (b)

Figure 5.9: Pool water turbidity during March (a) and October testing sessions (b).

During these tests the vehicle was remotely operated and some pipe grasping tasks were manually performed. Approximately 150 Gb of image dataset have been collected. A few fully automated grasping trials were also attempted but poor underwater visibility, spurious light reflections on the arm producing wrong ROI detection, difficulties in keeping the pipe in camera FoV, led to failures. Nevertheless, the experimental campaign permitted to gather more precise information about the grasping task, the environment and the behavior of the AUV. On the basis of the lessons learned, further improvements to the processing pipeline were planned and later developed and implemented in the system. The testing session therefore achieved the

expected results.

The most important outcome of these preliminary tests was the need of a robust pose estimation, even in case of temporary loss of the target object due to vehicle and arm movements. The vehicle odometry, indeed, was not reliable enough for manipulation purposes (as could be expected) and information coming from the vision system is the only reliable feedback. Due to these reasons, the necessity of a grasping frame tracking algorithm clearly emerged in this testing campaign.

At the end of October 2015 another testing session took place. Initial plan for this session was to perform grasping trials with the complete system working. However, conditions of water were as shown in figure 5.9.b. The dark color stain is actually the robot hand and the pipe is only visible as a very soft shade of yellow. These conditions precluded any chance to complete underwater trials. Some tests were performed out of water, with the pipe floating on the pool surface.

Goals of the out-of-water tests were the validation of the grasping system, working with information coming from the vision system. However, pipe detection algorithms, being designed for underwater operation, are not meant for working with direct sunlight illumination, where high gradient shadows occur. Therefore, initially out-of-water pipe detection was performed with the help of a checkerboard marker, as shown in figure 5.10.

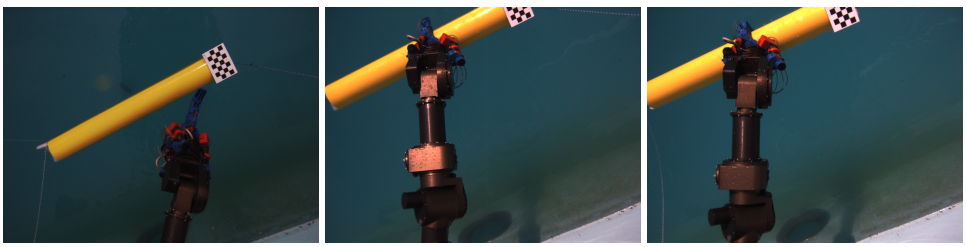


Figure 5.10: Sequence of a vision-driven successful grasp performed out of water surface, with the help of a checkerboard marker.

Taking advantage from the reconfigurability of the vision system, the pipe detection algorithm was then adapted to out-of-water conditions, thereby enabling trials without the checkerboard marker. Out-of-water calibration data were set to cameras

and image shooting parameters (exposure time, gain, color correction) were adapted to the new working environment. Furthermore, some image processing settings were modified in order to cope with the rather different light conditions, made of stronger contrasts and sharp shadows. The reconfigured vision system allowed many successful grasping trials, as shown in figure 5.11.

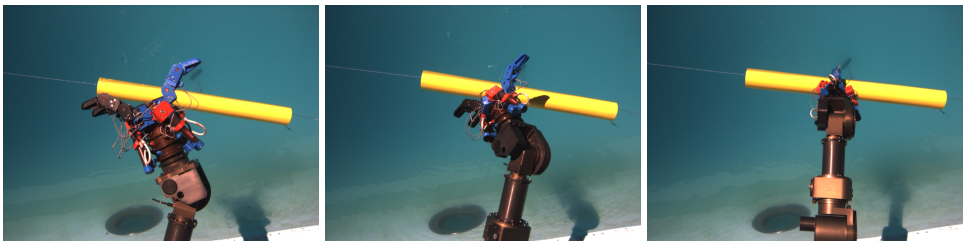


Figure 5.11: Sequence of a successfully grasping performed out of water surface.

During these experiments, the problem of arm and hand occlusions to the camera emerged. The arm is usually in a "home position" in which only part of the hand is visible to the cameras. When the arm approaches the pipe, a larger part of the image gets occluded. The pipe detection algorithm is robust to occlusions of part of the pipe but problems arise when the occluded part is the pipe terminal. Often it happens that only one pipe terminal is within the camera FoV. When the hand occludes the only visible terminal, the border of the hand is considered as the pipe terminal. In these situations, the pose estimation fails. Furthermore, sometimes the arm links may be confused for pipes, leading to erroneous ROIs in the early steps of the processing pipeline.

To solve both these problems, the developed solution was to mask out the arm and hand from the image. This improvement has been obtained by re-projecting in the image plane the approximate shape of the arm and its end-effector. Results of this arm-masking procedure in two different frames are shown in figure 5.12. The effectiveness of this solution has been tested on collected datasets and then validated in the next experimental campaigns.

Figure 5.12: Examples of arm-masking technique performed by re-projecting the manipulator approximate shape in the image plane.

5.2.3 Final experiment

As stated in the chapter introduction, the first result to be achieved in the MARIS project is the feasibility of grasping tasks by a single AUV. The expected success of a proof-of-concept experiment of a pipe grasping task, driven by information returned by a computer vision system, precedes the "cooperative" stage of the project. Furthermore, the experiment validates the vision system since it exploits the returned data in the vehicle and arm "free-floating" control loop [91].

The final experimental session took place at "Piscine di Albaro" testing site from December 15, 2015, lasted 3 days, and was preceded by system preparation activities, carried out by CNR. The AUV "Artù" was moved to the testing pool, assembled with the MARIS payload, made of arm, cameras and controlling canisters, and then calibrated again.

Testing days showed variable weather conditions, from sunny to cloudy, with comfortable temperatures. Pool water conditions were characterized by reasonable clear water with some dirty precipitate on the pool bottom (fig. 5.13). The water height was reduced with respect to previous performed tests by approximately 80 cm. The lower water depth led to some difficulties in pipe detection because of the reduced feasible workspace and the consequently reduced FoV of cameras, which were forced to look at a nearer pipe, often not fully contained in the image frame.

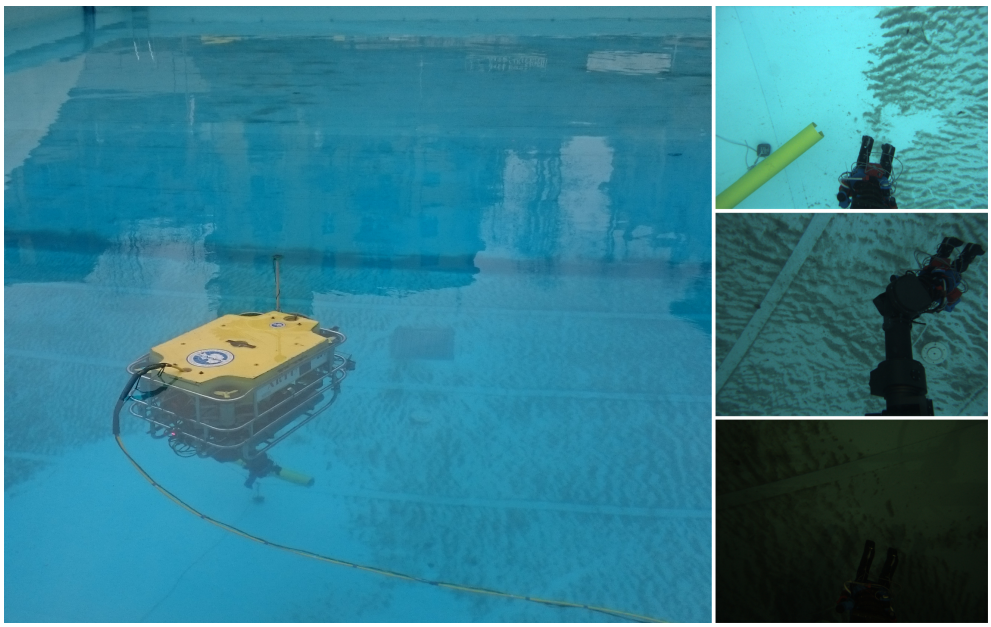


Figure 5.13: Water conditions of reduced turbidity during final experiments. Picture taken from out-of-water (left) and pictures shot by underwater cameras during day (top, centre right) and evening (bottom right).

