Sciences & Technologie – N°15, Juin (2001), pp. 83-89.

ERROR DETECTION BY ANTICIPATION
FOR VISION-BASED CONTROL

Reçu le 12/09/2000 – Accepté le 27/03/2001

Abstract

A vision-based control system has been developed. It enables a human operator to remotely direct a robot, equipped with a camera, towards targets in 3D space by simply pointing on their images with a pointing device. This paper presents an anticipatory system, which has been designed for improving the safety and the effectiveness of the vision-based commands. It simulates these commands in a virtual environment. It attempts to detect hard contacts that may occur between the robot and its environment, which can be caused by machine errors or operator errors as well.

Key words: Anticipatory systems, error detection, vision-based control, adaptive supervisory control, tele-robotics.

Résumé

Un système de contrôle à base de vision a été développé. Il permet à un opérateur humain de diriger à distance un robot équipé d'une caméra vers une cible dans l'espace 3D en cliquant simplement à l'aide d'un pointeur sur l'image correspondante transmise par le robot. Cet article présente un système anticipatoire conçu pour améliorer la sécurité et l'efficacité des commandes à base de vision. Il simule ces commandes dans un environnement virtuel. Il essaye de détecter les contacts durs qui peuvent survenir entre le robot et son environnement par suite d'erreurs causées par la machine ou par l'opérateur.

***Mots clés :** Systèmes anticipatoires, détection d'erreurs, control à base de vision, contrôle supervisé adaptative, télérobotique.*

A. ZAATRI
Département de Génie Mécanique
Faculté des Sciences de l'Ingénieur
Universite Mentouri
2500 Constantine, Algérie

H. VAN BRUSSEL
PMA Division
Mechanical Engineering Department
Katholieke Universiteit Leuven
Celestijnenlaan 300b
3001 Heverlee, Belgium

ملخص

لقد صممنا نظام تحكم مبني على الصورة – هذا النظام يُمكن معامل بشري أن يوجه عن بعد شخص آلي مجهز بآلة تصوير نحو أهداف في مجال ذو ثلاثة أبعاد بنقر الهدف الممثل في الصورة المرسلة من بعد. هذا المقال يقدم نظام تنبؤ قد صمم لرفع مستوى السلامة و النجاعة عند تطبيق أوامر التحكم المبنية على الصورة. هذا النظام يقوم في وسط خيالي بمحاكاة الأوامر المنبثقة عن نظام التحكم المبنية عن الصورة. ثم يحاول اكتشاف اللقاءات الصلبة و الخطيرة التي يمكن أن تحصل بين الشخص الآلي و محيطه بسبب أخطاء قد تحصل سواء من طرف الشخص الآلي أو من طرف المعامل البشري.

***الكلمات المفتاحية :** نظام تنبؤ، اكتشاف الأخطاء، تحكم بواسطة الصورة، تحكم توافقي بالرؤية الشاملة.*

Advanced research in human-machine interaction systems is promoting a new generation of systems that can be named "**Integrated Supervisory Control Systems" or ISCS**. ISCS are complex and synergistic Human Machine Interaction Systems (HMIS), which attempt to integrate multi-media techniques, modern simulation tools and high level programming techniques. They provide various control modes such as tele-manipulation, vision, speech, etc. Such systems are meant to carry out complex tasks and missions in various environments. ISCS are present in many important industrial domains such as nuclear power plants, space missions, military operations, aircraft industry, undersea, tele-surgery and assistance to disable people, etc. An initial evaluation about the state of the art and the trends in this field can be inferred by surveying some ISCS that have been described in the literature, e.g. [1, 2, 3, 4, 5].

This paper deals with a developed ISCS, which is characterized by multi-modal interaction and multi-level control. It presents an anticipatory system that has been designed and implemented for improving the safety of the controlled system by avoiding hard contacts with the environment when executing vision-based commands. According to our experiments, vision-based control reveals to be a very useful technique for directing the robot end-effector towards locations and objects to manipulate. Therefore, it is of prime importance to ensure effective and soft impacts of the robot with the work environment against both machine errors and human errors and mistakes. This anticipatory system enhances the safety of the overall system by simulating in a virtual environment the actions generated by the vision-based control. Nevertheless, an anticipatory system requires a model of the system, of the environment and of the sensors. It also requires a meta-reasoning capability for analysing the results of the simulated action and the consequences it implies upon the system safety, then deciding what to do.

