

Human-Machine Collaborative Surgery Using Learned Models

Nicolas Padoy, Gregory D. Hager
The Johns Hopkins University, Baltimore, MD, USA
{padoy, hager}@jhu.edu

Abstract—In the future of surgery, tele-operated robotic assistants will offer the possibility of performing certain commonly occurring tasks autonomously. Using a natural division of tasks into subtasks, we propose a novel surgical Human-Machine Collaborative (HMC) system in which portions of a surgical task are performed autonomously under complete surgeon’s control, and other portions manually. Our system automatically identifies the completion of a manual subtask, seamlessly executes the next automated task, and then returns control back to the surgeon. Our approach is based on learning from demonstration. It uses Hidden Markov Models for the recognition of task completion and temporal curve averaging for learning the executed motions. We demonstrate our approach using a da Vinci tele-surgical robot. We show on two illustrative tasks where such human-machine collaboration is intuitive that automated control improves the usage of the master manipulator workspace. Because such a system does not limit the traditional use of the robot, but merely enhances its capabilities while leaving full control to the surgeon, it provides a safe and acceptable solution for surgical performance enhancement.

I. INTRODUCTION

The increased use of tele-surgical robots, like the da Vinci® surgical system from Intuitive Surgical™, provides new ways to teach, assess and perform surgeries. On the one hand, large sets of objective performance data can be collected from the robots [1]. On the other hand, the technology will permit automation of specific surgical tasks in the near future [2]. It is however unclear how collected performance data can be used to design efficient human-machine collaborative systems that can adapt to the operator, to the surgical environment, and also provide seamless assistance during the surgery.

In this paper, we contribute to this new research area by proposing a Human-Machine Collaborative (HMC) system that learns, from surgical demonstration, how to *collaboratively* assist the surgeon during a tele-operated task. This is complementary to previous work focusing on pure automation [3], [4], [2]. Here, the task is broken down into manual and potentially automatable subtasks. The system then assists the operator by recognizing, in real-time, the completion of manual subtasks and automates the remaining ones. It also provides contextual Augmented Reality (AR) information in the operator’s view to reduce his/her cognitive workload. The subtasks to be automated were chosen so as to permit an optimized usage of the operator workspace. They involve transport tasks having no interaction with the environment. During tele-operation, transitions between manual and automated execution are automatic and seamless. Moreover, the operator has the possibility of intervening



Fig. 1. Configuration of the da Vinci robot: patient side manipulators plus endoscopic camera (left) and master manipulators (right).

during the automatic execution to correct the trajectory if needed.

We validate our approach using a da Vinci surgical robot [5] used exclusively for research purposes. The robot consists of two master console manipulators that tele-operate three patient-side manipulators under stereo endoscopic visualization (see Fig. 1). We first record kinematic data from the robot while an operator performs the task a few times and annotates segments of the data corresponding to subtasks that should be automated. We then propose an approach based on Hidden Markov Models (HMMs) to determine, in real-time, the completion of the current manual subtask. Once the transition to automatic execution is activated, the instrument motion corresponding to the automatic subtask is provided to the robot. This motion is learned from the demonstration data using a temporal curve averaging approach based on Dynamic Time Warping (DTW). Finally, for supervision purposes, we display an overlay of the planned trajectory in the stereo view of the operator.

We use two manipulative tasks to demonstrate our approach: moving three pins to the cardinal locations of a training pod and performing a running suture on a suturing pod. Our experiments show that such a Human-Machine Collaborative framework is intuitive and can highly improve the usage of the operator’s workspace, by reducing large movements from the master manipulators and thereby the need for clutching to readjust the hand positions.

The remainder of this paper is organized as follows. Related work is presented in Sec. II. The system setup as well as the task model are introduced in Sec. III. In Sec. IV, we present task completion recognition models and, in Sec. V, describe how the executed motions are learned from human demonstration data. Experiments and results are described in Sec. VI. Finally, conclusions and perspectives are given in Sec. VII.

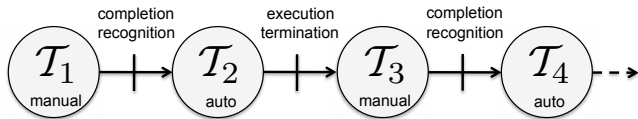


Fig. 2. Task model: transitions between manual and automatic subtasks.