The vehicle, operated in a ROV configuration, explores nearby until the target pipe has been detected by the vision system. As soon as the pipe pose is consistently estimated, the vision system returns the goal frame for the camera tracking and the vehicle becomes autonomous. The AUV control switches to “free-floating”, which means that the system comprising vehicle and arm is considered as a single entity with appropriate goals. In particular, the free-floating control aims at reaching two objectives: to keep the left camera centered on its tracking frame, in order to always have the best feasible view, and to reach the grasping frame with the robot end-effector to grasp the object.

The pipe detection and pose estimation algorithms used in the final experiment exploited a processing pipeline made of several components. Among the explored solutions described in chapter 3, the most reliable and robust algorithms have been

preferred. In particular, color and geometric shape have proven to be reliable features for object detection and pose estimation.

Raw images acquired by both cameras are initially preprocessed to remove lens distortion, apply Bayer filter and restore color components. Then, a Region of Interest (ROI) is searched by means of an approximately known color of the target object. At that point, the ROI is slightly expanded to ensure that pipe borders are within the considered area and an edge detector algorithm extracts the main border directions. Finally, the pipe 3D pose is computed by projecting and intersecting in the 3D scene the planes corresponding to detected edge directions. Constraints regarding the object size, which is a-priori known, are also exploited for a better robustness of the pose estimation. The arm re-projecting technique described in chapter 5.2.2 has been used to avoid wrong detections of the pipe terminals. Figure 5.14 shows the visual feedback of the processing algorithm. In particular the blue line represents the re-projection back to the image plane of the pipe longitudinal axis, based on the estimated 3D pose. The image shows that the re-projection matches with the pipe and qualitatively confirms the effectiveness of the algorithm.

Reliability of the returned grasping frame is enhanced by a tracking technique that filters out erroneous spike measurements and compensates for short interruptions, typically caused by loss of the pipe terminal in the image.

Trials have been performed with different weather conditions, leading to different kinds of illuminations. Experiments have been carried out till after sunset with natural ambient light slowly fading out and leaving space to artificial illumination (fig. 5.15). In deep waters, indeed, natural light is low to absent and cameras can only rely on AUV lights, with increased phenomena of backscattering due to closeness of light projection line and camera optical axis.

Pipe grasping has been successfully performed in 5 trials out of a total of 9 approaches to the target object where vision has been triggered for detection. Additional experiments have dealt with other aspects of the MARIS system. Successful grasps comprise both tests in daylight and after sunset (fig. 5.16), showing a great robustness to different light conditions.

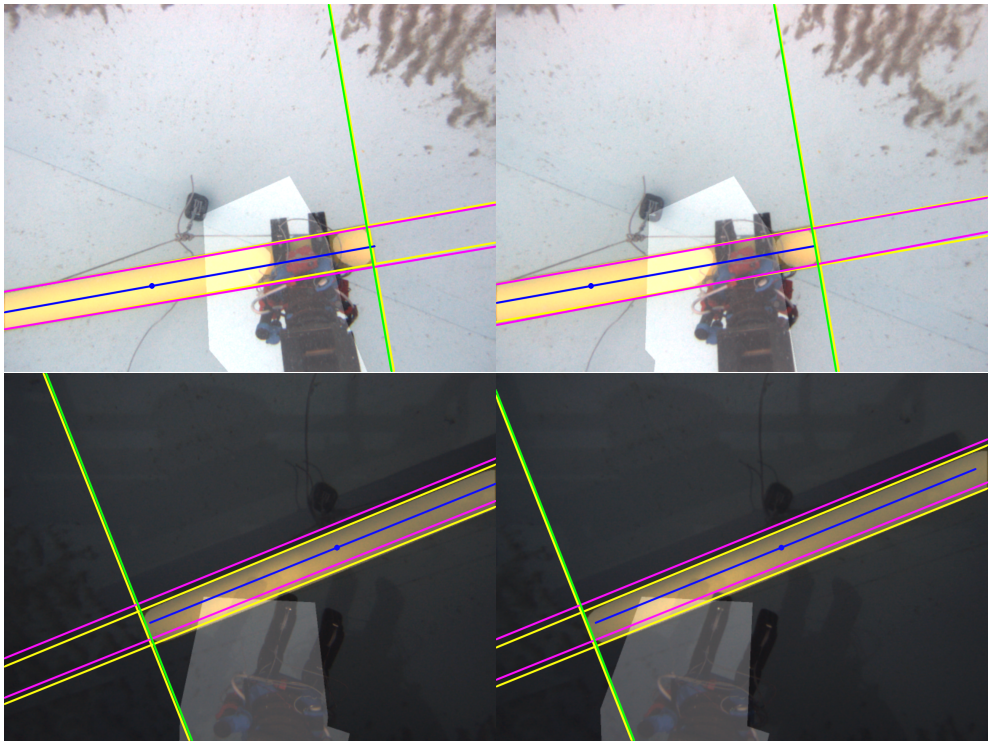


Figure 5.14: Stereo image processing for pipe detection during daylight (top) and evening (bottom) trials. Colored lines represent the detected and refined borders. The blue line represents the re-projection of the detected pipe in the image plane as a qualitative feedback.

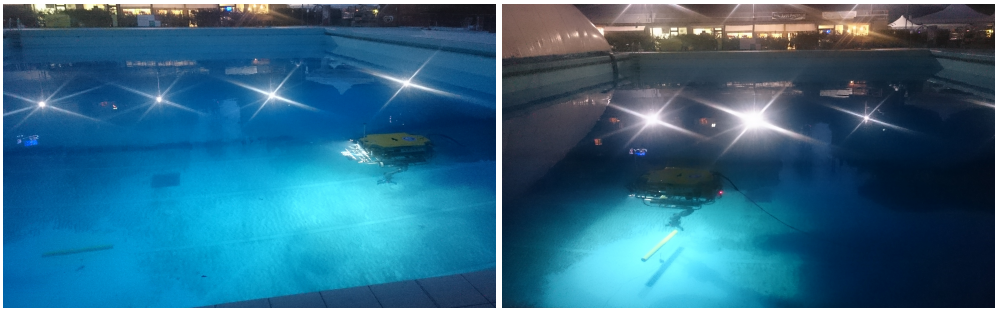


Figure 5.15: Experiments performed after sunset with low ambient light.