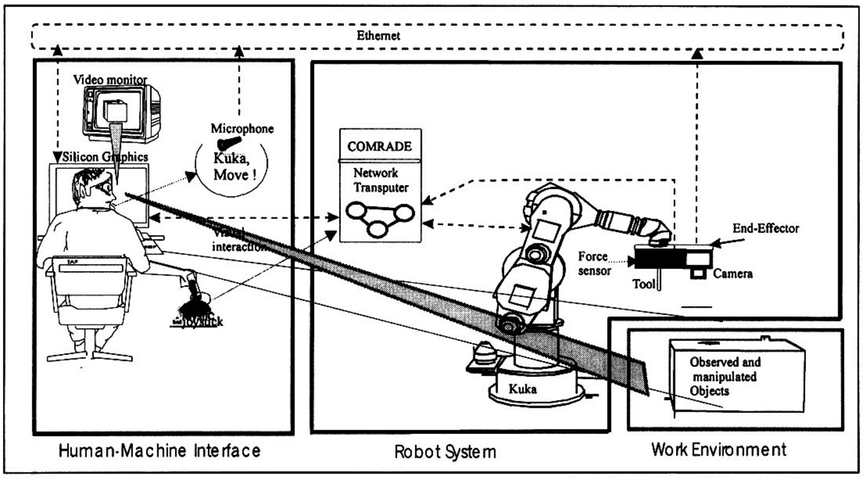


Figure 1: Overview of the experimental system.

In our implementation, by means of a rule-based decision system, the anticipatory system analyses the data acquired from the simulation in order to check whether an action generated by vision-based control is safe or not. Then it decides to enable the execution of this command in the real world or to alert the operator for handling the situation. This capability elevates the machine to the rank of an intelligent system since it emulates some processes that are involved in intelligent systems: perception, cognition and decision. It allows also the design and the implementation of co-assistance schemes where the human operator and the anticipatory system verify the results of each other. With co-assistance scheme, the anticipatory system and the human operator build up a high-level type of co-operation instead of one way or master-slave co-operation. Such schemes also illustrates a social-centered design.

This paper is organized as follows. Section 1 presents a brief overview of the ISCS. Section 2 presents the anticipatory system and section 3 concludes the paper.

1. OVERVIEW OF THE ISCS

1.1- Physical system

This physical system (see Figure 1) consists of a Kuka-361 industrial robot controlled by an interactive version of COMRADE [6]. COMRADE is a software package designed at K.U. Leuven to facilitate the development and tuning of compliant motion control. A force sensor and a CCD camera are mounted at the end-effector of the controlled system (the remote robot). The camera is linked to a video-monitor to display the captured scenes to the operator and to a frame grabber. The grabbed pictures are transmitted to a Silicon Graphics workstation, which hosts the image processing routines [7]. A human speech input system with a limited vocabulary is also implemented [8].

At the control site, the operator uses a 6 Degrees Of Freedom (DOF) hand controller, a vision module and a speech module to monitor the remote robot. Feedback information is ensured by direct vision if applicable, selected images sensed by the camera and other sensor measurements such as force and position. This physical system (see Figure 1) has already been described in more details in [5].

1.2- Virtual environment and simulation

Task simulation is an important tool for ISCS. It is usually used for visualising task achievements in virtual worlds before enabling their execution in the real world. For the system under consideration, we use the commercial 3D-robot simulation program IGRIP (Interactive Graphics Robot Instruction Program) as modelling tool and for visualisation. However, to build geometrical CAD (Computer Aided Design) models of planar objects found at the remote environment, a semi-automatic procedure has been developed [7]. 3D information about the objects in the robot environment and their positions with respect to the robot base is extracted from the stereo images provided by the vision system. They are transmitted to IGRIP to be added to the environment model. A fully graphical visualization of the robot with freely moving end effector acting in its work-cell environment has been implemented using IGRIP. For simulation of compliant-motion tasks, IGRIP has been complemented with ROSI (RObot reaction force SImulator) [9]. Developed in-house, ROSI is a unique simulation program enabling to accurately simulate reaction forces occurring during execution of compliant-motion tasks. Furthermore, on-line or off-line visualization of the real robot motion can be displayed and superimposed on the simulated robot motion.

