



# Transfer of learned dynamics between different surgical robots and operative configurations

Nural Yilmaz<sup>1,2</sup> · Jintan Zhang<sup>2</sup> · Peter Kazanzides<sup>2</sup> · Ugur Tumerdem<sup>1</sup>

Received: 3 March 2022 / Accepted: 8 March 2022  
© CARS 2022

## Abstract

**Purpose** Using the da Vinci Research Kit (dVRK), we propose and experimentally demonstrate transfer learning (Xfer) of dynamics between different configurations and robots distributed around the world. This can extend recent research using neural networks to estimate the dynamics of the patient side manipulator (PSM) to provide accurate external end-effector force estimation, by adapting it to different robots and instruments, and in different configurations, with additional forces applied on the instruments as they pass through the trocar.

**Methods** The goal of the learned models is to predict internal joint torques during robot motion. First, exhaustive training is performed during free-space (FS) motion, using several configurations to include gravity effects. Second, to adapt to different setups, a limited amount of training data is collected and then the neural network is updated through Xfer.

**Results** Xfer can adapt a FS network trained on one robot, in one configuration, with a particular instrument, to provide comparable joint torque estimation for a different robot, in a different configuration, using a different instrument, and inserted through a trocar. The robustness of this approach is demonstrated with multiple PSMs (sampled from the dVRK community), instruments, configurations and trocar ports.

**Conclusion** Xfer provides significant improvements in prediction errors without the need for complete training from scratch and is robust over a wide range of robots, kinematic configurations, surgical instruments, and patient-specific setups.

**Keywords** Surgical robotics · Dynamic identification · Tactile sensing · Transfer learning

## Introduction

Identification is a technique for obtaining models from measurements [1]. In robotics, dynamic identification is used for model-based control approaches like computed torque schemes [2] or external force estimation [3].

Robot-assisted minimally invasive surgery (RAMIS) systems enable surgeons to teleoperate patient side manipulators (PSMs), which are purpose-built robotic manipulators that can perform spherical motion about incision ports (remote centers of motion) and actuate articulated instruments that mimic the human wrist. Because the instruments pass through ports and trocars with diameters in the range of 5–15 mm, they have to be remotely actuated through cable transmissions. Mechanically, these robots significantly differ from industrial manipulators due to the use of flexible transmissions, counterbalancing weights and springs. Furthermore, they are always in contact with the patient body/trocar during operations. Therefore, more advanced identification techniques are required.

Dynamic identification is an active research subject in robotic surgery, especially with the widely adopted da Vinci Research Kit (dVRK) [4,5] and Raven II [6] open research platforms. In [7], identification of friction on a da Vinci instrument and in [8–10] parametric identification of the dVRK patient side manipulators were achieved with

---

✉ Nural Yilmaz  
nural.yilmaz@marun.edu.tr

Jintan Zhang  
jzhan247@jhu.edu

Peter Kazanzides  
pkaz@jhu.edu

Ugur Tumerdem  
ugur.tumerdem@marmara.edu.tr

<sup>1</sup> Department of Mechanical Engineering, Marmara University, Istanbul 34722, Turkey

<sup>2</sup> Department of Computer Science, Johns Hopkins University, Baltimore, MD 21218, USA

least square methods. In [11], a convex optimization-based dynamic identification package was proposed for the dVRK. In [12–14] identification of cable transmission dynamics on the Raven platform was also performed. However, all of these works employ model-based parametric approaches that are robot specific and do not take into account different surgical setups and patient interactions.

When the robot structures are complicated with several nonlinearities it is often more practical to obtain lumped models through nonparametric identification [1]. The methods for nonparametric identification include frequency response techniques [15], gray box [16,17] and black box models [17] including neural networks [3,18,19]. With the recent advances in machine learning and deep learning techniques, learning-based identification of robot manipulators is becoming more common [20–23]. In machine learning, a recently emerging trend is transfer learning, which aims to exploit skills learned in one application in other applications with similar characteristics for speed and efficiency [24]. In [21], transfer learning was applied to the identification of robot joint dynamics with harmonic drives.

We previously proposed a neural network-based dynamic identification method on dVRK for force estimation [25], and subsequently a correction network for trocar interactions on an abdominal phantom [26]. In this paper, we show for the first time that dynamic identification of surgical robots in clinical settings can be accelerated by transfer learning. More specifically, we show that models pre-trained in nominal conditions can be transferred and quickly fine-tuned for robots operating on phantoms with different instruments and kinematic configurations. Also, with the support of the dVRK community, we demonstrate dynamic identification using transferred models from other dVRK PSMs distributed around the world. While generally similar, dVRK systems can have variations in their mechanical structures, electronics, actuators, control systems as well as the instruments and this paper presents the first dynamic analysis comparison between different dVRK setups.

While least squares or convex optimization methods [8,11] use optimal or Fourier series-based excitation trajectories, our previous work involved identification with trajectories obtained from extensive training with human inputs through teleoperation. To scale this approach in this work, automated trajectories are also proposed and deployed on different dVRK platforms. These trajectories are shown to yield comparable identification results with teleoperation trajectories, thus eliminating the need for operator supervision in identification. Furthermore, it is also experimentally shown that the proposed method can achieve more accurate and faster intra-operative adaptation in a pseudo-clinical setting compared to a model-based approach.