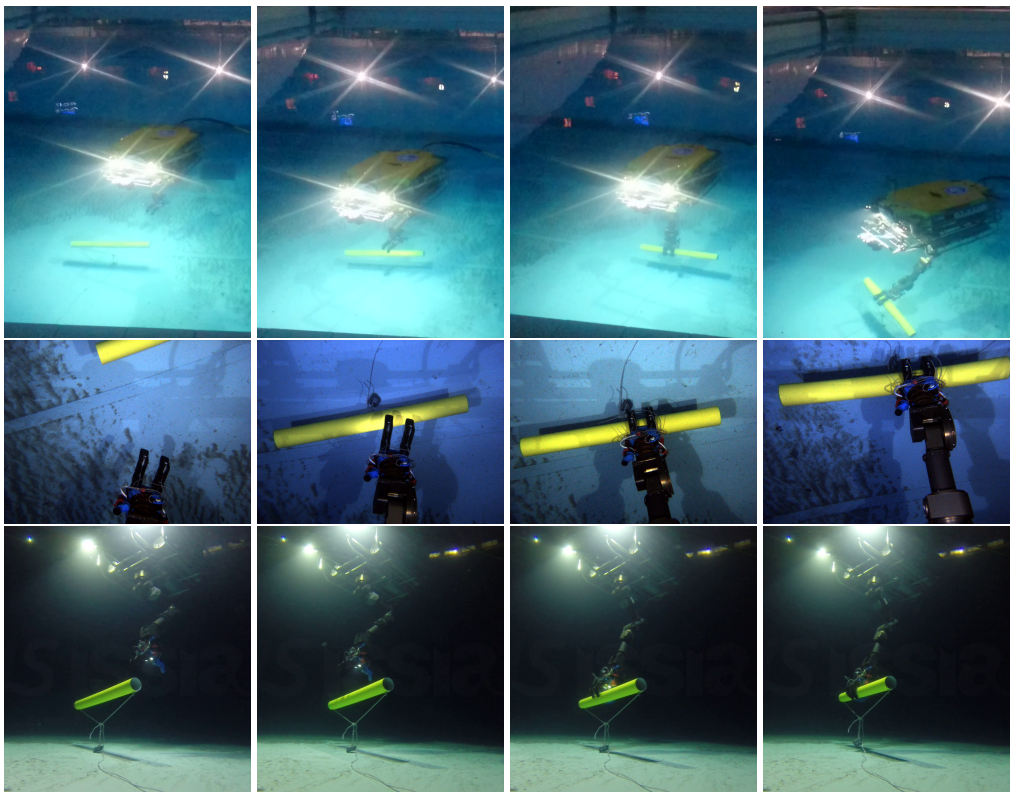
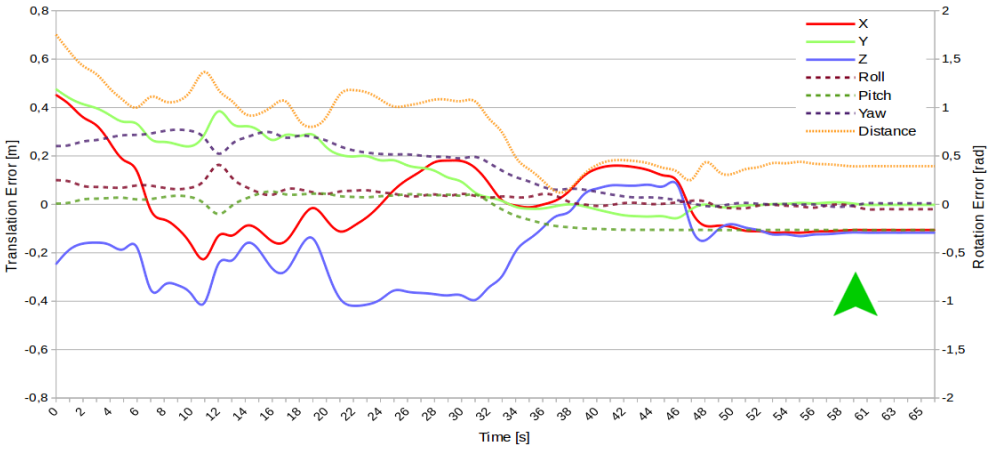


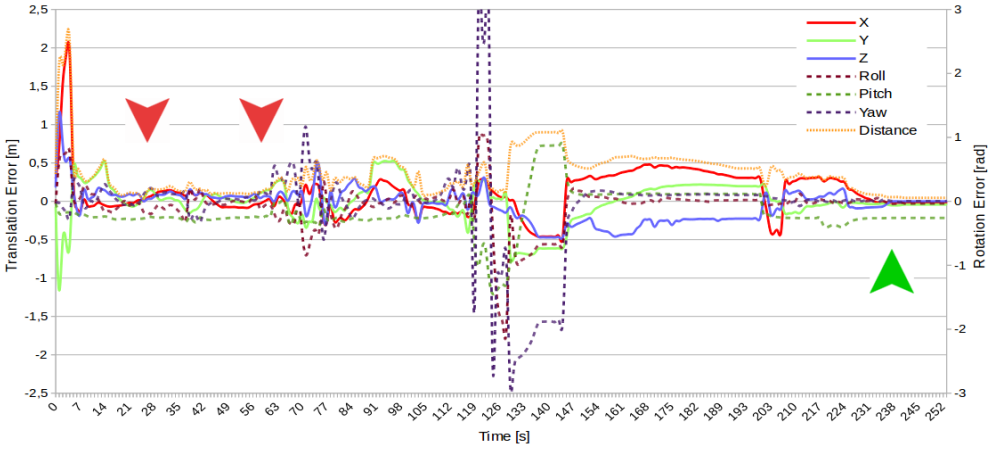
Figure 5.16: Successful pipe grasping sequence, from pipe approach (left) to pipe recovery (right). Central line shows underwater images which have been post-processed to enhance scene visibility. Bottom line frames have been recorded with a GoPro camera mounted on a small ROV (VideoRay)

Graphs in figure 5.17 refer to two of the five successful trials and show the trend of cartesian and angular errors between the grasping hand frame and the goal frame during the approach to the pipe. Error values are computed as a difference between pose of the arm end-effector and estimated pose of the pipe with respect to the vehicle frame. Values are thus affected by noisy and wrong estimations of the target object.

The graph in figure 5.17.a shows a successful approach to the pipe with decreasing angular and cartesian errors, until the pipe is actually grasped. Results of a second trial are presented in figure 5.17.b also highlighting some misses grasp opportunities.



(a)



(b)

Figure 5.17: Robot end-effector translation and rotation errors with respect to the grasping goal in two pipe-approaching trials. Green arrows approximately mark the time when the grasp took place and red ones the missed grasp opportunities.

Error values decrease for some time (in interval 25*s* - 60*s*) and the end-effector is close and aligned with the pipe, but the MARIS control system does not command grasp execution because of insufficient confidence. Only at about time 240*s* the grasp

is eventually performed with success.

The apparently low success rate of the vision-based grasping experiments is worth a deeper analysis, in particular on failure causes. Regarding pipe detection, a shallow pool reduces the distance between the target object laying on the floor and the vision system. Objects closer to the camera result bigger and parts of them are often outside the sensor field of view. The pipe detection algorithm requires part of object borders and at least one terminal. During the approach maneuver, movements of the vehicle and occlusions produced by arm and hand may significantly reduce the camera FoV, leading to unwanted loss of the pipe terminal. This problem, however, is only related to the shallow pool environment. In deeper waters, the AUV can easily keep the whole pipe in the cameras FoV by planning an appropriate approaching trajectory. Failures related to the described issue cannot be ascribed to the vision system, so the achieved result of 5 correct graspings can be considered a successful in-field validation of the proposed system.

During these long-lasting experiments, the vision system did not exhibit any hardware problem and showed a reliable behavior. The system was able to process images at roughly 7.5 Hz while at the same time images were stored and compressed with bzip algorithm. The file compression task saturated one of the eight available CPU logic cores. During the experiments the CPU temperature was logged and the resulting trend (fig. 5.18) showed a stable average value of 71°C, reached with a very short transient. This limit temperature is far lower than the CPU warning temperature of 82°C.

