Towards Robot Arm Training in Virtual Reality Using Partial Least Squares Regression

Benjamin Volmer[*] University of South Australia

Adrien Verhulst[†] The University of Tokyo.

Adam Drogemuller[‡] University of South Australia.

Bruce H. Thomas[§] University of South Australia

Masahiko Inami[¶] The University of Tokyo

Maki Sugimoto[‖] Keio University

Abstract

Robot assistance can reduce the user's workload of a task. However, the robot needs to be programmed or trained on how to assist the user. Virtual Reality (VR) can be used to train and validate the actions of the robot in a safer and cheaper environment. In this paper, we examine how a robotic arm can be trained using Coloured Petri Nets (CPN) and Partial Least Squares Regression (PLSR). Based upon these algorithms, we discuss the concept of using the user's acceleration and rotation as a sufficient means to train a robotic arm for a procedural task in VR. We present a work-in-progress system for training robotic limbs using VR as a cost effective and safe medium for experimentation. Additionally, we propose PLSR data that could be considered for training data analysis.

Index Terms: Human-centered computing—Human computer interaction (HCI)—Interaction paradigms—Virtual reality; Robot arm

1 Introduction

Performing a task of any variety requires cognitive effort, regardless of difficulty. While the assistance of another person can reduce ones workload, it requires someone else to be present. Robot assistance can replicate another user and thus reduce the workload of a task. To perform a task, the robot has to understand how to assist the user, either through a programmed sequence or from training.

However, having access to the robot for training may sometimes not be possible due to the cost or size of the robot. There is also the unlikely event of something going wrong, which may result in injury. We believe Virtual Reality (VR) is a safer and cost-effective approach to train a robot as the real world scenario can be replicated virtually, thus it can be controlled with ease and any issues will not result in injury. Additionally, the training from the VR system could be transferred to the physical robot system.

In this paper, we observe the possibility of a robotic arm being trained in VR through the use of Coloured Petri Nets (CPN) [5] and Partial Least Squares Regression (PLSR) [2]. CPN and PLSR were chosen as they have been used for training robot arms in a real world scenario [1]. There are also other methods for training a robot which will not be discussed in this paper, as we primarily focus on the CPN and PLSR concept. We present our work-in-progress system and further discussion on how PLSR data could be used to analyze training data.

*e-mail: benjamin.volmer@mymail.unisa.edu.au

†e-mail: adrienverhulst@star.rcast.u-tokyo.ac.jp

‡e-mail: adam.drogemuller@mymail.unisa.edu.au

§e-mail: bruce.thomas@unisa.edu.au

¶e-mail: inami@star.rcast.u-tokyo.ac.jp

‖e-mail: maki.sugimoto@keio.jp

Figure 1: User performing the training task. (Left) Real World scene, (Right) Virtual Scene.

2 Background

As robotic technology evolves, VR has been used in combination with robotics for many fields such as user training and rehabilitation [3, 4]. Robots can assist users, through programmed actions or provide users with extensions to their own movement. For instance, Saraiji et al. used robot arms that moved based on the user's legs to give them additional limbs to complete tasks [7].

CPN [5], see Fig. 2, is a graphical model representation often used for concurrent and synchronization systems. CPN primarily focuses on having *places*, *transitions* and *arcs*. Places are a state in the CPN represented by an eclipse. Transitions are an action between two places, represented by a rectangle. Finally, arcs are represented by arrows that can determine if a transition should occur.

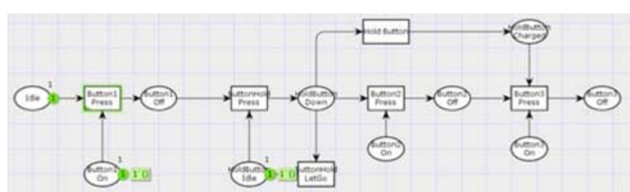

Figure 2: Coloured Petri Net (CPN) Diagram of this Study.

Partial Least Squares Regression (PLSR) [2] is a statistical comparison of two matrices, X and Y. From the comparison, various information can be obtained, such as the variance percentage of each measured component and predictions between the two matrices. Bonilla and Asada combined CPN and PLSR, to train a robot to assist users complete a ceiling tile procedural task and examined data

2019 IEEE Conference on Virtual Reality and 3D User Interfaces
23-27 March, Osaka, Japan
978-1-7281-1377-7/19/$31.00 ©2019 IEEE

such as the percent variance [1]. The future research of this paper differs from Bonilla and Asada's work by examining vector-based variables from a VR scene.

3 System Design

The system described in this paper was designed using Unity3D 2018[1] with real-time tracking using Optitrack and Oculus CV1. Tracking was performed by 10 Optitrack Prime 13[2] cameras and 6 Optitrack Prime 17W[3] cameras with approximately $10.3m^2$ ($4.37m \times 2.37m$) of tracking space. The user's body was tracked using an Optitrack Motion Capture Suit[4] with data collection mainly focusing on the user's hands, left shoulder and head. Two tables were setup to provide physical feedback to the user when performing the task (see Fig. 1). The robot arm was mapped to the user's left arm using the inverse kinematic functions of FinalIK[5].

