Bojan Šekoranja
Bojan Jerbić
Filip Šuligoj

ISSN 1333-1124
eISSN 1849-1391

VIRTUAL SURFACE FOR HUMAN-ROBOT INTERACTION

Summary

As cooperation between robots and humans becomes increasingly important for new robotic applications, human-robot interaction (HRI) becomes a significant area of research. This paper presents a novel approach to HRI based on the use of a virtual surface. The presented system consists of a virtual surface and a robot manipulator capable of tactile interaction. Multimedia content of the virtual surface and the option to manually guide the manipulator through space provide an intuitive means of interaction between the robot and the operator. The paper proposes shared workspaces for humans and robots to simplify and improve human-robot collaboration when performing various tasks utilizing a developed interaction model.

Key words: *robotics, human-robot interaction, virtual surface*

1. Introduction

Robotic technology has reached the level that enables new possibilities and new applications. As computers are becoming ubiquitous and mobile, they are becoming a part of our lives. New technologies enable a cascading shift in the fields of human activity to a kind of automated execution. One part of that is replacing human work with software solutions that can process information faster and more efficiently. But to replace the physical part of our activities, robotics has to be involved. Today, robotics has a crucial influence on the trends in technology development. The most significant trend is to bring robots into a human environment, which presents new challenges like dynamic changes and unstructured surrounding. But still, a great deal of robots has to be programmed by a qualified person who possesses certain technical knowledge of vectors, spatial transformations, and programming languages. That leads to the fact that unqualified users cannot operate robots, which results in the alienation of users from this technology. That raises a question how to simplify the user interface in such a manner that any user can interact with the robot. To achieve that, user interfaces have to be improved in a manner that people can use them intuitively and almost instantaneously [1]. This paper proposes the deployment of new approaches to enable shared workspaces for humans and robots and to simplify the human-robot interaction. An interactive table top with projected commands along with tactile sensing enable an intuitive and simple human-robot interaction.

2. Human-robot interaction

Robots generally treat humans as obstacles that need to be avoided. But the relation between people and robots can be turned into a relation between co-workers as long as the robot motions do not threaten the human. The robot working environments are slowly shifting from isolated areas with restricted human access to workspaces where motions of humans and robots overlap. Human operators can then take advantage of robot speed and accuracy. On the other hand, only the operator is able to perform certain complex manipulation and assembly tasks which require a complex perception analysis and the understanding of the surrounding world and to make decisions. The application of an industrial robot to do the tedious and repeating part of the work greatly alleviates the process for the human operator. The goal is to create teams of humans and robots that are efficient, effective and take advantage of skills of each team member [2]. Interactive working environments simplify the robot control and collaboration. A lot of work is done in interaction using vision systems, especially Kinect sensors [3,4]. Most vision systems base the interaction on some sort of sign or motion language as a means of issuing commands. An interactive table top with projected commands presents an intuitive method of interaction, as the user does not have to learn sophisticated command motions. Every displayed feature has a title that describes its purpose. If the title itself is not self-explanatory, additional information can be displayed. Also, an advantage of the projected interactive surface is the possibility of interactive user training. The developed framework is designed to include a multimedia content -film, sound, etc. The multimedia content enables new forms and techniques of teaching. This system also represents a modern educational material. Through an intuitive interaction the system acquires training capabilities in the form of a tutorial for users. New users do not have to study user manuals that consist of the text and figures because the required information will be displayed in the form of text image or video. Furthermore, the required information will be displayed only when needed, relieving the user of unnecessary data and providing only the vital information. New users can choose a “guide through mode” that also displays written information along with demonstrating the abilities and functionality of the robot. In a step-by-step tutorial, the user is guided through a work process of choice. As the user completes the tutorials, he/she is capable to perform tasks without guidance.

2.1 Interaction model

A hypothesis that the proposed interaction model can enable robots to collaborate with inexperienced users who have little or no robotics knowledge is proposed. When robots work in human environments and people can intuitively interact with them, the robots have to perceive and interpret the environment. Creating a model to recognize the intention of a human and to determine the appropriate reaction is the key part of human-robot interaction. The model takes into account the perceived information, tries to interpret and classify it and generate an appropriate action. The model is based on a premise that action is based on perception. The model relies on three forms of perception - touch, vision, and proprioception (robot position). The developed interaction model (Fig. 1) consists of the aforementioned interfaces that can all be used as a means of interaction. The model combines all the interface inputs and determines an appropriate action for the robot. This interaction model evaluates current system states and plans actions but it is also reactive as human behaviour is not always predictable. This reactivity is exhibited mostly through sudden contacts between a person and a robot. When this happens, the motion of the robot is immediately suspended so that the robot does not injure the user. The decision-making part of the model is primarily based on the first-order predicate calculus. Also, due to unpredictability, the model is partially based on probabilistic methods as there is a lot of uncertainty.