Table 5.1 summarizes data of the three experimental sessions performed to assess the underwater manipulation task within the MARIS project. Further experiments will be planned in sea waters with improved vehicle control algorithms, giving the opportunity to validate the vision system in several different scenarios.

Working in underwater environment is still a challenging task. Making every device ready for water leads to several difficulties that need to be addressed. Electronic components need to be sealed in appropriate canisters, generating heat dissipation

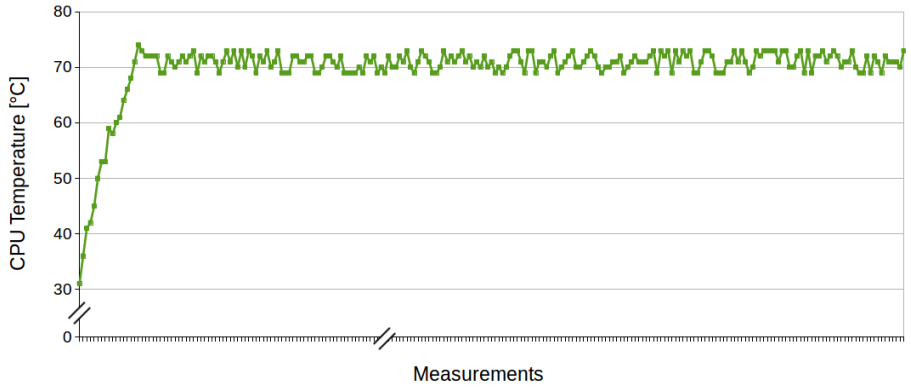


Figure 5.18: CPU temperature logged during the experiment. A stationary average value of 71°C is reached and maintained during the whole session.

Test Sessions	**March 2015**	**October 2015**	**December 2015**
Site	Pool	Pool	Pool
Water Depth	3.5 m		2.7 m
Water turbidity	Medium	High	Low
Weather	Cloudy to rainy	Sunny	Variable
Artificial illum.			YES
Testing hours	40	52	24
Collected data	150 Gb	78 Gb	228 Gb
Achieved results	Data collection	Out-of-water grasps	Underwater grasps

Table 5.1: MARIS experimental session data.

and maintenance issues. Furthermore, several components in the high end underwater vision system must still be developed as prototypes since adequate consumer products are not yet on the market. However, these are the same reasons that encourage research in this pioneering field, in order to increase the technical readiness level and robustness of developed prototypes and push them toward an industrial production.

Conclusion and future works

The main goal of this thesis was the development of a novel high performance underwater vision system for interventions. Although several steps forward have been made in technologies for ground, air and space robotics, underwater is still a marginally explored frontier, especially with regard to autonomous and interventions robotics. Autonomous vehicles, indeed, require a constantly updated representation of the environment and therefore a real-time processing of perceptual data. On the other hand, underwater intervention tasks are characterized by the same requirements in terms of perceptual accuracy as manipulation activities performed out of water.

The limitations of other sensors, like sonar and laser, mostly in term of poor resolution, variable perceptual range and high costs, have brought an increasing interest in computer vision solutions for underwater applications. However, moving solutions and approaches deployed in standard robotics, such as vision, to underwater scenarios is not easy due to peculiarities and limitations imposed by water. Of course, the applicability of computer vision in the underwater environment, like for any perceptual modality, will depend on the specific operational context, including the availability of natural or artificial light in the scene and the transparency status of water.

Although computer vision has already been exploited for underwater perception, most experiments have been performed in controlled environments, like pools, with

ideal conditions of illumination and visibility. Furthermore, most underwater computer vision applications are related to observational tasks, like seabed mapping or biological life monitoring, which do not require online processing nor the precision needed in manipulation and grasping activities.

Designing a computer vision system for underwater perception is therefore a challenging problem, especially if the final goal is to develop a reliable architecture able to cope with difficulties coming from real environments and different working scenarios. This thesis went through the analysis of the perception problem in real underwater environments with the purpose of understanding and facing some of the open issues that still prevent a larger diffusion of computer vision in submarine activities.

For this purpose, Chapter 1 has been devoted to an analysis of the state of art of the computer vision based solutions for underwater perception. The survey was focused on autonomous underwater vehicles for interventions and showed the limited maturity of these technologies and experiments: only few example prototypes have been developed in the last twenty years. The analysis has also considered monitoring and mapping applications with the aim of understanding the main issues related to underwater vision. This preliminary study has shown that the multiplicity of water and light conditions has led to different computational approaches in order to cope with unwanted reflections, color aberrations, image distortions and so on. The survey has shown also the lack of dataset publicly available: the dearth of shared experimental data, combined with the technical and logistic difficulties in performing submarine experiments, clearly prevents an extensive approach to underwater computer vision by the scientific community.

In order to cope with the lack of suitable datasets, in this thesis a low cost prototype has been designed and developed. The system has been conceived as a small, portable, self powered and self monitoring unit able to record image sequences of underwater environments and submerged objects for several hours. It was based on standard components like webcams, solid state drives, small form factor motherboard and low thermal-design-power CPU. The system, embedded and sealed in a transparent water resistant box, was equipped with a remote control unit and temperature and

humidity sensors for safe operations and self shutdown. The prototype was exploited in some testing campaigns performed in real environments (Garda Lake, Italy) and led to the collection of hours of image sequences.

The lessons learned in the prototype development, together with further considerations coming from state of art analysis and initial algorithmic investigations of the detection problem, led to the definition of technical requirements for the high end underwater computer vision system, that represents the main outcome of this thesis. In particular, the design of the high end computer vision system has addressed the limitation in computational power of ECUs generally used in underwater vehicles, that prevents a larger diffusion of computer vision for the specific operational context.

Although it should be desirable to exploit high end CPUs, power consumption constraints must be taken into account when working with autonomous vehicles in order not to excessively impact on mission time capabilities. Thermal balance of computational units, sealed in underwater containers, must be taken into account. Chapter 2 discussed these aspects and the design process of the final system. The philosophy behind the development was then the achievement of the best trade-off between computational power, energy power consumption and heat dissipation.

Image sensors have been chosen with high resolution and good frame rate to ensure enough precision in detection and pose estimation tasks. Due to the importance of color as a distinctive feature for region of interest detection in different water conditions, cameras are able to retrieve color images. The computational unit is made of multiple CPUs in a flexible architectural design which achieves some degree of failure awareness by means of realtime monitoring of the main sources of fault, characterizing underwater operations. The mechanical and thermal design of the canister allowed the deployment of a high performance Intel Core-i7 CPU, combined with auxiliary ATOM and ARM CPUs. The connection between systems and cameras exploits gigabit ethernet network interfaces with high bandwidth capabilities.

Both the first prototype and the final system have been tested in stressing conditions in order to validate the water resistance and to evaluate power consumption and adequate heat dissipation. Both developed systems have fulfilled the requirements. The high end vision system was able to support several processing pipelines,

described in Chapter 3, demonstrating its capability to adapt to different tasks and different scenarios. This result is a step forward in state of art of computer vision system for underwater autonomous vehicles.

The developed system has been then evaluated in real world environments. Chapter 4 has described a testing campaign that took place seawards of Portofino (Italy). In a real submarine scenario, at approximately 10m depth, a dataset of stereo images of submerged objects has been collected. The dataset has been publicly released and represents a unique and important contribution to the scientific community for research activities in the field of underwater object detection. Furthermore, collected images have been used as a test-bench for the development of pipe detection and pose estimation algorithms that could be computed on-board and in real-time on the proposed system.