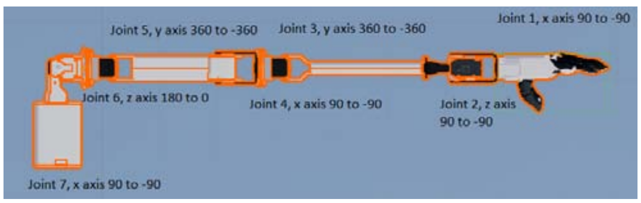

Figure 3: Virtual Robot Arm Joint data.

3.1 Task Design

We developed a task based upon the starting sequence for starting up a machine such as an aircraft. This was simplified to become like an arcade machine to develop a manageable CPN as shown in Fig. 2. Comparing the virtual scene and CPN, the orange square button represents button hold, button 1 is blue, button 2 is green and button 3 is red. For this task, the user's right arm was mapped to the avatar's right arm and their left arm controlled the robotic arm for training data. Once the user has completed the sequence, they move their hands to the blue boxes, which starts the next sequence.

3.2 Data Collection

As PLSR requires two matrices for observations, we collected data from the user for matrix X and from the robot arm joints for matrix Y (see Fig. 3). The user data we collected consisted of the square magnitude and acceleration on the x, y and z axis, alongside the square magnitude and angular rotation based on the local euler angles on all 3 axis. These 8 variables were collected from the user's right hand, left shoulder (where the robot arm was attached) and head, for a total of 24 data points. The user data obtained was based on the data collected by Bonilla and Asada [1]. Additionally, we gathered head movement data from the user for future observations on its effects on the PLSR model. The robot arm data consisted of 7 different joints which would each rotate on a single axis, for a total of 7 data points for the Y matrix (see Fig. 3). From this data, we plan to use the 24 data points from the user to examine the variance of the 7 robot joint rotations in PLSR. Through obtaining training data, the VR environment was tested, and the robot arm was found to accurately follow the user's left arm's motion. From initial observation, the system was easy and comfortable to use, thus we believe using a VR robot arm for training can be advantageous.

[1]https://unity3d.com/
[2]https://optitrack.com/products/prime-13/
[3]https://optitrack.com/products/prime-17w/
[4]https://optitrack.com/products/motion-capture-suits/
[5]http://www.root-motion.com/final-ik.html

3.3 PLSR Analysis Plan

With the obtained training data, we plan to observe gathered data in R Studio[6] using the pls package. From Bonilla and Asada's [1] study, we can observe that PLSR examines percentage of variance to describe the output data variance in relation to the input data. Furthermore, Mevik and Wehrens [6] have designed a guide to the pls package in R studio, which provides useful information on PLSR analysis. In particular, it describes the cross-validation data being able to represent how important each component is, when evaluating them for predictions. From these observations and the pls package guide, we aim to discover if the user data gathered can be used to predict the robot arm data and thus work towards training the robot arm in VR.

4 Conclusion

In this paper, we designed a VR procedural task based on a CPN. We gathered vector based training data in VR, inspired by current real world robotics [1]. Furthermore, we have proposed our plans for training a robot arm using PLSR data based upon existing research. For future research, we hope to further investigate PLSR elements and apply the training data gathered to the PLSR model. Additionally, we aim to use the PLSR analysis to train the robot in the VR application to determine if a CPN transition should occur or not. Finally, we wish to utilize our work obtained from the VR environment and apply it to a real robotic arm for testing, similar to the MetaArms system [7].

Acknowledgments

The authors thank the JST ERATO Grant Number JPMJER1701, Japan for supporting this project. Additionally, the Australian Research Council and the University of South Australia.

References

[1] B. L. Bonilla and H. H. Asada. A robot on the shoulder: Coordinated human-wearable robot control using coloured petri nets and partial least squares predictions. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 119–125, May 2014. doi: 10.1109/ICRA.2014.6906598

[2] V. Esposito Vinzi, W. W. Chin, J. Henseler, and H. Wang. *Handbook of partial least squares: Concepts, methods and applications*. Heidelberg, Dordrecht, London, New York: Springer, 2010.

[3] E. Freund and J. Rossmann. Projective virtual reality: bridging the gap between virtual reality and robotics. *IEEE Transactions on Robotics and Automation*, 15(3):411–422, June 1999. doi: 10.1109/70.768175

[4] A. Frisoli, L. Borelli, A. Montagner, S. Marcheschi, C. Procopio, F. Salsedo, M. Bergamasco, M. C. Carboncini, M. Tolaini, and B. Rossi. Arm rehabilitation with a robotic exoskeleleton in virtual reality. In *2007 IEEE 10th International Conference on Rehabilitation Robotics*, pp. 631–642, June 2007. doi: 10.1109/ICORR.2007.4428491

[5] K. Jensen and L. M. Kristensen. *Coloured Petri nets: modelling and validation of concurrent systems*. Springer Science & Business Media, 2009.

[6] B.-H. Mevik and R. Wehrens. Introduction to the pls package. 2015.

[7] M. Y. Saraiji, T. Sasaki, K. Kunze, K. Minamizawa, and M. Inami. Metaarms: Body remapping using feet-controlled artificial arms. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, UIST '18, pp. 65–74. ACM, New York, NY, USA, 2018. doi: 10.1145/3242587.3242665

[6]https://www.rstudio.com/