

PRONTO: a system for mobile robot navigation via CAD-model guidance

Antonios Gasteratos^{a,*}, Carlos Beltran^b, Giorgio Metta^c, Giulio Sandini^b

^aLaboratory of Electronics, Section of Electronics and Information Systems Technology, Department of Electrical and Computer Engineering, Democritus University of Thrace, GR-671 00 Xanthi, Hellas, Greece

^bLaboratory for Integrated Advanced Robotics, Department of Communication, Computer and System Sciences, University of Genoa, Viale Causa 13, I-16145 Genoa, Italy

^cMIT—Artificial Intelligence Laboratory, Humanoid Robotics Group, 200 Technology Square, MIT Building NE43, Cambridge, MA 02139, USA

Received 3 January 2001; revised 22 October 2001; accepted 29 October 2001

Abstract

This paper presents a vision system that finds and measures the location of 3D structures with respect to a CAD-model. The integration of a CAD-model to visual measurement and direct feedback of measurement results to the CAD is a key aspect. For the extraction of basic visual cues, independent and complementary modules are envisaged. The goal is that of navigating a legged robot into a ship structure using the pose estimated from visual landmarks extracted from the CAD-model. These are tracked in real-time by the vision system and are matched to the CAD-model. For the implementation of the vision system, commercial off-the-shelf parts were used along with a custom designed robot stereo head. The communication with other modules is done using simple ASCII commands on a fiber channel network with standard TCP/IP protocol. This allows easy debugging, straightforward development of applications software with little effort, and very high data rates. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Robot navigation; Computer aided design; Distributed systems; Active vision

1. Introduction

Over the last 20 years, the importance of computer-aided-design (CAD) has changed from computer aided drafting and designing of 3D objects to a key element of the manufacturing process. A CAD-system provides a computer representation of the geometry of the product and its characteristics, and it is the bond between all the industrial processes that follow the design such as construction, manufacturing, testing and control. Industries that use CAD systems in their working areas need a feedback to enable a comparison of designed and manufactured structures. Artificial vision driven by CAD information is an effective tool to establish this link [1]. The most common task where vision is employed in such sense is quality control, i.e. the matching of the image of the product to its model, in order to locate possible defects or errors during the manufacturing process [2–7]. However, in building and inspecting large structures such as ship bodies, the autonomy of a robotic

vehicle might be useful in many cases. Furthermore, the vision tool can be used for the task of dimensional measurements of parts. Using position-predefined landmarks usually performs navigating mobile robots in indoor environments. Autonomous robots hold a geometric map of the environment and use landmarks, such as walls or pillars, for navigation [8–11]. The robot assumes a rough position and matches the landmarks of its map to those detected by the vision system. The main problems are the difficulties due to changes and/or clutter of the background and the high computational demands. For example, a space application where the background is dark and the object consists of different surfaces with different characteristics requires dedicated hardware to run at frame rate [12]. Probably the most successful system that uses vision to control a mechanism is Fuerst's and Dickmanns' automatic car and air-vehicle approach using dynamic vision [13]. This integrates the dynamic aspects of a continuously operating system and image data to update the model description of the world.

In this paper, we present an active vision system that finds and measures the location of 3D structures with respect to a CAD-model named hereafter PRONTO. PRONTO is a subsystem of an integrated multi-computer distributed application called ROBVISION. The key aspect of ROBVISION is the integration of a CAD-model with visual

* Corresponding author. Address: Laboratory of Electronics, Department of Electrical and Computer Engineering, Democritus, University of Thrace, GR-671 00 Xanthi, Greece.

E-mail addresses: antonis@lira.dist.unige.it (A. Gasteratos), cbeltran@lira.dist.unige.it (C. Beltran), pasa@lira.dist.unige.it (G. Metta), sandini@lira.dist.unige.it (G. Sandini).

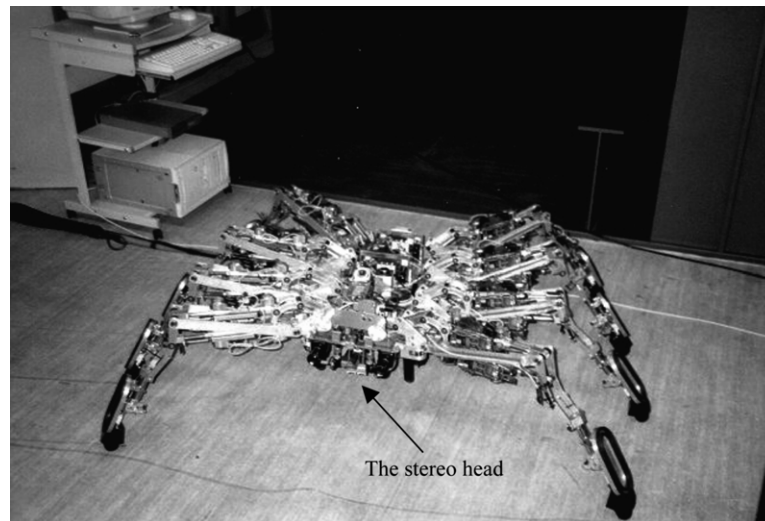


Fig. 1. The robot of the ROBVISION project (ROBUG) with the stereo robot head of PRONTO system. The robot possesses walking and climbing capabilities, which allows it to navigate into vessel parts with an average velocity of 3 cm/s.