Finally, the computer vision system has been integrated into the autonomous underwater vehicles developed by the MARIS consortium. The integration has comprised both a hardware stage, for components placement and adaptation, and a software stage for functional integration. Cameras mounting, orientation and distance between sensors have been adapted to the specific task the vehicle should perform. The canister has been placed on the bottom-centre of the vehicle payload for better weight balance and it has been connected to the robot LAN network and power supply. The vision system relies on the state-of-art robotic framework ROS, which simplified the communication between multiple agents and nodes.

Integration activities in the MARIS AUV demonstrated the flexibility of the developed system, which was able to communicate with the robotic arm equipping the vehicle, and return a grasping goal frame in less than one day of work. The system has then been used within the activities of the MARIS project and configured with an appropriate algorithmic suite to perform pipe detection and pose estimation. Several testing campaigns have been performed in a pool with the MARIS complete system and variable water turbidity, different weather conditions, both during day and night. The computer vision system, combined with the processing pipeline expressly deployed, has shown adequate functionality and performance, supporting the AUV in achieving the project goal, that is a reliable grasping of a submerged pipe.

Some research and development themes have been left open by this thesis. One of the major problems in experimenting with AUVs and ROVs is the limited network bandwidth between the vehicle and the remote monitoring station. Indeed, the umbilical cable carries a LAN network with bandwidth limited to few megabits, enough for command and feedback exchanges and console connection to onboard systems. This constraint generally prevents a smooth transmission of digital images collected by onboard cameras, leading to difficulties in remote monitoring. ARM board like Raspberry-Pi are now equipped with hardware H.264 encoders which are able to online encode and stream a video over network even with very low bandwidth. Furthermore, encoding images in standard formats like MPEG-4, makes the video stream playable over multiple platform like PCs and mobile devices. Another promising satellite application which exploits the image stream, could be the deployment of an intuitive HMI. Indeed, additional information regarding the status of the system and virtual representations of processing output, in a sort of Augmented Reality application, could be superimposed to images and delivered to the monitoring station through the vehicle umbilical cable. The proposed solution, could exploit open source multimedia frameworks like *gstreamer* [92], whose portability to ARM architectures has been verified. Developing an effective and intuitive HMI for underwater application is a still open and interesting research theme. In the last 15 years, very few endeavors [93, 94] have been focused on interaction with underwater vehicles, mainly due to technology limitations which could now be addressed.

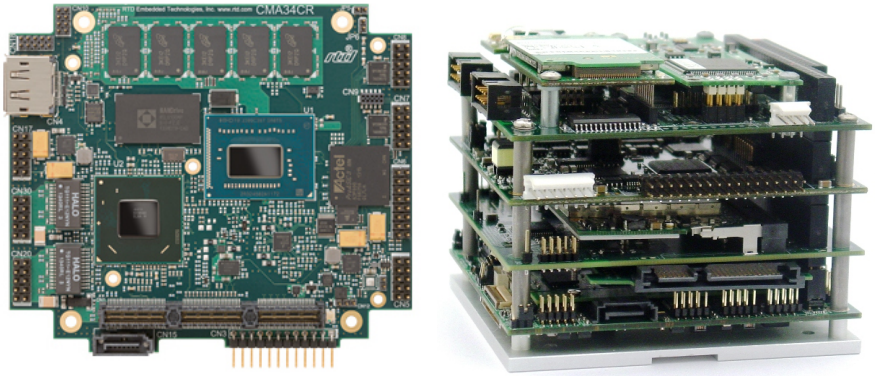
Finally, ARM-based devices recently released on market show highly increased performance with respect to first generation Raspberry-Pi boards. These computational units exploit up to 2GHz, quad-core ARM CPUs, DDR3 RAM and gigabit ethernet, while keeping the same thermal design power. These powerful devices could be exploited for camera images acquisition and, possibly, image distortion correction, which is the first step in the processing pipeline. Using the ARM board for camera frames acquisition is even more reasonable if images need to be encoded in a video stream, as suggested before. In fact, performing image acquisition and encoding on the same architecture aims at reducing the network load between vision ECUs.

A further advice is related to the main computational unit and the possibility of

increasing the CPU computational power. The system, as it was designed, is stable both underwater and in air but any change that increases the thermal design power of internal CPUs may lead to overheating. Furthermore, stability in air is guaranteed only if heavy load activities like image processing are suspended. Thus, a technology increasing robustness and safety of the system in critical situation is of interest. A promising solution could be the removal of air inside the canister and its substitution with mineral oil. Technology of liquid immersion cooling is already successfully applied to data centers with an increasing interest by large international players like 3M [95]. Mineral oil is dielectric so it is safe regarding electrical circuits and ensures a better transmission of heat from warm zones to cold canister walls, facilitating the heat dissipation. Of course, every cooling fan should be replaced with passive heatsinks. Furthermore, the removal of moving parts lead to better reliability, because of reduced risk of failures resulting in immediate overheating of electrical components. The trade-off of this solution regards maintenance, which becomes much more complicated. The vision system developed in this thesis has shown its reliability during the MARIS testing campaigns with no need of canister unsealing. For these reasons, placing the entire system in an oil bath could be a promising step forward, enabling higher CPU performance, increased heat dissipation and better reliability.

Possibility of future upgrades was one of the key requirements of the project, and it guided the components selection toward standard and off-the-shelves items. However, recently, some hardware producers [96, 97] of embedded systems started to release on market small form-factor boards, compliant to PC-104 standard, with high end CPUs (fig. 5.19.a). These systems offers advantages in terms of size, leading to underwater canisters of smaller diameter, extended temperature working range and modularity (fig. 5.19.b). The system proposed in this thesis, indeed, has been designed to equip middle size vehicles for interventions like “Artù”. However, there are families of smaller AUVs and underwater gliders with demanding tight constraints on payload size and weight. These vehicles are generally deployed in patrolling and monitoring applications and would benefit from availability of high performance vision systems. A complete re-design of the proposed system, based on newly released PC/104 components, can be tackled to extend the range of use at multiple kinds of

underwater vehicles. Trade-off regards overall costs of the system and possible difficulties in future upgrades and maintenance activities.



(a) (b)

Figure 5.19: Example of Intel Core-i7 PC-104 board, produced by RTD (a); representation of PC-104 stack concept (b).

Bibliography

[1] S.B. Williams, O.R. Pizarro, M.V. Jakuba, C.R. Johnson, N.S. Barrett, R.C. Babcock, G.A. Kendrick, P.D. Steinberg, A.J. Heyward, P.J. Doherty, I. Mahon, M. Johnson-Roberson, D. Steinberg, and A. Friedman. Monitoring of Benthic Reference Sites: Using an Autonomous Underwater Vehicle. *IEEE Robotics Automation Magazine*, 19(1):73–84, March 2012.

[2] D.A. Smale, G.A. Kendrick, E.S. Harvey, T.J. Langlois, R.K. Hovey, K.P. Van Niel, K.I. Waddington, L.M. Bellchambers, M.B. Pember, R.C. Babcock, et al. Regional-scale benthic monitoring for ecosystem-based fisheries management (EBFM) using an autonomous underwater vehicle (AUV). *ICES Journal of Marine Science: Journal du Conseil*, 69(6):1108–1118, 2012.

[3] D. L. Rudnick, R. E. Davis, C. C. Eriksen, D. M. Fratantoni, and Perry M. Underwater gliders for ocean research. *Marine Technology Society Journal*, vol. 38(n. 2):73–78, 2004.

