# A Visual Language for Robot Control and Programming: A Human-Interface Study

Gregory Dudek, Junaed Sattar and Anqi Xu
Center for Intelligent Machines, McGill University
Montréal, Québec H3A 2A7, Canada
{dudek,junaed,anqixu}@cim.mcgill.ca

***Abstract*— We describe an interaction paradigm for controlling a robot using hand gestures. In particular, we are interested in the control of an underwater robot by an on-site human operator. Under this context, vision-based control is very attractive, and we propose a robot control and programming mechanism based on visual symbols. A human operator presents engineered visual targets to the robotic system, which recognizes and interprets them. This paper describes the approach and proposes a specific gesture language called "RoboChat". RoboChat allows an operator to control a robot and even express complex programming concepts, using a sequence of visually presented symbols, encoded into fiducial markers. We evaluate the efficiency and robustness of this symbolic communication scheme by comparing it to traditional gesture-based interaction involving a remote human operator.**

## I. Introduction

Our work deals with the interaction between a robot and a human operator. We are interested in domains where the human and the robot work together at the same location to accomplish various tasks. In particular, our work deals with robot-human interaction underwater, where the available modes of communication are highly constrained and physically restricted. This paper describes an approach to controlling a robot by using visual gestures from a human operator (Fig. 1): for example to tell the robot to follow the operator, to acquire photographs, to go to a specified location, or to execute some complex procedure. In general the term *gesture* refers to free-form hand motions, but in this work we use to term *gesture* to refer to manual selection of symbolic markers.

We are currently developing an application in which a scuba diver underwater is joined by a semi-autonomous robot acting as an assistant. This application context serves as a motivation for the general communications problem. Conventionally, human scuba divers communicate with one another using visual hand gestures, as opposed to speech or writing. This is, of course, because the underwater environment renders acoustic and radio communication complex, costly and infeasible, and also because the physical and cognitive burden of writing or using other communication media is generally undesirable. Thus, gestures provide a natural mechanism for the diver to use in communicating with the robot. In fact, prior work involving human-robot interaction has already exploited the use of gestures for communication, although it is mediated by a human operator on the surface who interprets the gestures [**?**]. Alternative methods of controlling a robot underwater could include a keyboard or some other tactile device, but such methods can be unappealing since they entail costly waterproofing, requires physical contact between the operator and the robot, or necessitates some supplementary communications scheme between a remote device and the robot. In contrast, the proposed vision-based communication scheme can easily be materialized using cheap laminated paper, functions through passive sensing, and provides a direct interface between the user and the robot.

The authors gratefully acknowledge NSERC for supporting this research and Chris Prahacs for assisting with the robot experiments.

Fig. 1. A diver controlling the robot using visual cues.

While our approach was motivated by the desire to control a robot underwater, the methodology is more generally applicable to human-robot interaction contact. Traditional methods for human-robot interaction are based on speech, the use of a keyboard, or free-form gestures. Even in terrestrial environments each of these interfaces has drawbacks, including interference from ambient noise (either acoustic or optical), the need for proximity and physical contact, or the potential ambiguity in the transduction and interpretation process (i.e. both speech and gesture recognition systems can be error-prone). As such, our approach to robust non-contact visual robot control may have applications in diverse environments that are far more prosaic than the deep undersea.

Free-form hand gestures are clearly the most natural non-verbal means of communication. The difficulty with natural gestures is that they are very hard to interpret. This difficulty

stems from several factors, which include the repeatability of the gestures, the need to identify the operator's body parts in images with variable lighting and image content, and the possibility that the operators themselves are inconsistent with the gestures being used. These and other factors have made gesture interpretation a stimulating research area for over a decade (see Sec. II), but the unresolved challenges make it problematic for robust robot control.