measurements and direct feedback of the measurement results [14]. Accurate 3D measurements are carried out by the PRONTO vision system [15,16]. These, together with the 2D measurements of another vision system (V4R [17]), are used in order to determine the pose of the robot and, eventually, to guide the robotic vehicle [18]. The target application is to deliver work packages for inspection, welding and other tasks into the body of a large vessel during construction. Robustness is seen as the critical issue to obtain automated behavior and good accuracy of measurements. It is tackled by exploiting the redundancy and richness of image cues together with knowledge available from existing CAD-models in terms of object forms and locations. Weighting several visual cues and utilizing the knowledge acquired from the model in the integration process increases also reliability. Cues from the images and the model are exploited and integrated both at a global and at local level. The modularity of the toolbox is the basis for integrating the acquisition of visual information with tools of the CAD, the control and the engineering processes. The rest of the paper is organized as follows: Section 2 provides the description of the overall system PRONTO is part of; a

brief description of the feature extraction procedure from the CAD-model is given as well. The hardware of PRONTO, including the high accuracy stereo head, is provided in Section 3. Section 4 describes the software architecture, i.e. the vision and the communication subsystems, and finally, a discussion of the results is made in Section 5.

2. The ROBVISION project

The ROBVISION project stands for ‘Robust Vision for Sensing in Industrial Operations and Needs’. Relating to the task of navigating a robot into ship sections, the basic idea is to use vision in order to locate a feature in the image, to correlate this feature with that in the model, and to determine the pose of the robot. The key idea is to build a generic vision system for finding, measuring and locating objects. The only input is a CAD-model, as it is commonly used in industry. From this model, a list of features is extracted automatically and delivered to the vision systems. The features are then found on the image plane. Using the

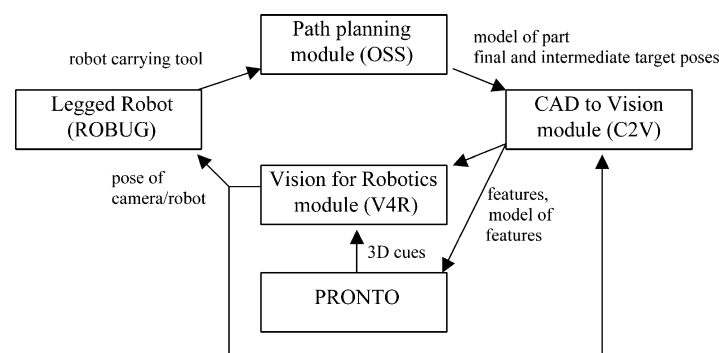


Fig. 2. Principal approach of the ROBVISION project including the major roles of each system/module.

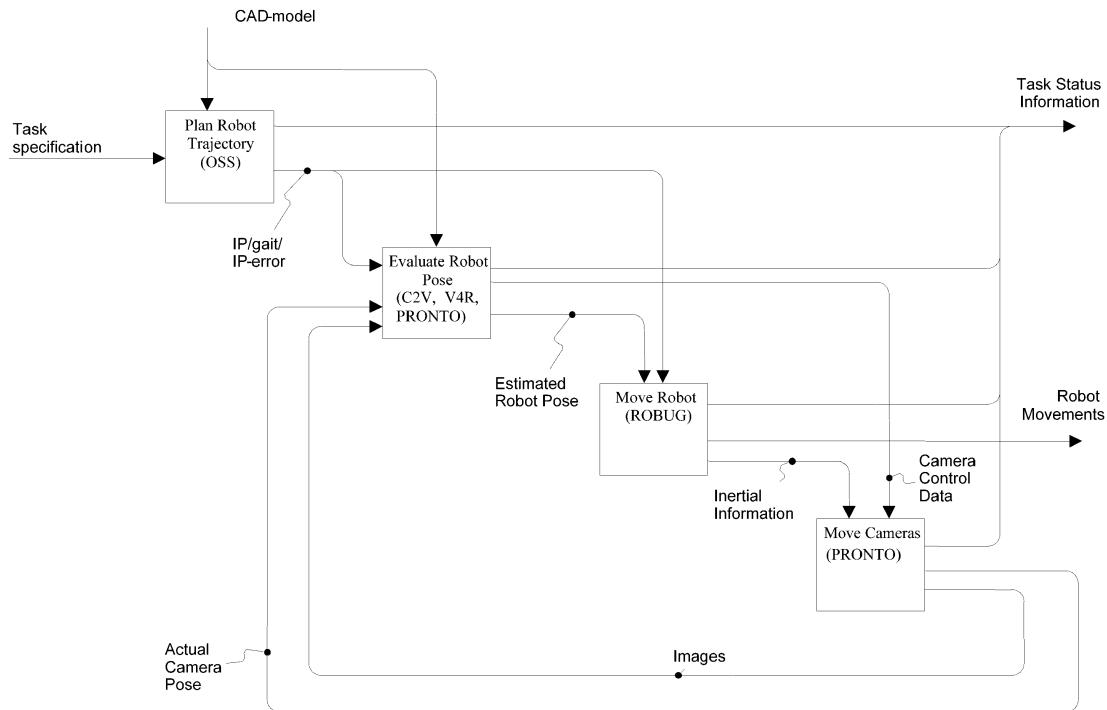


Fig. 3. Principal functional blocks of the ROBVISION project. Each block represents a main state of the system.

image data, the object can be located and the robot's pose can be calculated. The guidelines in this project were:

- vision: in particular measurement, tracking, and robustness;
- integration of modules: data fusion and similar techniques; and
- CAD systems and models and the respective architectures.