1.3- Integrated multi-modal operator interface

To access and to manage the different ISCS functionalities, an integrated multi-modal operator interface, hosted on the Silicon Graphics workstation, has been implemented. Operator can select the control modes they prefer or consider as best suited for achieving specific tasks. The available control modes range from tele-manipulation to practically autonomous control including traded, shared, superimposed, some interactive functions, Human Demonstration Programming (HDP) [9,10], vision-based [5,7] and speech-based controls. The operator interface serves for selection of and switching between interaction modes for simulation and control. To keep, as much as possible, similar conditions to supervise tasks in the real world as well as in the virtual world, the same multi-modal operator interface is employed.

In the designed ISCS, flexible mode management and change have been considered. For instance, the described ISCS can manage sequentially the selected control modes.

But, it can also be set to mixing modes for any controlled variables (joint or cartesian) at any time. This possibility enables the operator to intervene on-line during the wait-state or during autonomous operations.

This ISCS is enhanced by some other functions that can assist the operator such as the possibility to automatically centre the view of the camera on a relevant point designated by the operator anywhere in an image, focusing, for instance, on objects of interest.

2. ANTICIPATORY SYSTEMS

2.1- The click-and-move action with stiff contact avoidance

Let's recall how the vision-based command works. To direct the end-effector to a desire target, the operator clicks simply on the image of the desired target. The vision system estimates its 3D location by exploiting the stereovision effect. Corresponding points of this location in left and right images have to be matched. The Sequential Similarity Detection Algorithm (SSDA) is used for this purpose [5]. The result of this correspondence search obtained by the SSDA is visualised via a graphical operator interface window so that the operator can check its validity. If the operator detects a mismatch, the operator designates explicitly the location of the relevant object in the left and right images solving this way the correspondence problem.

Once the correspondence problem is solved, 3D location of the target is estimated. The end-tool is directed towards this location. But to ensure the safety of this command, a guarded motion is employed. By limiting the developed contact force to a certain value, the guarded motion, using force sensing, prevents hard contacts of the system with the environment.

Not withstanding the use of a guarded motion, there is still a possibility to get stiff impacts if the end-tool is directed at a relatively high speed towards the target. This dangerous situation may stem from two causes: estimation errors and human errors. Of course, to ensure a safe control of the impact due to estimation errors, one has to take into account the inaccuracies and the errors involved in the estimation of the impact point, such as the calibration error, the error of triangulation and the dimensions of the end-tool. This can be done by augmenting the combination of those errors by e.g. one to two centimetres in the robot workspace.

However, to realise both an effective and safe operation, which means avoiding stiff impacts while reaching the target location at a relatively fast speed, the motion towards the target location is decomposed in two phases. The first part is a fast approach that starts from the initial position to the neighbourhood of the target location. It is followed by the second part, which is relatively a slow motion very close to the location point. The contact occurs in a soft and compliant manner.

Moreover, we seek to improve the system safety against human errors and mistakes. In our context, the human operator can fail when detecting an error, when solving the correspondence problem, or can make mistakes when attempting to recover a machine error. To prevent and recover these errors and mistakes, the anticipatory system is designed.

2.2. The concept of an anticipatory system

The concept of anticipatory system is defined by R. Rosen [11] as follows. Given a dynamical system S. One has to associate another dynamical system M, which is a model of S. It is required, however, that if the trajectories of S are parameterised by real time, then the corresponding trajectories of M are parameterised by a time variable which goes faster than real time. That is, if S and M are started at same time, from equivalent states, and if time is allowed to run for a fixed interval T, then M will proceed further along its state trajectory than S. In this way, the behaviour of M can predict the behaviour of S.

Suppose that M is equipped with a set E of effectors, which allow it to operate either on S itself, or on the environmental inputs of S, in such a way as to change the dynamical properties of S (Figure 2). If we put this entire system into a single box, that box will appear to us to be adaptive system in which prospective future behaviours determine present changes of state.

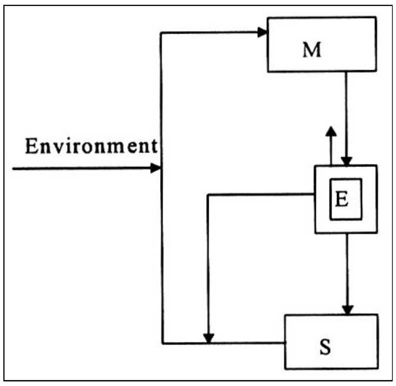


Figure 2: Principle of an anticipatory system.