Our approach is to use symbolic targets manipulated by an operator to affect robot control. By using carefully engineered targets, we can achieve great robustness and accuracy while retaining a large measure of convenience. In this paper, we both describe our approach and evaluate the precise extent to which it remains convenient and usable, and attempt to measure and quantify the loss of convenience relative to natural free-form gestures between humans. The symbolic tokens expressed by the targets are used to compose "command lists", or more accurately program fragments, in a robot control language we have developed called "RoboChat". One challenge has been to select an appropriate level of abstraction for RoboChat which is both expressive yet convenient, a domain-specific trade-off which is faced by conventional programming language designers as well.

Although the system is to be used on an underwater robot vehicle, we report a human interface performance evaluation conducted on dry land. This evaluation compares different operating modes employed to send commands in a simulated robotics context and, in some cases, includes a distractor task to replicate the cognitive load factors that arise underwater. We also report qualitative results from a test of the system with a fully deployed underwater robot, but due to the logistic constraints involved we were not able to run an actual multi-user performance evaluation underwater (the mere thought of getting that through an ethics approval process is outside the scope of this paper).

The symbolic targets we use are visual patterns that encode bit strings, but which also serve as *fiducial markers* in the video domain. Fiducial markers are visual targets that are robustly detectable and whose position can be accurately estimated. While we are examining different types of fiducials, this paper confines its attention to a class of markers called "ARTags" [**?**], which are composed of a two-dimensional configuration of achromatic binary blocks that redundantly encode binary digits with a range of roughly one through one thousand (Fig. 4).

## II. Related Work

Our work described in this paper is based on four principal ideas: a navigating underwater robot, the use of robust visual targets, gesture recognition in the abstract, and gestures for robot control.

Sattar *et al.* looked at using visual communications, and specifically visual servo-control with respect to a human operator, to control the navigation of an underwater robot [**?**]. In that work, while the robot follows a diver to navigate, their diver can only modulate the activities of the robot by making gestures that are interpreted by a human operator on the surface. Visual communication has also been used by several authors to allow communication between robots on land, or between robots and intelligent modules of the sea floor, for example in the work of Vasilescu and Rus [**?**].

The work of Waldherr, Romero and Thrun [**?**] exemplifies the explicit communication paradigm in which a robot is led through an environment and hand gestures are used to interact with it. Tsotsos *et. al* [**?**] considered an explicit gestural interface for non-expert users, in particular disabled children, based on a combination of stereo vision and keyboard-like input. As an example of implicit communication, Rybski and Voyles [**?**] developed a system whereby a robot could observe a human performing a task and learn about the environment.

Fiducial marker systems, as mentioned in the previous section, are efficiently and robustly detectable under difficult conditions. These marker systems depend on a unique encoding of information using a particular pixel pattern for the detection algorithm to work. Apart from the ARTag toolkit mentioned previously, other fiducial marker systems are available for applications similar to this work as well, although the levels of robustness, accuracy and usability vary among the toolkits. The ARToolkit marker system [**?**] consists of markers very similar to the ARTag flavour in that it contains different patterns enclosed within a square black border. ARToolkit works by outputting threshold values which measure the confidence of the presence of markers. Circular markers are also possible in fiducial systems, as demonstrated by the Photomodeler *"Coded Marker Module"* system [**?**]; the downside of using circular markers are the high rate of false positives and negatives, inter-marker confusion, and difficulty in pose estimation from a single visible marker.

Vision-based gesture recognition has long been considered for a variety of tasks and has proven to be a challenging problem with diverse well-established applications that have been enumerated and examined for over 20 years [**?**] [**?**]. The range of gestural vocabularies range from extremely simple actions (like simply fist versus open hand) to very complex languages such as the American Sign Language (ASL). ASL allows for the expression of substantial affect and individual variation, making it exceedingly difficult to deal with in its complete form. For example, Tsotsos *et al.* [**?**] considered the interpretation of elementary ASL primitives (i.e simple component motions) and achieved 86 *per cent* to 97 *per cent* recognition rates under controlled conditions.

Gesture-based robot control has been explored extensively. This includes explicit as well as implicit communication between human operators and robotics systems. Several authors have considered specialized gestural behaviours [**?**] or strokes on a touch screen to control basic robot navigation. Skubic *et al.* have examined the combination of several types of human interface components, with special emphasis on speech, to express spatial relationships and spatial navigation tasks [**?**].

## III. Methodology

Our approach to robot control using visual signalling is to have an operator describe actions and behaviours to the robot by holding up a sequence of engineered targets. Each target represents one symbolic token, and the sequence of tokens constitutes a program to modulate the robots behaviour. Because the user may be subject to conflicting demands (i.e. they may be cognitively loaded and/or distracted), we use only the token ordering to construct commands, while disregarding the timing of the presentation, the delay between tokens, and the motion of the targets within the field of view.

The specific fiducial markers we use are referred to as "ARTags". The sequence of markers constitutes utterances that express "commands" to the robot, formulated in a special-purpose language we have dubbed "RoboChat". While we refer to the utterances as commands, the semantics of the individual statements in RoboChat do not individually have to imply a change of program state.

In this paper we propose and define the RoboChat language, which is in turn used to control the robots operation by providing parameter input to RoboDevel, a generic robot control architecture [**?**]. RoboChat can be used to alter a single parameter in RoboDevel at once, or to iteratively and regularly vary parameters based on ongoing sensor feedback. In order to assess the RoboChat paradigm, we describe a series of human interaction studies, as well as a qualitative validation in the field using the AQUA underwater robot. The human interaction studies compare the performance of a user using ARTags and RoboChat with that of a diver controlling the robot using conventional hand gestures interpreted by a remote human operator. Factors that may relate to the usability of such signalling system are the extent to which the operator is distracted by other stimuli and activities, and the complexity of the vocabulary being used to control the robot. Thus we have performed some of our usability studies in the presence of a distractor task, and have also examined performance as a function of changing vocabulary size.

### A. ARTag fiducial markers

ARTags are digital fiducial markers similar to two-dimensional bar codes, engineered to be robustly recognized using efficient image processing operators [?]. The encoding process involves several algorithms that introduce redundancy to the 10-bit ID, ultimately outputting 36 bits of encoded data. Additionally, in normal operation, the likelihood of an incorrect recognition of the bit string is very low (much less than 1 *per cent*), so that the possibility of the robot receiving an erroneous command due to inter-marker confusion is slim to none.

### B. RoboChat grammar and syntax

Constructing an appropriate language for gestural control of a robot involves several competing priorities: the language must be abstract enough to be succinct, low-level enough to allow the tweaking of detailed parameters, reasonably easy to use for users with limited or no programming background, and flexible enough to allow for the specification of unanticipated behaviours by technically-oriented users conducting experiments. Two of these priorities dominate the RoboChat design: maintaining a minimal vocabulary size and allowing commands to be specified using as few markers as possible, even though commands may have optional parameters.

**BNF Grammar**

- block ::= explist [block] | funcdef [block]
- explist ::= exp [explist]
- funcdef ::= number *BEGIN* explist *END*
- exp ::= **ACTION** | number **PARAM** | number *CALL* | number *REPEAT* explist *END* | **CONDITION** *IF* explist [*ELSE* explist] *END*
- number number ::= [**UNIOP**] digit [modifier] | number **BINOP**
- modifier ::= **MODIFIER** [modifier]
- digit ::= **DIGIT** [digit]

Fig. 2. RoboChat BNF grammar.

Because this language is designed specifically to control robots, movement and action commands are treated as atomic structures in RoboChat. Different commands may require different arguments: for example, the `MOVE_FORWARD` command needs `DURATION` and `SPEED` arguments among other possible ones. Arguments are implemented as shared variables, thus after a parameter has been set once, all subsequent commands requiring this parameter will automatically acknowledge the previous value by default.

Although the structure of RoboChat is well-defined, the specific command vocabulary is task-dependent and can be expanded or substituted if necessary. However, RoboChat does have a core set of basic tokens, including numerical digits, arithmetic operators, and relational operators. Additionally, RoboChat defines a limited number of variables, including the aforementioned command parameters, as well as some general-purpose variable names.

RoboChat features two control flow constructs – the if-else statement, and the indexed iterator statement. The former construct allows the user to implement decision logic, while the latter immensely cuts down on the required number of tokens for repeated commands.

Arguably the most important feature of RoboChat is the ability to define and execute macros (i.e. macroinstructions). The user can encapsulate a list of expressions into a numerically tagged macro, which can then be called upon later. This feature allows the reuse of code, which is essential when trying to minimize the number of tokens needed to specify behaviour.

As seen from the BNF (Backus-Naur Form) grammar of RoboChat in Fig. 2, every construct is designed to minimize the number of tokens needed to express that construct. Reverse Polish notation (RPN) is heavily exploited to achieve this minimization – operators and operands are presented using RPN, eliminating the need for both an assignment

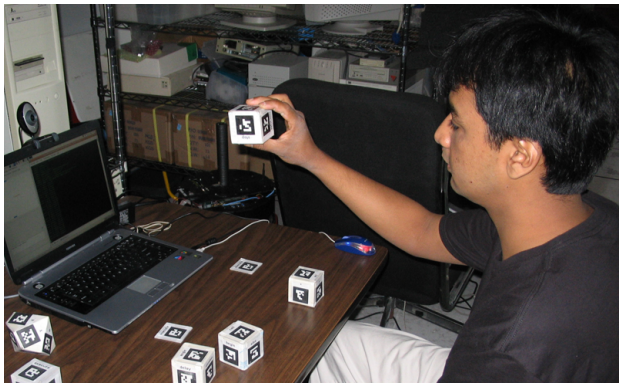

Fig. 3. A human-interface trial in progress.

operator and an end-of-command marker, while still allowing one-pass "compilation". Additionally, the use of RPN notation in the more abstract control flow constructs eliminate the need of various delimiters common to most programming languages, such as `THEN` or `{ ... }` (body code brackets).

RoboChat interprets the tokens in real time, but only executes the commands upon detection of the `EXECUTE` token. This feature allows for batch processing, and also enables the error recovery element, using the `RESET` token.

### C. Human Interaction Study

Two sets of studies were conducted using the proposed marker-based input scheme in combination with the RoboChat language, to assess their usability. In particular, the ARTag mechanism is compared to a hand gestures system, as competing input devices for environments unsuitable for the use of conventional input interfaces. The first study investigates the performance of the two systems under a stressful environment, similar to the one scuba divers must face underwater. The second study aims to compare the two input mechanisms in the presence of different vocabulary sizes. The main task in both studies is to input a sequence of action commands, with the possibility of specifying additional parameters.

The RoboChat format is used with both input devices, although in the case of the hand signal system, the gestures are interpreted by an expert human operator remotely, who subsequently validates the correctness of the input using the RoboChat syntax. This setup is realistic because in AQUAs case, the divers hand signals are interpreted by an operator on land, who then takes control of the robot. Also, the operator is not forced to be unbiased when interpreting gestures, because realistically the robot operator will guess and infer at what the diver is trying to communicate, if the hand gestures are ambiguously perceived.

Before starting each of the two sessions (using different input devices), participants are briefed on the RoboChat syntax, and are given the chance to practice using the devices and the RoboChat language, on a limited version of the experiment interface. This way, participant will have understood how the system works before attempting to carry out the full-blown experiments.

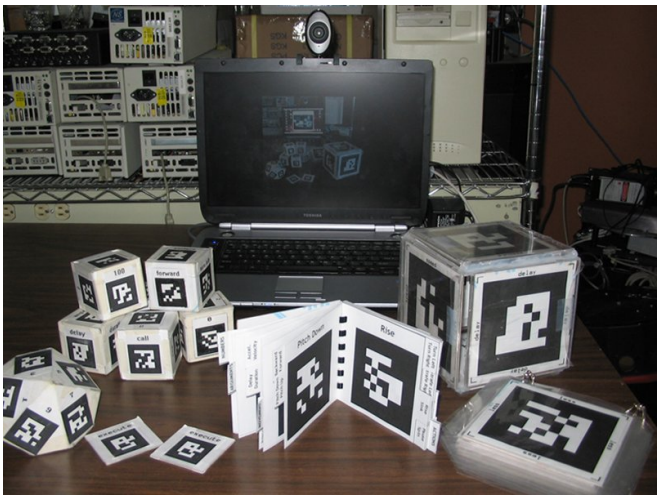

Fig. 4. Different ARTag structures used during the experiments.

*1) Study A:* In the first study, the ARTag markers are provided in groups of six, in a cubical dice configuration (Fig. 4). The participants are allowed to place the dice in any configuration in the provided work area, and they are encouraged to do so in a manner so that the cubes can be easily accessible. The hand gestures in this study are predetermined, and are visually demonstrated to the participants, who are then asked to remember all the gestures. During the experiment session, the participants must rely on memory alone to recall the gestures, much like the case for the AQUA divers.

The stress factor in the first study is introduced by asking participants to play a game of Pong (a classical 1970s table tennis video game [**?**]) during the experimental sessions. Several alternative distraction tasks were considered, and a discussion of why Pong is chosen is outside the scope of this paper, except to note that a suitable distractor task must be fairly accessible to all users, continually demanding of attention, yet still allow the core task to be achievable. This particular implementation of Pong uses the mouse to control the users paddle. As such, participants are effectively limited to using only one hand to manipulate the markers and to make out gestures, while constantly controlling the mouse with the other hand. But since some of the predefined hand gestures require the use of both hands, this distraction introduces additional stress for the participants in terms of the alternatively showing gestures and playing Pong.

In this study, the system (controlled by the operator for the hand gesture session) informs the participant when the entered command is incorrect, and proceeds onto the next command only after receiving the previous one correctly. The participants are told to complete the sessions as fast as possible, but also with as little error as possible as well.

*2) Study B:* The second study shares many similarities to the first, but the parameter of interest is no longer the participants concentration level, but rather the performance difference using different vocabulary sizes. Two vocabulary sets are given in this study the first set contains only 4 action

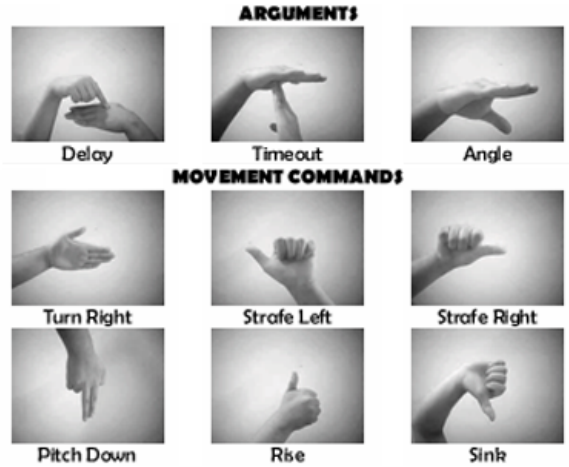


Fig. 5. A small subset of gestures presented to participants in study B.

commands, while the second includes 32. This distinction is mentioned to every participant so that they can use this information to their advantage.

Due to the increase in vocabulary size, the ARTag markers are provided in two different mediums – the digits are still offered on dice, while the whole vocabulary set is also organized into a flipbook (Fig. 4). The flipbook is separated by several high-level tabs into different token groups (such as digits, parameters and commands). Each tag sheet has a low-level tab on the side, listing the mappings for the two markers on both sides of the sheet. This feature halves the number of sheets needed, and by grouping similar mapping pairs into single sheets, it increases the access speed of the device.

The same vocabulary size issue arises for the hand gestures. Real scuba divers are required to remember many hand signals, but because it is unrealistic to ask participants to remember more than 50 different hand gestures under the experiments tight time constraints, a gesture lookup sheet is given to each participant (Fig. 5). The subjects are encouraged to familiarize themselves with this cheat sheet during the practice sessions, to ensure that they spend minimal time searching for particular hand signals.

There is no distraction factor in this second study, but at the same time, the system accepts incorrect commands without informing the participants or making them re-enter the commands. The users are informed of this criterion, and are recommended to constantly keep track of the entered tokens and try to make as few mistakes as possible.

## IV. Experimental Results

### A. Employed Criteria

Two criteria are used to compare the performance of the two input interfaces. The first criterion is speed, *i.e.* the average speed it takes to enter a command. A distinction is made between the two studies regarding this metric: in the first study, the input time per command is measured from the time a command is shown on screen until the time the command is *correctly* entered by the participant, whereas

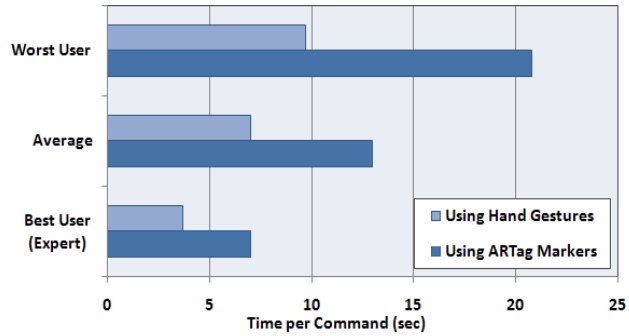


Fig. 6. Study A: Average time taken per command using ARTag markers and using hand gestures.

in the second study, the command speed does not take into consideration the correctness of the command.

The second study also uses the average time per individual tokens as a comparison metric. This metric demonstrates the raw access speeds of both input interfaces outside the context of RoboChat or any other specific environment.

The second criterion used to compare the two systems is the error rate associated to each input scheme. Once again, due to the distinction between how incorrect commands are treated between the two studies, results from this metric cannot be compared directly between studies. This criterion is used to look at whether the two schemes affect the user's performance in working with RoboChat differently.

In total, 12 subjects participated in study A, whereas 4 subjects participated in study B. One of the participants present in both studies has extensive experience with ARTag markers, RoboChat, and the hand gesture system. This expert user is introduced in the dataset to demonstrate the performance of a well-trained user. However, this user has no prior knowledge of the actual experiments, therefore is capable of exhibiting similar performance improvements throughout the sessions.

### B. Results: Study A

One obvious observation we can make from the performance data is that the gesture system allows for faster communication than the marker system. The ratio between the two input techniques for some users surpasses 3:1 favouring hand gestures, while data from other users (including those from the expert user) show ratios of lower than 2:1. Since all users have experience with primitive hand gestures, we can infer that it may simply be that those users who did almost equally well with markers as gestures adapted to the marker system more quickly. Thus, the data suggest that the ARTag markers are capable of matching half the speed of the hand gestures, even given only limited practice. It is worth noting that contrary to the hand gestures which are chosen to have intuitive and natural mappings to their corresponding tokens, the mappings between the ARTag markers and tokens are completely arbitrary.

To further substantiate the hypothesis that the enhanced performance of hand gestures is due to familiarity, note that Fig. 6 indicates that the spread of the average time per

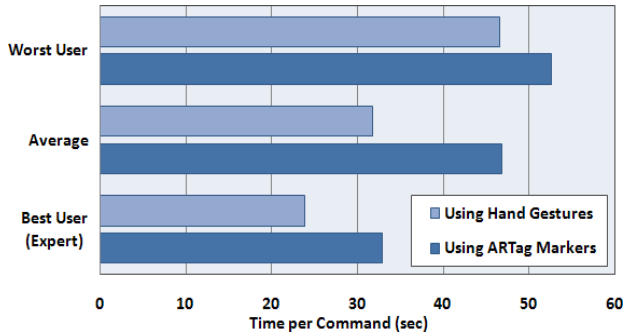


Fig. 7. Study B: Average time taken per command using ARTag markers and using hand gestures.

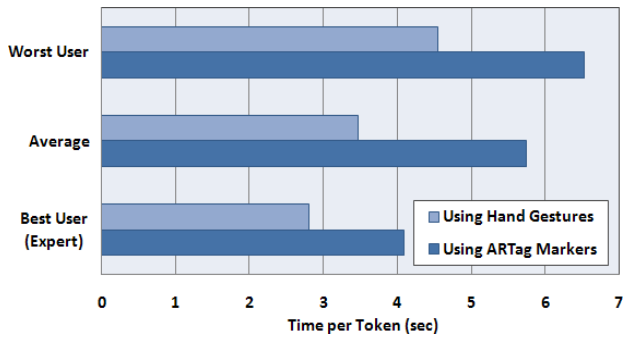


Fig. 8. Study B: Average time taken per token using ARTag markers and using hand gestures.

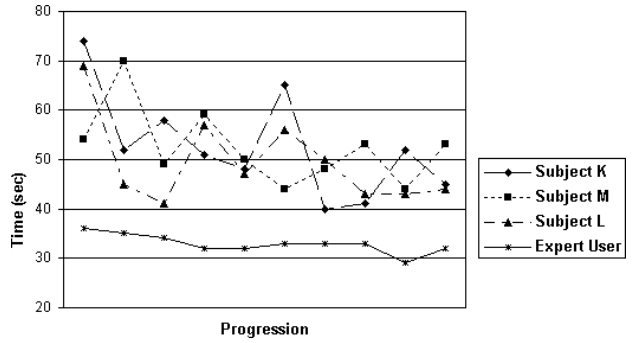


Fig. 9. Study B: Trial progression using tags.

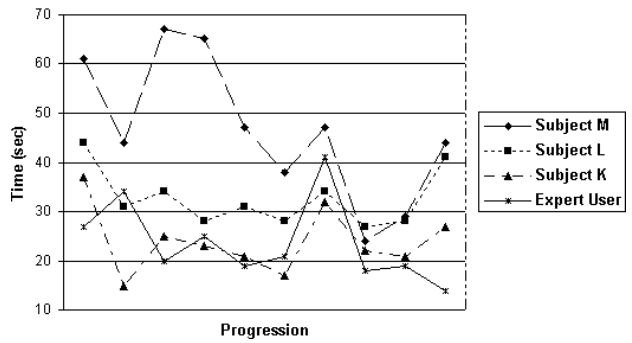


Fig. 10. Study B: Trial progression using gestures.

command using gestures ($\pm$ 3 seconds) is much smaller than that for markers ($\pm$ 8 seconds). Arguably the more sporadic spread for the markers is due to unfamiliarity with this new input interface.

The distraction task (playing Pong) also plays an important role in increasing the performance disparity between the two systems. For each token, the participants need to search through the entire ARTag vocabulary set for the correct marker, whereas the associated hand gesture can be much easily recalled from memory. Since the Pong game requires the participant's attention on an ongoing basis, the symbol search process was repeatedly disrupted by the distraction task, amplifying the marker search time.

In terms of the error rate associated with each system, all the participants displayed error rates of roughly 5 *per cent* for both systems. This finding is surprising and interesting, because even though the symbolic system is harder to learn, it does not seem to generate more errors than the gesture system, even for inexperienced users.

### C. Results: Study B

The data from study B suggests that the two input interfaces have very similar performances under the new constraints. Major contributing factors include the increase in the vocabulary size and the inclusion of many abstract action tokens (such as `RECORD_VIDEO` and `POWER_CYCLE`). This variation takes away the crucial advantage gestures had in the former study, and participants are now forced to search through the gesture sheet rather than remembering the many hand gestures. Essentially, in this study, the command speed criterion boils down to the search speed for each input device, and therefore depends on the reference structure, whether it is the ARTag flipbook or the gesture cheat sheet. And using the two engineered reference structures, the data of the experiments show that the speed performance of both input systems are actually very similar.