II. RELATED WORK

The recent introduction of robots into the surgery room has created the need for new techniques to train and evaluate surgeons. For this reason, surgical gesture modeling has attracted significant attention in the last years, and several methods, usually using Hidden Markov Models or variations thereof, have been proposed for off-line skill modeling and classification [6], [1], [7].

With the development of dexterous robots, different groups have proposed techniques to automate specific surgical tasks. An example of a task that has been addressed is knot tying. In [3], recurrent neural networks are used to learn a loop trajectory from demonstration. In [2], robot dynamics are learned to replay the trajectory at an increased speed. The needle insertion task has also attracted much attention: a geometric task model is designed in [4] to compute the path-planning of the needle insertion. In [8], a circular motion is automated for needle insertion after the surgeon has marked the insertion point with a laser-pointer. It remains however open research to design automation methods that either deal with the environment such as tissues and suture threads or rely on collaboration with the operator.

A natural way to allow collaboration between a robot and the operator is to change the interaction mode based on the current context. This has been demonstrated on a curve following task in microsurgery by using virtual fixtures to impose path constraints on the manipulator [9].

Context modeling for real-time recognition of the current surgical task has been addressed in [10], [11], [12], [13], [14], typically using automata or Hidden Markov Models. But, these approaches have not yet been tested for modifying the execution of the operation. In Sec. IV, we use a recognition approach inspired from [14], but in a simpler setting as our task model is sequential.

Finally, several assistance extensions have been proposed for the da Vinci robot. They primarily aim at displaying registered pre-operative images within the surgeon's view [15], [16] or at providing 3D user interactions [17]. Also, improvements for human-machine interactions with the da Vinci system, such as camera control while operating the tools at the same time, has been described in [18].

III. SETUP

A. Task Model

For the purposes of this paper, we consider a task to consist of an ordered set of subtasks that need to be executed in a sequential temporal order. To ease the presentation, we assume that the tasks alternate between a manual and an automated subtask, starting with a manual subtask. This can

#	Name	#	Name
1	Grab <i>pin1</i> from East	7	Pin <i>pin2</i>
2*	Move <i>pin1</i> to North	8*	Move tool back to East
3	Pin <i>pin1</i>	9	Grab <i>pin3</i>
4*	Move tool back to East	10*	Move <i>pin3</i> to South
5	Grab <i>pin2</i>	11	Pin <i>pin3</i>
6*	Move <i>pin2</i> to West	12*	Move tool back to East

TABLE I

PIN TASK DESCRIPTION. THE STAR (*) INDICATES THE AUTOMATED SUBTASKS.

be achieved by aggregating the consecutive subtasks of each type -manual or automated- as a single atomic subtask.

Let a task \mathcal{T} consist of a sequence of subtasks $\mathcal{T}_1, \dots, \mathcal{T}_N$, where $N = 2n$. $\mathcal{T}_2, \mathcal{T}_4, \dots, \mathcal{T}_N$ are assumed to be the automated subtasks. In this work, such subtasks are the ones which do not involve interactions with the environment or fine manipulations, but require larger motions instead.

Fig. 2 briefly illustrates the task model, which can be seen as an automata in which transitions are either triggered by the recognition of the manual subtask completion or by the termination of the automatic subtask.

B. Illustrative Tasks

The two tasks used to demonstrate our approach are illustrated in Figures 3 and 4. The first task, called *pin-task*, requires a single instrument and consists in displacing three pins to three cardinal locations (Fig. 3(a)). The task consists of six large transportation motions learned by the system and of six fine motions executed by the operator, consisting in grasping and pinning. A summary of the subtasks is shown in Table I. The motion performed by the instrument is displayed on Fig. 3(b), in which the transportation and fine motions are represented with different colors.

The second task, called *sut-task*, requires two instruments (referred to by *left* and *right*) and consists of performing a running suture with three needle insertions (Fig. 4(a)). It consists of five generic motions repeated at three successive locations. Three fine motions are performed by the surgeon: grasping needle with right instrument, inserting needle and grasping needle with left instrument. Two transportation motions are learned and automated: pulling thread out with left instrument and handing in the needle back to the right instrument at the location of the next suture point. All the subtasks, concatenated into manual/automated subtasks, are listed in Table II. The motions performed by the two instruments are displayed on Fig. 4(b), in which the transportation and fine motions are represented with different colors.