## Methods

### Transfer learning-based dynamic identification

The proposed method aims to identify the inverse dynamics model of a dVRK Patient Side Manipulator (PSM) in joint space by using neural networks which can later be transferred to other surgical setups and updated. The dynamic model of the PSM can be described by the following generalized equation:

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) + F(\dot{q}) + \tau_{\text{int}} + \tau_{\text{ext}} = \tau \quad (1)$$

where  $q$ ,  $\dot{q}$  and  $\ddot{q}$  denote the joint position, velocity and acceleration vectors.  $M$ ,  $C$  and  $G$  represent the mass/inertia matrix, Coriolis, centrifugal force/torque and gravity vectors, respectively, while  $F$  represents the friction force/torque vector,  $\tau_{\text{int}}$  represents the unmodeled internal robot dynamic forces/torques and  $\tau_{\text{ext}}$  represents any unmodeled external forces/torques. In [8,10,11], the inertial terms  $M$ ,  $C$  and  $G$  as well as friction were parameterized, and cable tension was simply modeled as a linear spring. In [9], the cable model obtained for a Raven robot [13] was applied to a dVRK. However, it may be difficult to capture the dynamics of different cable-pulley transmissions on the dVRK with a single parametric model; therefore a complete analytical description for  $\tau_{\text{int}}$  is difficult to obtain. Furthermore, during operation, the PSMs are always in contact with the patient at the trocar and depending on the configuration of the robot and interaction geometry with the patient, this effect may be impossible to capture analytically. In our previous work [25,26], we proposed the use of neural networks to nonparametrically identify the joint torques,  $\hat{\tau}$ , from joint position and velocity measurements. The trocar interactions or cable flexibilities can be internalized as a part of the internal dynamics of the robot,  $\tau_{\text{int}}$ , as assumed in our prior work. The external estimates to be used in operator feedback can then be obtained by subtracting from joint torque measurements the total estimated dynamic torques:  $\hat{\tau}_{\text{ext}} = \tau - \hat{\tau}$ . Also, in a surgical operation different instruments can be used whose dynamics can be significantly different. As dynamic identification is a computationally expensive and time-consuming task, in order to deal with the changes intraoperatively, we introduce transfer learning in this paper. Transfer learning refers to the transfer of trained networks between different robots/robot configurations and adaptation to any variations or changes in the setup with quick and efficient local updates.

Accurate models of PSMs can be identified in large workspaces in free-space motion with large networks given enough training and computation power. Networks trained before the surgical operation, in free space and without patient interactions, to capture robot dynamics, are referred to as FS in this paper. Inputs to the FS networks are PSM joint

