Vision-Force Interface for Path Specification in Tele-Manipulation

Camilo Perez Quintero[1], Masood Dehghan[1,2], Oscar Ramirez[1], Marcelo H. Ang[2] and Martin Jagersand[1]

***Abstract*— This work presents a novel vision-force based interface; allowing a local operator to visually specify a path constraint to a remote robot manipulator. By using our interface the operator is able to control a 7-DOF arm with only 2-DOF. Our interface aims to reduce task complexity and operator cognitive load. We illustrate our approach with 2 mock-up examples: 1) Closing a steam valve, 2) Gluing a T PVC pipe.**

I. Introduction

A key factor enhancing robots physical interaction performance is the establishment of intuitive and effective human-robot interfaces. These are capable of defining manipulation tasks easily and providing share control flexibility between the operator and the robot. For instance Leeper *et al.* [1] present an interface for human-in-the-loop grasp execution where different grasping strategies are provided to the operator to work in cluttered environments. The *VIBI interface* helps disabled users to visually select the object and the grasping approach by augmenting user capabilities through different levels of semi-autonomy [2]. Similar to the above examples, the majority of tele-manipulation interfaces in the literature rely only on vision-based approaches. However, as pointed out in [3], force interaction consideration could enhanced task performance substantially. Thus developing vision-force based methods are desirable. An interesting approach consist in generating visual-force active constraints [4], [5]. These constraints are typically used to restrict the movement of the robot to certain regions of its workspace or to guide the operator through a specific path [4]. Instead of achieving the complete task autonomously, these approaches aim to reduce task complexity and operator workload. Active constrains can be applied in many real world scenarios, for example polishing/finishing of mechanical parts [6], taping, welding, automotive inspection [7], robot-assistive surgery [8], or space exploration [9].

This work proposes a new vision-force interface that enables robots to perform manipulation task in the remote unstructured environment. The interface allows the operator to intuitively generate a 3D path constraint for the remote manipulator on top of the image stream coming from the remote location. Based on visual constraints defined by the human, active constrains are created to guide the operator in the remote location, improve operator's performance and reduce the workload.

[1]C. Perez, O Ramirez, M. Dehghan and M. Jagersand are with Department of Computing Science, University of Alberta, Canada, jag@cs.ualberta.ca

[2]M. Dehghan and M. H. Ang are with the Advacned Robotics Center, Department of Mechanical Engineering, National University of Singapore and SIMTech-NUS joint lab, Singapore, masood@nus.edu.sg

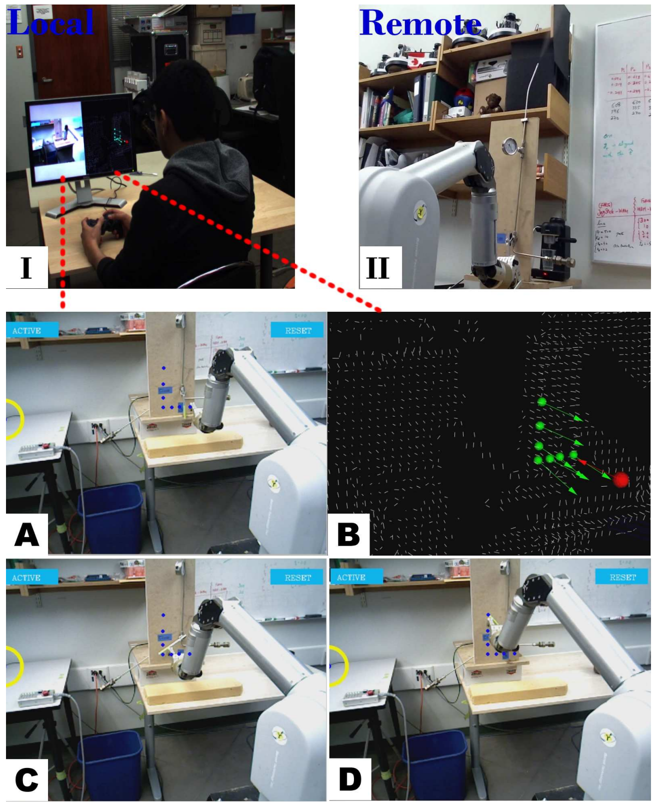

Fig. 1. The use of our proposed interface is illustrated by closing a steam valve in a remote location. I) The local operator is provided with a monitor displaying our interface and a game-pad for interaction. II) In the remote site a 7-DOF WAM arm is teleoperated with the aid of our interface to close a steam valve. **A** and **B** show the RGB and Point Cloud visualization of our interface. A) The operator defines a desired path in the 2D image by clicking (blue dots). B) The 3D path and the associated normals to the surface are calculated and visualized (green spheres and green arrows). C, D) Once the constrains are activated the user can complete the task by simply controlling 2-DOF: motion through the path direction and motion normal to the surface.