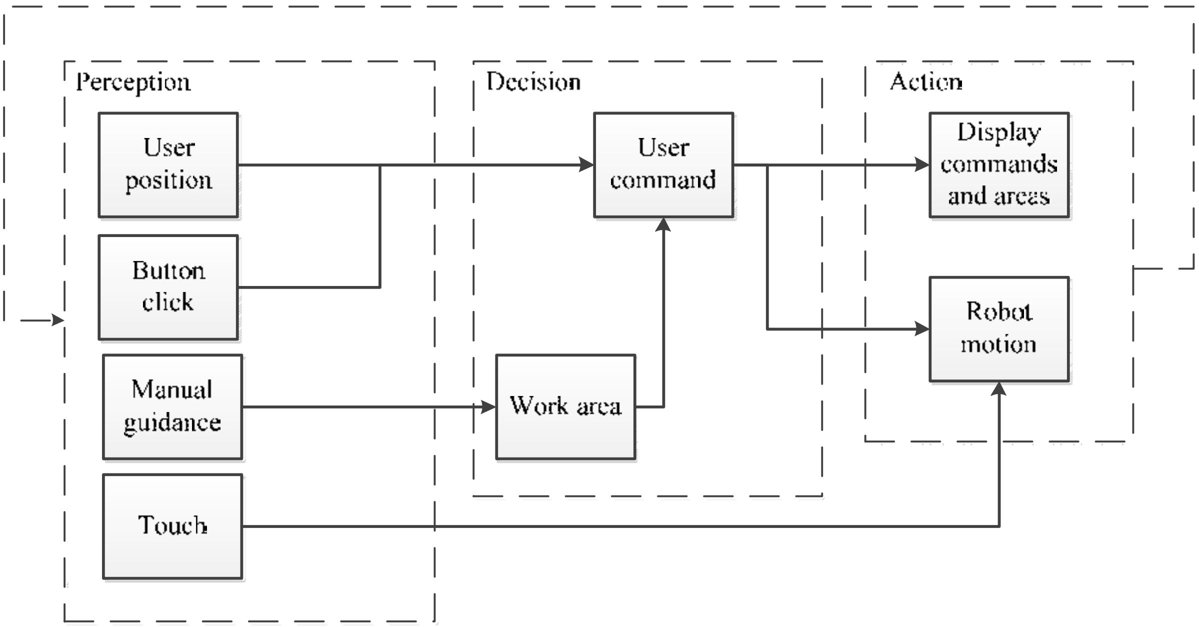

Fig. 1 Interaction model

2.2 Virtual surface

The proposed setup consists of a robot, a projector, a camera and a table top that serves as a working surface on which the image is projected (Fig. 2). An overhead projector is used to produce an interactive table top surface. The surface itself is projected in an overlapping work area of the robot and a human operator. A back projection solution was also considered but that setup would require a special transparent table top surface and a projector capable of projecting from a close distance underneath the table. As these requirements are not typical for industrial environments, it was ruled out and the original setup with the overhead projector is used.

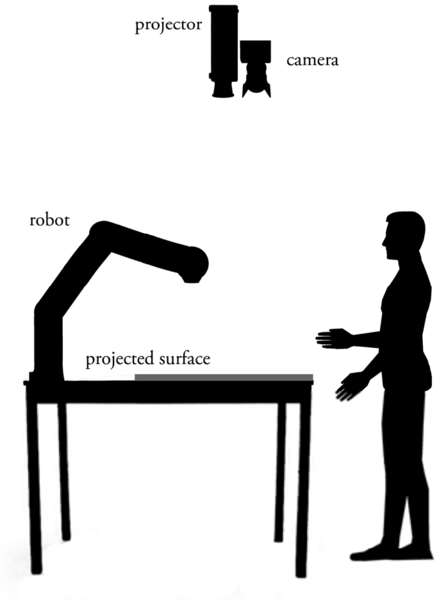

Fig. 2 Virtual surface

Virtual work area mesh, command buttons and specific work zones along with all other multimedia contents are created as a 2-D image in OpenCV. The constructed interface image is projected on the worktable and that produces an interactive surface. The stereovision system Bumblebee XB3 is used for 2-D motion tracking and range data acquisition to detect the user presence and interface inputs. Projected elements and user movements are observed by the stereovision system and based on the user's actions new elements are created, workplace positions are changed or commands are issued to a robot. In order to work properly, the system has to be initially calibrated. This involves two calibrations: the robot-surface and the stereovision-surface calibration. A coordinate system is projected onto the surface. Based on the projection of the coordinate system, the robot and the stereovision system coordinate systems are calibrated. Calibration between the projector and the stereovision system is important since it is a connection for interpreting the user's commands and actions. The robot-surface calibration is a prerequisite to linking the projected interactive areas with the robot coordinate system. The robot position is also used to make sure that the robot is not covering and shading any vital interactive areas. Furthermore, it is used to eliminate the false positive selection of commands instead of the user.

