# Intuitive Programming with Remotely Instructed Robots inside Future Gloveboxes

Ayan Ghosh
ayan.ghosh@sheffield.ac.uk
The University of Sheffield
Sheffield, United Kingdom

Sandor M Veres
s.veres@sheffield.ac.uk
The University of Sheffield
Sheffield, United Kingdom

Daniel Peredes-Soto
d.peredes-soto@sheffield.ac.uk
The University of Sheffield
Sheffield, United Kingdom

James E Clarke
j.e.clarke@sheffield.ac.uk
The University of Sheffield
Sheffield, United Kingdom

John Anthony Rossiter
J.a.rossiter@sheffield.ac.uk
The University of Sheffield
Sheffield, United Kingdom

## ABSTRACT

Our research aims at facilitating the design of '*Remotely Instructed Robots*' for future glove-boxes in the nuclear industry. The two main features of such systems are: (1) They can automatically model the working environment and relay that information to the operator in virtual reality (VR). (2) They can receive instructions from the operator that are executed by the robot. However, the deficiency of these kind of systems is that they heavily rely on knowledge of expert programmers when the robot's capabilities or hardware are to be reconfigured, altered or upgraded. This late breaking report proposes to introduce a third important advancement on remotely instructed robots: (3) Intuitive programming modifications by operators who are non-programmers but have basic knowledge of hardware, and most importantly, have experience of the weaknesses in particular handling tasks.

## CCS CONCEPTS

• **Computer systems organization** → Robotics; • **Human Centered Computing** → Robot Programming for Non-experts.

## KEYWORDS

Human-Robot Interaction, Nuclear glove-boxes, Intuitive interfaces, Language based interactions

**ACM Reference Format:**
Ayan Ghosh, Sandor M Veres, Daniel Peredes-Soto, James E Clarke, and John Anthony Rossiter. 2020. Intuitive Programming with Remotely Instructed Robots inside Future Gloveboxes. In *Companion of the 2020 ACM/IEEE International Conference on Human-Robot Interaction (HRI '20 Companion), March 23–26, 2020, Cambridge, United Kingdom.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3371382.3378326

## 1 INTRODUCTION

Gloveboxes for treating nuclear waste require personnel to put their hands in dangerous environments as they regularly come into close proximity to nuclear materials. This is done with assumed levels of radiation under acceptable limits. However, accidents happen, and then the allowed radiation limits may exceed. A big hazard is the puncturing of an operator's glove by a sharp item and also moving parts that can tear the glove open. This makes working within a glovebox particularly hazardous. Due to the nature of this working environment, personal protective equipment is required, by which an operator's dexterity and the task's visibility may be impaired. In addition, the working environment within a glovebox can be cramped and restrictive to movements, and the views provided by the glovebox windows can be limited and obscured by dust or dirt.

For all these reasons, robotic enhancement of gloveboxes has great potential to improve operational safety and reliability [1]. With recent advent of sensor technologies, which can be implemented inside or outside the glovebox, depending on radiation levels, the way forward is to use multi-arm robotic manipulators inside gloveboxes and operate them remotely through intuitive and effective interfaces, which would ensure productivity.

In the nuclear industry it is common to rely on tele-operation and there are a multitude of available solutions [1, 4]. Mostly hand held controllers are in use for various tele-operations [13, 16]. However, recently, Jang et al. [6] developed a hands-free *leap motion* based tele-operation system [2], where the operator's hand gestures are translated into movements of the robot. There also exist exoskeleton glove interfaces [5, 9] with haptic force feedback to remotely tele-operate robotic systems.

Even with these improvements, glovebox operators typically work in shifts of 6 to 8 hours usually, and under such circumstances human-controlled tele-robotics can cause muscle fatigue [11] leading to [7] high cognitive load on the operator. Therefore, to achieve high productivity and reduce operator fatigue, the focus needs to be on developing interactions, which can improve perception of the operators and significantly reduce fatigue. With this vision, the authors developed Remotely Instructed Robots (RIRs), which still rely on human intelligence but can accept high and low level instructions (such as "*pick up object B and place it into container C*") from the operator through voice and execute tasks based on operators' descriptions and at a variety of complexity levels. The general idea is a virtual reality model of the working environment presented to the operator, and also rendering the actual robot's status as a digital twin. The operator can point to locations, interact

[1] https://www.gamechangers.technology/challenges/gloveboxes/


*HRI '20 Companion, March 23–26, 2020, Cambridge, United Kingdom*

ACM ISBN 978-1-4503-7057-8/20/03.
https://doi.org/10.1145/3371382.3378326

with the created virtual model in VR, and then communicate their intent to the system through voice.