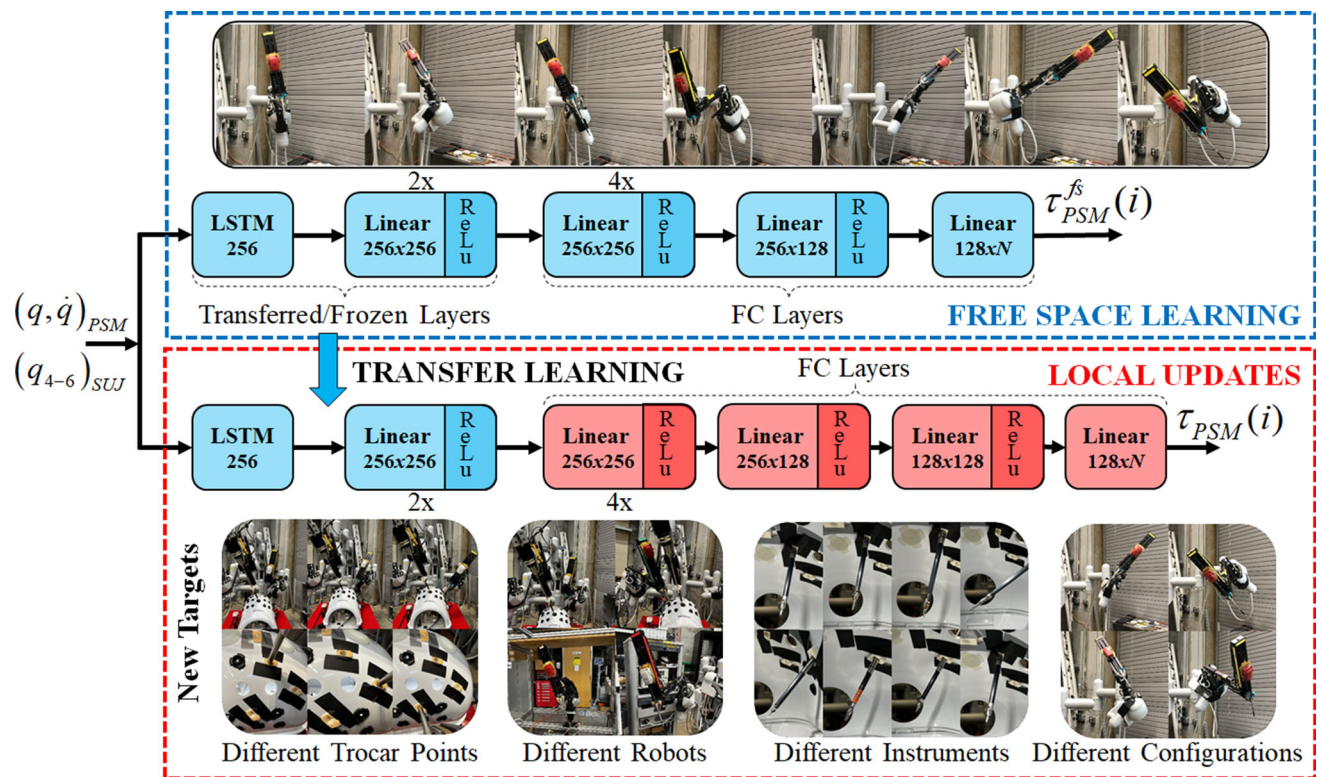


Fig. 1 Transfer learning-based dynamic identification (Xfer) on the dVRK

positions and velocities, and the last three joint positions of the setup joints (SUJ), which determine orientation of the PSM base and thus the effect of gravity. The outputs of the networks are joint torque predictions. During training, robot joints are grouped based on their joint position, velocity and torque ranges ( $G_1:J_1; J_2, G_2:J_3, G_3:J_4, G_4:J_5; J_6$ ), and separate but identical networks, implemented using PyTorch, are trained for each group. In each network of FS, as shown in Fig. 1, we used an LSTM layer to capture critical details from time-series data. The LSTM output with 256 hidden dimensions is forwarded to fully connected (FC) layers. There are 8 FC layers in total with ReLU activation functions and the output layer is also a linear layer for regression.

Once such a model is obtained, it is later transferred to a robot which is deployed for operation (with possible variations as shown in Fig. 1) and locally updated. In transfer learning, some layers of the transferred networks are frozen by setting their learning rates to zero and the rest are trained to update the network. Also, extra layers may be added to the transferred networks and their weights can be determined with local training. In our implementation, we maintain the LSTM layer as well as the first two feedforward layers from FS, and append 6 feedforward layers to the end of the transferred networks, which can be updated with new training data. This forms our transfer learning method (Xfer), as shown in Fig. 1.

## Experiment design

To evaluate the performance of transfer learning and show its potential as a plug and play identification tool across the dVRK community, several experiments were designed and conducted.

First, we examined the identification of different instruments using a common free space network and instrument-specific transfer learning network inside the phantom to investigate performance deterioration during intraoperative instrument changes. Second, we examined identification performance with different port placements and kinematic configurations to evaluate the combined effect of patient interactions and gravity. Third, to evaluate the feasibility of robot to robot transfer learning, we compared the identification results using free space networks trained on PSMs from the dVRK community on our systems, and then applied transfer learning to investigate the performance improvements. Additionally, to see the combined effect of all the variations, we implemented a network from another PSM in the dVRK community on our setup, with a different instrument, in the abdominal phantom. Fourth, we performed Pareto optimal hyper-parameter tuning with different batch, window, and sample sizes to reduce intra-operative data collection to 15 s and training times to about 30 s. Fifth, we compared our

results to a model-based parametric Least Squares identification method used in the literature [8,11].

We generated seven excitation paths in joint space, around 15 min in total, to move the robot in free space. For the experiments excluding robot to robot transfer, we replayed each excitation path in 7 different configurations to collect training data (PSM joint positions and velocities, and setup joint positions) to enable our FS network to learn the gravitational force changes in the robot dynamics. The free space excitation data are split into training (13x7 min), evaluation (1x7 min), and test (1x7 min). In addition, a 1 min teleoperation data is generated for testing. All free space training data collection is done using a Large Needle Driver (LND) and the training process takes approximately 5 h. For the first two transfer learning experiments, three manual teleoperation paths in joint space are recorded and replayed inside the phantom. The data are split into training (3 min), evaluation (1 min), and test (3 min). The transfer network training takes 1 min with this dataset. For the robot to robot transfer learning experiment, we only used one FS configuration from the other dVRK sites because most were unable to provide setup joint positions. To generate excitation paths that could be replayed at different locations with different PSMs, we used a random number generator with a nonrepeatable uniform

output and smoothed the output signal with two low pass filters. Joint limits were considered to provide a motion within the robot workspace.

To evaluate the performance of the proposed method, the normalized root-mean-square errors (NRMSE) between the actual and estimated joint forces/torques were calculated using the formula given in [10]. Each experiment was repeated ten times and error means and standard deviations are presented in the tables. The results are discussed in the next section.

## Results

### Different instruments

To demonstrate the effectiveness of the transfer learning network to adapt to potential intraoperative instrument changes, we experimented with the following 8 instruments inside the phantom through a common port: Large Needle Driver (LND), Small Clip Appliers (SCA), Resano Forceps (RF), Monopolar Curved Scissors (MCS), Maryland Bipolar Forceps (MBP), Prograsp Forceps (PF), Cadiere Forceps (CF), and Long Tip Forceps (LTF), which are shown to the left of Table 1.