The virtual surface is used as an interaction interface between the system and the user. It is designed solely to help and guide the user through specific tasks. In order to improve interaction and free the user from information overflow, the interface itself is content aware. This means that only the commands and workspace areas related to the task on hand are projected. For example, if the on-going task is to assemble product A, only the product A-related content is projected. This includes projected work zones and other commands in the form of projected buttons. Everything is related to the on-going process. As the process is advancing, the interactive surface changes to follow along. The advanced user always has the option to access other commands and interactive features from other parts of the process and to change the process sequence if needed.

The interactive surface is user-oriented and it adapts to the user position in the working area and can be customized to a great extent. User oriented surface means that the position of projected command elements is dependent on the user position in space. The intention is to make commands available to the user at all times. For example, if the commands were projected on a fixed position, as the user moves around the table in some areas he would have to reach very far or would not even be able to activate any command. Some research surveys indicate the content-orientation as the most important element that needs to be dealt with [5]. Also, there is a possibility that other items such as the robot or work objects cover the area of commands and they become inaccessible. In order to avoid all of the above mentioned disadvantages of the fixed position, the commands and other interactive areas are auto-oriented and user-dependent and are projected near the user. The user interface is set to be change-resistant. This is set in order to avoid rapid changes in the positioning of the projected interface interaction elements. If the interface is not change-resistant, the position of the elements rapidly shifts around from one position to the other. This would present a frustrating feature and users would have difficulties to adapt to that kind of interface. The interface has to react predictably and accommodate the user in such a manner that a user can intuitively predict the position of the elements after a short period of adaptation to the system functionality. Therefore, detection areas are intentionally overlapping to integrate the described change resistance.

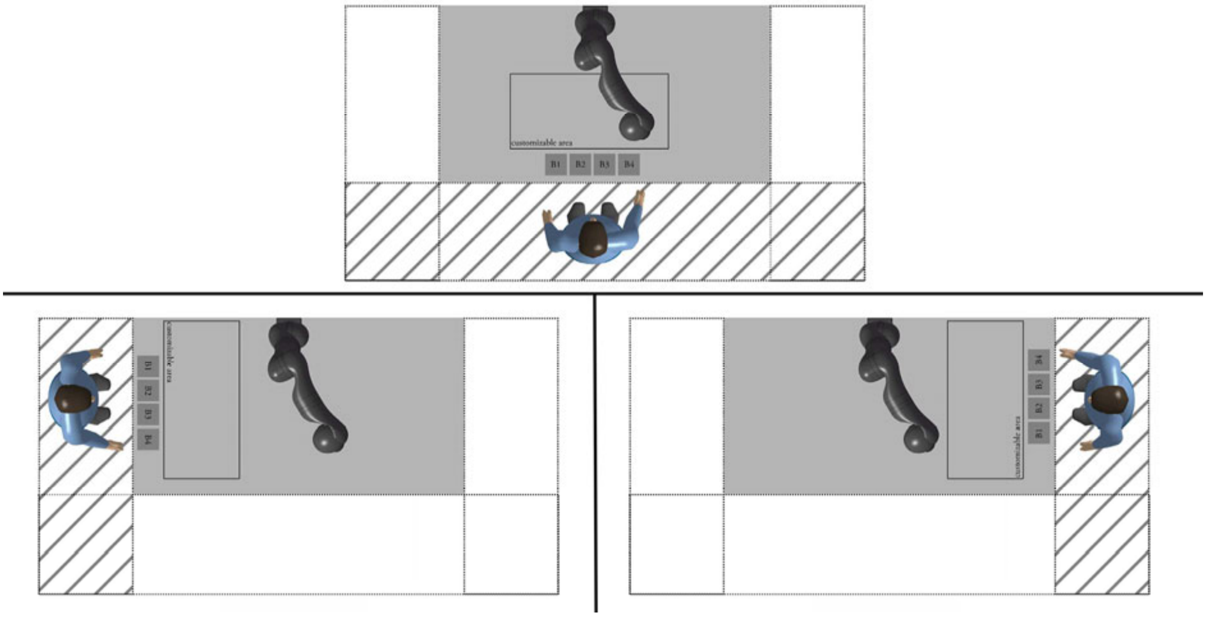

Fig. 3 User dependent interface

The work area is divided into several regions (Fig. 3) that determine the orientation and position of the interface elements in order to make them more easily accessible to the user. Some workplace positions are indifferent to user positions. The workplace positions which can be shifted also move with the user position. This means that the projected elements can have two different position attributes – a fixed position or a relative position. Fixed attribute means that the object position is independent of the user position and has the same position in relation to the robot base. Relative attribute means that the object is positioned on the projected surface depending of the user position.