Interestingly enough, the data spread between systems are actually reversed. With the exception of the expert user, the average command and token speeds for all the participants using ARTag markers are almost identical, whereas the same speeds using gestures are now erratic between individuals. This result can be blamed on the fact that since the gestures are not kept in memory, different subjects adapt to the cheat sheet setup at different speeds.

One interesting observation from Fig. 9 is that the command speeds for the non-expert users seem to all feature a similar decaying factor. This gradual improvement suggests that the ARTag marker system is still relatively quick to learn. This hypothesis is reinforced by observing the constant command speeds of the expert user.

In comparison, Fig. 10 shows that the command speeds for the gesture portion of the study are more or less linear, with a chaotic nature due to the spread explanation above. This result further strengthens the theory presented in the previous section, that all the participants have prior experience with the general hand gesture system, and only have to learn the particular supplied gestures.

The two vocabulary sets employed did not have any negative impacts on the subjects' performances. Even though

most participants acknowledged the fact that the initial commands only uses a small subset of the larger vocabulary set, the difference in the input speed is too little to be significantly commented.

The expert user's data shows almost a 1:1 ratio between the two command input speeds. Because the expert user is familiar with all the specified hand gestures as well as the configuration of the ARTag flipbook, his data suggests that the ARTag markers can rival the gesture system in terms of speed, given enough training.

As for errors in the different sessions, despite that most commands were entered correctly, the `RESET` token was employed at several occasions. This result simply says that without distractions, RoboChat can be used easily without committing any non-reversible errors.

### D. RoboChat field trials

The results from our controlled usability study were corroborated by field trials in which the visual symbol-driven RoboChat interface was used on a fully autonomous robot operating underwater. These tests were conducted both in a large enclosed swimming pool at a depth of roughly 2m, and in a large open water lake at a depth that ranged from 0m to 6m. While the demands imposed by the experimental conditions precluded quantitative measurements like those in the preceding section, both the hand signals and RoboChat symbols were employed. The simple fact that the RoboChat system makes a tether unnecessary makes it very valuable. The subjective response of two divers familiar with controlling the robot is that the RoboChat system is easy and convenient to use. In general, the RoboChat controller may reduce the cognitive load on the diver, but it does imply an additional risk if the device showing the symbolic markers becomes lost or inaccessible. The system was also tested in a terrestrial environment in conjunction with a Nomadics SuperScout robot (controlled using the RoboDevel software package). This interface also appeared to be convenient, but probably would of been more effective in combination with a keyboard for low-level programming.

## V. Summary and Conclusions

We have presented a visual communication and programming paradigm for mobile robots based on visual tags. This system is optimized for operating in underwater environments, but can be used in other contexts. We evaluated it qualitatively in the field and using a controlled human interface study. This method of sending commands also enables us to operate the robot without a tether.

The use of symbolic markers is very convenient and provides several important advantages. It is somewhat surprising that such a system has not been exploited more heavily in the past, especially given its effectively as reflected from our study data. The experimental results demonstrate that the tag-based system can be at least as efficient as traditional gestural communication protocols, given enough training to the assistant. It also eliminates the need for the human operator, thereby reducing the sources of error due to communication.

Possible future research include the investigation of multiple markers to send compound commands. Another fertile topic is to combine robust symbolic targets with simple free-form motions to increase the vocabulary in an intuitive manner. We are also exploring the use of alternative marker systems that degrade more gracefully under impaired visibility. It may also be appropriate to exploit probabilistic reasoning, for example in the form of a Markov model, to improve the robustness of the language over sequences of several tokens (although this approach would imply losing some expressive power).