II. Interface Description

The general task that we are addressing is to teleoperate a robot manipulator while guiding it through a desired path (see Fig. 1). A block diagram of our proposed interface is shown in Fig. 2.

A. *Visual Interface and Vision Module*

The *Visual Interface* is shown in Fig. 1 A and B. Both visualizations RGB and point cloud are register such that for each pixel in the 2D image a 3D point exists in the point cloud visualization. The interface is equipped with a drawing tool that allows the operator to define a desired trajectory on the image stream from the remote site. The interface also

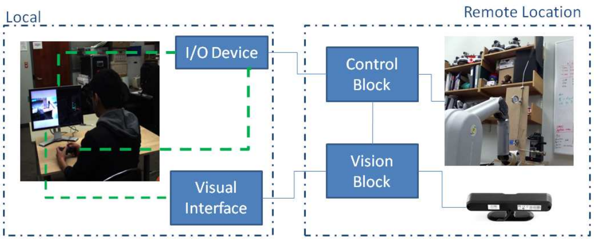

Fig. 2. System overview. Left: The I/O devices that the local operator interacts with are the mouse, screen and game-pad. The first two are used for monitoring the remote location and for specifying the desired path constraints. Right: The vision and control modules are in charge of processing the RGB-D sensor information and constraining the 7-DOF WAM arm to the specified path.

provides a real time point cloud visualization in which the operator can corroborate visually the selected 3D path and the corresponding estimate of the surface normals.

The *Vision module* streams the video coming from an RGB-D sensor in the remote location to the local operator through the visual interface. In our setup we utilize a Kinect sensor. The Vision module feeds the *Control module* with the 3D path coordinates and corresponding surface normals with respect to the robot reference frame. The 3D path coordinates are found using the direct correspondence of the RGB sensor and depth sensor. A common approach to calculate surface normals consists of fitting a plane to the neighborhood P_i of the target point p_i. P_i is either formed by the k nearest neighbors of p_i or by points within a radius r from p_i. Given P_i, its covariance matrix $\boldsymbol{C}_i \in \mathbb{R}^{3x3}$ is computed (see [10] for details). The eigenvectors of the covariance matrix, $\boldsymbol{v}_{i,0}$, corresponding to the smallest eigenvalue $\lambda_{i,0}$, can be used as an estimate of the target normal $\hat{\boldsymbol{n}}_i$ [10].

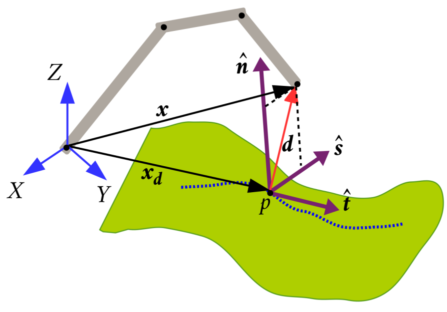

Fig. 3. Schematic of the path on a surface and construction of the corresponding path reference frame.

B. Control Module

The goal of the *Control Module* is to constrain the robot end-effector to the path on a surface, while maintaining a contact force (F_n) and keeping the tool frame aligned with the surface normal. This is achieved by projecting the motion onto orthogonal directions along the tangential ($\hat{\boldsymbol{t}}$), normal ($\hat{\boldsymbol{n}}$) and side directions ($\hat{\boldsymbol{s}}$) as shown in Fig. 3. A dedicated controller is then designed for each direction. An additional controller is also required for orienting the robot's tool. The proposed controller architecture is shown in Fig. 4 which is composed of four sub-controllers, namely 1) path controller, 2) normal force controller, 3) tangential force controller and 4) orientation controller. The inputs to the controller are (a) the desired path, which is a collection of 3D points on the surface ($\{\boldsymbol{x}_{d_i}\}$) and (b) the corresponding normal directions at each point ($\{\hat{\boldsymbol{n}}_i\}$).

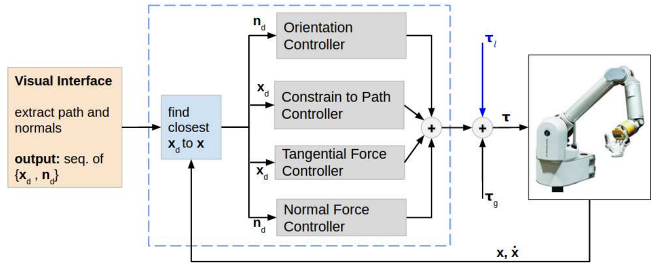

Fig. 4. Control architecture of the remote robot.