[4] R.B. Wynn, V.A.I. Huvenne, T.P. Le Bas, B.J. Murton, D.P. Connelly, B.J. Bett, H.A. Ruhl, K.J. Morris, J. Peakall, D.R. Parsons, E.J. Sumner, S.E. Darby, R.M. Dorrell, and J.E. Hunt. Autonomous Underwater Vehicles (AUVs): Their past,

present and future contributions to the advancement of marine geoscience. *Marine Geology*, 352:451–468, 2014. 50th Anniversary Special Issue.

[5] D.R. Yoerger, A.M. Bradley, M. Jakuba, C.R. German, T. Shank, and M. Tivey. Autonomous and Remotely Operated Vehicle Technology for Hydrothermal Vent Discovery, Exploration, and Sampling. *Oceanography*, 20, March 2007.

[6] M. Ludvigsen, B. Sortland, G. Johnsen, and H. Singh. Applications of Geo-Referenced Underwater Photo Mosaics in Marine Biology and Archaeology. *Oceanography*, 20, December 2007.

[7] G. Casalino, M. Caccia, A. Caiti, G. Antonelli, G. Indiveri, C. Melchiorri, and S. Caselli. MARIS: a National Project on Marine Robotics for InterventionS. In *Proc. of the IEEE Mediterranean Conference on Control & Automation (MED)*, 2014.

[8] R. Conti, E. Meli, A. Ridolfi, and B. Allotta. An innovative decentralized strategy for I-AUVs cooperative manipulation tasks. *Robotics and Autonomous Systems*, 72:261–276, 2015.

[9] J. Pérez, J. Sales, A. Peñalver, J.J. Fernández, P.. Sanz, J. García, J. Martí, R. Marín, and D. Fornas. Robotic Manipulation Within the Underwater Mission Planning Context. In G. Carbone and F. Gomez-Bravo, editors, *Motion and Operation Planning of Robotic Systems*, volume 29 of *Mechanisms and Machine Science*, pages 495–522. Springer International Publishing, 2015.

[10] V. Rigaud, E. Coste-Maniere, M.J. Aldon, P. Probert, M. Perrier, P. Rives, D. Simon, D. Lang, J. Kiener, A. Casal, J. Amar, P. Dauchez, and M. Chantler. UNION: underwater intelligent operation and navigation. *IEEE Robotics & Automation Magazine*, 5(1):25–35, 1998.

[11] P.J. Sanz, P. Ridao, G. Oliver, G. Casalino, Y. Petillot, C. Silvestre, C. Melchiorri, and A. Turetta. TRIDENT An European project targeted to increase the autonomy levels for underwater intervention missions. In *Proc. of the MTS/IEEE OCEANS'13*, 2013.

[12] H.H. Wang, S.M. Rock, and M.J. Lee. Experiments in automatic retrieval of underwater objects with an AUV. In *Proc. of the MTS/IEEE OCEANS'95. Challenges of Our Changing Global Environment.*, volume 1, pages 366–373 vol.1, Oct 1995.

[13] S.K. Choi, G.Y. Takashige, and J. Yuh. Experimental study on an underwater robotic vehicle: ODIN. In *Proc. of the Symposium on Autonomous Underwater Vehicle Technology. AUV '94*, pages 79–84, Jul 1994.

[14] D.M. Lane, J.B.C. Davies, G. Casalino, G. Bartolini, G. Cannata, G. Veruggio, M. Canals, C. Smith, D.J. O'Brien, M. Pickett, G. Robinson, D. Jones, E. Scott, A. Ferrara, M. Coccoli, R. Bono, P. Virgili, R. Pallas, and E. Gracia. AMADEUS: advanced manipulation for deep underwater sampling. *IEEE Robotics Automation Magazine*, 4(4):34–45, Dec 1997.

[15] D.M. Lane, G. Bartolini, G. Cannata, G. Casalino, J.B.C. Davies, G. Veruggio, M. Canals, and C. Smith. Advanced manipulation for deep underwater sampling: the AMADEUS research project. In *Proc. of the IEEE International Conference on Control Applications*, volume 2, pages 1068–1073 vol.2, Sep 1998.

[16] B. Gilmour, G. Niccum, and T. O'Donnell. Field resident AUV systems - Chevron's long-term goal for AUV development. In *Proc. of the IEEE/OES Autonomous Underwater Vehicles (AUV)*, 2012.

[17] P. Ridao, M. Carreras, D. Ribas, P. Sanz, and G.O. Codina. Intervention AUVs: The Next Challenge. *IFAC World Congress. Keynote in the Marine Robotics Workshop*, 2014.

[18] S.C. Yu, T.W. Kim, A. Asada, S. Weatherwax, B. Collins, and J. Yuh. Development of High-Resolution Acoustic Camera based Real-Time Object Recognition System by using Autonomous Underwater Vehicles. In *Proc. of the MTS/IEEE OCEANS'06*, pages 1–6, 2006.

[19] P. Jonsson, I. Sillitoe, B. Dushaw, J. Nystuen, and J. Heltne. Observing using sound and light: a short review of underwater acoustic and video-based methods. *Ocean Science Discussions*, 6(1):819–870, 2009.

[20] A. Gordon. Use of laser scanning system on mobile underwater platforms. In *Proceedings of the Symposium on Autonomous Underwater Vehicle Technology, AUV'92.*, pages 202–205. IEEE, 1992.

[21] P.J. Sanz, A. Penalver, J. Sales, D. Fornas, J.J. Fernandez, J. Perez, and J.A. Bernabe. GRASPER: A Multisensory Based Manipulation System for Underwater Operations. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 1–9, 2013.

[22] T. Nicosevici, N. Gracias, S. Negahdaripour, and R. Garcia. Efficient three-dimensional scene modeling and mosaicing. *Journal of Field Robotics*, 26(10), 2009.

[23] R. Eustice, H. Singh, J. Leonard, M. Walter, and R. Ballard. Visually Navigating the RMS Titanic with SLAM Information Filters. In *Proc. of Robotics: Science and Systems*, Cambridge, USA, June 2005.

[24] T. Rahman, J. Anderson, P. Winger, and N. Krouglicof. Calibration of an underwater stereoscopic vision system. In *Proc. of the MTS/IEEE OCEANS'13*, pages 1–6, Sept 2013.

[25] D.C. Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971.

[26] D. Gonzalez-Aguilera, J. Gomez-Lahoz, and P. Rodriguez-Gonzalvez. An Automatic Approach for Radial Lens Distortion Correction From a Single Image. *IEEE Sensors Journal*, 11(4):956–965, April 2011.

[27] R. Garcia and N. Gracias. Detection of interest points in turbid underwater images. In *Proc. of the MTS/IEEE OCEANS'11*, pages 1–9, 2011.

[28] D. Lee, G. Kim, D. Kim, H. Myung, and H.-T. Choi. Vision-based object detection and tracking for autonomous navigation of underwater robots. *Ocean Engineering*, 48:59–68, 2012.

[29] S. Bazeille, I. Quidou, and L. Jaulin. Color-based underwater object recognition using water light attenuation. *Intelligent Service Robotics*, 5:109–118, 2012.

[30] M. Prats, J.C. Garcia, S. Wirth, D. Ribas, P.J. Sanz, P. Ridao, N. Gracias, and G. Oliver. Multipurpose autonomous underwater intervention: A systems integration perspective. In *Proc. of the IEEE Mediterranean Conference on Control & Automation (MED)*, pages 1379–1384, July 2012.

[31] J. Aulinas, M. Carreras, X. Llado, J. Salvi, R. Garcia, R. Prados, and Y.R. Petillot. Feature extraction for underwater visual SLAM. In *Proc. of the MTS/IEEE OCEANS'11*, pages 1–7, 2011.

[32] A. Olmos, E. Trucco, and D. Lane. Automatic man-made object detection with intensity cameras. In *Proc. of the MTS/IEEE OCEANS'02*, volume 3, pages 1555–1561, Oct 2002.