**Table 1** NRMSE mean (standard deviation) of joint force/torques with different instruments

|       | Joint 1      |              | Joint 2       |               | Joint 3       |               |
|-------|--------------|--------------|---------------|---------------|---------------|---------------|
|       | FS (%)       | Xfer (%)     | FS (%)        | Xfer (%)      | FS (%)        | Xfer (%)      |
| (LND) | 9.120(0.934) | 6.155(0.537) | 14.273(2.475) | 10.440(1.965) | 20.322(2.804) | 16.493(1.997) |
| (SCA) | 9.149(0.976) | 6.446(0.859) | 14.438(2.483) | 10.153(1.715) | 20.685(2.373) | 17.250(3.139) |
| (RF)  | 9.185(0.946) | 6.108(0.488) | 14.226(2.451) | 10.282(2.021) | 20.647(2.969) | 17.623(2.183) |
| (MCS) | 8.788(0.890) | 6.028(0.452) | 14.444(2.603) | 10.403(2.156) | 21.178(2.407) | 19.619(2.106) |
| (MBF) | 8.967(0.914) | 6.252(0.423) | 14.382(2.500) | 10.537(2.262) | 20.770(2.689) | 17.295(1.984) |
| (PF)  | 9.115(0.929) | 6.081(0.484) | 14.208(2.331) | 10.226(1.964) | 20.423(2.937) | 16.993(1.795) |
| (CF)  | 9.153(0.928) | 6.164(0.430) | 14.551(2.515) | 10.524(2.232) | 20.871(3.286) | 17.370(1.954) |
| (LTF) | 9.057(0.888) | 6.060(0.494) | 14.518(2.451) | 9.949(1.881)  | 19.275(2.628) | 16.015(2.281) |

|       | Joint 4       |              | Joint 5       |               | Joint 6       |               |
|-------|---------------|--------------|---------------|---------------|---------------|---------------|
|       | FS (%)        | Xfer (%)     | FS (%)        | Xfer (%)      | FS (%)        | Xfer (%)      |
| (LND) | 6.785(1.196)  | 6.097(1.100) | 15.232(0.900) | 4.261(0.429)  | 21.460(2.250) | 4.875(1.470)  |
| (SCA) | 11.014(1.802) | 7.546(2.144) | 13.943(2.294) | 8.454(3.371)  | 12.115(1.496) | 8.707(3.404)  |
| (RF)  | 7.288(0.891)  | 6.103(0.935) | 14.902(1.028) | 5.812(1.185)  | 14.704(0.987) | 6.141(1.482)  |
| (MCS) | 8.509(1.319)  | 7.231(1.329) | 8.525(1.203)  | 5.882(0.484)  | 9.105(1.132)  | 8.788(1.461)  |
| (MBF) | 10.147(1.474) | 7.052(1.555) | 19.765(1.430) | 11.795(1.124) | 25.630(1.758) | 11.080(1.797) |
| (PF)  | 7.635(1.079)  | 6.675(1.155) | 8.707(0.987)  | 12.336(1.627) | 19.572(1.322) | 7.826(1.342)  |
| (CF)  | 7.183(1.139)  | 6.361(1.333) | 16.417(1.072) | 11.966(1.101) | 20.302(0.923) | 12.511(1.670) |
| (LTF) | 9.961(1.440)  | 7.504(1.240) | 11.192(1.308) | 6.362(2.300)  | 17.562(0.912) | 13.139(2.271) |

**Table 2** NRMSE mean (standard deviation) of joint force/torques with different PSMs (R1, R2), port placements (P1–P4) and kinematic configurations

|       | Joint 1       |              | Joint 2       |              | Joint 3       |               |
|-------|---------------|--------------|---------------|--------------|---------------|---------------|
|       | FS (%)        | Xfer (%)     | FS (%)        | Xfer (%)     | FS (%)        | Xfer (%)      |
| R1:P1 | 10.285(0.860) | 7.610(1.096) | 11.378(0.927) | 8.618(0.971) | 12.915(1.436) | 9.505(0.713)  |
| R1:P2 | 12.560(1.116) | 7.786(0.960) | 14.802(1.000) | 9.098(1.316) | 14.326(1.376) | 9.742(0.513)  |
| R1:P3 | 9.235(0.964)  | 8.188(0.948) | 13.806(1.704) | 9.807(1.634) | 13.093(1.293) | 9.924(0.473)  |
| R2:P4 | 10.048(0.751) | 8.371(0.589) | 12.236(1.401) | 9.338(0.785) | 12.941(1.350) | 11.064(0.577) |

|       | Joint 4      |              | Joint 5       |              | Joint 6       |              |
|-------|--------------|--------------|---------------|--------------|---------------|--------------|
|       | FS (%)       | Xfer (%)     | FS (%)        | Xfer (%)     | FS (%)        | Xfer (%)     |
| R1:P1 | 5.932(0.740) | 4.444(0.442) | 19.231(0.972) | 4.391(0.321) | 27.422(2.204) | 4.616(0.527) |
| R1:P2 | 7.151(1.104) | 5.004(0.532) | 19.997(2.367) | 4.279(0.301) | 31.501(2.393) | 4.442(0.476) |
| R1:P3 | 6.466(0.703) | 5.070(0.440) | 18.734(0.718) | 4.302(0.290) | 27.630(2.321) | 4.511(0.498) |
| R2:P4 | 6.555(0.766) | 5.117(0.624) | 19.311(0.935) | 4.676(0.718) | 28.663(2.573) | 4.698(0.427) |

**Table 3** NRMSE mean (standard deviation) of joint force/torques for FS and robot-to-robot Xfer at configurations (1) and (2) shown on left

| Network Name      | Free Space    |               | Xfer         |              |
|-------------------|---------------|---------------|--------------|--------------|
|                   | excita.(%)    | teleop.(%)    | excita.(%)   | teleop.(%)   |
| JHU PSM3 config1  | 12.998(0.358) | 13.070(0.408) | 6.889(0.190) | 7.600(0.560) |
| JHU PSM3 config2  | 13.502(0.404) | 13.562(0.382) | 6.970(0.150) | 7.723(0.477) |
| WFU PSM1 config1  | 17.442(1.834) | 19.434(0.764) | 6.784(0.482) | 7.813(0.530) |
| WFU PSM2 config1  | 17.061(1.610) | 19.410(0.600) | 7.398(0.358) | 8.465(0.693) |
| WFU PSM1 config2  | 17.468(1.848) | 20.043(0.785) | 6.855(0.450) | 8.024(0.493) |
| WFU PSM2 config2  | 17.154(1.609) | 19.902(0.532) | 7.315(0.254) | 8.408(0.647) |
| OU PSM1 config1   | 25.449(2.392) | 27.015(0.776) | 7.777(0.383) | 9.041(0.646) |
| OU PSM2 config1   | 17.716(1.362) | 18.030(0.848) | 7.663(0.366) | 8.636(0.693) |
| OU PSM1 config2   | 25.743(2.371) | 28.097(0.909) | 7.771(0.252) | 9.112(0.723) |
| OU PSM2 config2   | 18.099(1.320) | 18.932(0.909) | 7.720(0.305) | 8.661(0.666) |
| UCL PSM1 config1  | 15.570(0.862) | 14.746(0.472) | 8.075(0.364) | 8.391(0.740) |
| UCL PSM3 config1  | 34.341(3.624) | 31.895(1.211) | 7.620(0.315) | 8.134(0.632) |
| UCL PSM1 config2  | 15.298(0.813) | 14.492(0.349) | 8.291(0.374) | 8.548(0.704) |
| UCL PSM3 config2  | 34.129(3.349) | 32.140(1.067) | 7.666(0.233) | 8.081(0.642) |
| WPI PSM1 config1  | 16.716(0.604) | 17.506(0.459) | 9.847(0.254) | 9.756(0.552) |
| WPI PSM3 config1  | 14.660(0.429) | 15.735(0.580) | 7.643(0.464) | 8.120(0.689) |
| WPI PSM1 config2  | 16.964(0.679) | 17.998(0.543) | 9.853(0.258) | 9.833(0.592) |
| WPI PSM3 config2  | 14.749(0.448) | 16.379(0.583) | 7.824(0.706) | 8.261(0.725) |
| SSSA PSM1 config1 | 23.794(0.873) | 22.325(0.610) | 7.113(0.425) | 7.792(0.529) |
| SSSA PSM1 config2 | 24.105(0.781) | 23.280(0.650) | 7.208(0.368) | 7.829(0.548) |
| JHU PSM1 config1  | 13.468(1.318) | 15.758(0.263) | 4.598(0.095) | 8.449(0.415) |
| JHU PSM1 config2  | 13.101(1.475) | 15.537(0.216) | 4.570(0.117) | 8.473(0.566) |

Table 1 shows the NRMSEs of identification using FS, trained with LND, and tested on all instruments. For different instruments, it would be natural to see errors in the last 2 joints, however, interestingly, when the LND instrument is dismounted and remounted on the PSM for testing, variations can occur in the dynamics of the same instrument as well. Table 1 shows that FS identification results are not reliable for any of the instruments after remounting, including the LND, and with Xfer, significant improvement in all joints can be seen and most importantly these improvements are more pronounced in the last two joints.

### Different port placements and configurations

To demonstrate the effectiveness of the transfer learning network to adapt to different port placements, we selected four different trocar insertion points, as shown to the left of Table 2, using the LND instrument. For PSM1 and PSM2 with different port placements, we used the same free space network but different transfer learning networks for each port.

As shown in Table 2, for all the ports, the NRMSEs of torque identification using Xfer are lower than that of FS, which implies that Xfer is capable of adapting to changes in gravitational force vector, friction, as well as kinematic configuration, since some of the test ports require changes to the PSM body frame from the training configuration.

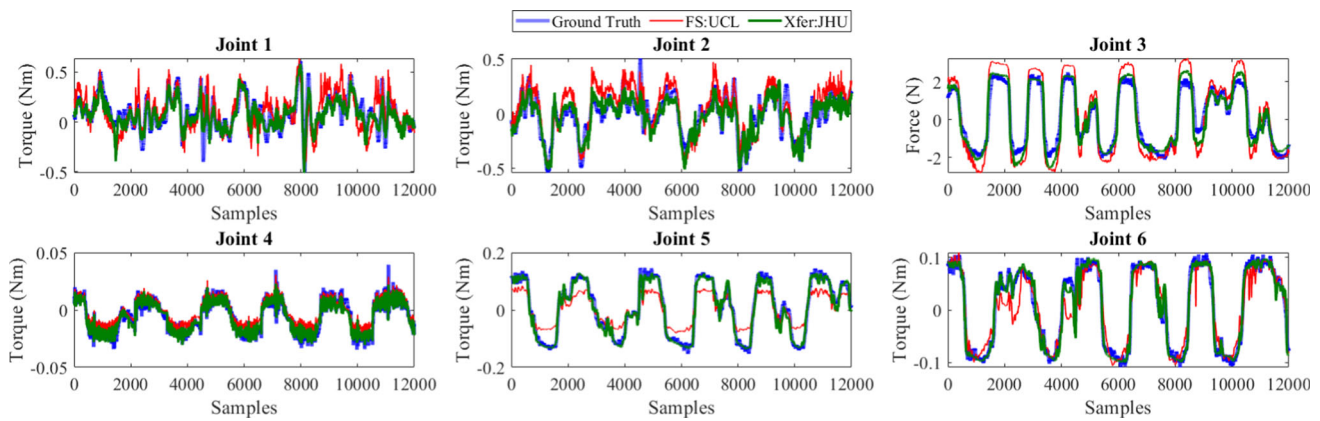
### Robot to robot transfer learning

To test robot to robot transfer learning, we collected data from the dVRK community. For community members that do not