To develop a seamless interaction with RIR manipulators, two aspects have been addressed in the past: (1) sensing and perception inside gloveboxes to produce a VR model in realtime (2) VR interaction by pointing within the virtual environment and voice. This paper proposes a third innovation that we identified from practical industry experience: (3) intuitive reconfiguration and programming by non-expert operators is needed. The benefits, which such novel HRI can bring about, include cost effectiveness, increased safety, higher productivity and reduced operator workload. The main purpose of this paper is to introduce an approach to designing an implementing an intuitive programming interface for non-expert users of RIRs.

## 2 INTUITIVE PROGRAMMING FOR RIR

The nuclear industry is conservative in the uptake of robotic solutions as these are designed in controlled environments and therefore do not cater for dynamic real-world deployments, where, without expert programmers, systems often become limited. In order to adapt changes in robotic capabilities and hardware, it is necessary to have a simplified programming interface of RIRs, so that non-programmer operators can easily create program sequences, according to their needs. This will significantly reduce the time and cost of operations, by enabling rapid programming and training without specialised skills. Primarily, it removes the requirement for a trained programmer to interpret user requirements and then translate them into lines of code for a compiler, because this process is prone to communication errors, leading to slipped/missed requirements. Although previous research [8, 12, 12] have highlighted the need for intuitive programming interfaces for non-expert robot users, the topic has not been researched thoroughly in the field of robotics for nuclear industry.

Thus, the main focal point is to design an operator's programming interface through natural language programming (NLP) [3, 10, 14, 15], which is an ontology assisted way of programming using controlled natural language sentences. Each sentence can carry out an atomic robotic action in the environment and an ontology is basically a formal description of the data structures that can be used in sentences. Figure 1 depicts an example of NLP ontology classes for RIR's perception and sensing system, where ">..>" indicates class hierarchies, "@" is for class properties with types after ":". A set of NLP sentences is provided to the robot user, with the associated ontology. Relying on these, users are able to create a customised configuration of their system by arranging the sentences. The user can quickly understand how the system has been designed, and how it will behave in operation, without having the knowledge of low level program steps. Documentations of industry procedures become convenient. The syntax of the program scripts needs no further descriptions or comments to be understood, so the program itself can be converted directly into an easily readable document for the operators and managers. Figure 2 depicts an example arrangement of sentences that can be used to design RIR's perception system.

```
>counter
        @value : number

>spatial model
  @name: string
  >>surfaces model
        @point cloud: real_array
        @texture addons: mapping<string,image frame>
    >>>object set model
                @objects:  list<physical object>
                @object headings:  list<RealArray>
                @locations:  list<RealArray>
                @relative positions: TextArray

>global world model
  @local worlds: list<local world model>
  @model correspondences: mapping<string,string>

>local world model
  @name: string
  @separable objects:  object set model

>typed input
  @value: string

>camera state
  @euler angles:RealArray
  @positon vector:RealArray

>camera images
  @time stamp:  RealArray
  @camera names: list<string>
  @camera positions: list<camera state>
  @camera data: list<image frame>

>image frame
  @number of rows: integer
  @number of columns: integer
  @number of colours: integer
  @data: IntArray8

>image survey
  @images: list<camera images>
```

**Figure 1: Ontology classes for RIR's perception and sensing system**

```
Initialise counter Count by 1.

Define Ims as an 'image survey'.
Initialize Lm as a "local world model'.
Note that "process needs speeding up".

While (Count is less than 30.
        && not (Lm is complete relative to Ims.) )
   [
    Obtain current camera images Cams. Preprocess Cams.
    Add Cams to image survey Ims. Integrate Ims into Lm.
    Move cameras into new view based on Ims and Lm.
    If ('process needs speeding up') then [Increment Count by 3.]
      otherwise [Increment Count.] .
   ].
```

**Figure 2: Example sentences for robot perception based on Figure 1 ontology**

## 3 CONCLUSION AND FUTURE WORK

This late breaking report introduced the concept and the need for an intuitive programming interface to configure robotic manipulators inside nuclear gloveboxes, which can be used by non-expert robot users. Such an interface is currently being implemented in our laboratory environment. The next step will be to quantitatively and qualitatively evaluate the interface with our end users. Hence we are in the process of setting up a test bed in collaboration with our industry partners, with whom trials will be carried out. Using our demonstrator, placed at our test bed, industrial users will be able to appreciate the benefits of our system for their adoption. They will be able to enhance their system's AI capabilities with precision and also expand their operational capabilities.