The interface also enables the creation of new elements. New elements that can be created are specific action zones for the robot. In these user defined areas, the robot places objects, picks them up or performs any other specific task. The creation of these zones is also done in an intuitive manner using the vision system. As the user's hand is positioned on the interactive surface space, the area is created and projected simultaneously. When constructing new zones, a work area mesh is projected in order to help the user to decide where the robot can and should perform the action.

2.3 Tactile interaction

Tactile interaction has an increasingly significant role as humans work closer to robots. This way of interaction provides direct haptic communication. As work areas of robots and people overlap, some collisions are likely to occur. The vision system and the user detection prevent most of them. But if external factors affect the vision system, a collision can occur. To ensure that those collisions are not harmful for the human operator, the system has to ensure that a minimal force is exerted when they happen. In the proposed solution, which is applicable to any robot regardless of its sensory equipment and hardware capabilities, the robot is covered with a soft "skin" containing capacitive sensors. It consists of two layers, a conductive metal foil/mesh and rubber foam. The layers are formed to accommodate to the robot outer surface and to cover all larger and potential contact areas. When the sensor registers a change in capacitance, that means the user is present. Depending on the threshold activation values of the sensor, either contact with the user is established or the user is very close to the robot. Automatically, a stop command is issued to the robot. This is intended to prevent any damage to the human operator that enters the robot working area. Once the operator departs from the robot, the process is continued.

The other benefit of tactile interaction is a possibility to guide the robot by hand. Instead of navigating the robot through space and recording points using a teach pendant, the operator can replicate the same guiding of the robot by hand To simply drag the robot using haptic feedback is a far more intuitive manner of navigating. Also, further interactive features can be activated this way. For instance, by placing the robot tool centre point (TCP) into some area, it recognizes the task it has to perform and can execute that task autonomously. If there are multiple tasks to be performed in a small region of space, the positioning must be precise enough to activate the specific assignment. As the robot coordinate system is calibrated with the projected surface, the position of the robot indicates whether the robot is in a specific zone.

3. Implementation

3.1 Capacitive sensor

As a touch sensor, a simple capacitive sensor design is used to aid human-robot interaction. The capacitive sensor uses human body capacitance as an input and provides the robot with information of its immediate environment. As a concept, capacitive sensing is simple. Every object has its own capacitance. This capacitance is the ability of an object to hold a charge, and when a person comes in contact with an object, the capacitance changes. For this technical solution, an Arduino microcontroller, MPR121 Capacitive Touch Sensor Breakout Board, and a flexible conductive layer are used. The metal foil/mesh is the sensing part connected to MPR121 that reacts to changes in the environment. Rubber foam is used as an insulator between the conductive layer and the surface of the robot arm to eliminate any interference. A change in capacitance on the metal part of each sensor is measured and processed. Once the sensory information is processed, an associated message is sent to the robot. All messages between Arduino and the robot are exchanged via TCP/IP communication (socket messaging). The received message is then interpreted in the robot controller and adequate behaviour is exhibited [6].

3.2 Compliance and manual guidance

The robot used in this study is a Kuka lightweight robot LWR 4+ with 7 degrees of freedom. An additional axis provides greater flexibility than that of the common 6-axis robot. The arm itself weighs about 16 kg, making the robot light in comparison to industrial robots. This fact is favourable in case of a collision as the robot has a smaller mass and therefore lower inertia and collision force. Also the arm outer body is rounded and blunt without sharp edges that could injure the user. The Kuka lightweight robot is capable of measuring torques in all joints and running control loops at high frequency [7]. That can produce adequate compliant behaviour and enable smooth manual guidance to the desired position in space. A programmer or user can move the robot intuitively and quickly to the desired position. All of this sums up as a safer, fast-teaching and simple-control robot.

With regards to the final position when the user displaces the robot, there is a distinction in interpreting the user command. If the end position in space is arbitrary, the robot will continue the previous motion from the current position. If the end position is in a specific area, then an appropriate robot task is activated. The probability of requesting the specific service is calculated depending on the position in which the robot was left in space after dragging. If the point belongs to a specific area (pick-up area, drop-off area) or is a work point, or is very close to either one or the other, then the manual guidance motion is interpreted as a command. The probability that the point in space belongs to a certain area is exponentially dependent on the distance from that area. This allows the user some error in positioning but rapidly decreases the probability as the error increases. To determine the probability of a robot position being interpreted as a part of area or a workspace, a simple equation is used:

$$P(A)=e^{-d/a} \tag{1}$$

The denominator a is set with respect to defined zones and their spatial information.