How is this information to be used to modify the properties of S through the effector system E? According to [11], one way is to partition the state space of S (and hence of M) into regions corresponding to "desirable" and "undesirable" states. The virtual system M is used to detect the undesirable regions and actions are therefore taken to prevent and to avoid their occurrence in the real system S.

2.3- Design and implementation

The implemented anticipatory system makes use of the scheme presented in figure 2. The components of a generic anticipatory system (figure 3) can be easily recognized in figure 2. S corresponds to the image denoted Real World, M to the image denoted Virtual World. The "undesirable" state for this system is the occurrence of a hard contact with the environment and any other unexpected and unsafe behavior. Nevertheless, while in the anticipatory system, such as described in the previous section, functioning of the real system and its model are parallel; in our implementation, they function in a sequential way. The anticipatory system necessitates the following steps:

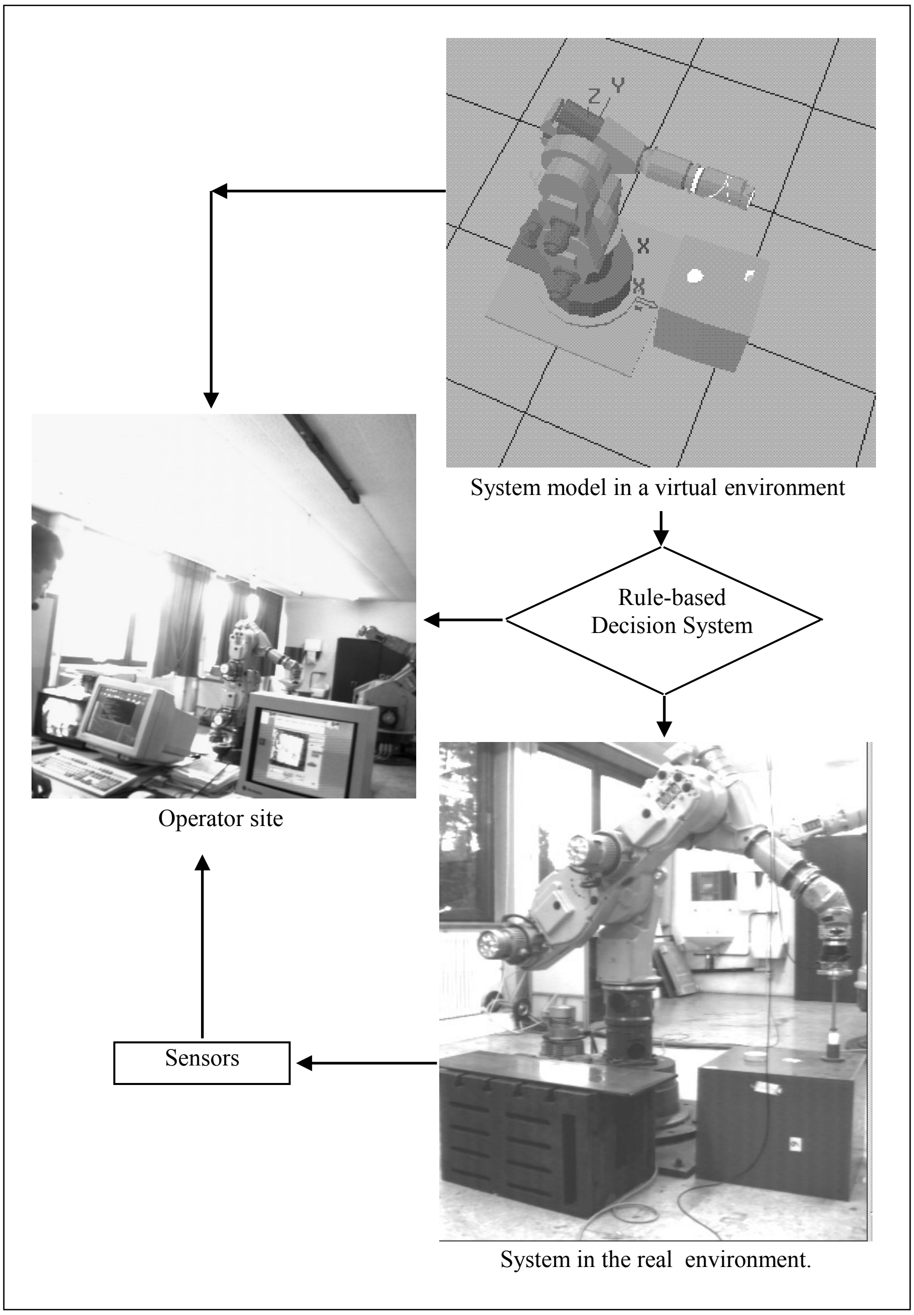


Figure 3: Scheme used by the anticipatory agent.