have the setup joints (or do not have the setup joint controller), the PSM body frame is in its normal vertical position with respect to the base frame, i.e., there is no rotation between the PSM body frame and the world coordinate frame, and thus the setup joint positions are set to zero. The network name in Table 3 follows the convention: University Name (A), Robot Name (B), and Configuration ID (C). This means that the network is trained on data collected from PSM B at school A, and tested on configuration C using our dVRK setup. The universities providing training data are: Worcester Polytechnic Institute (WPI), Wake Forest University (WFU), Óbuda University (ÓU), University College London (UCL), and Sant'Anna School of Advanced Studies (SSSA). The average of the NRMSEs for all joints are provided in Table 3.

These results show that the NRMSE of free space torque identification using FS networks trained on data from the community are much higher than the FS network directly trained on our dVRK setup (PSM1) at JHU, which implies that there are significant variations in the robot dynamics. Such differences are corrected using the transfer learning (Xfer) network, as it shows significant and consistent improvement in torque identification when tested on both excitation and recorded teleoperation paths.

To demonstrate the effectiveness of transfer learning to adapt to identification in different kinematic configurations for which the FS networks were not trained, we picked two configurations of the setup joints to train our transfer learning network, as shown to the left of Table 3. Configuration 1 is in the standard position  $[0\ 0\ 0]$  rad in the last 3 axes. Configuration 2 has a joint position of  $[-0.141, -0.435, -0.538]$  rad in the last 3 axes. Table 3 shows that the transfer learning network was able to correct the gravitational or inertial varia-



**Fig. 2** Robot to robot transfer learning identification results: FS: UCL and Xfer: JHU on a JHU PSM utilizing Resano Forceps inside an abdominal phantom with a different configuration

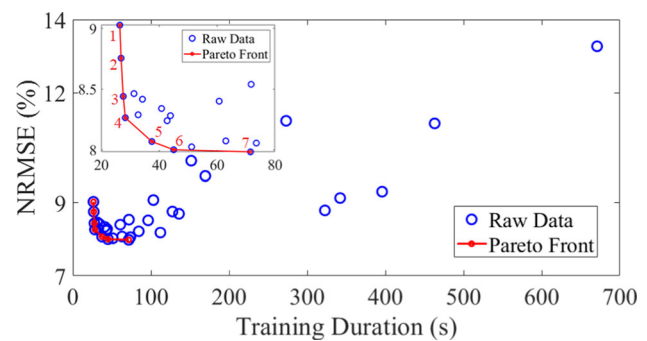
tions in the dynamics resulting from training and testing with different robots and kinematic configurations. The NRMSE of free space torque identification using a network trained on configuration 1 data from different robots within the dVRK community shows consistent improvement after the transfer learning network update, when tested at the target JHU robot in configuration 2.

In addition, though our free space network is trained for seven configurations (six more configurations than other dVRK users in the community), results from Table 3 show that transfer learning networks with local updates on other robots' free space networks are capable of achieving roughly the same dynamic identification performance, with comparable NRMSE. This suggests that it may not be necessary to train the FS network using multiple SUJ configurations, but rather that a single configuration may be sufficient, with transfer learning (Xfer) to adapt to a new configuration.

Finally, to evaluate the combined effect of all variations, a network trained on a UCL PSM was implemented on a JHU PSM with a different kinematic configuration, inside an abdominal phantom and with a different instrument. Figure 2 shows the time series results from the experiment and the first two rows of Table 5 provide the NRMSEs for each joint. There are significant errors in joints 3–6 for the FS network, which are compensated by Xfer.

### Hyper-parameter optimization

We investigated the possibility of improving intra-operative data collection and training times for transfer learning through hyper-parameter tuning. The hyper-parameters used in the tuning process are batch, LSTM window and training sample sizes. To see the effect of training sample size, the training dataset was divided into segments ranging from 4 to 450 k samples where 1 k samples roughly corresponds to 1 s of data collection. Then, we set window size to 1500 and



**Fig. 3** Pareto front analysis

**Table 4** Pareto front parameters

| P | NRMSE (%) | Train. time (s) | Batch size | Win. size |
|---|-----------|-----------------|------------|-----------|
| 1 | 9.0270    | 26.4030         | 16         | 500       |
| 2 | 8.7546    | 26.8510         | 32         | 300       |
| 3 | 8.4400    | 27.6010         | 64         | 200       |
| 4 | 8.2639    | 28.3090         | 32         | 400       |
| 5 | 8.0680    | 37.4790         | 16         | 400       |
| 6 | 8.0002    | 45.0140         | 8          | 500       |
| 7 | 7.9829    | 71.5840         | 4          | 400       |

batch size to 1 and trained the networks for each training set separately. In the training results, the NRMSE is over 20% with 4 k samples while the networks generated with 5–450 k samples provide similar results with NRMSE less than 10%. Therefore, we decided to focus on sample sizes of 15 k. In the next step, we changed window sizes between 100 and 500 and batch sizes between 1 and the highest possible batch size (restricted by sampling size/window size). Figure 3 shows all results (blue circles) as a function of NRMSE and training time and identifies the Pareto front (red line) for optimal hyper-parameter selection, with red circles (numbered from 1 to 7) corresponding to Pareto optimal solutions (see Table 4).