To calculate the distance from the area for a current position of the robot, a set of equations is proposed:

$$(y_1 \leq y \leq y_2) \wedge (x_1 \leq x \leq x_2) \leftrightarrow \text{inside the square} \tag{2}$$

$$(y_1 \leq y \leq y_2) \wedge \neg(x_1 \leq x \leq x_2) \rightarrow d=\min(\text{abs}(x-x_i),\ i=1,2) \tag{3}$$

$$\neg(y_1 \leq y \leq y_2) \wedge (x_1 \leq x \leq x_2) \rightarrow d=\min(\text{abs}(y-y_i),\ i=1,2) \tag{4}$$

$$\neg(y_1 \leq y \leq y_2) \wedge \neg(x_1 \leq x \leq x_2) \rightarrow d^2=\min((x-x_i)^2+(y-y_j)^2\ ;\ i=1,2;j=1,2) \tag{5}$$

Once the distance and the associated probability are calculated, an assumption of the user's intention is made. If the probability is within the task-defined values, the conclusion is that the user intended to issue a command through the manual guidance of the robot and adequate behaviour of the robot is exhibited.

3.3 Visual tracking

Object tracking in a 2-d image plane can be defined as localization or trajectory estimation of a moving object. Most common tasks of object tracking are used in motion-based recognition, automated surveillance, video indexing, human-computer interaction, human-robot interaction, traffic monitoring, and vehicle navigation. Solving the problem of object localization and tracking can be achieved by using unique visual features of the object. As shown in [8], common visual features for tracking are: colour, edges, optical flow, and texture. Colour feature is used to locate an object with a specified colour value in a scene [9]. To eliminate numerous false readings and at the same time to diminish the lighting impact, the image is filtered based on the colour (hue), saturation, and value. All colours in the image except the ones targeted can be removed by using range filters,. Edges can be used for motion tracking by applying one of many edge detection algorithms to acquire the desired feature [10]. Optical flow and texture can also be used for the purpose of tracking [11, 12].

Colour tracking was considered for this application but that is impractical since a user should always be wearing the same colour. Due to the aforementioned reasons, frame differencing [13] is used for capturing motion since the only moving objects in the scene should be the user and the robot. Frame differencing is fast and reliable when it comes to tracking constantly moving objects as it constantly evaluates the difference between the last two consecutive frames [14], or, as seen in [15], with a separation of more than two frames, for the detection of slowly moving objects. Pixels inside the difference frame, which have values higher than the specified minimum, are then considered to be part of a moving object.

$$I_{diff}(x,y) = I_n(x,y) - I_{n-1}(x,y), \tag{6}$$

where I_n is the n-th or the last recorded image and I_{n-1} is the one before the last. For implementation in different environments, it is very important that object tracking is impervious to illumination conditions. Because of that the difference image is converted to a binary state with a dynamic sensitivity threshold value λ.

$$I_{bi}(x,y) = [\ 1 \text{ if } I_{diff}(x,y) > \lambda,\ 0 \text{ otherwise}] \tag{7}$$

The dynamic sensitivity threshold value λ is calculated as a function of overall pixel values in the last known static image and is constantly updated to accommodate the lighting and the scene changes.

$$\lambda = f\,(I_{stat}\,(x,y)) \tag{8}$$

To remove random noise from the image $I_{bi}(x,y)$, a simple blur function is used. After the motion has been captured and the image processed, the object is determined based on the outer contour of the motion. Contour object representation is used since it is ideal for tracking objects that have complex non-rigid shapes [16], like people in this case.