Both *normal* and *tangential* controllers allows the operator to vary contact force and tool acceleration through the path. Given the desired path, the tangential direction at each point is computed first by $\hat{\boldsymbol{t}}_i = \boldsymbol{x}_{d_i} - \boldsymbol{x}_{d_{i-1}}$. From this and the given normal direction, the path reference frame is fully characterized by $\{\hat{\boldsymbol{n}}, \hat{\boldsymbol{s}}, \hat{\boldsymbol{t}}\}$, where $\hat{\boldsymbol{s}} = \hat{\boldsymbol{n}} \times \hat{\boldsymbol{t}}$ (see Fig. 3).[1] The *path controller* is responsible for keeping the end-effector constrained to the path by applying corrective forces in the $\hat{\boldsymbol{s}}$ direction. To this end, let $\boldsymbol{d} = \boldsymbol{x} - \boldsymbol{x}_d$ be the minimum distance of the end-effector from the path. A PD controller is designed to minimize the projection of $\boldsymbol{d}$ onto $\hat{\boldsymbol{s}}$. Similarly, a PD controller is used as the *orientation controller* to align the end-effector normal to the surface.

III. Experiments

We test our interface by completing two anecdotal examples: 1) Closing a steam vale. 2) Gluing a T PVC pipe. Demonstration of our interface for these tasks can be seen in [11].

A. Closing a steam valve

Similar to the the ball valve task in the DARPA robotics challenge [12], the steam valve task consists of closing a ball valve 90° clockwise to stop a steam leak. In our case the required force to activate the valve was 7 N. Figure 1.II shows the remote setup; after the path is defined by the local operator, the robot moves to the initial position on the path and then the force controllers are activated. Using the game-pad, the operator can move the robot's end-effector forward or backward through the path direction and also through the surface normal direction. The controller provides sufficient compliance to protect the robot in case of undesired collisions.

[1] $\hat{\boldsymbol{n}}, \hat{\boldsymbol{s}}, \hat{\boldsymbol{t}}$ represent the normal, side and tangential directions of the path, respectively.

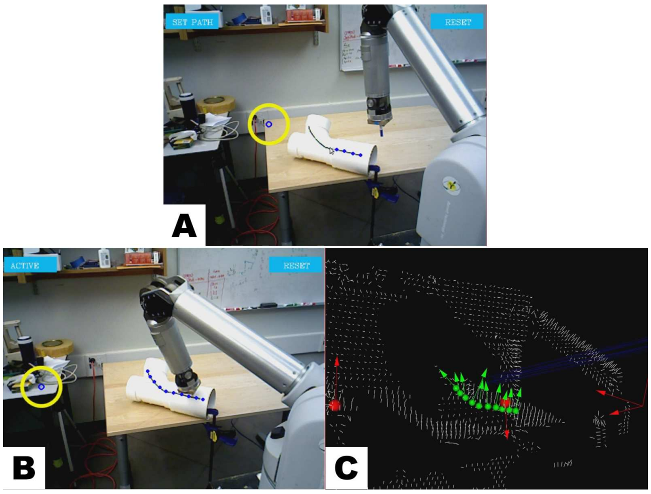

Fig. 5. Gluing a T PVC pipe: A) The user defines a path on top of the black crack. B) Once the constraints are activated, the user can apply glue by moving the end-effector through the crack. C) The scene point cloud is visualized with the selected path (green spheres) and the calculated normals (green arrows).

B. *Gluing a T PVC pipe*

Our interface allows to specify different magnitudes of normal contact force to complete tasks like welding, polishing, painting, etc. which require moving through a specific path while making contact with a surface. In this example the objective is to apply glue to a crack on a PVC pipe, see Fig. 5. During our tests we notice a drift of approximately 1^{cm} to 2^{cm} between the real path to the one followed by the robot. This is due to the Kinect depth error (from a few millimetres up to about 4 cm at the maximum range of the sensor) and calibration issues between the Kinect sensor and the robot.