1) Build up a virtual world including the model of the controlled system, conform to the real world and to the real controlled system.

2) Once the operator selects a location of an object by clicking on its image, the anticipatory system automatically starts the anticipated motion towards the selected location in the virtual world.

3) The anticipatory system, by means of the rule-based system, analyses the data obtained during the virtual evolution of the action. Events and situations are extracted. It checks the consistency of the virtual action by comparing it with a referential model of real and safe execution.

4) Finally, the anticipatory system decides to send the action for real execution or alerts the operator to handle the problem.

2.4- The reference model

The implementation of the anticipatory system requires a model of a successful safe evolution of the click-and-move action. As described earlier in section 2.1, the reference model is composed of two parts. The first part consists of an approach to the vicinity of the target location. This should be a free motion with relatively high speed. The second one consists of the pre-impact motion. This should be a relatively slow motion close to the impact location. From the evolution of the positions, contact forces and torques, and correlation between them; rules to identify the model for a referential safe click-and-move action are established. They constitute the rule-based decision system of the anticipatory system.

Consider the practical and successful example of a real execution of a click-and-move action with stiff impact avoidance (see Figure 4). Although, the force sensor measures six components (three forces and three torques), in the configuration associated with this example, we restrict our analysis to the temporal evolution of the vertical force and vertical position, which are the most meaningful components. Figure 4a shows an example of the temporal evolution of the vertical position *Z(t)* of the end-effector during a real execution of the click-and-move action. It shows also in figure 4b the corresponding vertical contact force *Fz(t)* of the end-effector with the environment. The vertical position *Z(t)* of the end-effector in Figure 4a can be split in four different parts. In the first part, the robot is in its initial position (1230 mm), there is no motion yet in the vertical direction. In the second part, the position decreases with a constant velocity, which is expressed by the slope of the curve. This part corresponds to the fast approach motion towards the target in free space. In the third part that corresponds to the pre-contact situation, *Z(t)* decreases but with a low speed. The fourth and horizontal part corresponds to the soft impact followed by a return motion to the initial position. Correspondingly, the figure 4b shows the temporal evolution of the vertical contact force *Fz(t)*. It can be split in four parts as for *Z(t).* The first part shows a zero force corresponding to a fixed end-effector position. The second and the third parts correspond to the motion in free space with high and low velocities. It shows small force variation around zero bounded by ≈ 2 N due to the non-zero acceleration. In the fourth part a contact force of

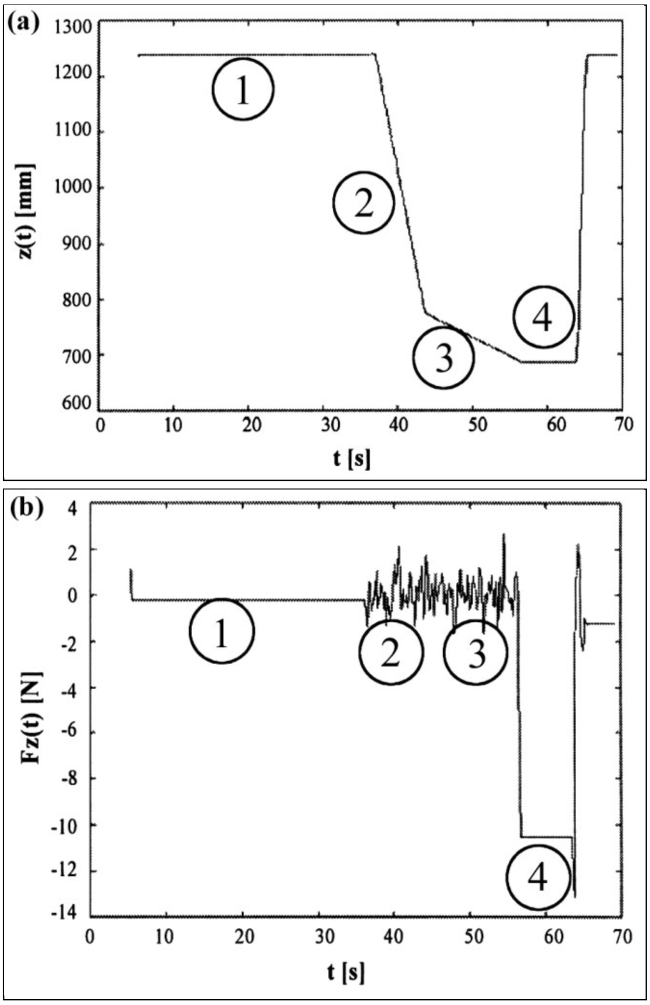


Figure 4 : **(a)** Vertical position of the end-effector ; **(b)** Contact force *Fz(t)*.