[33] J.P. Queiroz-Neto, R. Carceroni, W. Barros, and M. Campos. Underwater stereo. In *Proceedings of the 17th Brazilian Symposium on Computer Graphics and Image Processing*, pages 170–177, 2004.

[34] V. Brandou, A.-G. Allais, M. Perrier, E. Malis, P. Rives, J. Sarrazin, and P.-M. Sarradin. 3D Reconstruction of Natural Underwater Scenes Using the Stereovision System IRIS. In *Proc. of the MTS/IEEE OCEANS'07*, pages 1–6, 2007.

[35] R. Campos, R. Garcia, and T. Nicosevici. Surface reconstruction methods for the recovery of 3D models from underwater interest areas. In *Proc. of the MTS/IEEE OCEANS'11*, pages 1–10, 2011.

[36] A. Leone, G. Diraco, and C. Distante. Stereoscopic System for 3-D Seabed Mosaic Reconstruction. In *Proc. of the IEEE International Conference on Image Processing (ICIP)*, volume 2, pages 541–544, Sep. 2007.

[37] J. Fernandez, M. Prats, P.J. Sanz, J. C. Garcia, R. Marin, M. Robinson, D. Ribas, and P. Ridao. Grasping for the Seabed: Developing a New Underwater Robot Arm for Shallow-Water Intervention. *IEEE Robotics Automation Magazine*, 20(4):121–130, Dec 2013.

[38] M. Prats, J.C. Garcia, J.J. Fernandez, R. Marin, and P.J. Sanz. Advances in the specification and execution of underwater autonomous manipulation tasks. In *Proc. of the MTS/IEEE OCEANS'11*, pages 1–5, June 2011.

[39] M. Prats, D. Ribas, N. Palomeras, J. García, V. Nannen, S. Wirth, J. Fernández, J. Beltrán, R. Campos, P. Ridao, P. Sanz, G. Oliver, M. Carreras, N. Gracias, R. Marín, and A. Ortiz. Reconfigurable AUV for intervention missions: a case study on underwater object recovery. *Intelligent Service Robotics*, 5(1):19–31, 2012.

[40] M. Massot-Campos, G. Oliver-Codina, H. Kemal, Y. Petillot, and F. Bonin-Font. Structured light and stereo vision for underwater 3D reconstruction. In *Proc. of the MTS/IEEE OCEANS'15*, 2015.

[41] F. Bonin-Font, A. Cosic, P.L. Negre, M. Solbach, and G. Oliver. Stereo SLAM for robust dense 3D reconstruction of underwater environments. In *Proc. of the MTS/IEEE OCEANS'15*, pages 1–6, May 2015.

[42] B. Allen, R. Stokey, T. Austin, N. Forrester, R. Goldsborough, M. Purcell, and C. von Alt. REMUS: a small, low cost AUV; system description, field trials and performance results. In *Proc. of the MTS/IEEE OCEANS'97*, volume 2, pages 994–1000 vol.2, Oct 1997.

[43] R.J. Komerska and S.G. Chappell. A Simulation Environment for Testing and Evaluating Multiple Cooperating Solar-powered AUVs. In *Proc. of the MTS/IEEE OCEANS'06*, 2006.

[44] Benchmark comparison between Core 2 Duo and Pentium CPUs. `http://www.cpu-world.com/benchmarks/browse/188_10, 188_11,447_20/?c_test=3`.

[45] J. Albiez, A. Duda, M. Fritsche, F. Rehrmann, and F. Kirchner. CSurvey—An autonomous optical inspection head for AUVs. *Robotics and Autonomous Systems*, 67:72–79, 2015.

[46] M. Jacobi. Autonomous inspection of underwater structures. *Robotics and Autonomous Systems*, 67:80–86, 2015.

[47] M. Novi, F. Pacini, G. Paoli, G. Saviozzi, G. Ballini, A. Caiti, F. Di Corato, D. Fenucci, S. Grechi, R. Reggiannini, and F. Carrai. Project V-FIDES: An innovative, multi purpose, autonomous underwater platform. In *Proc. of the MTS/IEEE OCEANS'15*, 2015.

[48] N. Stilinovic, D. Nad, and N. Miskovic. AUV for diver assistance and safety - Design and implementation. In *Proc. of the MTS/IEEE OCEANS'15*, pages 1–4, May 2015.

[49] Y. Nishida, J. Kojima, Y. Ito, K. Tamura, H. Sugimatsu, Kangsoo Kim, T. Sudo, and T. Ura. Development of an autonomous buoy system for AUV. In *Proc. of the MTS/IEEE OCEANS'15*, 2015.

[50] A. Marouchos, B. Muir, R. Babcock, and M. Dunbabin. A shallow water AUV for benthic and water column observations. In *Proc. of the MTS/IEEE OCEANS'15*, pages 1–7, May 2015.

[51] J. Santos-Victor and J. Sentieiro. The role of vision for underwater vehicles. In *Proceedings of the Symposium on Autonomous Underwater Vehicle Technology, AUV '94*, pages 28–35, Jul 1994.

[52] K. Konolige. Small Vision Systems: Hardware and Implementation. In Yoshiaki Shirai and Shigeo Hirose, editors, *Robotics Research*, pages 203–212. Springer London, 1998.

[53] A. Rowe, C. Rosenberg, and I. Nourbakhsh. A low cost embedded color vision system. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 208–213, 2002.

[54] A. Rowe, C. Rosenberg, and I. Nourbakhsh. A Second Generation Low Cost Embedded Color Vision System. In *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR Workshops.*, pages 136–136, June 2005.

[55] F. Oleari, D. Lodi Rizzini, and S. Caselli. A Low-Cost Stereo System for 3D Object Recognition. In *Proc. of the Inernational Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 127–132, Cluj-Napoca, Romania, Sept 2013.

[56] D. Valeriani, F. Oleari, D. Lodi Rizzini, and S. Caselli. A Viewpoint Planning and Navigation Algorithm for Mobile Robots using Depth Images. In *Proc. of Australasian Conference on Robotics and Automation (ACRA)*, Sept 2013.

[57] R. Szabó and A. Gontean. Full 3D Robotic Arm Control with Stereo Cameras Made in LabVIEW. In *Federated Conference on Computer Science and Information Systems (FedCSIS)*, pages 37–42, 2013.

[58] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng. ROS: an open-source Robot Operating System. In *ICRA Workshop on Open Source Software*, 2009.

[59] W. Meeussen, M. Wise, S. Glaser, S. Chitta, C. McGann, P. Mihelich, E. Marder-Eppstein, M. Muja, V. Eruhimov, T. Foote, et al. Autonomous door opening and plugging in with a personal robot. In *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pages 729–736, 2010.

[60] N. Hudson, T. Howard, J. Ma, A. Jain, M. Bajracharya, S. Myint, C. Kuo, L. Matthies, P. Backes, P. Hebert, et al. End-to-end dexterous manipulation with deliberate interactive estimation. In *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pages 2371–2378. IEEE, 2012.

[61] R. Eustice, H. Brown, and A. Kim. An overview of AUV algorithms research and testbed at the University of Michigan. In *Proc. of IEEE/OES Autonomous Underwater Vehicles, (AUV'08)*, pages 1–9. IEEE, 2008.

[62] H. C. Brown, A. Kim, and R. M. Eustice. An overview of autonomous underwater vehicle research and testbed at PeRL. *Marine Technology Society Journal*, 43(2):33–47, 2009.