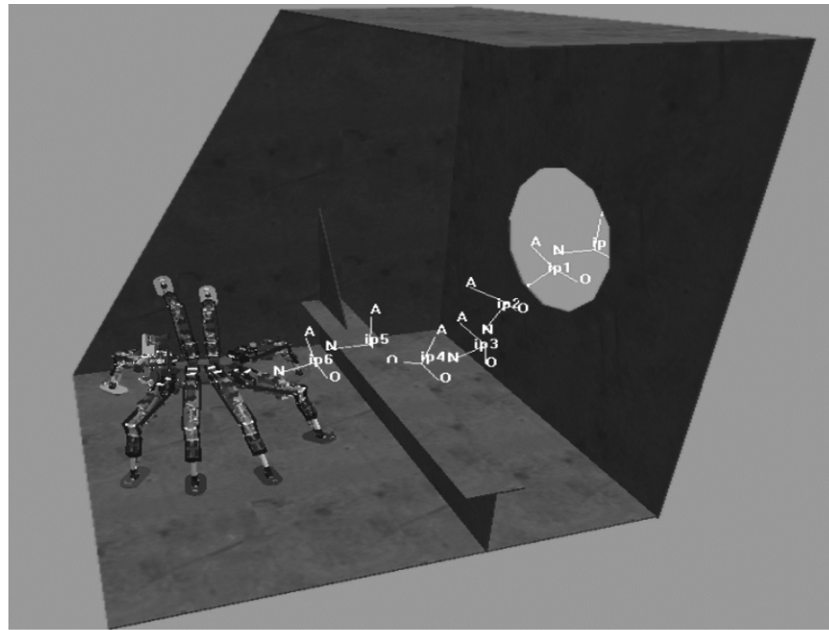
In detail, the role of the different subsystems in the ROBVISION architecture is as follows: a system serves for the off-line planning of the robot's path (OSS). It includes the walking robot (ROBUG) shown in Fig. 1 and its control. Competence in basic vision techniques is granted by the PRONTO system. Visual cues and tracking expertise are provided by the system V4R. The CAD to Vision (C2V) system contributes the modeling expertise (see for instance Fig. 2). More specifically OSS provides the model of the ship section that the robot should enter. A human operator at OSS defines the target and eventual intermediate poses needed to navigate the robot, i.e. the path planning. The system C2V automatically extracts features from the CAD-model and makes them available to the vision systems V4R and PRONTO. These determine the pose of the robot relative to the ship section. The pose is then sent to the robot in order to correct errors while following the planned trajectory towards the target. The potential uses for such a tool are quite diverse. The principal capability is to use a CAD-model to find features in images and to

return the position and orientation measured back into the CAD-model. Fig. 2 outlines the idea and methodology of the approach.

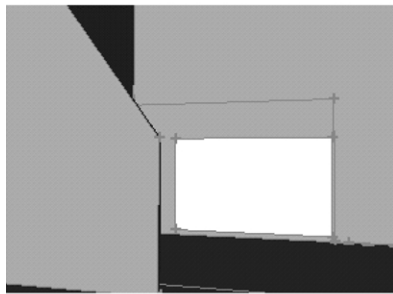
Fig. 3 shows the basic functional blocks and the commands between these blocks. The CAD-model describes a ship section. The coordination of the different subsystems is done by an extra system called Supervisor and the status of all components is reported through the Supervisor to a Human Machine Interface (HMI). It should be noted that the system design is flexible for usage with other system modules. For example, the pose feedback from the vision system can be used for walking robots as well as other mobile platforms. Any extra module can be easily connected to the existing components. Information distributed from one subsystem to the others, such as images, robot pose, camera pose, etc. can be easily passed to any extra subsystem. Each ROBVISION subsystem runs on its own computer that is interconnected with others via an optical network using the standard Transmission Control Protocol/Internet Protocol (TCP/IP protocol). The commands are sent in ASCII format. These allow easy human inspection (e.g. easy debugging) and, moreover once specified other modules can replicate them.

2.1. Operating the vision systems using the CAD-model

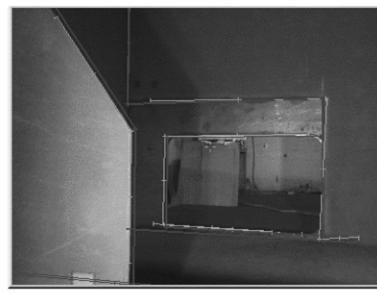
The C2V subsystem concentrates on determining the reference features, which are defined as a coherent set of: (i) an intermediate point, (ii) the corresponding robust features that should be seen in a view and (iii) a robot



(a)



(b)



(c)

Fig. 4. (a) The robot moves towards a predefined path, planned offline; (b) for a certain viewpoint a view is generated by the CAD-model, as it would be seen by the vision systems; (c) the CAD-model is overlapped onto the corresponding image to calculate the pose of the robot.

gait. The features are geometric references determined from the CAD-model of the work piece, i.e. surface boundaries and intersections, represented as lines, circles, half circles and junctions (intersections of line segments). All the geometrical features are provided relatively to a predetermined world coordinate system. The planning subsystem by OSS provides the CAD-model of the environment with some intermediate target poses and corresponding gaits (Fig. 4a). C2V delivers features in the 3D description available in the CAD-model (Fig. 4b). The vision systems obtain this model and perform the projections onto the image plane (Fig. 4c).