about –10 N is sensed in the vertical direction announcing a contact with the environment. This contact force attempts to increase but the value of –10 N is the limit force enabled by the control program of the robot. For this reason, the robot quickly returns to its initial position as imposed by the control program. The pic shown after the –10 N is due to the end-effector acceleration when leaving the contact.

Because the simulation exploits the models of the 3D vision and force sensing, the events and the situations encountered in the virtual world are compared to those expected in a reference model. This comparison is made by the rule-based decision system, which analyses the signals resulting from a virtual execution of the action. By analogy to the real world, any abnormal or unsafe achievement of the action in the virtual world will be considered as undesirable in the real world. The simulation is conducted autonomously by the anticipatory system, so that the operator can be completely free to do other things and may be alerted if there is any detectable risk or problem.

2.5- Application examples

The anticipatory system has been tested in different cases. Two examples are given to illustrate it. The first example corresponds to a safe realisation of a click-and-move action while the second corresponds to the occurrence of a stiff contact.

Figures 5 shows the evolution of the vertical position of the virtual end-effector obtained during a simulation of a click-and-move action. The simulation uses 3D vision and force sensing to compare the events and situations encountered in the virtual world with those expected in the real world. When the simulation is completed, the anticipatory system via the rule-based decision system checks the safety of this action by comparing it to the reference model. This example fulfils the consistency test with respect to the model. Thus, the rule-based decision system will allow the execution of this action.

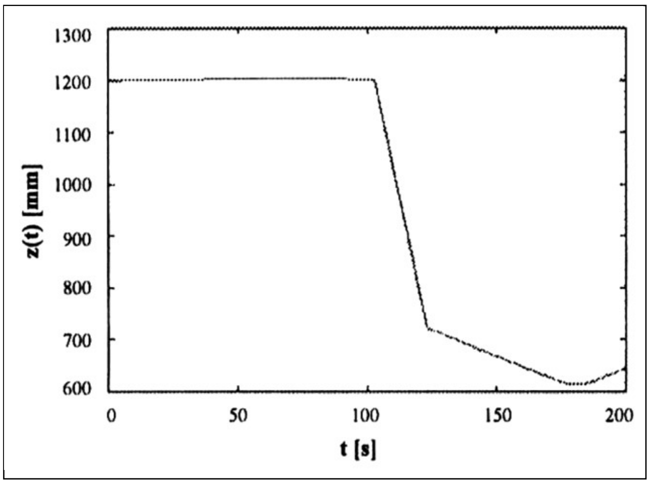
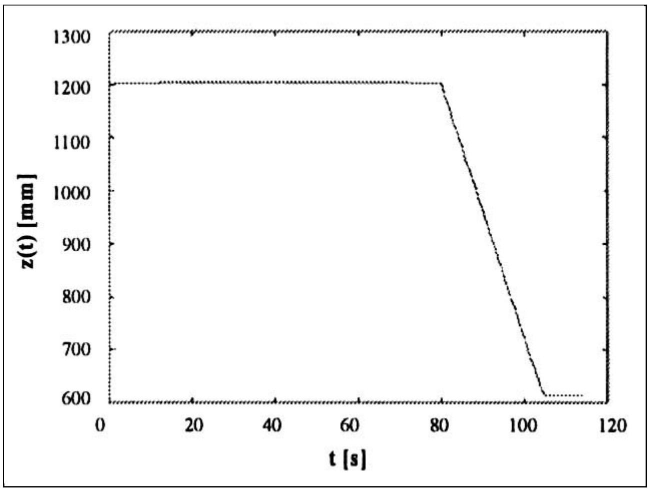


Figure 5: Example of a soft contact.

Figure 6 corresponds to another example where the consistency tests are not fulfilled. The evolution of *Z(t)* does not present the expected reduction of speed near the impact point. Thus it corresponds to a hard impact. The execution of this command in the real world is not allowed by the rule-based system.