IV. Future Work & Conclusions

We present the design and implementation of a force-vision-based interface that enables the users to visually specify the path constraints for tele-manipulation. We noticed that by planing the robot trajectory beforehand, the task is carefully laid out and obstacles that could be hit during direct teleoperation would be avoided during the task specification. We see three future work directions to improve our interface: 1) We are planning to perform a user study to compare the effectiveness of our proposed interface in unilateral (game-pad) and bilateral (control device with force feedback) configurations. Although in a direct teleoperation a bilateral configuration may surpass the unilateral, we believe that by using our interface the unilateral configuration could achieve a similar performance. 2) Allowing the operator to specify multiple tasks. Figure 6 shows a cake cutting mock-up task. The task consist of performing an initial round cut in the middle (blue path) and then a slice cut (red path). Multiple paths specifications enables to break down a complex task into several simple sub-tasks. 3) For the tasks that require high precision, we are planning to include two eye-in-hand cameras and develop a hybrid image based visual servoing [13] combined with our current implementation to improve the precision.

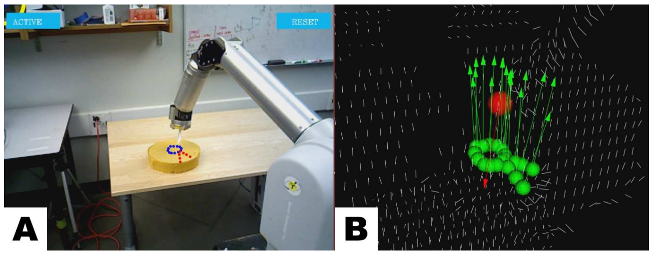

Fig. 6. A) A circular path (blue dots) and a triangular path (red dots) are defined; the user can switch between both the paths to complete the cutting task. B) The scene point cloud is visualized with both paths (green spheres) and the calculated normals (green arrows).

References

[1] A. Leeper, K. Hsiao, M. Ciocarlie, L. Takayama, and D. Gossow, "Strategies for human-in-the-loop robotic grasping," in *Human-Robot Interaction (HRI), 2012 7th ACM/IEEE International Conference on*, March 2012, pp. 1–8.

[2] C. P. Quintero, O. Ramirez, and M. Jagersand, "Vibi: Assistive vision-based interface for robot manipulation," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, May 2015, pp. 4458–4463.

[3] G. Tholey, J. P. Desai, and A. E. Castellanos, "Force feedback plays a significant role in minimally invasive surgery: results and analysis," *Annals of surgery*, vol. 241, no. 1, pp. 102–109, 2005.

[4] S. A. Bowyer, B. L. Davies, and F. Rodriguez y Baena, "Active constraints/virtual fixtures: A survey," *Robotics, IEEE Transactions on*, vol. 30, no. 1, pp. 138–157, 2014.

[5] T. Xia, S. Leonard, I. Kandaswamy, A. Blank, L. Whitcomb, and P. Kazanzides, "Model-based telerobotic control with virtual fixtures for satellite servicing tasks," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, May 2013, pp. 1479–1484.

[6] A. Y. C. Nee, *Handbook of Manufacturing Engineering and Technology*. Springer Publishing Company, Incorporated, 2014.

[7] D. Nakhaeinia, R. Fareh, P. Payeur, and R. Laganiere, "Trajectory planning for surface following with a manipulator under rgb-d visual guidance," in *Safety, Security, and Rescue Robotics (SSRR), 2013 IEEE International Symposium on*, Oct 2013, pp. 1–6.

[8] A. M. Okamura, "Methods for haptic feedback in teleoperated robot-assisted surgery," *Industrial Robot: An International Journal*, vol. 31, no. 6, pp. 499–508, 2004.

[9] S. Vozar, S. Leonard, P. Kazanzides, and L. Whitcomb, "Experimental evaluation of force control for virtual-fixture-assisted teleoperation for on-orbit manipulation of satellite thermal blanket insulation," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, May 2015, pp. 4424–4431.

[10] R. B. Rusu, *Semantic 3D object maps for everyday robot manipulation*. Springer, 2013.

[11] https://webdocs.cs.ualberta.ca/~vis/HRI/VIFS.wmv, May 2016.

[12] M. DeDonato, V. Dimitrov, R. Du, R. Giovacchini, K. Knoedler, X. Long, F. Polido, M. A. Gennert, T. Padır, S. Feng, *et al.*, "Human-in-the-loop control of a humanoid robot for disaster response: A report from the darpa robotics challenge trials," *Journal of Field Robotics*, vol. 32, no. 2, pp. 275–292, 2015.

[13] R. T. Fomena, C. P. Quintero, M. Gridseth, and M. Jägersand, "Towards practical visual servoing in robotics," in *Tenth Conference on Computer and Robot Vision, CRV 2013, Regina, Saskatchewan, Canada, May 28-31, 2013*, 2013, pp. 303–310.