C. Robotic environment

In our setup, the pod used to accomplish the task can be manipulated by two patient side manipulators (PSMs) having each 7 degrees of freedom (DOFs). Each PSM controls a surgical instrument. In our case the PSMs have the same instruments during the task, namely two large needle drivers. The 7th degree of freedom corresponds to

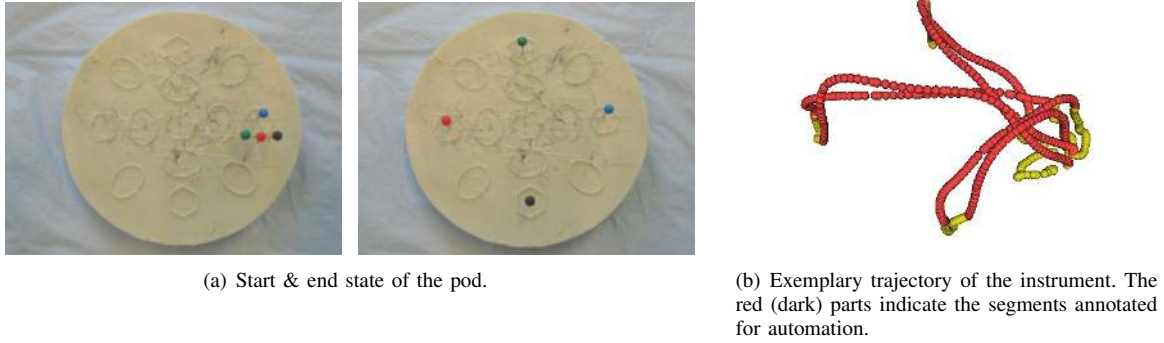


Fig. 3. Pin Task, performed with a single instrument.

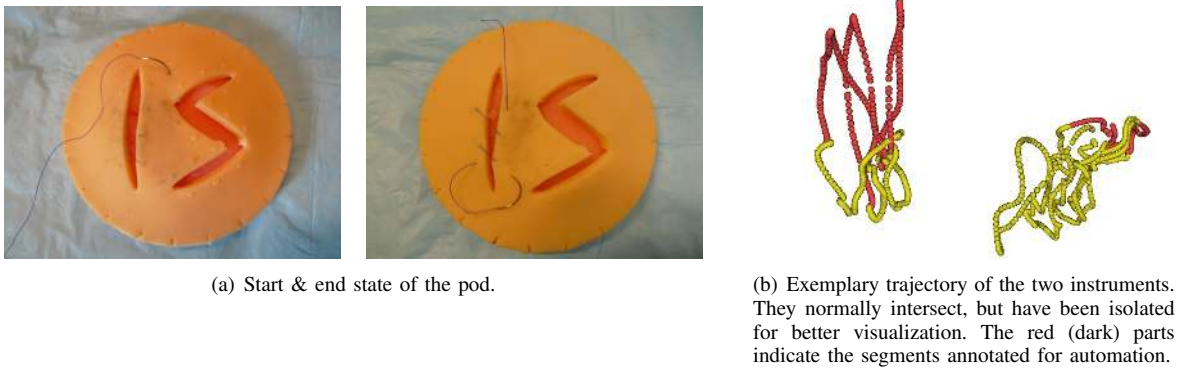


Fig. 4. Suturing Task, performed with two instruments.

#	Name
1	Grasp needle (RT) from pod, move to 1st suture point (RT), Insert needle (RT), grasp it (LT)
2*	Pull thread out (LT), move back to 2nd suture point (LT)
3	Grasp needle (RT) from (LT), Insert needle (RT), grasp it (LT)
4*	Pull thread out (LT), move back to 3rd suture point (LT)
5	Grasp needle (RT) from (LT) Insert needle (RT), grasp it (LT)
6*	Pull thread out (LT), move back to pod end point (LT)

TABLE II

SUTURING TASK DESCRIPTION. (*) INDICATES AUTOMATED SUBTASKS. (RT) STANDS FOR "RIGHT TOOL" AND (LT) FOR "LEFT TOOL".