The Co-operative Error Recovery Scheme (CERS) is a co-operative human-machine interaction technique that aims to improve the robustness of control algorithms [5]. The main feature of the CERS is that it enables a gradual and adjustable operator intervention. The CERS is conceptually simple and very efficient in practice and it actually successfully used. Applied to the click-and-move action, it works as follows (Figure 7). From an image, the operator directs the robot towards relevant elements. To extract the features associated with those elements (locations and objects), the computer initiates the CERS by running automatically the most automated procedure involved in this task, assumed to be Pr1 in figure 7. The result is graphically visualised so that the operator can check it. If the operator accepts the result, the job can continue. Otherwise, the operator selects another less automated procedure (assumed to be Pr2). If this procedure gives an acceptable result, the job can continue. Otherwise, the operator selects another gradually less automated procedure (Pr3). This process is repeated until the problem is solved. The last option corresponds to the case where the robot is fully tele-manipulated.

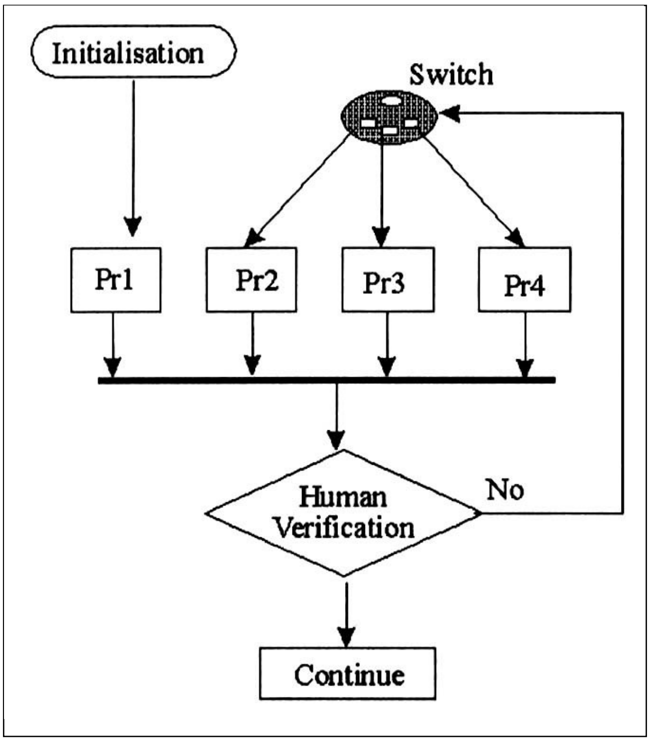


Figure 7: Co-operative Error Recovery Scheme.

This process increases the safety of the system. However, in this design, the human is still seen as the ultimate supervisor on whom the ISCS has to rely. This conception does not agree with some recent results, which prove that the operator in highly sophisticated Human-Machine Interaction Systems can be more fallible than automation [1,12,13]. That is why the CERS is enhanced by adding the anticipatory system (Figure 8). At this stage, the anticipatory system supervises the human operator. This double control illustrates the human-machine co-assistance paradigm. This design corresponds to a social-centered approach.