The loop between the vision systems and C2V is as follows:

1. The robot has been moved and is located in an unknown position.
2. From the knowledge of the former pose PRONTO calculates camera pose, which is passed to the C2V subsystem.
3. C2V provides 3D geometric features detected in the CAD-model using the present view of the cameras (which we will call 'viewpoint' hereafter).
4. If the features or images are recognized by the vision systems, the viewing direction is subsequently maintained while the robot moves towards its target. Go to 2 and continue the loop from there.
5. If the features or images are not recognized by the vision subsystem, the cameras move to another viewing direction. This new viewpoint is suggested by the C2V subsystem evaluating areas that are rich with features to render the task of the vision system easier. Go to 3 and continue the loop from there.

3. The hardware architecture

The PRONTO system comprises several tasks that in

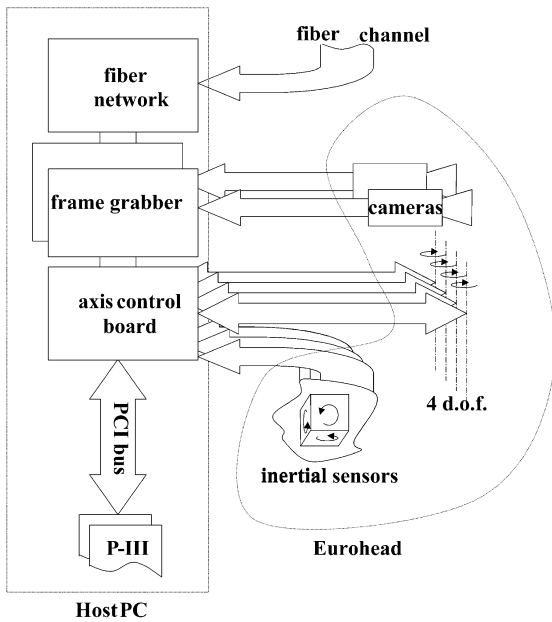
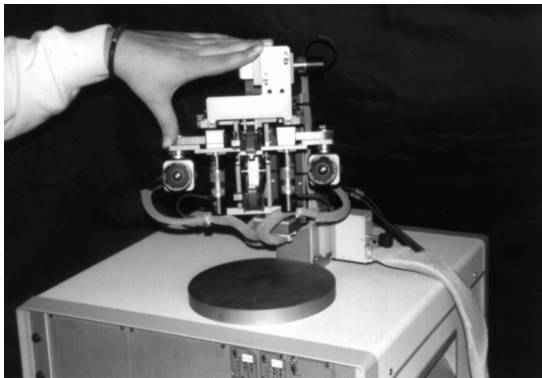


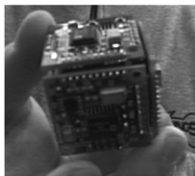
Fig. 5. A block diagram of the principal parts of the PRONTO hardware.

brief are:

- head control;
- head calibration;
- inertial sensor data acquisition, distribution to other systems and use for image stabilization (see description later on);
- image acquisition and delivery to the other systems;
- image processing;



(a)



(b)

Fig. 6. (a) The Eurohead and (b) the inertial sensors system.

- 3D measurements; and
- communications and synchronization with V4R vision system;

In order to accomplish these tasks, PRONTO is based on a two-processors computer architecture and a high-resolution stereo head, named the 'Eurohead'. An abstract view of the hardware of the system is depicted in Fig. 5. For the simultaneous acquisition of the stereo images, the system is equipped with a pair of frame grabbers. An axis control board, capable of controlling up to eight axes, is utilized to handle the four degrees of freedom (dof) of the head. Moreover, the axis control board comprises analog to digital converters (ADCs), which enables the reading of the signals of external sensors. The Eurohead is shown in Fig. 6a. It has been designed and implemented to be an accurate vision-based measuring device. For the control of its four dof, i.e. the pan, the tilt and the two camera pans (vergence), four DC motors with harmonic drive reduction gear are used. These actuators have been chosen according to their mechanical characteristics. Due to their harmonic drive gearing, they provide high reduction ratios in a single stage, zero backlash and, high precision. Teeth belts have been used for the movement transmission from the actuator to the joints. This gives better results in term of accuracy than usual gearing transmission. The specifications of the head are summarized in Table 1. The head was carefully designed in order to be compact, portable and low weight. Its dimensions are 209 mm × 222 mm × 185 mm and its weight is about 3 kg. It carries standard CCD cameras of 752 × 582 resolution and 4.8 mm lenses. Moreover, the head employs three piezoelectric gyroscopes [19]. Each sensor along with the driving and filtering electronics is mounted on a card of about 3.5 cm in size. Three cards are arranged so that they form a small modular cube as the one shown in Fig. 6b. In this manner, the sensing elements are able to measure motion along three orthogonal axes. The cube is mounted on the top of the head, so that it monitors the external disturbances that are subjected to the head. The signals end up to the ADCs of the axis control board and are used to compensate the head movements due to such external disturbances.