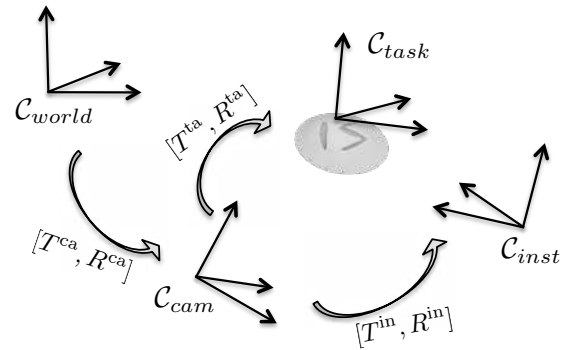


Fig. 5. Coordinate systems illustration.

the opening of the instrument grasper. We also assume for simplification that the left (resp. right) master manipulator controls the left (resp. right) PSM, even though more general tele-operation configurations are possible using the da Vinci robot [5]. During tele-operation, the instruments and the pod are observed by a stereo endoscopic camera, which can be moved using a specific 6 DOF manipulator.

Four main coordinate systems (or frames) are of importance in this setup (see Fig. 5). The task coordinate system C_{task} is specific to the task and independent of the robot initial kinematics. The camera coordinate system C_{cam} indicates the position of the stereo endoscopic camera and

the instrument coordinate systems C_{inst}^j indicates the position of instrument j , with $j \in \{0, 1\}$ denoting the left or right instrument. Finally, the origin of the world, for instance representing the base of the robot, is denoted by C_{world} .

We denote by $[T, R] \in \mathbb{R}^3 \times SO^3$ 3D transformations composed of a translation T and a rotation R . In the following, we assume that we know the transformations $[T_t^{inj}, R_t^{inj}]$ from C_{cam} to C_{inst}^j and $[T_t^{ca}, R_t^{ca}]$ from C_{world} to C_{cam} at each time t . They are collected using the da Vinci research interface [19]. Additionally, we need the transformation $[T_t^{ta}, R_t^{ta}]$ from C_{cam} to C_{task} . We obtain this transformation by placing the training pod at a reference world position before starting the task and by using $[T_t^{ca}, R_t^{ca}]$ to track the camera motion. For such short tasks, we assume

the pod to remain at its initial reference position. For longer tasks, where the pod could get displaced, the corresponding transformation could be provided by visual pod tracking. We also collect the linear/angular velocities of the two patient side manipulators in Cartesian space, the state of the graspers and for workspace analysis the Cartesian positions of the two master manipulators. We collect this data at a frequency of 40 Hz.

Finally, the stereo endoscopic camera has been calibrated. This permits us to overlay 3D augmented reality information such as trajectories within the stereo view of the surgeon.

D. Demonstration data

An operator is asked to demonstrate each task M times, yielding a series of M recordings of the task $\{\mathbf{r}^k\}_{1 \leq k \leq M}$. Each recording corresponds to a multi-dimensional time-series $\mathbf{r}^k = \{\mathbf{r}_1^k, \dots, \mathbf{r}_{|\mathbf{r}^k|}^k\}$ of length $|\mathbf{r}^k|$ taking its values $\mathbf{r}_t^k = (\hat{\mathbf{r}}_t^k, \dot{\mathbf{r}}_t^k)$ in $\mathbb{R}^{26} = \mathbb{R}^{14} \times \mathbb{R}^{12}$ for $1 \leq t \leq |\mathbf{r}^k|$. The projection $\hat{\mathbf{r}}_t^k$ contains the six-dimensional Cartesian velocities as well as the gripper state for the two instruments. This data will be used by the recognition system. The projection $\dot{\mathbf{r}}_t^k$ contains the six-dimensional Cartesian positions of the two instruments that will be used for learning control trajectories. Positions are expressed in the task coordinate system to be independent of the robot initial kinematics.

For each \mathbf{r}^k , the parts of the task that should be automated are labeled by the operator. This decomposes \mathbf{r}^k into N data segments $\{\mathbf{r}^{(k,1)}, \mathbf{r}^{(k,2)}, \dots, \mathbf{r}^{(k,N)}\}$ corresponding to the subtasks $\mathcal{T}_1, \dots, \mathcal{T}_N$. The next sections explain how

- to train a recognition model H^i determining the completion of manual subtask \mathcal{T}_{2i+1} , $1 \leq i \leq n$. To do so, we use an HMM built from the training data $\{\hat{\mathbf{r}}^{(k,2i+1)}, \dot{\mathbf{r}}^{(k,2i+2)}\}_{1 \leq k \leq M}$.
- to compute a trajectory for subtask \mathcal{T}_{2i} , $1 \leq i \leq n$, that will be automatically executed by the instrument. To do so, we use a temporal curve averaging technique based on DTW, applied on the data $\{\dot{\mathbf{r}}^{(k,2i)}\}_{1 \leq k \leq M}$.