**Table 5** Mean (standard deviation) NRMSE of joint force/torques in trocar

|              | TD (s) | J1 (%)        | J2 (%)        | J3 (%)        | J4 (%)        | J5 (%)        | J6 (%)        |
|--------------|--------|---------------|---------------|---------------|---------------|---------------|---------------|
| NN (FS)      | 2100   | 14.195 (1.94) | 13.784 (1.92) | 15.944 (0.76) | 8.331 (1.03)  | 15.666 (0.93) | 10.992 (1.61) |
| Xfer (3 min) | 60     | 8.294 (0.64)  | 9.533 (0.94)  | 9.183 (1.06)  | 5.061 (0.75)  | 5.706 (1.02)  | 5.527 (0.91)  |
| Xfer (15 s)  | 28     | 10.609 (0.89) | 11.302 (0.58) | 10.419 (1.28) | 5.791 (0.80)  | 10.009 (1.41) | 11.906 (0.78) |
| LS (FS)      | 514    | 12.902 (1.38) | 12.933 (1.63) | 13.621 (1.04) | 13.571 (1.91) | 21.770 (2.49) | 22.147 (2.45) |
| LS (3 min)   | 228    | 8.558 (0.81)  | 9.674 (0.98)  | 12.938 (1.12) | 7.690 (1.34)  | 18.436 (2.17) | 20.524 (2.46) |
| LS (15 s)    | 18     | 16.070 (4.29) | 15.287 (3.79) | 26.528 (3.93) | 8.679 (1.25)  | 22.355 (3.16) | 23.319 (3.31) |

Least square (LS) versus proposed neural network (NN) in FS and with transfer learning (Xfer). *TD* training duration

As we place equal emphasis on error and training duration, a batch size of 16–32 and window size of 400 provides a Pareto optimal solution with training times of 28–38 s.

### Transfer learning versus least squares identification in a clinical scenario

To demonstrate the advantages of the proposed method over conventional model-based parametric identification approaches in a clinical setting, we used data from the final experiment in Section “Robot to robot transfer learning” with different robots, instruments and an abdominal phantom. We implemented a model-based parametric identification using an LMI reformulation of the popular Least Squares (LS) method proposed in [8] and implemented in [11] on dVRK. We also implemented Xfer with the Pareto optimal hyper-parameter set P4 obtained in Table 4, with only 15 s of intraoperative data. Table 5 shows the identification errors comparing Xfer with LS. The first row of the table shows the errors for the proposed network (NN) trained with comprehensive free space data obtained from UCL, with the second and third rows showing the errors for Xfer trained with 3 min and 15 s of intraoperative data, respectively. The fourth row shows the errors for LS with identification performed using the same free space data. The fifth and sixth rows show the errors for LS where identification was performed with 3 min and 15 s of intraoperative data, respectively. The results show that the proposed networks have comparable performance with LS when both utilize FS datasets from UCL. LS has better performance in the first three joints, whereas the neural networks have better performance in the wrist, where modeling is prone to errors due to nonlinear behavior. Moreover, with both 3 min and 15 s of intraoperative data, Xfer outperforms LS decisively. Furthermore, for 15 s and 3 min of intra-operative data, Xfer training was completed in 28 s and 60 s, respectively, whereas the LS identification took 18 s and 228 s. LS also had much higher standard deviations in identification errors, indicating low robustness. This shows that it is more feasible to use the proposed transfer learning approach over model-based approaches in clinical settings.

## Conclusion

In this work, we proposed a self-supervised transfer learning approach to transfer learned dynamics between different surgical robots within the dVRK community, or different configurations of the same robot (with different instruments, kinematic configurations or port placements). Our studies show that transfer learning provides significant improvements in prediction errors without the need for complete training from scratch and is robust over a wide range of robots, kinematic configurations, surgical instruments, and patient-specific setups. Furthermore, our work shows the potential of achieving such accuracy with only 10–15 s of data collection and online training in the surgical theater, given more computational power and parallel processing. We hope to convert this work into a software package that the dVRK community can use for quick dynamic identification. Immediate applications of this method are in the development of force estimation and haptic teleoperation algorithms that can run on the dVRK.

**Acknowledgements** This work was supported in part by NSF OISE 1927354. Haoying (Jack) Zhou from Worcester Polytechnic Institute (PI: G. Fischer), Scotty Chung from Wake Forest University (PI: P. Brown), Tamás D. Nagy from Óbuda University (PI: T. Haidegger), Solene Dietsch from University College London (PI: D. Stoyanov), and Andrea Mariani from Sant’Anna School of Advanced Studies (PI: A. Menciassi) provided training data for this study.

## Declarations

**Conflict of interest** The authors declare that they have no conflict of interest.

**Human and animal participants** This article does not contain any studies with human participants or animals performed by any of the authors.