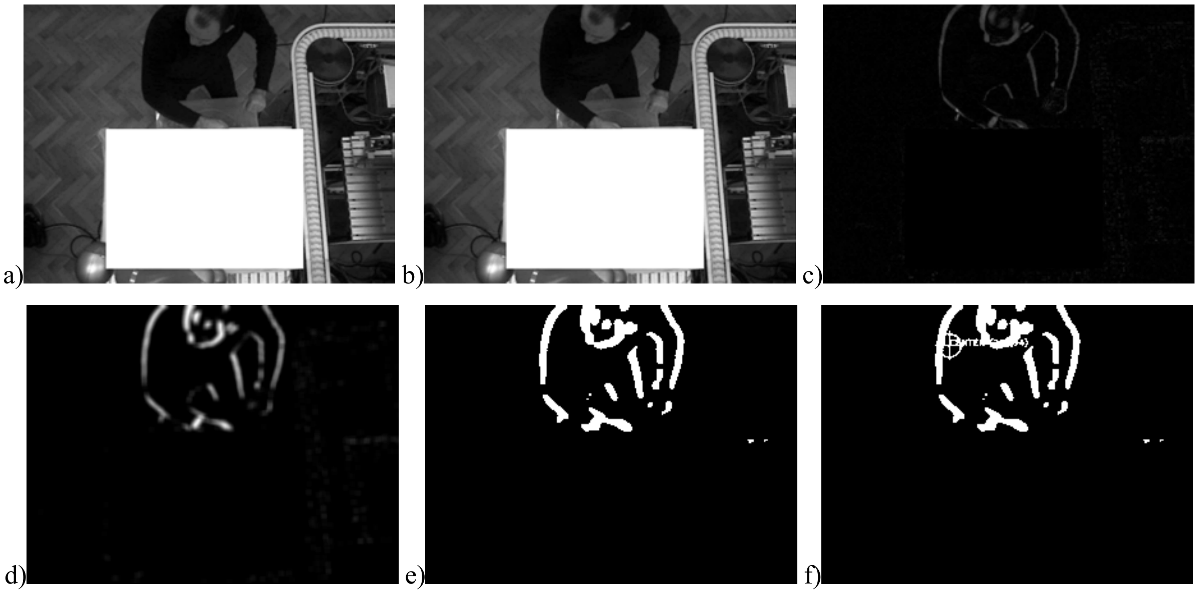

Fig. 4 Visual tracking: a)Frame - I_n, b)Frame (I_{n-1}), c)Difference frame (I_{diff}), d)Blurred difference frame (I_{blur}), e)Binary frame (I_{bi}), f)Biggest contour centroid

The system presented in this paper uses two separate motion-tracking systems for the purpose of human-robot interaction. A motion capture camera is positioned above the work area and set up to capture and differentiate the worktable from the area around it. The first motion tracking is focused on the area around the worktable and performs the user tracking. The user's position relative to the worktable determines the position of projected commands (buttons). The second motion tracking system follows the user's hand in the area above the worktable. By using a combination of motion tracking and stereovision 3-d data, the system determines whether the command button on the interactive worktable has been pressed. In situations when there are multiple moving objects in the scene, the system uses the one that has the biggest surface area. As the robot position is known, the algorithm is adapted to mask and eliminate the robot motion.

3.3.1 User tracking

The point that is considered to represent the user is calculated as a centroid (geometric centre) using the following equation:

$$C(x,y)=\frac{\sum_{i=1}^{n}P_i(x,y)}{n}. \tag{9}$$

where *C(x,y)* is the centroid of a set of n points located at the positions $P_i(x,y)$ along the outer contour of the shape extracted from the motion detection. In that way, if a user's limb is to intersect with another working area, the system will not move the graphic interface as long as the geometric centre of the user is still in the original one. Based on the number of different work areas A_n around the worktable, command buttons are projected accordingly if

$$C(x,y)\in A_1\wedge A_2\wedge\ldots\wedge A_n. \tag{10}$$

3.3.2 Button click

Giving a command or pressing a button on the interactive screen is accomplished by using a combination of 2-d and 3-d data. When using only 2-d motion tracking for finding the user's hand inside the interactive work area it is difficult to precisely determine when the user has pressed the surface. Rectangular command buttons projected on the surface are located and their contour is inspected. When the buttons are not pressed, their contours consist of four lines. When the hand is moving and it is located above the button, the number of contour lines in the specified area changes and it is different from four. If this condition is met, the stereovision system starts to take measurements of the area. If the hand is pressed to the table, meaning that the user wanted to press the button, all 3-d points in this area should be at table level. A distance of a few centimetres above the table is also interpreted as a valid result to compensate for the measurement errors and different values of hand thickness. In order to ensure the stability of the system and to avoid false reading, the time delay condition is also induced as the button should be pressed for a minimum amount of time. By using the data from a stereovision camera, with the aforementioned conditions satisfied, the virtual button pressing is accomplished.

3.3.3 Area specification

When the user wants to specify the area on the worktable, he/she needs to press the area command button. When the area command button is pressed, the system waits for the user to press the surface of the worktable to specify the upper right corner of the area and after that to press the lower left corner of the desired area. These two points define the diagonal of a rectangular area. The system is programmed to work in the same way as the button click routine. The only difference is that the point of the motion contour which is the most distant from the user's position is used for tracking and commands.

3.3.4 Object localization

Work objects can be randomly placed on the worktable so they have to be localized by using a vision system. As they are placed in designated work areas, this reduces the processing time of the entire localization.

4. Application

The majority of processes, regardless of the application, have a specific flowchart. The process itself is predefined to a great extent and consists of specific actions. As the sequence of activities is usually strictly defined, the user can customize the process by requesting specific actions from the robot. The user can also change the sequence to a certain extent but only within the allowed boundaries of the specific process. The user participates in the process, especially in the parts of the process that are difficult and unaffordable to automate, partially as an executor of specific actions, but mostly as an overseer in the form of requesting services and assigning tasks to the adaptive system. The user interacts with the system through the interactive table top surface. In industrial environments this interaction provides hybrid assembly systems where humans work in close collaboration with robots and take advantage of robot speed and accuracy to perform the tedious and repeating part of the work. Utilization of an industrial robot as a highly flexible and sophisticated tool (third hand) makes the process easier for the human operator. The rest of the system, which delivers parts to the user on request and performs actions that can be performed autonomously, is an adaptive robotic assembly system capable of adjusting the process to accommodate the user request [17,18] and plan the robot trajectories [19].