In order to guarantee real-time communication, image transmission and synchronization with the other subsystems of the ROBVISION network, a fiber channel was chosen as the physical network layer. This has a capacity of full duplex 1 Gbps, delivering up to 200 Mbps. Such a capacity is sufficient for the transmission of the images (752 × 582 × 1 byte) at frame a rate of 25 frames per second and, moreover, for the communication with the other subsystems with the standard TCP/IP protocol.

4. The software architecture

PRONTO performs a large number of complex and

Table 1
Synopsis of the stereo head characteristics

	Range (°)	Velocity (°/s)	Acceleration (°/s ²)	Resolution (°)
Pan	± 45	≥ 73	≥ 1600	0.007
Tilt	± 60	≥ 73	≥ 2100	0.007
Vergence	± 45	≥ 330	≥ 5100	0.03

interrelated tasks as mentioned in Section 3. The synchronization of these tasks and the amount of information that should be acquired, processed and distributed to other subsystems, led to designing the software as a multi-thread distributed object-oriented application. The programming language was C++ with the distributed programming technology DCOM (Distributed Component Object Model) [20]. The use of such a technology enables the design of modular software architectures and permits a scalable hardware architecture, which allows increasing the computational power by easily adding more computers, whenever needed. Modules within DCOM are defined and can be accessed only through their interface whilst the implementation is safely hidden and encapsulated. Interfaces are standardized at a binary level so that the standard is both platform and language independent. Moreover, methods on objects can be called transparently whether they are either in the same process of the caller or in another process within the same machine or in a different machine in a network. The modularity of the system allows us reusing any component written for a specific task, throughout the system. Although PRONTO uses internally a distributed programming technology, this was not the policy in the ROBVISION project. It was decided to use a rather simple communications schema, implemented, as explained before, through plain ASCII commands on TCP/IP protocol. The choice of the TCP/IP protocol is due to its availability in most of operating systems and due to its successful use in similar projects (see for example Ref. [21]). An interesting point in ROBVISION is that of integrating different operating systems; all machines run Windows NT except one that runs SuSE Linux. In the case of PRONTO, the decision of using NT was taken mainly because it allows us to reuse existing software as well as to exploit the implemented software in future applications. Moreover, the device drivers of specific hardware (the axis control board and the frame-grabbers) were only available for this operating system. Software code has been shared with other partners in an effort to homogenize it in the whole project. Performance has been the second big objective in the implementation. For that reason, special care has been devoted to the optimization of the code, so that shorter execution circles are achieved. In the next subsections, the main units of PRONTO software are described. These are the Vision (containing also the head control) and the Communication units. These two main threads are constantly activated and communicated to each other by means of data queues.

4.1. Vision

The 3D features are found and measured by PRONTO system, since this comprises stereoscopic vision. V4R measures 2D features, utilizing the images from the left camera sent by PRONTO through the fiber channel. V4R fuses the measurements with a Least-Mean-Squares (LMS) method and provide the robot pose. Moreover, it updates this pose periodically every 120 ms by tracking the 2D features. At the same time, PRONTO assists the tracking by stabilizing the images. The stabilization is performed in two loops. The fast one utilizes direct feedback from the inertial sensors and the slow one uses the estimated robot pose to move the head towards the viewpoint. A velocity command is given to the corresponding motors every 40 ms (fast loop) which is equal to: $v = K\omega$, where K is a constant gain and ω is the velocity measured by the corresponding gyro. In the slow loop, the system obtains the estimated robot pose every second and it moves the head accordingly towards the viewpoint.

The 3D features are used to estimate with high accuracy the robot pose in the following cases: (i) at the initial position; (ii) whenever a new viewpoint is provided by C2V; (iii) to recover the robot pose whenever it is lost. The 3D features include only junctions, as lines have proved not to be reasonable for the pose estimation [22]. The scanning of the 3D features requires the robot to be stationary. As far as the 3D junctions identification and the measurement algorithm are concerned, a robust and quick edge detector [23] followed by a fast Hough technique [24] is used to extract the lines on the image planes of a stereo pair. The CAD-model provides the lines and the related junctions and the extracted lines and the junctions are projected on the image plane and associated to the former using a weighted LMS method. The match of the lines is done using a greater weight for their slope, rather than their distance from the center of the image (these are the two sufficient parameters to describe a line on image plane). In this way, the system is proved to be robust in detecting the lines, even if they are translated with respect to the expected position on the image plane. A closed loop method is then applied, so that by moving simultaneously the four dof of the head, the junction is put at the principal point of the image in both images. When this is the case, the two cameras are verging on a certain junction and the direct kinematics of the head is used to determine the 3D position of the junction relatively to the head [15].

Several experiments have proven the accuracy, repeatability and robustness of our system, as well as of the whole ROBVISION system. The results of the experiments for the Eurohead have been presented in Ref. [16]. It can be summarized here that the stereoscopic vision system relies on the high accuracy of the head encoders, which has been validated experimentally. The main factors that affect the measurements are small variations in the vergence angle or small horizontal deviations of the principal point. On the

Table 2

Generic format of the communication commands exchanged between the several modules of ROBVISION network, generated by PRONTO

0 0 0 0 X X	, PRONTO	, Operator	, 34	; LogScene...	;
Number of bytes, 6	Source Name	Destination Name	Message Sequence Number	Command ID, arguments	
bytes					
long					

other hand, variations in pan and tilt and vertical deviations of the principal point do not significantly affect the measurement, because these are eliminated by the head calibration method presented in Ref. [16]. As a result, PRONTO is very accurate in the absence of false correspondences. However, the use of the CAD-model to correlate the located features eliminates false correspondences. The average accuracy at typical distances from the head (about 1–1.5 m) was 2.2 cm, whilst the biggest error reported was about 15 cm in one case. The whole ROBVISION system was tested in both static and dynamic scenes. The pose, integrating 3D and 2D measurements, has been found to have a standard deviation of 5–35 mm, with respect to a reference position, which depends on the relative location of the features. The distance to the features was in the range of 1–3 m and the worst mean distance at this range was 0.5 cm. The fact that the pose estimation is more accurate and consistent than the individual measurements verifies the main aspect of the ROBVISION project, which is the robust pose identification, through the redundancy of the visual measurements.

4.2. Communication with the other systems

The vision algorithms conform an important part of PRONTO; the other big software part is the communication within ROBVISION. These are implemented in a component called CommManager. The CommManager acts as an interface between ROBVISION and the PRONTO visual and control system. It manages a group of TCP sockets that implement the low level communications with other ROBVISION components, by sending and receiving the commands. In order for the commands to be directly understandable by a human operator, it was chosen to use an ASCII format. Therefore, a complete protocol was designed and implemented for the ROBVISION communications necessities. A generic message sent by our system is presented in Table 2. The NumberOfBytes field denotes the total number of bytes in this message, which include all the data contained in the attached commands and the responses. The SourceName is PRONTO as the messages originate from our system, and the DestinationName is the system the message is sent to. Clients sending messages to PRONTO indicate their name in the SourceName field accordingly. An exemplar subset of PRONTO's commands is given in Appendix A. The CommManager utilizes a friendly interface, which allows the user a surveillance of the network status, the state in which the system runs as well

as of all the incoming/outgoing PRONTO's commands. It should be noted that some of the information sent in the messages is redundant. For example, in a point-to-point connection using sockets, it is obvious who is the sender and who is the receiver. But the main idea was to create a generic protocol that could be used in other kind of communications, such as a connectionless (e.g. UDP) or broadcasting.

The CommManager implements internally the ROBVISION communications protocol. It is an event oriented software component, which waits for events from the sockets or from other threads within PRONTO (image acquisition, synchronization). The socket layer was implemented using the Microsoft Foundation Classes socket library. When the CommManager receives a message, it is internally preprocessed in the socket. A determination of the type of message is done and data is extracted from it. The corresponding function in the CommManager is called afterwards to process that data. The CommManager updates the received data in memory spaces shared with other PRONTO threads. These memory spaces are blocking objects that serialize the access to important data within PRONTO. These data can be classified into three types:

- images;
- camera pose; and
- synchronization data.

In order for the other vision system (V4R) to provide an accurate robot pose, the features found by PRONTO and the corresponding measurements should be labeled with the image number on which they were found. Therefore, PRONTO needs to be fully synchronized with V4R. For this reason, along with each image, the actual camera pose respecting to a common reference frame (the robot's head base), has to be associated. Consequently, V4R is able to project the CAD data into the image plane and search for relevant features on it.

Inside ROBVISION, the subsystems act as clients and servers at the same time. The main idea is that a subsystem operates as server supplying requested data owned by it to the other systems and as a client inquiring data owned by the others. A system can send a message asking to receive certain data with a certain frequency. More than one systems can acquire the same 'data service', as for example, the estimated robot pose from V4R, which is needed to all the systems to perform different tasks (C2V: features extraction, ROBUG: walking, PRONTO: 3D measurements).

5. Discussion

An active vision system for robot navigation was presented. The system is integrated on a multi-computer distributed network. The visual system can perform accurate 3D measurements, which are used to estimate robustly the pose of the robot. Moreover, the system performs several other vision tasks including: the control of the stereo head; image stabilization by means of the inertial information; image acquisition and delivery to the other subsystems. For the communications with the rest of the network ASCII commands with a standard TCP/IP protocol on a fiber channel network were used. Several experiments have proven the accuracy, repeatability and robustness of our module, as well as of the whole ROBVISION system. The aspects related to the distributed computational architecture of the system are worth stressing. Although there is not a common system bus, the different modules work effectively together at a reasonable frame rate thanks to the fast network connection. In terms of modularization, each subsystem could not be badly influenced by what other modules do. For instance, the legged robot controller could not break down just because of a failure in a visual process. None of the subsystem is running a ‘hard real-time’ OS though it is fair to say that the time critical control aspects are carried out independently by the control boards (e.g. PID position/velocity controller). Where needed, the use of suitable techniques in software circumvented time constraints, e.g. where images need synchronization, an appropriate reference frame was used instead.

Future work includes the integration of linear accelerometers and inclinometers to the existing gyros, in order to perform a more efficient and accurate image stabilization. Moreover, since the robustness of the pose estimation technique relies on the redundancy of the measured features, more 3D features extracted by stereoscopic vision, such as lines in 3D and the normal of surfaces, are still under investigation. This should enhance the pose estimation in areas poor of features. Together with the improvements in the vision capabilities, future work is necessary to improve accuracy, error recovery and fault tolerance behaviors. Due to the high modularity of the system, the integration with extra sensors can be considered feasible and easy to implement.

Acknowledgements

The work presented in this paper has been supported by the Esprit project ROBVISION (EP-28867). The Training and Mobility of Researchers Networks VIRGO and SMART II support Dr Gasteratos and Mr Beltran. The authors would like to thank the ROBVISION group for their assistance in writing this article and the anonymous referees for their constructive criticism that improved the presentation of the paper.

Appendix A

RequestedData,Feature,<ObjectName>,<CameraID>,<FeatureID>,<Found3D>,<[x y z]>;

With this command PRONTO reports the feature that was found as well as its distance from the base of the stereo head. Feature and Found3D are fixed strings. This message is the answer to a previous message Monitor, AllFeatures.

Example:

```
000087, PRONTO, V4R, 5; RequestedData,
Feature, ship, 1, 12, Found3D, [300.232
968.242 2501.333];
```

RequestedData,CameraPose,<CameraID>,<[x y z r p y]>;

With this message PRONTO reports the position of the camera with its $x y z$ and roll, pitch, yaw position with respect to the base of the robot head. CameraPose is a fixed string. This message is a periodical answer to a previous message Monitor, CameraPose.

Example:

```
000095, PRONTO, V4R, 56; RequestedData,
CameraPose, 1, [230.342 1032.654
2666.433 0.534 1.53463 0.03];
```

ReadyToRun;

With this command PRONTO reports to the Supervisor that it is ready to operate.

Example:

```
000038, PRONTO, Supervisor, 3; Ready-
ToRun;
```

Monitor,CameraPose,<CameraID>,<Frequency>;

This is an inquire command to obtain the actual camera pose with respect to the base of the robot head. The answer is supplied by a RequestedData message where the data field will be the position and the roll–pitch–yaw angles of the camera $[x y z r p y]$. The response message will be send with a frequency defined by the \langle Frequency \rangle field (in Hz).

Example:

```
000043, V4R, PRONTO, 3; Monitor, Camera-
Pose, 1, 3;
```

Monitor,AllFeatures,<Frequency>;

This command tells PRONTO to start sending the features that it has found. The features information is sent with a RequestedData, Feature response with a frequency defined by the \langle Frequency \rangle field (in Hz).

Example:

```
000042, V4R, PRONTO, 5; Monitor, AllFea-
tures, 3;
```


References

- [1] O. Knudsen-Neckelmann, Industrial Vision, PhD Thesis, Technical University of Denmark, Copenhagen, 1998.
- [2] M.R. Stevens, J.R. Beveridge, Localized scene interpretation from 3D models, range, and optical data, *Computer Vision and Image Understanding* 80 (2000) 111–129.
- [3] F. Prieto, P. Boulanger, H.T. Redarce, R. Lepage, Visual system for fast and automated inspection of 3D parts, *International Journal of CAD/CAM and Computer Graphic* 13 (1998) 211–227.
- [4] J.H.M. Byne, J.A.D.W. Anderson, A CAD-based computer vision system, *Image and Vision Computing* (1998) 533–539.
- [5] A. Heikki, CAD model-based planning and vision guidance for optical 3D co-ordinate measurement, PhD, University of Oulu, Oulu, 1997.
- [6] M. Zapp, H. Janocha, Geometry measurement as integrated part of the manufacturing process using a moved CCD camera, *SPIE Proceedings* (1995) 350–361.
- [7] K. Bubna, C.V. Stewart, Model selection in computer vision, in: R. Fisher (Ed.), *CV Online: On-Line Compendium of Computer Vision*, 2000 <http://www.dai.ed.ac.uk/CVonline/>.
- [8] A. Kosaka, J. Pan, Purdue experiments in model-based vision for hallway navigation, *Proceedings of Workshop on Vision for Robots in IROS'95*, 1995, pp. 87–96.
- [9] D.S. Kim, R. Nevatia, Recognition and localization of generic objects for indoor navigation using functionality, *Image and Vision Computing* 16 (1998) 729–743.
- [10] S. Shah, J.K. Aggarwal, Mobile robot navigation and scene modeling using stereo fish-eye lens system, *Machine Vision and Applications* 10 (1997) 159–173.
- [11] R. Graf, M. Rieder, R. Dillmann, A new driving concept for a mobile robot, *IFAC Workshop on Intelligent Components for Vehicles—ICV 98*, Sevilla, Spain, 1998, pp. 115–120.
- [12] P. Wunsch, G. Hirzinger, Real-time visual tracking of 3-D objects with dynamic handling of occlusion, *ICRA'97* (1997) 2868–2873.
- [13] S. Fuerst, E.D. Dickmanns, A vision based navigation system for autonomous aircraft, *Robotics and Autonomous Systems* 28 (1999) 173–184.
- [14] M. Vincze, M. Ayromlou, S. Galt, A. Gasteratos, C. Gramkow, N. Hower, S. Hoffgaard, O. Madsen, R. Martinotti, O. Neckelmann, G. Sandini, R. Waterman, M. Zillich, RobVision—visually guiding a walking robot through a ship structure, *First International Conference on Computer Applications and Information Technology in the Maritime Industries COMPIT 2000*, Berlin, Germany, 2000.
- [15] A. Gasteratos, R. Martinotti, G. Metta, G. Sandini, Precise 3D measurements with a high resolution stereo head, *IWISPA 2000*, Pula, Croatia, 2000, pp. 171–176.
- [16] A. Gasteratos, G. Sandini, On the accuracy of the Eurohead, *Technical Report, TR 2/00, LIRA-Lab, DIST, University of Genova, Genova, 2000*.
- [17] M. Zillich, M. Vinze, M. Ayromlou, D. Legenstein, A framework for model-based vision for robotics (V4R), *24th OAGM Workshop*, Vienna, Austria, 2000, pp. 177–184.
- [18] S. Galt, D.S. Cooke, G. Sandini, A. Gasteratos, C. Beltran, An advanced walking vehicle with autonomous navigation capabilities, *Third International Conference on Climbing and Walking Robots (CLAWAR'2000)*, Madrid, Spain, 2000, pp. 731–739.
- [19] F. Panerai, G. Metta, G. Sandini, Visuo-inertial stabilization in space-variant binocular systems, *Robotics and Autonomous Systems* 30 (2000) 195–214.
- [20] Microsoft Corporation, DCOM—The Distributed Component Object Model, 1996, <http://www.microsoft.com/TechNet/winnt/Winntas/technote/dcomwp.asp>.
- [21] A.F.T. Winfield, O.E. Holland, The application of wireless local area network technology to the control of mobile robots, *Microprocessors and Microsystems* 23 (2000) 597–607.
- [22] A. Gasteratos, C. Beltran, Finding the 3D orientation of a line using hough transform and a stereo pair, *Technical Report, TR 3/00, LIRA-Lab, DIST, University of Genova, Genova, 2000*.
- [23] S.M. Smith, J.M. Brady, SUSAN—a new approach to low level image processing, *International Journal of Computer Vision* 23 (1997) 45–78.
- [24] S.J. Perantonis, B. Gatos, N. Papamarkos, Block decomposition and segmentation for fast hough transform evaluation, *Pattern Recognition* 32 (1999) 811–824.



Antonios Gasteratos was born in Athens, Hellas on 10th March 1972. He received the Diploma and the PhD degrees both from the Department of Electrical and Computer Engineering, Democritus University of Thrace, Hellas, in 1994 and 1998, respectively. During 1999–2000, he was with the Laboratory for Integrated Advanced Robotics, Department of Communication, Computer and System Science, University of Genoa, Italy, as a post-doc fellow with a European TMR grant. Currently he carries out his mandatory service at the Corps of Research and Informatics of the Hellenic Army. Also he is a research fellow at the Laboratory of Electronics, Department of Electrical and Computer Engineering, Democritus University of Thrace. His research interests include robotics, with emphasis on robot vision; fuzzy and non-linear image processing; digital VLSI design and computer architectures. He is a member of the IEEE and the Technical Chamber of Greece (TEE).



Carlos Beltran was born in Valencia, Spain on 7th February 1972. He received the Diploma and the BSc degrees both in Computer Engineering from the Polytechnic University of Valencia in 1996 and from the University of Valencia in 1998, respectively. Currently he lives in Italy where he works towards his PhD at the Laboratory for Integrated Advanced Robotics, Department of Communication, Computer and Systems Science, University of Genova. His research interests include computer vision, space vehicles, wavelets, robotics and space physics



Giorgio Metta was born in Cagliari, Italy, in 1970. He received the MS degree in Electronic Engineering from the University of Genova, Italy, in 1994, discussing a thesis on visual servoing and robot manipulation. In 1995, he was with the Department of Computer Science, Leeds University, England, funded by a grant from the EU-HCM SMART project. After his mandatory national service he did in the Italian Navy Chief Staff, in 1996 he joined the Laboratory for Integrated Advanced Robotics (LIRA Lab, Genova) and he received the PhD degree in 1999. During the three-year course, he investigated issues related to the development of sensori-motor coordination from both the artificial and the computational neuroscience point of view. For this reasons, he built an anthropomorphic head–arm robotic setup. Currently he is with the Laboratory of Artificial Intelligence at the Massachusetts Institute of Technology, as a post-doc fellow.



Giulio Sandini teaches the course of Natural and Artificial Intelligent Systems for students of the Electronics and Computer Science curriculum. He currently coordinates the activity of researchers at the Laboratory of Integrated Advanced Robotics (LIRA Lab) where research related to robotics and computational neuroscience is carried out. Among the ongoing projects, the control of a binocular head using space-variant, anthropomorphic sensors and the guidance of mobile robots on the basis of visual information.

Giulio Sandini has been a member of program committees of international conferences and chairman and co-chairman of international conferences and workshops.