## REFERENCES


[1] Jordan Allspaw, Jonathan Roche, Nicholas Lemiesz, Michael Yannuzzi, and Holly A Yanco. 2018. Remotely teleoperating a humanoid robot to perform fine motor tasks with virtual reality–. In *Proceedings of the 1st International Workshop on Virtual, Augmented, and Mixed Reality for HRI (VAM-HRI)*.

[2] Laura Cancedda, Alberto Cannavò, Giuseppe Garofalo, Fabrizio Lamberti, Paolo Montuschi, and Gianluca Paravati. 2017. Mixed Reality-Based User Interaction Feedback for a Hand-Controlled Interface Targeted to Robot Teleoperation. In *International Conference on Augmented Reality, Virtual Reality and Computer Graphics*. Springer, 447–463.

[3] Michael D Ernst. 2017. Natural language is a programming language: Applying natural language processing to software development. In *2nd Summit on Advances in Programming Languages (SNAPL 2017)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[4] Peter F Hokayem and Mark W Spong. 2006. Bilateral teleoperation: An historical survey. *Automatica* 42, 12 (2006), 2035–2057.

[5] Haiying Hu, Jiawei Li, Zongwu Xie, Bin Wang, Hong Liu, and Gerd Hirzinger. 2005. A robot arm/hand teleoperation system with telepresence and shared control. In *Proceedings, 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. IEEE, 1312–1317.

[6] Inmo Jang, Joaquin Carrasco, Andrew Weightman, and Barry Lennox. 2019. Intuitive Bare-Hand Teleoperation of a Robotic Manipulator Using Virtual Reality and Leap Motion. In *Annual Conference Towards Autonomous Robotic Systems*. Springer, 283–294.

[7] Kanav Kahol, Mario J Leyba, Mary Deka, Vikram Deka, Stephanie Mayes, Marshall Smith, John J Ferrara, and Sethuraman Panchanathan. 2008. Effect of fatigue on psychomotor and cognitive skills. *The American Journal of Surgery* 195, 2 (2008), 195–204.

[8] Ying Siu Liang, Damien Pellier, Humbert Fiorino, and Sylvie Pesty. 2017. Evaluation of a robot programming framework for non-experts using symbolic planning representations. In *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 1121–1126.

[9] Neal Y Lii, Zhaopeng Chen, Benedikt Pleintinger, Christoph H Borst, Gerd Hirzinger, and Andre Schiele. 2010. Toward understanding the effects of visual- and force-feedback on robotic hand grasping performance for space teleoperation. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 3745–3752.

[10] Nicholas K Lincoln and Sandor M Veres. 2013. Natural language programming of complex robotic BDI agents. *Journal of Intelligent & Robotic Systems* 71, 2 (2013), 211–230.

[11] Nurhayati Mohd Nur, Siti Zawiah Md Dawal, Mahidzal Dahari, and Junedah Sanusi. 2015. Muscle activity, time to fatigue, and maximum task duration at different levels of production standard time. *Journal of physical therapy science* 27, 7 (2015), 2323–2326.

[12] Eric M Orendt, Myriel Fichtner, and Dominik Henrich. 2016. Robot programming by non-experts: Intuitiveness and robustness of one-shot robot programming. In *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 192–199.

[13] Daniel Rakita, Bilge Mutlu, and Michael Gleicher. 2018. An Autonomous Dynamic Camera Method for E ective Remote Teleoperation. (2018).

[14] S.M. Veres. 2008. *Natural Language Programming of Agents and Robotic Devices: Publishing for Humans and Machines in sEnglish*. SysBrain Ltd.

[15] S. M. Veres. 2010. Theoretical foundations of natural language programming and publishing for intelligent agents and robots. In *TAROS 2010, Towards Autonomous Robotic Systems, UK Conference*. Plymouth, UK.

[16] David Whitney, Eric Rosen, Elizabeth Phillips, George Konidaris, and Stefanie Tellex. 2017. Comparing robot grasping teleoperation across desktop and virtual reality with ros reality. In *Proceedings of the International Symposium on Robotics Research*.