[63] C. Roman, G. Inglis, and J. Rutter. Application of structured light imaging for high resolution mapping of underwater archaeological sites. In *Proc. of the MTS/IEEE OCEANS'10*, pages 1–9. IEEE, 2010.

[64] Lawrence Berkeley National Laboratory. iperf3: A TCP, UDP, and SCTP network bandwidth measurement tool. https://github.com/esnet/iperf, March 2015.

[65] A. Tirumala, F. Qin, J. Dugan, J. Ferguson, and k. Gibbs. Iperf: The TCP/UDP bandwidth measurement tool. *https://iperf.fr/*, 2005.

[66] G. Conte, S. Zanoli, A.M. Perdon, G. Tascini, and P. Zingaretti. Automatic analysis of visual data in submarine pipeline inspection. In *Proc. of the MTS/IEEE OCEANS'96*, volume 3, pages 1213–1219 vol.3, Sep 1996.

[67] P. Rives and J.-J. Borrelly. Underwater pipe inspection task using visual servoing techniques. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 1, pages 63–68 vol.1, Sep 1997.

[68] R. Bradbeer, S. Harrold, F. Nickols, and L.F. Yeung. An underwater robot for pipe inspection. In *Proc. of Mechatronics and Machine Vision in Practice*, pages 152–156. IEEE, 1997.

[69] A. Shukla and H. Karki. A review of robotics in onshore oil-gas industry. In *Proc. of the IEEE International Conference on Mechatronics and Automation (ICMA)*, pages 1153–1160. IEEE, Aug 2013.

[70] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, Nov 2000.

[71] J.M. Lavest, G. Rives, and J.T. Lapresté. Underwater camera calibration. In *Computer Vision—ECCV 2000*, pages 654–668. Springer, 2000.

[72] A. Agrawal, S. Ramalingam, Y. Taguchi, and V. Chari. A theory of multi-layer flat refractive geometry. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3346–3353. IEEE, 2012.

[73] F. Oleari, F. Kallasi, D. Lodi Rizzini, J. Aleotti, and S. Caselli. An Underwater Stereo Vision System: from Design to Deployment and Dataset Acquisition. In *Proc. of the MTS/IEEE OCEANS'15*, pages 1–5, 2015.

[74] Colorimetry – Part 4: CIE 1976 L*a*b* Colour space, ISO 11664–4:2008 (CIE S 014–4/E:2007) CIE, Vienna, 2006.

[75] S.M. Pizer, E.P. Amburn, J.D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J.B. Zimmerman, and K. Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39(3):355–368, 1987.

[76] D. Lodi Rizzini, F. Kallasi, F. Oleari, and S. Caselli. Investigation of Vision-based Underwater Object Detection with Multiple Datasets. *International Journal of Advanced Robotic Systems (IJARS)*, 12(77):1–13, May 2015.

[77] C. Ancuti, C.O. Ancuti, T. Haber, and P. Bekaert. Enhancing underwater images and videos by fusion. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 81–88, 2012.

[78] M. Ebner. *Color constancy*, volume 6. John Wiley & Sons, 2007.

[79] R.O. Duda, P. E Hart, and D.G. Stork. *Pattern classification*. John Wiley & Sons, 2012.

[80] R. Ugolotti, Y. S.G. Nashed, P. Mesejo, S. Ivekovi, L. Mussi, and S. Cagnoni. Particle Swarm Optimization and Differential Evolution for model-based object detection. *Applied Soft Computing*, 13(6):3092–3105, 2013.

[81] F. Oleari, F. Kallasi, D. Lodi Rizzini, J. Aleotti, and S. Caselli. Performance Evaluation of a Low-Cost Stereo Vision System for Underwater Object Detection. In *IFAC World Congress.*, 2014.

[82] F. Kallasi, D. Lodi Rizzini, F. Oleari, and J. Aleotti. Computer Vision in Underwater Environments: a Multiscale Graph Segmentation Approach. In *Proc. of the MTS/IEEE OCEANS'15*, pages 1–6, 2015.

[83] H. Hirschmuller. Stereo Processing by Semiglobal Matching and Mutual Information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, Feb 2008.

[84] S. Birchfield and C. Tomasi. A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(4):401–406, Apr 1998.

[85] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[86] G. Casalino, E. Simetti, N. Manerikar, A. Sperinde, S. Torelli, and F. Wanderlingh. Cooperative Underwater Manipulation Systems: Control Developments within the MARIS project. *IFAC-PapersOnLine*, 48(2):1–7, 2015. 4th IFAC Workshop on Navigation, Guidance and Controlof Underwater Vehicles.

[87] J. Evans, P. Redmond, C. Plakas, K. Hamilton, and D. Lane. Autonomous docking for Intervention-AUVs using sonar and video-based real-time 3D pose estimation. In *Proc. of the MTS/IEEE OCEANS'03*, volume 4, pages 2201–2210, Sept 2003.

[88] G. Marani, S.K. Choi, and J. Yuh. Underwater autonomous manipulation for intervention missions AUVs. *Ocean Engineering*, 36(1):15–23, 2009. Autonomous Underwater Vehicles.

[89] J.R. Bemfica, C. Melchiorri, L. Moriello, G. Palli, and U. Scarcia. A three-fingered cable-driven gripper for underwater applications. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[90] ROS Camera Calibration Tool. `http://http://wiki.ros.org/camera_calibration`.

[91] E. Simetti, G. Casalino, S. Torelli, A. Sperinde, and A. Turetta. Floating Underwater Manipulation: Developed Control Methodology and Experimental Validation within the TRIDENT Project. *Journal of Field Robotics*, 31(3):364–385, 2014.

[92] G. Sundari, T. Bernatin, and P. Somani. H.264 encoder using Gstreamer. In *Proc. of the International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, pages 1–4, March 2015.

[93] P. Ridao, J. Battle, J. Amat, and M. Carreras. A distributed environment for virtual and/or real experiments for underwater robots. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, volume 4, pages 3250–3255, 2001.

[94] G.B. Meo. The HMI of an experimental underwater vehicle for archeological survey, inspection and remote touring of important submarine sites and finds. In *Proc. of the MTS/IEEE OCEANS'04*, volume 2, pages 818–821 Vol.2, Nov 2004.

[95] Two-Phase Immersion Cooling. `http://www.3m.com/3M/en_US/novec/products/engineered-fluids/immersion-cooling/`.

[96] RTD Embedded Technologies, Inc. `http://www.rtd.com`.

[97] ADL Embedded Solutions. `http://www.adl-usa.com`.

Acknowledgements

It was an amazing and unbelievably short period and now it is coming to the end. During these years, I have received many precious gifts that made me grow personally and professionally. Everything started with my advisor, prof. Stefano Caselli. He believed in me when I was a student and he guided me through several projects with patience and willingness. Some further events have really changed my life.
Thank you Stefano.

My bosses in Elettric80, Mimmo and Vittorio, are two open minded and modern people, without their support nothing of this would have been possible.

I would like to thank my colleagues of our research group for having shared this adventure with me. Dario and Jacopo have always pushed my lazy mind to new horizons. Fabjan, Giorgio and Riccardo are great programmers, smart roboticists and, above all, good friends.

I do not want to forget my colleagues in Elettric80 that saw me suddenly disappear from office for several days: I was simply playing with yellow submarines!
Massi, your friendship is simply a privilege.

I am grateful to my family, for never having forgotten to remind me the madness needed to go on with a Ph.D. Looking in my grandparents' eyes all the astonishment and pride was the best gift I have ever received.

Erica, Ilaria, Giulia and Eleonora, you fill my days with endless wonder.
My heart is definitely with you.

I am close to the soul of the braves that struggle against life surprises, smiling every day. Thanks for the precious inspiration.