## References

- Hollerbach J, Khalil W, Gautier M (2008) Model identification. Springer Handbook of Robotics, Springer, Berlin Heidelberg, pp. 321–344

2. Khosla PK, Kanade T (1985) Parameter identification of robot dynamics. In: Conference on decision and control (CDC). IEEE, pp 1754–1760
3. Smith AC, Hashtrudi-Zaad K (2005) Application of neural networks in inverse dynamics based contact force estimation. In: Conference on control applications (CCA). IEEE, pp 1021–1026
4. Kazanzides P, Chen Z, Deguet A, Fischer GS, Taylor RH, DiMaio SP (2014) An open-source research kit for the da Vinci® surgical system. In: IEEE International conference on robotics and automation (ICRA). Hong Kong, China, pp 6434–6439
5. D’Ettorre C, Mariani A, Stilli A, Rodriguez y Baena F, Valdastrì P, Deguet A, Kazanzides P, Taylor RH, Fischer GS, DiMaio SP, Menciassi A, Stoyanov D (2021) Accelerating surgical robotics research: a review of 10 years with the da Vinci Research Kit. *IEEE Robot Autom Mag* 6:66
6. Hannaford B, Rosen J, Friedman DW, King H, Roan P, Cheng L, Glozman D, Ma J, Kosari SN, White L (2012) Raven-II: an open platform for surgical robotics research. *IEEE Trans Biomed Eng* 60(4):954–959
7. Mahvash M, Gwilliam J, Agarwal R, Vagvolgyi B, Su L.-M, Yuh DD, Okamura AM (2008) Force-feedback surgical teleoperator: controller design and palpation experiments. In: 2008 Symposium on haptic interfaces for virtual environment and teleoperator systems. IEEE, pp 465–471
8. Fontanelli GA, Ficuciello F, Villani L, Siciliano B (2017) Modelling and identification of the da Vinci research kit robotic arms. In: IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE, pp 1464–1469
9. Sang H, Yun J, Monfaredi R, Wilson E, Fooladi H, Cleary K (2017) External force estimation and implementation in robotically assisted minimally invasive surgery. *Int J Med Robot Comp Assist Surg* 13(2):1824
10. Piqué F, Boushaki MN, Brancadoro M, De Momi E, Menciassi A (2019) Dynamic modeling of the a Vinci Research Kit arm for the estimation of interaction wrench. In: International symposium on medical robotics (ISMR). IEEE, pp 1–7
11. Wang Y, Gondokaryono R, Munawar A, Fischer GS (2019) A convex optimization-based dynamic model identification package for the da Vinci Research Kit. *IEEE Robot Autom Lett* 4(4):3657–3664
12. Haghighipanah M, Miyasaka M, Hannaford B (2017) Utilizing elasticity of cable-driven surgical robot to estimate cable tension and external force. *IEEE Robot Autom Lett* 2(3):1593–1600
13. Miyasaka M, Haghighipanah M, Li Y, Hannaford B (2016) Hysteresis model of longitudinally loaded cable for cable driven robots and identification of the parameters. In: International conference on robotics and automation (ICRA). IEEE, pp 4051–4057
14. Li Y, Miyasaka M, Haghighipanah M, Cheng L, Hannaford B (2016) Dynamic modeling of cable driven elongated surgical instruments for sensorless grip force estimation. In: International conference on robotics and automation (ICRA). IEEE, pp 4128–4134
15. Guillaume P, Pintelon R, Schoukens J (1996) Accurate estimation of multivariable frequency response functions. *IFAC Proc* 29(1):4351–4356
16. Östring M, Gunnarsson S, Norrlöf M (2003) Closed-loop identification of an industrial robot containing flexibilities. *Control Eng Pract* 11(3):291–300
17. Wernholt E (2004) On multivariable and nonlinear identification of industrial robots. PhD thesis, Linköping University, Linköping, Sweden
18. Karakasoglu A, Sudharsanan SI, Sundareshan MK (1993) Identification and decentralized adaptive control using dynamical neural networks with application to robotic manipulators. *IEEE Trans Neural Netw* 4(6):919–930
19. Pham D, Oh S (1994) Adaptive control of a robot using neural networks. *Robotica* 12(6):553–561
20. Su H, Qi W, Hu Y, Sandoval J, Zhang L, Schmirander Y, Chen G, Aliverti A, Knoll A, Ferrigno G, De Momi E (2019) Towards model-free tool dynamic identification and calibration using multi-layer neural network. *Sensors* 19(17):3636
21. de Gea Fernández J, Yu B, Bargsten V, Zipper M, Sprengel H (2020) Design, modelling and control of novel series-elastic actuators for industrial robots. In: *Actuators*, vol 9. Multidisciplinary Digital Publishing Institute, p 6
22. Bargsten V, de Gea Fernandez J, Kassahun Y (2016) Experimental robot inverse dynamics identification using classical and machine learning techniques. In: *International symposium on robotics (ISR)*. VDE, pp 1–6
23. Shareef Z, Mohammadi P, Steil J (2016) Improving the inverse dynamics model of the KUKA LWR IV+ using independent joint learning. *IFAC-PapersOnLine* 49(21):507–512
24. Lu J, Behbood V, Hao P, Zuo H, Xue S, Zhang G (2015) Transfer learning using computational intelligence: a survey. *Knowl Based Syst* 80:14–23
25. Yilmaz N, Wu JY, Kazanzides P, Tumerdem U (2020) Neural network based inverse dynamics identification and external force estimation on the da Vinci Research Kit. In: *International conference on robotics and automation (ICRA)*. IEEE, pp 1387–1393
26. Wu JY, Yilmaz N, Tumerdem U, Kazanzides P (2021) Robot force estimation with learned intraoperative correction. In: *2021 International symposium on medical robotics (ISMR)*. IEEE, pp 1–7

**Publisher’s Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.