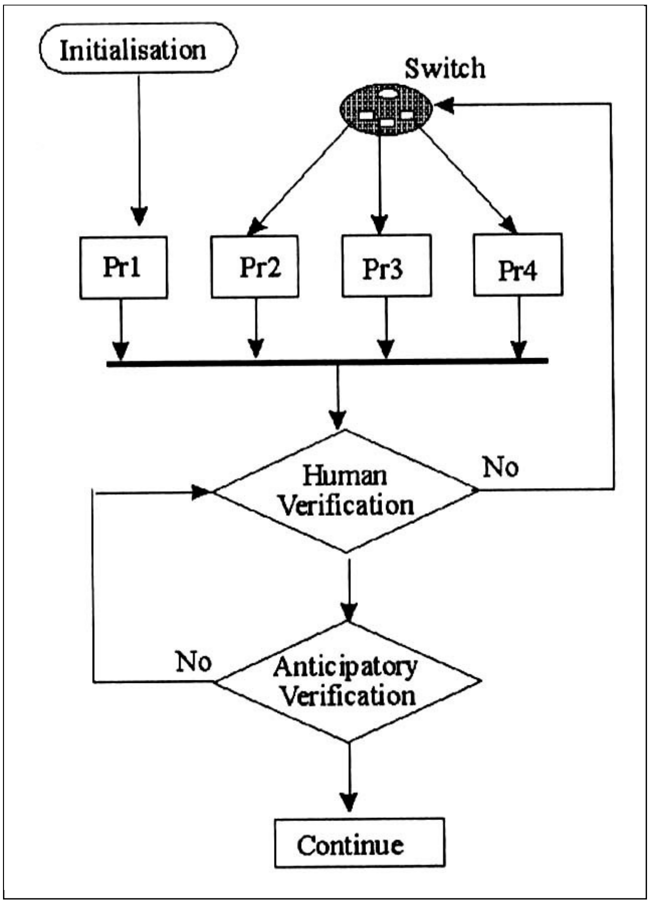


Figure 8: Co-assistance Error Recovery Scheme.

3. CONCLUSION

This paper has presented a technique developed for improving the performance of an ISCS, which is characterised by a multi-modal interaction and control. To this end, an anticipatory system has been designed and implemented in order to improve the safety and the efficiency of the click-and-move actions generated by the vision-based control system. By combining the vision and force sensors, both in the real world and in the virtual world, the anticipatory system simulates the vision-based commands, checks their impact on the system and environment safety, then enables or not their execution in the real world.

Experiments have confirmed the capability of the anticipatory system for detecting errors stemming from autonomous computational routines as well as errors and mistakes stemming from human operator.

The combination of the anticipatory system with the cooperative error recovery scheme improves remarkably the safety of the overall system and illustrates human-machine co-assistance principle.

Acknowledgments

This work was sponsored by the Belgian Program on Inter-University Attraction Poles initiated by the Belgian State –Prime

Minister's Office- Science Policy Programming. The scientific responsibility is assumed by its authors.

REFERENCES

[1]- Sheridan T.B., "Tele-robotics, Automation, and human supervisory control", MIT Press (1992).

[2]- Backes P.G., Beahan J. and Bon B., "Interactive command building and sequencing for supervised autonomy", Proc. IEEE Int. Conf. on Robotics and Automation, Atlanta, (1993,) pp.795-801.

[3]- Backes P.G. and Tsao K.S., "UMI: an interactive supervisory and shared control system for telerobotics", in: Proc. of IEEE Int. Conf. on Robotics and Automation, Cincinnati, Ohio, (1990), pp. 1096-1101.

[4]- Hirzinger J., Heindl K., Landzettel and Brunner B., "Multisensory shared autonomy- a key issue in the space robot technology experiment ROTEX", Proc. of IEEE/RSJ, Int. Conf. on Intelligent Robots and Systems, Raleigh, NC, (1990), pp. 221-230.

[5]- Zaatri A. and Van Brussel H., "An integrated multi-modal control architecture for Tele-autonomous Systems", Proc. IAS-5, Sapporo, Japan, (1998), pp. 775-783.

[6]- Van de Poel P., Witvrouw W., Bruyninckx H. and de Schutter J., "An environment for developing and optimising compliant robot motion tasks", in: International Conference on Advanced Robotics, Tokyo, (1993), pp.738-741

[7]- Zaatri A. and Van Brussel H., "Investigations in tele-robotics using co-operative supervisory modes of control", in: Proc. of the SPIE Conf, Pittsburg, PA, (1997), pp-41-52.

[8]- Viljoen J. et *al*, "Force controlled robot programming using natural language", Master's thesis, dept of computer science, Katholieke Universiteit Leuven, Belgium (1997).

[9]- Graves S., "An environment for compliant motion programming by human demonstration", Proc. of IEEE/RSJ, Int. Conf. on Intelligent Robots and Systems, (1996), pp. 795-801.

[10]- Wang Q. and de Schutter J., "Force controlled robot: Programming by Human Demonstration", Internal technical report 95R64, KULeuven, PMA (1995).

[11]- Rosen R., "Anticipatory Systems: Philosophical, Mathematical, and Methodological Foundations", Pergamon Press (1985).

[12]- Prahbu P.V. and Prahbu G.V., "Human Error and User-Interface Design", in: M. Helander, T. K. Landauer, P. Prahbu, eds. Handbook of Human-Computer Interaction, Elsevier Science B.V., (1997), pp. 485-501.

[13]- Rasmussen J., "Information Processing and Human-Machine Interaction; An approach to cognitive engineering", North-holland, Elsevier Science Publishing Co (1986). ☐