As interactive tools are slowly entering the medical field, this can provide an extension of those tools directly applicable in surgery, specifically in operating rooms. Any instrument table can become an interactive surface. Equipment in the operating room is subjected to medical asepsis, referring to practices used to promote or induce asepsis in an operative field in surgery or medicine to prevent infection. All instruments are sterilized prior to the procedure, personnel wear sterile clothes and gloves and additionally special pants, blouse, mask and caps, which is mandatory in operating rooms. The table surface is sterilized, but the system can also be adapted in such a manner that the surgeon does not have to make contact with the surface. A surgical team comprises a surgeon, an assisting surgeon or a resident, a surgical technician and an anaesthesiologist (Fig 5). In the proposed concept, some team members are replaced by a medical Robotic Neurosurgical Navigation (RONNA) system (Fig 5). The system RONNA is a new application of robotics in neurosurgery based on a dual-arm configuration with coordinated navigation and a new localization method. The use of robots is not limited only to the replacement of stereotactic frames, but robots can also be used as an assisting technology. With two robots collaborating with each other, a robotic assistant can easily perform some strenuous procedures difficult to be performed precisely by a human surgeon alone. In addition, the robotic assistant can perform the task in a simplified, faster and more precise way [20]. The interactive system is proposed as an upgrade to the current interface (tablet) in order to simplify system control.

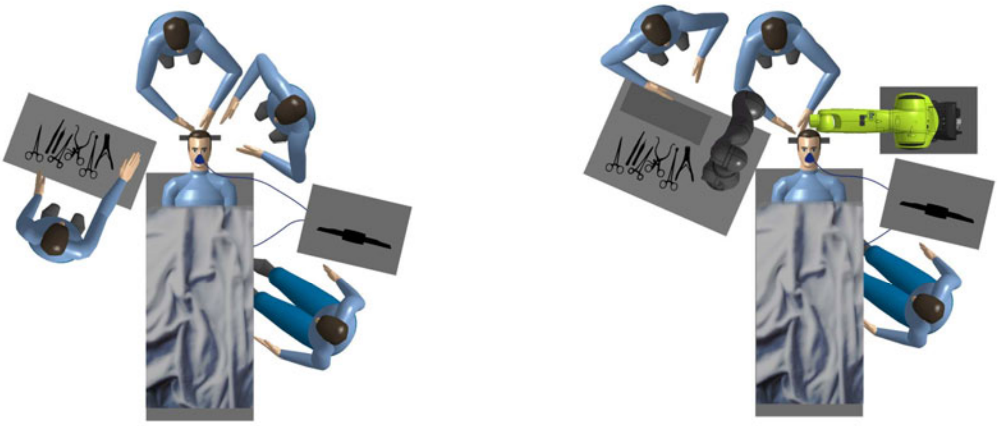

Fig. 5 Operating room

The interactive surface allows the customization of each operating procedure and an individual approach to each patient. The process itself is very similar to that used in the production process. The robots are used as task executors; they hand instruments to the surgeon and assist in the parts of the operating procedure in which they are superior to the surgeon. As the interface can be customized, it can be adjusted to display the specific data of the patient just when needed in the operating procedure. Therefore, the CT, MRI scans or any other relevant information about the patient can be presented to the surgeon. This system can be adapted as an interactive guide for young surgeons and other medical personnel. The interface is able to project any multimedia content. Therefore, videos and instructions can be integrated in form of interactive tutorials. This tutorial can be designed to follow specific operating procedures and to tutor and train medical staff.

5. Conclusion

The developed model of interaction provides an interactive environment in which a person can intuitively issue commands to the robot in several ways. The virtual surface created to provide the user with all the necessary information is also the main interface by means of which the user issues commands to the system. As the interface provides a multimedia content, the user does not have to possess any previous knowledge as he can follow the tutorial part of the process. Also a safer working envelope of the robot is ensured due to the activation of stop command on any contact with a human. A touch-based movement of the robot means an operator can simply guide the robot by hand in any direction. This can serve as a foundation to simplify robot programming. Due to its simplicity, robotics could find application in all sorts of tasks that could be further improved with the use of robots. Robotics shows potential to be used in numerous fields, ranging from every day service robotics to very specific fields that would benefit from high precision of robots, like neurosurgery and other minimally invasive surgical procedures. This requires the blending of multiple scientific directions, from technical to medical and social studies, to better comprehend human communication patterns and to embed them in control modules. As this is an interdisciplinary field, the research potential is immense and widely applicable.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the support of the Croatian Scientific Foundation through the research project ACRON - A new concept of Applied Cognitive RObotics in clinical Neuroscience.