IV. RECOGNITION

The objective of the recognition system is to determine the completion of each manual task and to perform a seamless transition to automatic control. The operator should naturally perform a continuous motion as if he would start and perform the next subtask manually, until automation takes over the control of the trajectory. To trigger a seamless transition, we need to determine the instant when the operator has reached a point where the automatic trajectory can be initiated. This is done by using a real-time measure of the subtask completion, computed from a temporal model based on HMMs.

A Hidden Markov Model is defined formally as a quintuplet $\lambda = (S, A, O, B, \pi)$ where S is the number of states $x \in \{1, \dots, S\}$ in the model, A the transition probability matrix between the states, modeling the topology, and O the space of observations, in our case \mathbb{R}^{14} . B is the observation model, indicating for any observation $\mathbf{o} \in O$ and state x the probability $B_x(\mathbf{o}) = P(\mathbf{o} | x)$ that \mathbf{o} can be observed by x . π is a probability distribution over the initial states.

To measure the subtask completion, we use an HMM H^i that is constructed from the concatenation of two HMMs H_0^i and H_1^i modeling respectively the manual subtask \mathcal{T}_{2i+1} and the subtask to be automated \mathcal{T}_{2i+2} . H_0^i and H_1^i are built respectively from the data $\{\hat{\mathbf{r}}^{(k,2i+1)}\}_{1 \leq k \leq M}$ and $\{\dot{\mathbf{r}}^{(k,2i+2)}\}_{1 \leq k \leq M}$. For our experiments, we use left-right HMMs and mixture of Gaussians as observation distributions. We initialize the parameters as in [14]: the number of states is determined from the length of the training sequences and the probabilities are initialized from the data by splitting the data evenly in as many sequences as available states. Then, expectation-maximization is applied to refine the parameters. When the two HMMs are concatenated, the last state of H_0^i is modified to have a transition to the first state of H_1^i . The transition probability is chosen so that the expected time in H_0^i equals the mean duration of the subtask, computed from the training data.

For each state x of HMM H^i , we define the binary indicator function

$$\gamma(x) = \begin{cases} 0 & \text{if } x \in H_0^i \\ 1 & \text{if } x \in H_1^i \end{cases} \quad (1)$$

We define the probability θ_t of having completed the manual subtask at time t by the probability of having reached a state of the HMM that corresponds to the task to be automated:

$$\theta_t = \sum_{x=1}^S \gamma(x) P(X_t = x | \mathbf{o}_{1:t}) \quad (2)$$

Here, $\mathbf{o}_{1:t}$ indicate the observations up to current time t and X_t is a random variable denoting the HMM state at time t . This expression is computed using the forward probabilities of the HMM [20].

Finally, the decision of task completion is given by averaging over a short temporal interval δ , using a decision threshold β ($0 < \beta < 1$):

$$\text{completion} \Leftrightarrow \frac{1}{\delta} \sum_{i=t-\delta}^t \theta_i > \beta \quad (3)$$

V. AUTOMATION

A. Control

The robotic arms are controlled using the da Vinci research interface [19]. For each instrument, we use a control mode that superimposes a Cartesian motion $\Delta = [\Delta T | \Delta R]$ to the motion caused by the surgeon's manipulation of the master manipulator. In our case, the motion Δ is given in the camera coordinate system \mathcal{C}_{cam} .

Let $\{\{T_t | R_t\}\}_{1 \leq t \leq \tau}$ be a learned trajectory, expressed in the task coordinate system \mathcal{C}_{task} and computed as explained in the next section. This trajectory has been normalized by subtracting its initial position, so that $T_1 = 0, R_1 = Id$. For a given instrument, we execute this relative trajectory at the position of the tool tip at the instant when the automatic execution is started, which we denote by T_1^{in} expressed in \mathcal{C}_{task} . Our experiments have shown such execution to be a natural way to create a seamless transition between the manual control and the automatic execution. The fact that

the trajectory is executed relative to the current tool position when automatic control is initiated leaves the flexibility to the operator to vary his/her manipulation from the exact demonstrated task, e.g. by inserting the needle at different positions.

The superimposed motion is then given for $1 \leq t \leq \tau$ by

$$\begin{cases} \Delta T_t = R_t^{\text{ta}} \cdot (T_t + T_1^{\text{in}}) + T_t^{\text{ta}} \\ \Delta R_t = R_t^{\text{ta}} \cdot R_t \cdot R_1^{\text{in}} \end{cases} \quad (4)$$

B. Average Motion Computation

We explain in this section how we learn, for a given instrument, the motion $\{[T_t|R_t]\}_{1 \leq t \leq \tau}$ to be provided to the robot. For an automatic subtask \mathcal{T}_{2i} , the motion is computed from the data $\{\tilde{\mathbf{r}}^{(k,2i)}\}_{1 \leq k \leq M}$, in which we only use the Cartesian positions and orientations of the considered instrument. In the following, the rotations are represented by quaternions and we denote this seven-dimensional position data by $\{\tilde{\mathbf{r}}^k\}_{1 \leq k \leq M}$, dropping the index $2i$.

We learn a meaningful motion from the demonstrated sequences by using a temporal curve averaging method that has been first presented in [21] and then applied successfully on continuous data and also on binary data [22]. The method consists of an iterative procedure similar to an expectation minimization algorithm, in which all trajectories are temporally synchronized to a reference average trajectory using dynamic time warping [23] before being averaged. The resulting average is used as reference within the next iteration. Using this approach, the average has a length τ equal to the mean length of the input trajectories. Average trajectories generated by this approach can be seen in Fig. 6.

We briefly summarize the approach below:

- Let $\tilde{\mathbf{r}}^{\text{ref}}$ be a reference sequence. Temporally warp $\tilde{\mathbf{r}}^k$ to $\tilde{\mathbf{r}}^{\text{ref}}$ using DTW and denote the warping function by $h^k(t)$.
- Compute the average timeline with respect to $\tilde{\mathbf{r}}^{\text{ref}}$ as

$$h(t) = \frac{1}{M} \sum_{k=1}^M h^k(t) \quad (5)$$

- Compute so-called shift functions $u^k(t) = h^k(h^{-1}(t))$ that permits to transform $\tilde{\mathbf{r}}^k$ to the average timeline.
- Compute the average

$$\tilde{\mathbf{r}}^{\text{avg}} = \frac{1}{M} \sum_{k=1}^M \tilde{\mathbf{r}}^k(u^k(t)) \quad (6)$$

- Replace $\tilde{\mathbf{r}}^{\text{ref}}$ by $\tilde{\mathbf{r}}^{\text{avg}}$ and iterate until convergence.

As initial reference, we recursively merge the input sequences two-by-two, using the same method. We adapted the approach to the motion data in the following way: first, we only use 3D position data for the time-warping synchronization, as position is the dominant feature and we notice that the orientation information does not play a major role for the synchronization in our case. Second, some of the steps above require either interpolating between two quaternions or averaging multiple quaternions. We use respectively SLERP interpolation [24] and spherical averaging [25]. After having

computed the averaged trajectory in the task coordinate system, we normalize it by subtracting its initial position.

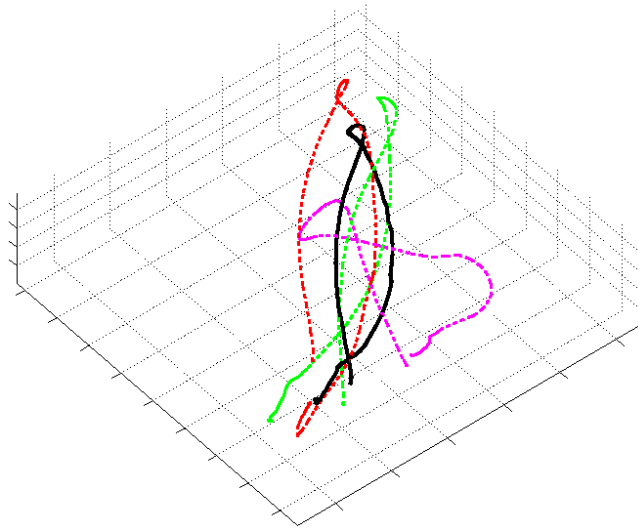
It has been noticed in the literature that if the data is high dimensional or contains multiple local variations, the DTW synchronization can be incorrect. More advanced temporal synchronization techniques like [26] have been proposed for such cases and can alternatively be used. This is however not the case with our data. We show in Fig. 6(a) how the approach provides a smooth averaged motion from several demonstrated sequences. Fig. 6(b) highlights the rotational motion of a learned trajectory.

VI. EXPERIMENTS

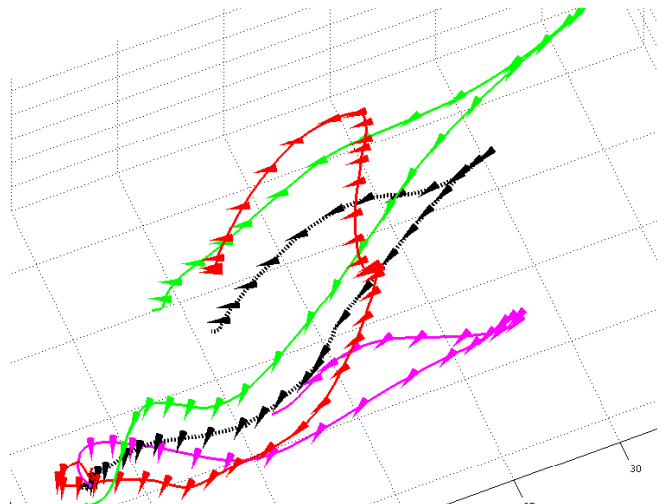
We have implemented this HMC approach in a modular application based on the CISST libraries [27], using a da Vinci robot. The application contains five interacting threads with roles: 1) completion recognition, 2) path planning, 3) robot control, 4) visual overlay and 5) main task control.

For the experiments, the authors of this paper have performed each task five times to build the task models. The illustrative video provided with this paper shows the view of an operator using our Human-Machine Collaborative system while performing the two illustrative tasks. Within this view, the label "manual" is displayed when the operator is performing the motion. The computed probability of the operator having completed the subtask is also provided. When the control switches to automatic motions, the label "auto" is displayed, as well as the planned trajectory. The displayed trajectory does not exactly overlay with the real position, so as not to interfere with the view of the operator. It is translated to the top of the view instead. This serves the purposes of supervision and cognitive workload reduction of the operator. The video shows seamless transitions between manual operation and automatic execution. It also shows that the da Vinci robot can be operated freely, e.g. by rotating its endoscopic camera. Robustness of the approach to outliers, such as repetitive unsuccessful trials to insert the needle or unexpected closing/opening of the gripper, are illustrated too. We display in Fig. 8 several images taken during the performance of the tasks with the HMC system.

Additional operators have been asked to use the system and have found the transitions seamless and intuitive. It can however happen during the first trial that an operator who has never performed the specific task beforehand does not exactly perform the expected motion. In that case, the completion may be recognized with a short delay, resulting in a trajectory that is slightly translated from the correct and expected position. This requires an adjustment of the tool position in the end. This shows that either a few trials may be necessary for the user to learn how to perform the task correctly (in a way that is similar to the demonstrated data), or that the task models should be built by including data demonstrated by the current operator and illustrating his/her style. In our experiments, the HMMs models use a mixture of two Gaussians and we trigger automatic execution using $\beta = 0.9$. We average θ_t over 1 second.

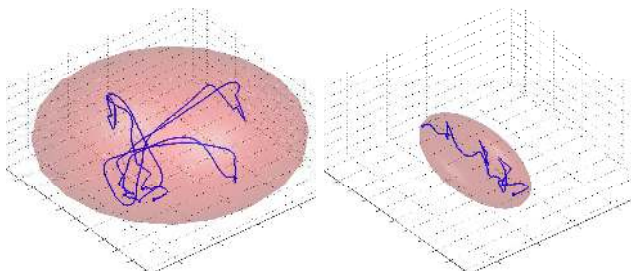


(a) Input trajectories (dotted lines) and average trajectory (black continuous line).

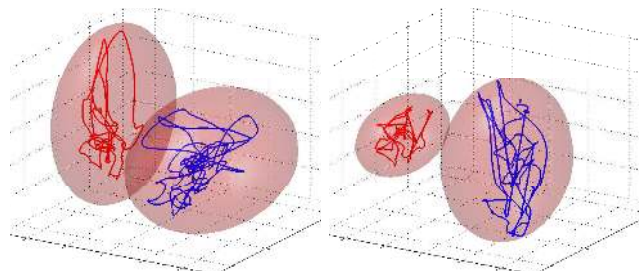


(b) Input trajectories (continuous lines) and average trajectory (black dotted line). Emphasis on rotational motion, illustrated by displaying one frame axis over time.

Fig. 6. Examples of average trajectories.



(a) