Smart Cable-Driven Camera Robotic Assistant

I.Rivas-Blanco, C. López-Casado, C. J. Pérez-del-Pulgar, F. García-Vacas, J.C. Fraile and V.F. Muñoz, *Member, IEEE*

***Abstract*— This paper describes the mechanical design and the cognition system of a novel concept of camera robotic assistant that combines the advantages of intra-abdominal devices and autonomous camera navigation. The robotic assistant is composed of a magnetic intra-abdominal camera robot with two internal cable-driven degrees of freedom and an external robot that handles an external magnet. The intelligence of the robot is implemented in a cognitive architecture based on a long-term memory that stores the robot's knowledge and in learning capabilities to improve the robot's behavior. The navigation strategy combines a reactive control based on instrument tracking and a proactive control based on predefined behaviors, depending on the actual state of the task. The robot's learning capabilities include a semantic learning capability, to adapt the camera's behavior to the surgeon's preferences, and a reinforcement learning, to improve the camera navigation strategy. The paper details both the hardware implementation of the system and the software implementation in a Robotic Operating System (ROS) architecture. The cognition system and the performance of the cable-driven mechanism have been validated with a set of *in vitro* experiments. Moreover, the camera robot has been evaluated through an *in vivo* experiment in a pig.**

***Index Terms*—Mechatronics, medical robotics, cognitive robotics, robot control, robot motion.**

I. INTRODUCTION

MINIMALLY invasive surgery is widely accepted globally and provides numerous benefits for patients. Nevertheless, minimally invasive surgery remains a more challenging procedure for surgeons than open surgery, as it limits the number of interventions that can be performed. One of the main limitations when performing minimally invasive surgery is the need to use an endoscope to have visual feedback of the operating field. First, motion of the endoscope is restricted by the access port, which only provides four degrees of freedom (DoFs): two rotations around the entry point, a rotation around the instrument longitudinal axis, and a translation along this axis. This fact unavoidably limits both the field of view, as there are many blind areas that are unreachable for the camera, and the perspective of the operating area, i.e., the angle from which an object or organ is viewed. Second, conventional laparoscopy requires an assistant devoted to handling the endoscope. Fatigue and stress, especially in long interventions, may affect the quality and stability of the image. Moreover, it has been demonstrated that an efficient gaze strategy improves laparoscopic performance [1], so cooperation between surgeon and assistant is fundamental.

One solution to augment the field of view during laparoscopic procedures is the use of flexible endoscopes. These devices provide additional DoFs but are complex to use; lack of rigidity affects image stability, and they have fixed visual horizons that force the surgeon to adjust for tilted or inverted views [2]. Another approach is the use of intra-abdominal devices. These devices are introduced into the abdominal cavity through one of the incisions surgeons make to insert the surgical tools or through natural orifices, commonly the esophagus. Different forms of attachment to the abdominal wall have been explored, such as suturing [3] or needle locking [4], but magnetic interaction is the only one that allows continuous motion of the devices [5]. In this last approach, the base of the devices is provided with permanent magnets. By means of an external magnetic handle, the camera can be displaced along the abdominal wall, making it possible to reach areas unattainable by traditional endoscopes [6]. Moreover, this solution releases an entry port, which may be avoided or used for an additional tool if required. The feasibility of magnetic devices has been validated in human clinical experiences [7] and it has been demonstrated that this approach provides a wider field of view with respect to a fixed device [8]. Some authors have designed devices with two DoFs (pan and tilt) activated by internal motors [9][10][11]. Although these solutions augment the field of view compared with traditional endoscopes, they still require an assistant to manage the external handle.

As regards the navigation of the camera, there exist surgical robots specifically designed to hold the endoscope; this improves operative efficiency and reduces the need for operating room staff [12]. These systems typically provide one or more direct control interfaces, such as head [13] or eye [14] trackers, or voice-activated control [15], through which the surgeon directly commands the robot to make a specific motion of the camera through a set of predefined orders. Although these methods have succeeded in replacing medical staff, they introduce extraneous devices and they require very specific

This work was supported by the Spanish national projects under action DPI2010-21126-C03-01 and DPI2013-47196-C3-1R.

I. Rivas-Blanco, C. López-Casado, C.J. Pérez-del-Pulgar, and V.F. muñoz are with the System Engineering and Automation Department, University of Malaga, Andalucía Tech, Malaga 29071, Spain (e-mail: irivas@uma.es; mclopezc@uma.es; carlosperez@uma.es; vfmm@uma.es).

F. García-Vacas is with the Mechanical Engineering and Fluid Mechanics Department, University of Malaga, Andalucía Tech, Malaga 29071, Spain (e-mail: fgv@uma.es).

J.C. Fraile is with the System Engineering and Automation Department, University of Valladolid, Valladolid, Spain (e-mail: jcfraile@eii.uva.es).

instructions that also distract the surgeon. Moreover, camera movements are constricted to a limited number of commands, such as right, left, or inwards, indicating the desired direction of motion. Conversely, automatic guidance of the camera reduces the surgeon workload, who can concentrate on the important surgical task. Automatic strategies can be divided into two approaches: instrument tracking and navigation based on the surgical workflow.

Instrument tracking is a guidance approach that involves following the tips of the surgical instruments [16][17][18]. Many instrument-tracking approaches rely on image processing techniques, generally using colored markers to identify tool position [19]. The advantage of this approach is that no sensors are required, but tracking is only possible when tools are inside the camera field of view. Weede *et al.* [20] improve camera guidance by predicting tool motion using Markov models. Unlike direct control interfaces, camera motion is automatic and does not require constant supervision by the surgeon. However, the guidance strategy follows simple and rigid rules, such as tracking one tool or the middle point of both tools, or zooming in or out so that both tools are always in the camera field of view [21].

Some authors employ probabilistic classifiers to model surgical tasks [22]. In this way, the guidance strategy is made dependent on the actual state of the task. Ko *et al.* [23] propose a camera guidance method based on pointing to a specific surgical site, depending on the task state. Similarly, in our previous work [24], we proposed a method to improve traditional instrument-tracking methods by defining different tracking points, depending on the actual state of the task. Moreover, this work includes a learning mechanism based on taking into account particular guidance preferences of different surgeons to improve the behavior of the robot. Although these methods consider the surgical workflow to adapt the camera guidance, the camera is supposed to be in a static position during the execution of every maneuver or subtask. Despite providing automatic guidance of the endoscope, all these solutions are actually slaves to their rigidly preprogrammed behaviors, and they lack the intelligence and awareness to be considered autonomous [25]. Moreover, tracking methods follow the same strategy independently of the task being performed. In addition, approaches based on probabilistic classifiers consider the actual conditions of the scene to decide how to move the camera, but behaviors of the camera are limited to the number of states of the overall task, and the system does not contemplate changes in the preprogrammed surgical plan or different ways of working of surgeons.

This paper proposes a novel concept of camera robotic assistant that combines the advantages of intra-abdominal devices and autonomous camera navigation systems. The control architecture of this smart camera robotic assistant is shown in Fig. 1. The cognition system implements the robot's base of knowledge and reasoning functions to infer the optimal camera view for each state of the intervention, based on the environmental information provided by a perception system, and sends the desired motion of the camera to the camera robotic assistant. The human-robot interaction module allows

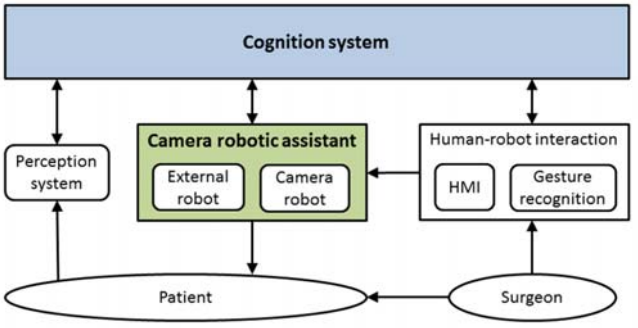

Fig. 1. Control architecture for a smart camera robotic assistant.

for communication between the camera robotic assistant and the surgeon. This interaction includes a human- machine interface (HMI) to directly command the robot a particular motion, and a gesture recognition system to be able to follow the surgical workflow. The recognition system, which has been further studied in our previous work [24][26][27], interprets the surgeon's maneuvers to estimate the actual state of the task. Finally, the perception system acquires dynamic information of the environment; thus, it provides a snapshot of the surgical scenario in every instant of the intervention. There are numerous works in the literature related to object identification in real applications, such as automatic gauze detection [28], needle detection [29] or automatic surgical tool tracking [30][31]. There are also studies addressing the problem of bleeding detection in real environments [32].

Based on the state of the art, the development of the smart camera robotic assistant defined in Fig. 1 still requires a further research on two scopes: the design of the camera robotic assistant and the cognition system. First, current intra-abdominal devices are handled by hand, so they cannot perform automatic movements. Second, current navigation strategies are very rigid and do not offer enough flexibility for a highly dynamic environment such a surgical intervention. This work proposes a camera robotic assistant composed of an intra-abdominal magnetic camera robot and an external robot with a magnetic holder attached at its end effector. Thus, this solution allows autonomous camera navigation. Moreover, the camera robot is provided with two cable-driven DoFs. This approach avoids the need to include internal motors that would augment the size and weight of the device. Regarding the camera navigation, this work proposes a cognition system that combines a reactive behavior based on instrument tracking with a proactive behavior based on the surgery workflow. Hence, the camera robotic assistant is able to accommodate the camera view to the present state of the task, but with enough flexibility to be able to adapt its behavior in unplanned or unforeseen situations. Object recognition in real surgical scenarios is out of the scope of this work, so for the experiments the perception system has been reduced to a vision algorithm that tracks a set of color markers.

The rest of this paper is organized as follows. Section II introduces the mechanical design of the camera robotic assistant, along with the geometric model of the task. Section

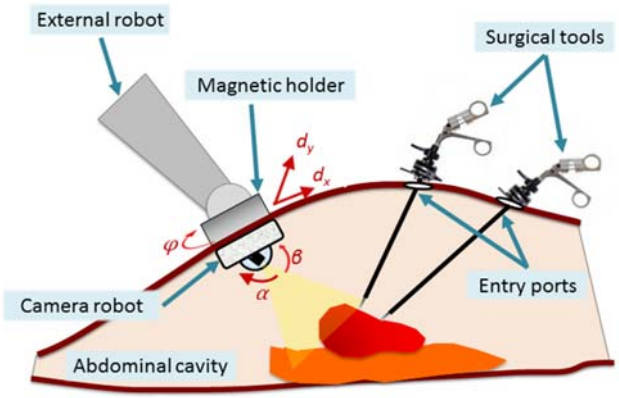

Fig. 3. Camera robotic assistant overview.

III details the robot cognition and the codification of the long-term memory and the learning mechanisms. Implementation of the system is described in Section IV, presenting the experimental setup and the hardware and software architectures. Experiments are described in Section V, and conclusions are presented in Section VI.

II. Smart Cable-Driven Camera Robotic Assistant

This section describes the mechanical design of the camera robotic assistant and the geometric model of the task.

A. Mechanical Design

Fig. 2 shows an overview of the camera robotic assistant. The robotic assistant is divided into two components: an external robotic arm and an internal camera robot. Both devices are coupled through a magnetic holder, which is attached to the end effector of the robotic arm through its upper side, and is provided with a set of permanent magnets on its lower side for magnetic coupling of the camera robot. The camera robot is introduced into the abdominal cavity through one of the incisions the surgeon makes to introduce the surgical tools (entry port). Once inside the abdominal cavity, it is attached to the abdominal wall through magnetic coupling with the external holder, thanks to a set of magnets placed on its base. Thus, the robotic assistant has six DoFs: two displacements along the abdominal wall surface (d_x and d_y) and a pan rotation (φ), performed by the external robot, internal roll and tilt rotations (α and β, respectively), and a digital zoom that simulates the inward–outward motion of traditional endoscopes. Displacement along the abdominal wall makes it possible to reach any area inside the abdominal cavity, while pan rotation (performed by rotating the external robot's last joint) controls the image horizon. In addition, tilt and roll DoFs allows viewing the operating area from different perspectives, which may prove really useful in 2D environments, where depth perception is lost.

The mechanical design of the camera robot is depicted in Fig. 3. The robot is 30 mm × 27 mm × 88 mm, and the neodymium magnets are 20 mm in diameter and 5 mm high. The camera (model MO-F3506LSC-3T, Misumi Electronics Corp., Taiwan), which is only 8 mm in diameter and 10.5 mm long, has a resolution of 400 TV lines, a frame rate of 50 fps, and a focus distance between 5 and 10 cm, appropriate for providing high-resolution images in a laparoscopic environment. The camera is provided with six white LEDs, so no additional light is required during an operation.

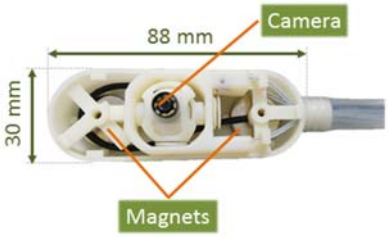

Fig. 4. Top view of the camera robot.

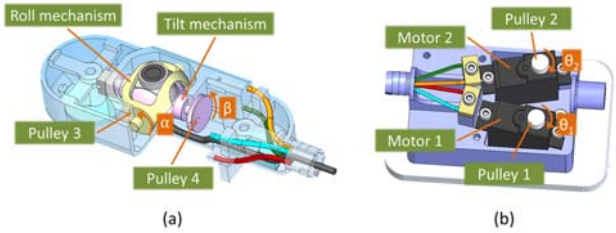

Fig. 2. Cable-driven actuation system: (a) driven side; (b) driver side.

The cable-driven system to actuate roll and tilt rotations is illustrated in Fig. 4. The driven side is composed of two concentric mechanisms (roll mechanism and tilt mechanism) that move the camera in the two directions indicated by rotations α and β, and the driver side is composed of two motors (motor 1 and motor 2) in charge of actuating the system. The actuation system is based on motion transmission between two driver pulleys (pulley 1 and pulley 2) and two driven pulleys (pulley 3 and pulley 4). On the driven side, blue and red cables are attached to the pulley of the roll mechanism (pulley 3) and green and orange cables are attached to the pulley of the tilt mechanism (pulley 4). On the opposite side, the driver side, blue and red cables are attached to the pulley of motor 1 (pulley 1), and green and orange cables are attached to the pulley of motor 2 (pulley 2). Thus, an incremental rotation, θ_1, of motor 1 transmits an incremental rotation, α, of the roll mechanism, and an incremental rotation, θ_2, of motor 2 transmits an incremental rotation, β, of the tilt mechanism. Assuming no slip of the cables in the mechanisms and negligible cable thickness, the equations describing the transmission of motion from the driver side to the driven side are:

$$\alpha = \frac{D_1}{D_3}\theta_1 \quad (1)$$

$$\beta = \frac{D_2}{D_4}\theta_2 \quad (2)$$

where D_1, D_2, D_3 and D_4 are the diameters of pulley 1, pulley 2, pulley 3, and pulley 4, respectively.

B. Geometric Model of the Task

The geometric model of the task environment is depicted in Fig. 5, where the global reference frame is marked {0} and the 2D image reference frame is marked {I}. The origin of system {0} is located at the central contact point of the camera robot

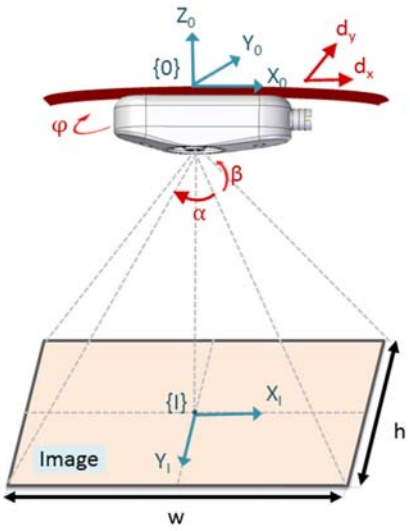

Fig. 5. Geometric model of the task environment.

with the abdominal wall, and the origin of system {I} is located at the center of the image. The image width and height are denoted w and h, and depend on the image resolution. Displacements d_x and d_y cause a displacement of the image on axes X_0 and Y_0, respectively, while a pan rotation φ causes a rotation of the image around axis Z_0. In addition, a roll rotation provokes a displacement of the image in the X_I and Y_I directions, respectively. Finally, image zoom is computed digitally as:

$$\begin{pmatrix} w_z \\ h_z \end{pmatrix} = \frac{1}{zoom}\begin{pmatrix} w \\ h \end{pmatrix} \quad (3)$$

where w_z and h_z are the image width and height, respectively, for a particular zoom level.

III. Robot Cognition

As shown in Fig. 6, the robot cognition is composed of a *long-term memory*, which encodes information for long periods of time, and a *working memory*, in which knowledge from the perception system and the long-term-memory is combined to provide a unified representation of the current situation. Long-term memory is divided into *procedural memory*, which is associated with unconscious knowledge, such as motor skills, and *semantic memory*, which stores conscious knowledge [33]. Procedural memory contains the basic knowledge of how to select and perform basic actions or behaviors, i.e., it contains the sequences of situation action rules to perform navigation routines. *Reinforcement learning* algorithms allow the system to improve the robot behavior by an iterative process based on rewards associated with the action rules that form the base of knowledge. Semantic memory contains general knowledge that forms the basis of the robot world understanding. Semantic learning allows information to be added to the robot knowledge base during the assistant's life-time.

Next, a mathematical formulation of the robot cognition is presented. First, codification of the semantic and procedural memory is described, followed by an analysis of the reinforcement learning algorithm.

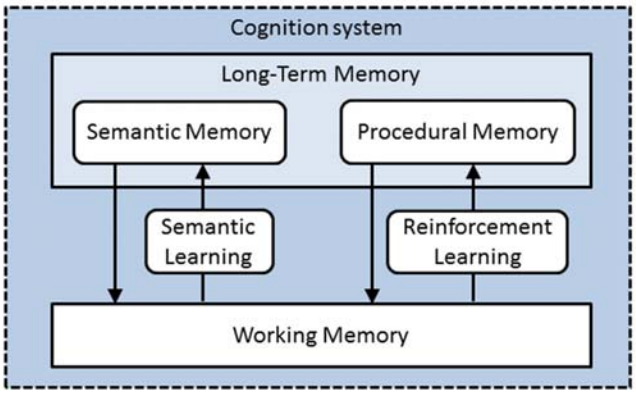

Fig. 6. Description of the cognition system.

A. Semantic Memory

Semantic memory stores the declarative knowledge of the system, modeled as a database composed of a set of tables or semantic units $\{S_1, S_2, S_3\}$. S_1 stores knowledge for object recognition through visual algorithms; S_2 stores knowledge about the surgical protocol; and S_3 stores the parameters of the camera view for each state of the protocol. Each semantic unit S_i, with $i = 1, 2, 3$, is defined by a set of attributes and each row of the semantic unit represents a particular object, event, or state.

Semantic unit S_1 stores visual information of the objects to be recognized. Hue/saturation/value (HSV) is the most suitable color space for color-based image segmentation [34], so visual object tracking is performed with color marks coded in HSV coordinates. Hence, each row j belonging to S_1 represents a marker M_j to be tracked, with HSV coordinates in the range represented by H_{min}–H_{max}, S_{min}–S_{max}, and V_{min}–V_{max}, respectively. Thus, row j of S_1 is defined as:

$$S_{1j} := \langle M_j, H_{\min j}, H_{maxj}, S_{minj}, S_{maxj}, V_{minj}, V_{maxj} \rangle \quad (4)$$

Semantic unit S_2 is the semantic unit that stores information related to the surgical protocol. Most surgical procedures can be described following a standardized protocol in which the global task is divided into a set of subtasks or states, modeled as a state diagram, where evolution to a particular state is triggered by a transition condition. Hence, S_2 contains n rows, with n the number of states of the overall task or protocol, each described by a task state and the transition that allows the evolution from the previous state. Thus, row j of S_2 is defined as:

$$S_{2j} := \langle state_j, transition_j \rangle \quad (5)$$

Finally, semantic unit S_3 contains information about the desired camera view for each state of the task, so it contains the value of the assistant DoFs for each state, i.e., α, β, φ, and zoom. According to (5), robot displacements d_x and d_y depend on the actual scene, so are not included in the semantic unit. Thus, row j of S_3 is defined as:

$$S_{3j} := \langle state_j, M_j, \alpha_j, \beta_j, \varphi_j, zoom_j \rangle \quad (6)$$

where M_j is the marker that is the focus of attention of $state_j$, chosen from the objects defined in semantic unit S_1. Semantic units S_1 and S_2 are input offline during the assistant design

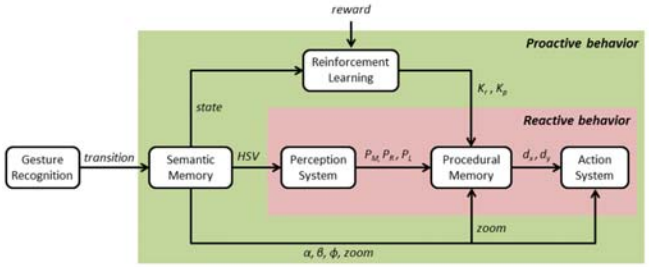

Fig. 7. Camera navigation strategy.

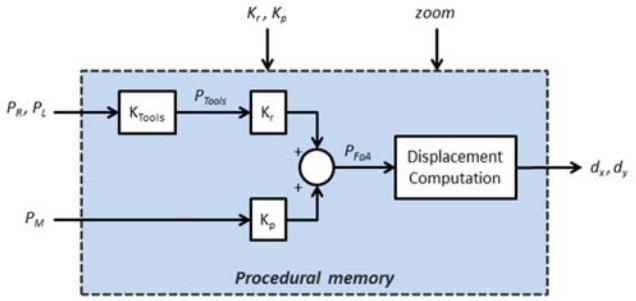

Fig. 8. Procedural memory control scheme.

process. However, the data of S_3 greatly depend on the preferences of each particular user. It has been demonstrated that eye gaze patterns are different for expert surgeons than for novices [35], so the assistant needs an online semantic learning mechanism that allows knowledge to be acquired from users. Thus, during the semantic learning process, the user sets the values of S_3 manually using the human–machine interface.

B. Procedural Memory

Current navigation strategies for camera guidance in laparoscopic surgery are divided into two groups: instrument tracking and navigation based on the surgical workflow. The first approach can be considered as a reactive behavior, while the second corresponds to a proactive control. Both approaches are based on very rigid preprogrammed actions, and they lack the intelligence and awareness to be considered autonomous. While a reactive behavior has flexibility enough to track the surgical tools wherever they are, it lacks the capacity of adapting the viewpoint in response to the task currently being performed. Conversely, a proactive camera behavior has the flexibility to offer different camera views depending on the task state, but the behavior within a particular state cannot be changed. Thus, a camera navigation strategy that combines both approaches would be able to offer a proactive behavior without losing the advantages of reactive instrument tracking.

Under these considerations, the navigation strategy proposed for the camera guidance is described in Fig. 7; this strategy combines a proactive behavior with a reactive behavior. Owing to its nature, each behavior has a different execution cycle: whereas the proactive behavior runs every time the gesture recognition system recognizes that a maneuver has been completed and triggers a transition to jump to the following state, the reactive behavior runs every system cycle. When a transition is triggered by the gesture recognition system, the cognition system infers the next state of the task according to data stored in semantic unit S_2 (5). With the current state of the task, the system infers the values of DoFs *α, β, φ,* and *zoom* according to data stored in semantic unit S_3 (6), and sends it to the action system to execute the corresponding robot actions. Moreover, the current state is also used to send to the perception system the *HSV* coordinates of the marker to be tracked, according to the information stored in semantic unit S_1 (4). The state is also sent to the reinforcement learning module, which along with a *reward* value, computes the value of constants K_r and K_p, needed in the procedural memory module. These two modules will be described next. Then, for each cycle of the system, the perception system outputs the instantaneous position of the marker tracked in the actual state, P_M, and the positions of the tips of the right and left tools, P_R and P_L, respectively. These variables are used along with the zoom value to compute the corresponding displacement of the camera robot, d_x and d_y, which is sent to the action system.

Fig. 8 shows the control scheme of the procedural memory module of Fig. 7. The aim of this control scheme is to compute the external robot displacements d_x and d_y so that the camera keeps in its field of view the marker set in the semantic memory for the actual state and the surgical tools. Thus, P_{Tools}, defined as the image point to be tracked regarding only the surgical tools, is computed as:

$$P_{Tools} = K_{Tools}\begin{pmatrix} P_R \\ P_L \end{pmatrix} \tag{7}$$

where $K_{\text{Tools}} = \frac{1}{\delta_R+\delta_L}(\delta_R \quad \delta_L)$, where δ_R and δ_L are binary variables that are true when the right tool and left tool, respectively, are moving, and false when they are not. Next, P_{Tools} must be weighted with P_M to provide a global point to track that combines the proactive and reactive behaviors. This weighing is done using constants K_r and K_p, which weight the contribution of the reactive and the proactive behavior, respectively, to the global behavior of the robotic assistant. Thus, the global point to track, i.e., the global focus of attention of the image P_{FoA}, is computed as:

$$P_{FoA} = K_r \cdot P_{Tools} + K_p \cdot P_M \tag{8}$$

where $K_r, K_p \in [0,1] / K_r + K_p = 1$. These values are learned with the reinforcement learning algorithm described in the following section. Finally, point P_{FoA} must be transformed into camera displacements. This is done as follows:

$$\begin{pmatrix} d_x \\ d_y \end{pmatrix} = \frac{K_{pm} R_{0I}}{zoom} P_{FoA} \tag{9}$$

where K_{pm} is a constant to convert a distance in the image space (given in pixels) into the Cartesian space (given in millimeters), and R_{0I} is the rotation matrix between systems {0} and {I}.

C. Reinforcement Learning

Robot behavior greatly depends on the values of K_r and K_p: high values of K_r make the camera follow the instruments independently of the state of the task, a behavior usually more comfortable for novices, while low values of K_r make the navigation strategy highly dependent on the actual state, a behavior more appropriate for expert surgeons. Therefore, for

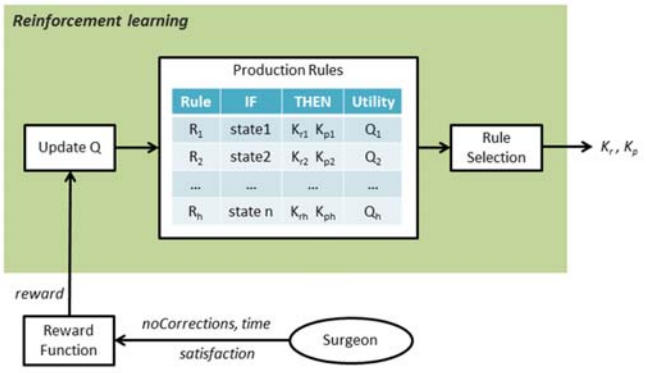

Fig. 9. Reinforcement learning algorithm.

each user, the system must learn the values of K_r and K_p that provides the best robot behavior, i.e., the values that maximize user performance and satisfaction. The process of learning these values is performed using a reinforcement learning algorithm. This method is based on a set of production rules that are balanced depending on a reward given by the environment based on the last action performed by the robot [36]. Reinforcement learning is performed following a Q-learning technique [37], which is based on associating a value Q_h, called the utility, to each rule R_h.

The complete process of the reinforcement learning algorithm is described in Fig. 9. Production rules are composed of two parts: a precondition ("IF") and an action ("THEN"). Preconditions depend on the current state of the task, while actions store the possible values of K_r and K_p, which are discretized into m values as: $K_r = \{K_{r1}, K_{r2}, \dots, K_{rm}\}$ and $K_p = \{K_{p1}, K_{p2}, \dots, K_{pm}\}$, where $K_{pi} = 1 - K_{ri}$. Therefore, each production rule R_h has the following syntax:

$$R_h: IF\ state = i\ THEN\ K_r = K_{rj}\ and\ K_p = K_{pj} \qquad (10)$$

where $1 \leq i \leq n$, with n the number of states of the task, and $1 \leq j \leq m$. Thus, the production system is composed of n × m rules. The final goal of the learning process is to infer the action that maximizes the global robot behavior for each state. Thus, the reward value is computed whenever the surgeon finishes a particular state. The reward function is designed as a fuzzy model that evaluates the robot behavior during the performance of a particular task:

$$reward = fuzzy(noCorrections, time, satisfaction) \qquad (11)$$

where *noCorrections* is the total number of corrections the user makes to the camera view using the human–machine interface, time is the total time the user spends in completing the state, and satisfaction is a qualitative variable that measures how satisfied is the user with the camera's behavior. The value of this variable is set by the user after the completion of each state, as it represents the user's perception of the quality of the camera view provided by the system. Fuzzy rules are designed so that the user satisfaction predominates over the other two variables. The time performance and the number of corrections give an idea of whether the camera behavior has helped the surgeon during the task. However, these two variables are highly affected by the surgeon's dexterity and the particular conditions in which the task is performed, especially in unpredictable environments, such as surgical interventions. Thus, although these two variables are taken into account to adjust the reward, they are highly related to surgeon satisfaction. This can be clearly appreciated in Fig. 10, which shows the behavior of the fuzzy model: reward is maximum for maximum satisfaction, decreases slightly with the number of corrections and the performance time, and is a minimum for minimum satisfaction and maximum number of corrections and performance time. For intermediate values of satisfaction, the reward value follows a similar tendency to the number of corrections and the performance time: it decreases with user satisfaction.

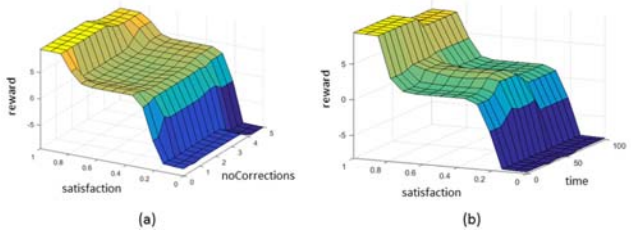

Fig. 10. Fuzzy model surfaces: (a) satisfaction vs. noCorrections; (b) satisfaction vs. time.

TABLE I
PROCEDURAL AND REINFORCEMENT LEARNING ALGORITHM

1: **if** new transition condition **do**
2: infer next state using (5)
3: retrieve M, α, β, φ and zoom using (6)
4: retrieve HSV components of object using (4)
5: compute reward function following (11)
6: update utility values (12)
7: select a rule following an ε-greedy policy
8: **for each** system cycle **do**
9: compute d_x and d_y using (9)
10: **end for**
11: **end if**

The reward value is then used to update the utility Q_h of each rule of the production system, following the equation:

$$Q_{h\_new} = (1 - \tau) \cdot Q_{h\_old} + \tau \cdot (reward + \mu \cdot EFR) \qquad (12)$$

where $0 \leq \tau \leq 1$ is the learning rate, $0 \leq \mu \leq 1$ is the discount factor, and EFR is the expected future reward, computed as the maximum utility for the next state. The learning rate determines the importance of new knowledge over old information, while the discount factor determines the importance of future rewards. Finally, rule selection is performed using the ε-greedy policy [38]. With this policy, a random rule is selected with ε probability, and the rule with the highest utility value is selected with probability 1 − ε.

Summarizing, the process of procedural memory with reinforcement learning follows the algorithm described in Table I. When a new transition condition is triggered by the gesture recognition system, the system infers the new task state using data from the semantic memory, in particular, using (5). With the new state, the system retrieves the object to track in the new state, M, and the value of the robotic assistant DoFs α, β, φ, and zoom using (6). Then, using (4), the HSV value of object M is retrieved so that the perception system can track the object. Then, the reward value associated with the last state is computed using the fuzzy model described in (11). With this reward value, the system updates the utility value of each rule in the production system using (12). Finally, the system selects the following action, i.e., the values of K_r and K_p for the next

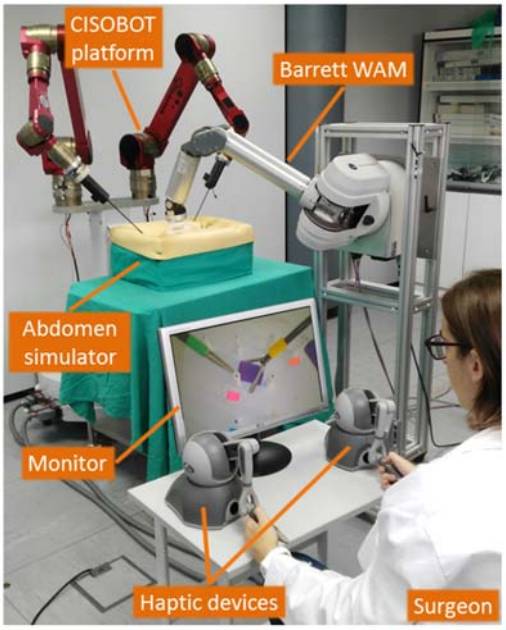

Fig. 12. Robotic assistant experimental setup.

state, following an ε-greedy policy. Then, for each system cycle, the system computes the camera displacements d_x and d_y using (9).

IV. Implementation

The setup used for the experiment is shown in Fig. 11. To operate the surgical tools, the experimental surgical robotic platform CISOBOT, developed in the University of Malaga [39], is used. This platform is composed of two six-DoFs robotic arms, to the end effectors of which have been attached two robotic graspers. These robotic graspers were manufactured by motorizing the opening and closing of commercial surgical grasper tools. Two haptic devices are used to teleoperate the tools. For natural and ergonomic teleoperation, the camera image is displayed on a monitor placed in front of the user and between the haptic devices. The external robot is a Barrett WAM (Barrett Technology, Inc.), a seven-DoFs cable-driven robot, which exhibits zero backlash and has low friction and low inertia. The main advantage of this robot is that the initial position of the end effector can be placed manually. The abdomen simulator is a methacrylate box with a flexible cover that simulates the abdominal wall. Next, the software architecture implemented in Robotic Operating System (ROS) and the implementation of the reward function are further described.

A. Software Architecture

The system software has been implemented in ROS, an open-source framework specially designed for writing robot software [40]. A ROS architecture is composed of a set of nodes that exchange data through a publisher–subscriber communication using streaming topics. Each node is an isolated process that controls a particular device or system component. Communication among nodes of the same architecture is achieved through the ROS master, also called roscore. The ROS

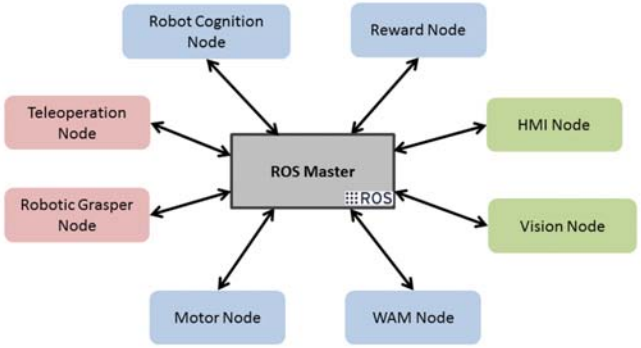

Fig. 11. Implementation in a ROS architecture.

architecture implemented in this work is shown in Fig. 12, and is composed of the following nodes:

- *Robot Cognition Node*. Robot cognition is implemented in SOAR, a general cognitive architecture designed to handle a full range of capabilities of intelligent agents [41]. SOAR is based on a production system, and uses explicit production rules to govern its behavior. Thus, SOAR allows the implementation of the robot cognition described in Section III, except for the fuzzy model that computes the immediate reward for reinforcement learning, which is an input to the node.
- *Reward Node*. This node implements the fuzzy model that computes the environment reward of the reinforcement learning algorithm. This is done in a MATLAB environment.
- *Human–Machine Interface Node*. The human–machine interface is also implemented in a MATLAB environment, and it manages the surgeon's corrections to the system.
- *WAM Node*. This node implements the motion planning algorithm of the Barret WAM robot and commands the robot to produce the corresponding motion. It is written in C++ using the real-time control library *libbarrett*. Robot displacement along the abdominal wall is based on the hybrid force–position control developed in our previous work [42]. This algorithm includes a torque compensation module that controls the robot orientation to maintain the magnetic interaction between the external holder and the camera robot and to avoid injury to the patient.
- *Vision Node*. This node inputs the digital image from the camera and implements the color segmentation algorithm for object detection. The code is written in C++ using the real-time computer vision library *openCV*. As this node handles the camera image, it also implements the image display and the digital zoom.
- *Motor Node*. This node controls the rotation of the motors responsible for the roll and tilt of the camera robot. This node is implemented in the Arduino environment using the servo motor control library *servo*.
- *Robotic Grasper Node*. Control of the motors in charge of the opening and closing of the robotic graspers is written in C++ using the same library as for the motor node, *servo*.
- *Teleoperation Node*. The node in charge of controlling the CISOBOT, including processing the inputs from the haptic devices, computing the desired motion of the instrument's tip and sending the corresponding motion to the robot's joints is