REFERENCES

[1] O. Schlatter, B. Migge, and A. Kunz, “User-aware content orientation on interactive tabletop surfaces.” In Cyberworlds (CW), 2012 International Conference on, pp. 246-250, 2012.

[2] J. Scholtz, “Theory and evaluation of human robot interactions,” in System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on, pp. 125-135, 2003.

[3] M. G. Jacob, Y.-T. Li, and J. P. Wachs, “Gestonurse: a multimodal robotic scrub nurse,” in Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction, pp. 153–154, 2012.

[4] T. Stipančić, B. Jerbić, A. Bučević, and P. Ćurković, “Programming an industrial robot by demonstration “, In Proceedings of the 23rd International DAAAM Symposium, vol. 23, no. 1, pp. 15-18, 2012.

[5] Madni, T.M.; Sulaiman, S.B.; Tahir, M., "Content-Orientation for Collaborative Learning Using Tabletop Surfaces," Information Science and Applications (ICISA), 2013 International Conference on, pp. 1-6, 2013.

[6] B. Šekoranja, D. Bašić, M. Švaco, F. Šuligoj, and B. Jerbić, “Human-Robot Interaction Based on Use of Capacitive Sensors,” Procedia Eng., vol. 69, pp. 464–468, 2014.

[7] R. Bischoff, J. Kurth, G. Schreiber, R. Koeppe, A. Albu-Schäffer, A. Beyer, O. Eiberger, S. Haddadin, A. Stemmer, G. Grunwald, and others, “The KUKA-DLR Lightweight Robot arm-a new reference platform for robotics research and manufacturing,” in Robotics (ISR), 2010 41st international symposium on and 2010 6th German conference on robotics (ROBOTIK), pp. 1–8, 2010.

[8] A. Yilmaz, O. Javed, and M. Shah, “Object tracking: A survey,” ACM Computing Surveys, vol. 38, no. 4, pp. 13–26, 2006.

[9] F. Šuligoj, B. Šekoranja, M. Švaco, and B. Jerbić, “Object Tracking with a Multiagent Robot System and a Stereo Vision Camera,” Procedia Engineering, vol. 69, pp. 968–973, 2014.

[10] N. Senthilkumaran and R. Rajesh, “Edge detection techniques for image segmentation–a survey of soft computing approaches,” International Journal of Recent Trends in Engineering, vol. 1, no. 2, pp. 250-254, 2009.

[11] D. Decarlo, D. Metaxas, „Optical Flow Constraints on Deformable Models with Applications to Face Tracking“, Int. J. Comput. Vision 38, no. 2, pp. 99-127, 2000.

[12] V. Takala and M. Pietikainen, “Multi-object tracking using color, texture and motion,” in Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on, pp. 1–7, 2007.

[13] J. Ramesh, H. Nagel, "On the Analysis of Accumulative Difference Pictures from Image Sequences of Real World Scenes," Pattern Analysis and Machine Intelligence, IEEE Transactions on , vol. 1, no.2, pp. 206-214, 1979.

[14] I. Haritaoglu, D. Harwood, and L. S. Davis, “W 4: Real-time surveillance of people and their activities,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 22, no. 8, pp. 809–830, 2000.

[15] N. Prabhakar, V. Vaithiyanathan, A. Prakash Sharma, A. Singh, P. Singhal, „Object Tracking Using Frame Differencing and Template Matching“, Research Journal of Applied Sciences, Engineering and Technology, vol. 4, no. 24, pp. 5497-5501, 2012.

[16] M. Yokoyama and T. Poggio, “A contour-based moving object detection and tracking,” in Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on, pp. 271–276, 2005.

[17] M. Švaco, B. Šekoranja, and B. Jerbić, “Autonomous Planning Framework for Distributed Multiagent Robotic Systems,” in Technological Innovation for Sustainability, Springer, pp. 147–154, 2011.

[18] C. Castejón, G. Carbone, J. G. Prada, and M. Ceccarelli, “A multi-objective optimization of a robotic arm for service tasks,” Strojniški vestnik-Journal of Mechanical Engineering, vol. 56, no. 5, pp. 316–329, 2010.

[19] P. Ćurković, B. Jerbić, T. Stipančić "Coordination of Robots with Overlapping Workspaces Based on Motion Co-Evolution", International journal of simulation modelling, vol. 12 , no 1, pp 27-38, 2013

[20] “Ronna” [Online]. Available: http://ronna.fsb.hr/. [Accessed: 16-Oct-2014].

Submitted: 03.6.2014
Accepted: 04.3.2015

Bojan Šekoranja
Bojan Jerbić
Filip Šuligoj
Faculty of Mechanical Engineering and
Naval Architecture,
University of Zagreb
Ivana Lučića 5, Zagreb