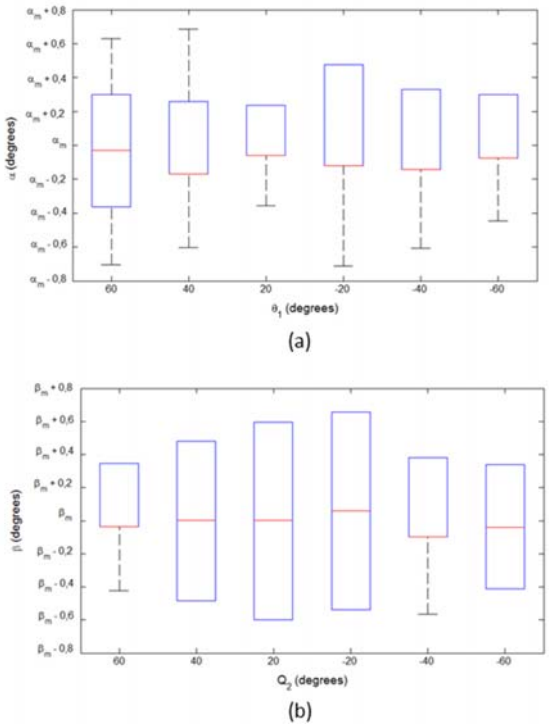

Fig. 14. Distribution of the experimental data for the analysis of repeatability of cable-driven DoFs: (a) roll; (b) tilt.

implemented using a Simulink real-time windows target (SRTWT), which provides runtime for Microsoft Windows operative systems and is able to execute Simulink diagrams in real-time. The haptic devices are connected to the SRTWT using a C++ application as a gateway between both haptic devices and the SRTWT.

V. Experiments & Results

This section presents the experimental results of the smart camera robotic assistant proposed in this work. First, an analysis of the camera robot is reported. Then, the design of the experiment for the validation of the navigation strategy is described, followed by the experimental results. Finally, the reinforcement learning algorithm is analyzed.

A. Camera Robot Analysis

The camera robot analysis has been divided into two steps: an *in-vitro* experiment to study the behavior of the internal DoFs, and an *in-vivo* experiment to evaluate the viability and quality of the camera robot. Control of the internal roll (α) and tilt (β) is performed in an open loop, owing to the lack of a mechanism to measure camera orientation in a real environment. Thus, the behavior of these DoFs has been studied with an analysis of repeatability. For this experiment, six particular positions of the motors 1 and 2, θ_1 and θ_2, respectively, were chosen, namely ±60º, ±40, and ±20º. Each of these positions was repeated ten times, measuring α and β for each of them, for motor 1 and 2, respectively. Fig. 13 shows the distribution of the experimental data for each DoF, where α_m and β_m represent the mean value of α and β, respectively, for each set of data. As it can be appreciated in the figure, the values

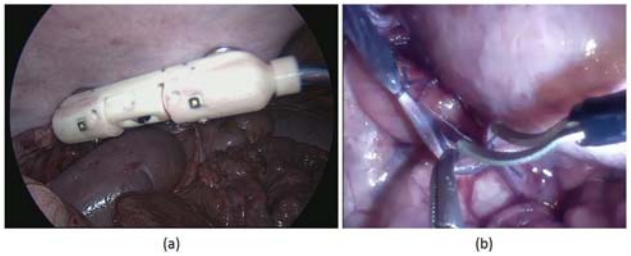

Fig. 13. In-vivo experiment in a pig: (a) snapshot of the camera robot, and (b) image of the camera.

Table II. Comparison between camera robot and conventional endoscope

	Camera robot	Conventional endoscope
Image quality	3	5
Image stability	5	4
Field of view	5	3
Camera handling	4	3

measured are in the range [α_m–0.7, α_m+0.7] and [β_m–0.6, β_m+0.65], for α and β, respectively. Taking into account that maximum values of α and β are ±30 degrees for both angles, the maximum errors are 2.3% and 2% for α and β, respectively. These are acceptable errors for moving the camera in a surgical intervention, as this task does not require an accuracy as high as is required for other surgical tasks, such as moving the instruments. In fact, for a standard camera height of approximately 10 cm, an error of 0.7º in the orientation of the camera means an error of 1.22 millimeters in the area displayed by the camera, which can be considered a negligible error.

On the other hand, an *in-vivo* experiment was carried out in the Instituto Andaluz de Cirugia, Malaga, Spain. For this experiment, two expert surgeons performed a laparoscopic varicocelectomy in a pig using the camera robot presented in this work and a conventional endoscope. The main goals of the experiment were to demonstrate the viability of using a camera robot with magnetic anchoring to the abdominal wall, and to test the quality of the camera in a real environment. Fig. 14(a) shows a first prototype of the camera robot inside the abdominal cavity during the intervention. The camera robot was introduced through a 3 cm incision at the beginning of the procedure, through which the camera wires exited the abdomen. In order to avoid gas escapes, the incision was closed with a purse-string suture to seal the tissue around the wires. At the end of the intervention, the surgeons reported that there were no gas escapes during the procedure, and that the displacement of the camera did not damage the pig's abdominal wall.

To evaluate the camera, the surgeons were asked to complete a survey to compare the camera robot with conventional endoscopes for five different features: image quality, image stability, field of view, illumination, and camera handling. Each surgeon was asked to grade each feature with a score from one to five, with one indicating "wholly dissatisfied" and five indicating "highly satisfied." The results are reported in Table II, which lists the mean scores for the two surgeons. The results reveal that the weak point of the camera robot is the image quality. However, surgeons reported that, although not comparable with an endoscope, the camera robot provides a

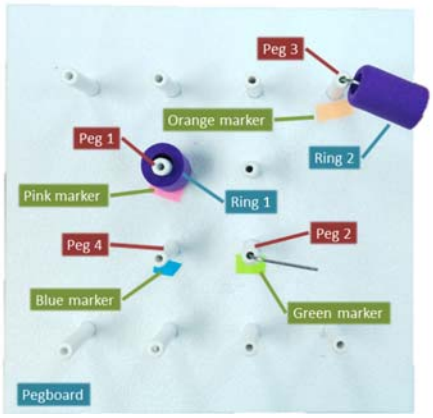

Fig. 15. Initial view of the experiment setup.

stable image with sufficient quality to perform a real operation. A snapshot of the image provided by the camera robot is shown in Fig. 14(b). Moreover, image stability with the camera robot was better than with the conventional endoscope, as the latter suffers from the inherent instability of human holding. On the other hand, the strengths of the camera robot are the field of view and the camera handling. As the motion of the camera is not restricted by the entry port, it can provide more points of view than a conventional endoscope, making it possible to reach areas inaccessible for endoscopes. As regards camera handling, although the camera robot has a more tedious insertion procedure, once inside the abdominal cavity it can be easily moved along the abdominal wall, and the camera does not fog up during the intervention. By contrast, the conventional endoscope fogs up very frequently, forcing surgeons to remove them for cleaning during surgery.

B. Cognitive Camera Navigation

The objective of this experiment was to test the smart camera robotic assistant proposed in this work. The experiment was designed according to the following criteria: the task needed to be as realistic as possible, to enable extrapolation to real surgical environments, but easy enough to be performed by inexpert personnel, and it needed to be repeated a significant number of times to demonstrate that the results were reliable. Under these considerations, the experiment was inspired by the first task in the SAGES manual of skills in laparoscopic surgery [43], peg transfer. The main purpose of this training task is to exercise depth perception in a two-dimensional environment, so it is a suitable task to validate whether the navigation strategy proposed in this work helps users to perform the task. Moreover, this task reproduces similar tools and camera movements than real tasks like needle transfer or picking a gauze and transferring it to the other tool. The experiment setup is illustrated in Fig. 15. The task is performed over a commercial pegboard with two cylindrical rubber rings (ring 1 and ring 2). Color markers (pink, blue, orange, and green) are used to identify particular areas within the pegboard. The colors were chosen to be clearly different in HSV coordinates. The main technical difficulty in extrapolating this experiment to a real environment is that of identifying the areas of interest inside a real abdominal cavity. As mentioned in Section I, there are many studies addressing this problem [28][29][30][31][32].

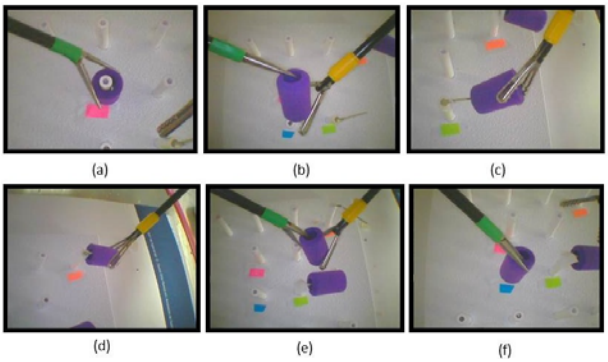

Fig. 16. Camera view for each stage of the task: (a) Stage 1; (b) Stage 2; (c) Stage 3; (d) Stage 4; (e) Stage 5; (f) Stage 6.

The experiment involves performing a pick and place task with each ring. It has been divided into six stages, where each stage is associated with the following task:

- *Stage 1*. Picking ring 1 from peg 1 with the left tool. The stage is completed when the ring is completely off the peg.
- *Stage 2*. Transferring ring 1 from the left tool to the right tool. The stage is considered finished when the ring is held by the right tool and the left tool is not in contact with the ring.
- *Stage 3*. Placing ring 1 onto peg 4. The stage is completed when the ring is on peg 4 and the right tool is not touching it.
- *Stage 4*. Picking ring 2 from peg 3 with the right tool. As in Stage 1, the stage is completed when the ring is completely off the peg.
- *Stage 5*. Transferring ring 2 from the right tool to the left one. This stage is considered finished using the same criteria as Stage 2.
- *Stage 6*. The last task involves placing ring 2 on peg 2. After this stage is completed, the complete experiment is considered finished.

A snapshot of each stage is shown in Fig. 16. Transitions to jump from one stage to the following are triggered when the gesture recognition system recognizes that the maneuver associated with the stage has been finished. As stated, gesture recognition is out of the scope of this work, as it has been further studied in our previous work [24][26][27], so for these experiments, transitions are triggered manually. During the performance of the task, the robotic assistant navigates the camera autonomously according to the algorithm of TABLE I. If the user is not comfortable with the camera view provided by the assistant, he or she may input particular orders to the camera through a graphical human–machine interface. This interface allows any of the six DoFs of the assistant to be changed in discrete steps: d_x, d_y, α, β, φ and zoom. The experiment was performed by five inexpert users. Each user performed two experiments: one with the reinforcement learning module disabled, to compare the behavior of the assistant with different predefined values of K_r and K_p (experiment 1), and the other with the reinforcement learning module enabled, to analyze the convergence of the algorithm (experiment 2). Next, these two experiments are further described, along with an analysis of their results.

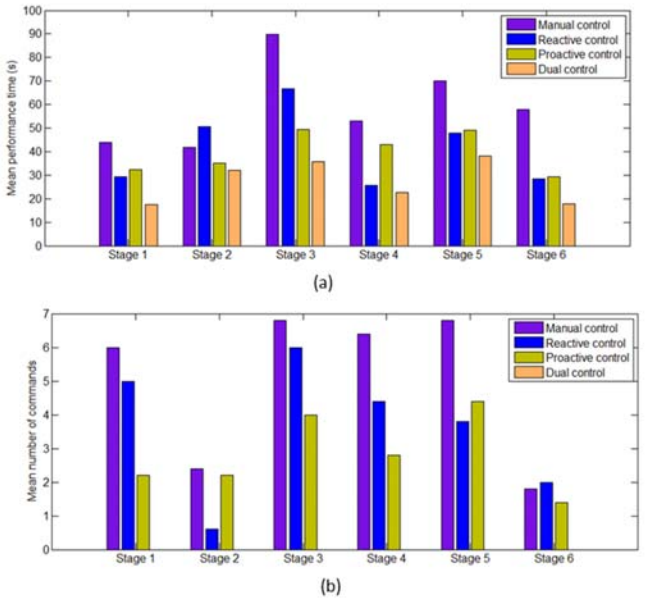

Fig. 17. Results for experiment 1: (a) mean performance time; (b) mean number of corrections.

Table III. DESCRIPTION OF EXPERIMENT 1

K_r	K_p	Number of trials	Label
0	0	5	Manual control
1	0	5	Reactive control
0	1	5	Proactive control
0.5	0.5	5	Dual control

C. Experiment 1

As described in Section III(B), the camera navigation strategy combines two types of behavior: a reactive one, which tracks the surgical tools, and a proactive one, which modifies the camera view depending on the current state of the task. Thus, the camera position depends on the contribution of each behavior to the global comportment of the robotic assistant, according to (8). Hence, the objective of this experiment is to compare the task performance using different behavior contributions parameters K_r and K_p. For the first experiment, each user performed 20 trials of the task described in the previous section, divided into sets of five trials, each with a different predefined robotic assistant behavior, as described in Table III. The experimental results are evaluated using two variables that allow an objective evaluation of the user performance: the number of commands used by the users to adapt the camera view during each state, and the time spent on completing each state. Unavoidably, the user's ability to perform the task and to handle the teleoperation system affects the results, even for two trials with identical camera view conditions. Therefore, to minimize the effects of external factors on the experimental results, each user spent a training period to get used to the teleoperation system and to perform the task with dexterity.

Experimental results are shown in Fig. 17, where the mean performance time and the mean number of corrections of the five users are represented. On the one hand, it can be observed that both the time and the number of corrections are greatest for manual control, decrease for reactive and proactive control, and are least for dual control. Thus, these results demonstrate the efficacy of combining a reactive and a proactive behavior in the camera navigation strategy, as dual control shows the better of the task performance. Moreover, in such case, the number of corrections is reduced to 0, letting the user concentrate on the important surgical task.

On the other hand, a close analysis of each subtask reveals that reactive and proactive control strategies are more or less useful depending on the task being performed. Picking and placing tasks, i.e., stages 1, 3, 4 and 6, are performed with fewer camera corrections using the proactive control, as these stages are performed around a particular area and tools do not usually move out of the camera's field of view. However, in transferring tasks (stages 2 and 5), both tools are interacting and may not be always moving around the same area, so a more reactive behavior is more comfortable for users. For example, the ring may be transferred around the picking peg or around the placing peg. Moreover, during the ring transfer it may happen that the object falls. This situation is not contemplated by the proactive control but the reactive control will follow the tools to pick up the ring again. Hence, in these states, reactive contribution is more critical than proactive contribution. Performance time is less significant in comparing the different control strategies, as it is more affected by the user's ability and the difficulty of the task. However, the tendency is similar to that of the number of corrections.

D. Experiment 2

The previous experiment revealed that the contribution of reactive and proactive control strategies, K_r and K_p, respectively, should be dependent on the current task stage. The reinforcement learning algorithm described in Section III(C) chooses, for each stage of the task, the values of K_r and K_p that maximize the system reward. Thus, the goal of this experiment is to analyze if the camera robotic assistant is able to learn the customized values of parameters K_r and K_p that most satisfies each user. For this second experiment, each user performed 15 trials of the task with the following values of parameters of (12): a learning rate (τ) of 0.3 and a discount factor (μ) of 0.9. As regards the parameter for the selection of the following action, a value of $\varepsilon = 0.4$ is used for the first ten trials, to allow exploration of new actions, and the last five trials are performed with $\varepsilon = 0.2$, to reinforce the values of K_r and K_p with higher utility. Finally, the reward computed with the fuzzy model of (11) ranges from −10 to 10, where −10 is the highest penalization the system may obtain, and 10 is the highest reward. In a surgical scenario, environmental conditions may randomly change from one intervention to another. Thus, long-term learning should prevail over the most recent information. Hence, the value of τ has been chosen so that old information prevails over new knowledge, and the value of μ makes the system strive for a long-term reward. Conversely, values of τ approaching 1 would make the system consider only the most recent information, and values of μ approaching 0 would consider only the current rewards. Moreover, the value of ε determines the learning speed of the algorithm. While a smaller

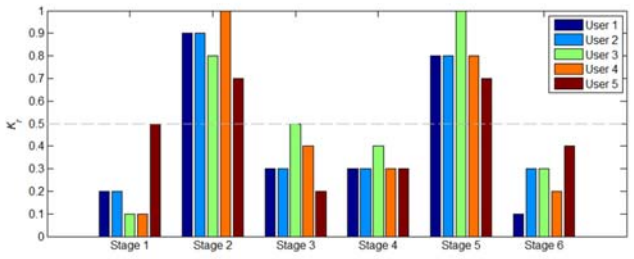

Fig. 18. Customized values for K_r for each user and each stage.

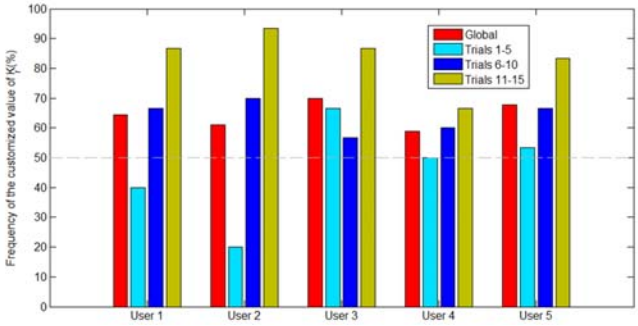

Fig. 19. Frequency of the customized value for the 15 trials performed by each user.

value of ε favors the exploration of new rules, higher ε values favors the convergence of the algorithm.

Fig. 18 shows the customized values of K_r for each user (please note that $K_p = 1 - K_r$), defined as the mode of the results of the 15 trials. As it can be seen, stages 1, 3, 4 and 6 have K_r values lower or equal to 0.5, as for this stages users appreciate the proactive control contribution over the reactive one. Conversely, stages 2 and 5 have K_r values closer to 1, as for these stages users feel more comfortable with the camera tracking the surgical tools. It can be appreciated that the value of K_r is slightly different among users. This is because of the particular preferences of each user, and because of the influence of the initial values of K_r output by the algorithm, as when a particular value has a positive reward, it is more likely to be chosen again. These values of K_r will be used by the procedural memory to set the global focus of attention during the normal performance of the system.

Fig. 19 shows the frequency of the customized values of K_r for each user (the mean of the six stages is represented). The purpose of this figure is to show the evolution of the election of the final customized value of K_r during the learning period. The red bar represents the global frequencies for the 15 trials, which are 64.4%, 61.1%, 70%, 58.9%, and 67.8% for Users 1 to 5, respectively. The other three bars represent the evolution of the data during the 15 trials. As can be seen from the figure, during the first five trials, the frequency of the customized value of K_r is lower and, as the system learns, the frequency increases, reaching the maximum for the last five trials.

Fig. 20 shows the particular results for Stage 4 of User 4, with the value of K_r and the associated *reward* of each of them represented in blue and yellow, respectively (this is a representative case, but results of other stages and users are similar). As can be seen in Fig. 18, the customized value for User 4 and Stage 4 is $K_r = 0.3$. Fig. 20 shows how this value has

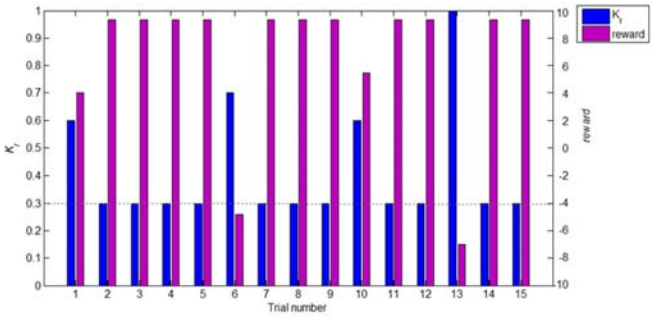

Fig. 20. Reinforcement learning algorithm analysis for stage 4 of user 4: K_r and *reward* for each trial of the experiment.

a higher reward than the other values of K_r. According to (12), a rule with a high reward has a higher utility value; thus the value of K_r associated with this rule is more likely to be chosen in following trials. This fact can be appreciated in the Figure, as the most rewarded value, $K_r = 0.3$, is the most output value by the learning algorithm in the set of trials. However, owing to the reinforcement learning nature, even within the last five trials, new action exploration is presented. This is why the algorithm chose a random value of K_r in Trial 13, which in this case is 0.1. However, the user penalizes this value with a negative reward; therefore this value is highly unlikely to appear again.

Analyzing the results of the experiments, we can observe that, with the values chosen for parameters τ, μ and ε of (12), the speed of the learning algorithm prevails over its accuracy. Higher values of ε will make the algorithm less dependent on initial values, but appreciably slower. For the particular application of moving the camera in a laparoscopic procedure, a high accuracy of the customized values of K_r and K_p is not crucial, as similar values of these parameters will provide similar behaviors of the camera. However, it is important that the system is able to provide a comfortable camera view with a short learning period.

VI. Conclusion

This paper presents the mechanical design and the cognition system of a novel concept of camera robotic assistant that combines the advantages of intra-abdominal devices with the advantages of autonomous camera navigation systems. The robotic assistant is composed of a magnetic intra-abdominal device provided with a high-resolution camera and an external robot to handle the camera. Thus, the robotic assistant has six DoFs, allowing the robot to reach any area inside the abdominal cavity and to provide different perspectives of the operating area, an important factor in 2D vision, where depth perception is lost. A novel cable-driven mechanism actuates the internal DoFs, avoiding the need to use internal motors. This actuation mechanism allows the DoFs to reach a set of positions with a good repeatability. Robot navigation is performed autonomously following a cognitive architecture that provides the navigation strategy with enough flexibility to adapt to different users and to react to unforeseen or unprogrammed situations. Experimental results show that the navigation strategy proposed in this work provides a camera view that

helps the users to perform a surgical procedure. Moreover, the results show that users do not need to give corrections to the system to change the camera view, thus allowing them to concentrate on the important task. Results also demonstrate that camera navigation provides the system with means to react to unforeseen situations. The reinforcement learning algorithm has also been validated, demonstrating that with a short training period of just 15 trials the algorithm provides customized values of K_r and K_p to provide the camera behavior that most suits each user.

Future advances of this work will include a redesign of the camera robot device to reduce friction forces, and the implementation of a perception system able to track objects in real surgical scenarios. Furthermore, an analysis of the viability of integrating the whole camera robotic assistant in an operating room must be carried out, along with the development of a fault tolerance system to guarantee the safety of the patient and of the medical staff.

REFERENCES

[1] S.J. Vine, R. J. Chayton, J. S. McGrath, R. S. W. Masters, M. R. Wilson, "Gaze training improves the retention and transfer of laparoscopic technical skills in novices," *Surg Endosc*, vol. 27, no. 9, pp. 3205-3213, 2013.

[2] V. Karimyan, M. Sodergren, J. Clark, G.Z. Yang, and A. Darzi, "Navigation systems and platforms in natural orifice translumenal endoscopic surgery (NOTES)," *Int. Journal of Surgery*, vol. 7, no. 4, pp.297-304, 2009.

[3] T. Hu, P.K. Allen, N.J. Hogle, D.L. Fowler, "Insertable surgical imaging device with pan, tilt, zoom and lighting," *Int. Journal of Robotics Research*, vol. 28, no. 10, pp. 1373-1389, Oct. 2009.

[4] C.A. Castro, S. Smith, A. Alqassis, T. Ketterl, Y. Sun, S. Ross, A. Rosemurgy, P.P. Savage, R.D. Gitlin, "MARVEL: a wireless miniature anchored robotic videoscope for expedited laparoscopy," *IEEE Int. Conf. Robotics and Automation*, RiverCentre, May 2012, pp. 2926-2931.

[5] Valdastri P, Quaglia C, Buselli E, Arezzo A, Di Lorenzo N, Morino M, Menciassi, P. Dario, "A magnetic internal mechanism for precise orientation of the camera in wireless endoluminal applications," *Endoscopy*, vol. 42, no. 6, pp. 481-486, 2010.

[6] A.C. Lehman, J. Dumpert, N.A. Wood, L. Redden, A.Q. Visty, S. Farritor S, B. Varnell, D. Oleynikov, "Natural Orifice Cholecystechtomy Using a Miniature Robot," *Surg Endosc*, vol. 23, pp.260-266, 2009.

[7] J. Cadeddu, R. Fernandez, M. Desai, R. Bergs, C. Tracy, S.J. Tang, P. Rao, M. Desai, D. Scott, "Novel Magnetically Guided Intra-abdominal Camera to Facilitate Laparoendoscopic Single-Site Surgery: initial human experience," *Surg Endosc*, vol. 23, no. 8, pp.1894-1899, 2009.

[8] M. Fakhry, B. Gallagher, F. Bello, G.B. Hanna, "Visual Exposure Using Single-Hand Magnet-Driven Intra-abdominal Wireless Camera in Miniamally Access Surgery," *Surg Endosc*, vol. 23, pp. 539-543, 2009.

[9] M. Simi, M. Silvestri, C. Cavallotti, M. Vatteroni, P. Valdastri, A. Menciassi, P. Dario, "Magnetically Activated Stereoscopic Vision System for Laparoscopic Single-Site Surgery," *IEEE/ASME Trans. on Mechatronics*, vol. 18, no. 3, pp. 1140-1151, 2013.

[10] A. C. Lehman, K. A. Berg, J. Dumpert, N. A. Wood, A. Q. Visty, M. E. Rentschler, S. R. Platt, S. M. Farritor, D. Oleynikov, "Surgery with cooperative robots," *Comput. Aided Surg*, vol. 13, pp. 95–105, 2008.

[11] D. L. Fowler, T. H. T. Nadkarni, P. K. Allen, N. J. Hogle, "Initial trial of a stereoscopic, insertable, remotely controlled camera for minimal access surgery", *Surg. Endosc*, vol. 24, pp. 9–15, 2009.

[12] R.H. Taylor, P. Kazanzides, "Medical Robotics and Computer-Integrated Interventional Medicine", *Adv. in Computers*, vol. 73, pp. 219-260, 2008.

[13] D. Sharma, C. Brown, C. Kouriefs, H. Sood, P. Grange, H. Patel (2009) Initial experience with the freehand robotic camera holder in laparoscopic urology, Journal of Endourology 23(s1):A1-A403. doi:10.1089/end.2009.2003.supp.

[14] C. Staub, S. Can, B. Jensen, A. Knoll, S. Kohlbecher, "Human-Computer interfaces for interaction with surgical tools in robotic surgery," *IEEE/EMBS Int. Conf. on Biomedical Robotics and Biomechatronics*, Rome, Italy, 2012, pp. 81-86.

[15] V.F. Muñoz, J.M. Gomez-DeGabriel, I. García-Morales, J. Fernandez-Lozano, J. Morales, "Pivoting motion control for a laparoscopic assistant robot and human clinical trials", *Adv. Robotics*, vol. 19, no. 6, pp. 694-712, 2005.

[16] K. Omote, H. Feussner, A. Ungeheuer, K. Arbter, G.-Q. Wei, J.R. Siewert, G. Hirzinger, "Self-guided robotic camera control for laparoscopic surgery compared with human camera control," *Am. J. Surg*, vol. 177, pp. 321–324, 1999.

[17] S.M. Ali, L.A. Reisner, B. King, A. Cao, G. Auner, M. Klein, A.K. Pandya, "Eye gaze tracking for endoscopic camera positioning: An application of a hardware/software interface developed to automate aesop," *Stud. Health Technol. Inform.*, vol. 132, pp. 4–7, 2007.

[18] M. Azizian, M. Khoshnam, N. Najmaei, R.V. Patel, "Visual servoing in medical robotics: A survey. Part I: Endoscopic and direct vision imaging—Techniques and applications," *Int. J. Med. Robot. Comput. Assist. Surg*, vol. 11, no. 1, pp. 67-79, 2013.

[19] A. Casals, J. Amat, and E. Laporte, "Guidance of an assistant robot in laparoscopic surgery," *IEEE Int. Conf. on Robotics and Automation*, vol. 1, Apr 1996, pp. 895–900.

[20] O. Weede, A. Bihlmaier, J. Hutzl, B.P. Müller-Stich, H. Wörn, "Towards Cognitive Medical Robotics in Minimal Invasive Surgery" *Proc. of the Conf. on Adv. in Robotics*, Pune, India, July 2013, pp. 1–8.

[21] B.W. King, L.A. Reisner, A.K. Pandya, A.M. Composto, R.D. Ellis, M.D. Klein, "Towards an autonomous robot for camera control during laparoscopic surgery," *J. Laparoendosc. Adv. Surg. Tech.*, vol. 23, pp. 1027–1030, 2013.

[22] J. Rosen, J.D. Brown, L. Chang, M.N. Sinanan, B. Hannaford, "Generalized approach for modeling minimally invasive surgery as a stochastic process using a discrete Markov model," *IEEE Trans Biomed Eng*, vol. 53, no. 3, pp. 399–413, 2006.

[23] S.Y. Ko, J. Kim, D.S. Kwon, and W.J. Lee, "Intelligent interaction between surgeon and laparoscopic assistant robot system," *IEEE Int. Workshop on Robots and Human Interactive Comunication*, pp. 60- 65, Oct. 2005.

[24] I. Rivas-Blanco, B. Estebanez, M. Cuevas-Rodriguez, E. Bauzano, V.F. Muñoz, "Towards a cognitive camera robotic assistant," *IEEE RAS & EMBS Int. Conf. on Biomedical Robotics and Biomechatronics*, Sao Paulo, Brasil, August 2014, pp. 739-744.

[25] A. Pandya, L. A. Reisner, B. King, N. Lucas, A. Composto, M. Klein, R.D. Ellis, "Review A Review of Camera Viewpoint Automation in Robotic and Laparoscopic Surgery", *Robotics*, vol. 3, pp. 310-329, 2014.

[26] E. Bauzano, B. Estebanez, I. Garcia-Morales, and V.F. Muñoz- Martinez, "Collaborative Human-Robot System for HALS suture procedures" *IEEE Systems Journal*, vol. 10, no. 3, pp. 957-966, 2014.

[27] B. Estebanez, P. del Saz-Orozco, I. Rivas-Blanco, E. Bauzano, V.F. Muñoz, I. García-Morales, "Maneuvers recognition in laparoscopic surgery: artificial neural network and hidden Markov model approaches," *IEEE RAS/EMBS Int. Conf. on Biomedical Robotics and Biomechamtronics*, Rome, Italy, 2012, pp. 1164-1169.

[28] A. García-Martínez, C.G. Juan, N.M. Garcia, J.M. Sabater-Navarro, "Automatic detection of surgical gauzes using computer vision", *23rd Mediterranean Conference on Control and Automation (MED)*, Torremolinos, Spain, June 2015, pp. 747-751

[29] A. Rodriguez-Morales, L. Lovstakken, I.K. Ekroll, H. Torp, "Needle detection by image source localization", *IEEE Int. Ultrasonic Symposium (IUS)*, 2015, pp. 1-4

[30] M. Sahu, A. Mukhopadyay, A. Szengel, S. Zachow, "Adressing multi-label imbalance problem of surgical tool detection using CNN", *Int. J Comput Assist Radiol Surg*, 2017.

[31] L. Zhang, M. Ye, P.L. Chan, G.Z. Yang, "Real-time surgical tool tracking and pose estimation using a hybrid cylindrical marker", Int J Comput Assist Radiol Surg, 2017.

[32] F. Deeba, S.K. Mohammed, F.M. Bui, K.A. Wahid, "Learning from imbalanced data: A comprehensive comparison of classifier performance for bleeding detection in endoscopic video", *5th Int. Conf. on Informatics, Electronics and Vision (ICIEV),* Dhaka, Bangladesh, May 2016, pp. 1006-1009.

[33] W. Wang, A-H. Tan, L-N. Teow, "Semantic Memory Modeling and Memory Interaction in Learning Agents," *IEEE Transactions on Systems, man and cybernetics*, vol. 99, pp. 1-14, 2016.

[34] S. Bhaskaran, S. Devi, S. Bhatia, A. Samal, L. Brown, "Mapping shadows in very high-resolution satellite data using HSV and edge

detection techniques," *Applied Geomatics*, vol. 5, no. 4, pp. 299-310, 2013.
[35] M. Wilson, J. McGrath, S. Vine, J. Brewer, D. Defriend, R. Masters, "Psychomotor control in a virtual laparoscopic surgery training environment: gaze control parameters differentiate novices from experts," *Surg Endosc*, vol. 24, pp. 2458–2464, 2010.
[36] Y. Song, Y-B. Li, C-H. Li, G-F. Zhang, "An efficient initialization approach of Q-learning for mobile robots," *Int. Journal of Control, Automation, and Systems*, vol. 10, no. 1, pp. 166-172, 2012.
[37] M.L. Littman, "Value-function reinforcement learning in Markov games," *Journal of Cognitive Systems Research*, vol. 2, pp. 55-66, 2001.
[38] M. Tokic, G. Palm, "Value-Difference Based Exploration: Adaptive Control between Epsilon-Greedy and Softmax," *Proc. of the 34th Annual German Conf. on Adv. in artificial intelligence*, Berlin, Germany, 2011, pp. 335-346.
[39] C.J. Pérez-del-Pulgar, I. García-Morales, I. Rivas-Blanco, V.F. Muñoz, "Navigation method for teleoperated single-port access surgery with soft tissue interaction detection," *IEEE Systems Journal*, in press, 2016.
[40] A. Koubaa, "Robotic Operating System (ROS)," in *Studies in Comutational Intelligence*, vol. 625, Springer Int. Publishing, 2016.
[41] J. Laird, "Extending the Soar Cognitive Architecture", Proc. of the first AGI conference, 2008, pp. 224-235.
[42] I. Rivas-Blanco, E. Bauzano, M. Cuevas-Rodriguez, P. del-Saz-Orozco, V.F. Muñoz, "Force-Position control for a miniature camera robotic system for single site surgery", *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, Tokyo, Japan, Nov. 2013, pp. 3065-3070.
[43] I. Choy, A. Okrainec, "Fundamentals of laparoscopic surgery-FLS", *The SAGES manual of Quality, Outcomes and Patient Safety*, vol. 46, pp. 461-471, 2012.

I. Rivas-Blanco received her M.S. degree in industrial engineering from the University of Malaga, Spain, in 2010.

In 2011, she joined the Medical Robotics Research Group of the University of Malaga and she is currently working toward the Ph.D. degree in mechatronics. In 2013, she was a Visiting Researcher in The Biorobotics Institute of the Scuola Superiore Sant'Anna. Her main research interests include robotic systems for minimally invasive surgery and cognitive strategies for medical robotics.

M. López-Casado received her M.S. degree in telecommunication engineering from the University of Malaga in 2004.

In 2004, she started working with the Andalusian Center for Innovation and TIC, where she participated in the design and implementation of a wireless system for biomedical signals. From 2005 to 2011, she was the coordinator of the R&D department in a private company, where she participated in several regional and national projects related to medical systems. Since 2011, she has been part of the Medical Robotics Research Group of the University of Malaga.

C.J. Pérez-del-Pulgar received his M.Sc. and Ph.D. degrees in computer science from the University of Malaga, Malaga, Spain, in 2004 and 2016, respectively.

In 2004, he was given a permanent position on the research support staff at the University of Malaga. Additionally, since 2010, he has been a Part-Time Assistant Lecturer in the Electrical Engineering Faculty, where he is responsible for various subjects related to automation and robotics. In 2014, he was a Visiting Researcher in the Telerobotics and Haptics Laboratory at the European Space Agency, European Space Research and Technology Centre. His research interests include machine learning, robotics, haptics, control, and automation. He has more than 20 publications on these topics and has been involved in more than ten Spanish and European projects.

F. García-Vacas received his M.S. and Ph.D. degrees in industrial engineering from the University of Malaga in 2000 and 2012, respectively.

He is currently an Associate Professor in the Department of Mechanical Engineering at the University of Malaga and is working on a variety of projects related to mechanical engineering, biomechanics, and surgical devices. He has led and participated in more than 20 cooperation contracts with private companies, and he is an inventor on more than 15 patents. He has authored or co-authored more than 30 journal articles, conference papers, and book chapters and has been involved in more than ten Spanish and European projects. His research interests include prosthetics design, surgical robotics, and process automation.

J.C. Fraile received his Ph.D. degree from the Faculty of Engineering at the University of Valladolid in 1987. Since 1992, he has been an Associate Professor for Control and Robotics at the University of Valladolid. His current research interests involve: rehabilitation robotics, robots for surgery, biomechanical modeling, and haptic display technologies. He has authored and co-authored more than 70 journal and conference articles.

V.F. Muñoz (M'94) received his M.Sc. degree in computer science and his Ph.D. degree from the University of Malaga, Spain, in 1990 and 1995, respectively.

Through a research fellowship, he joined the Research Group of Systems Engineering and Automation, University of Malaga, where he began his research in 1991 on the problem of mobile robot navigation. In December 1996, he became an Associate Professor at the University of Malaga. In 1997, he started his research on the application of robots in surgery. He completed this research in 2004 with the construction and use of a minimally invasive surgery robot assistant in human clinic trials. He is currently a Full Professor in the Faculty of Electrical Engineering, University of Malaga.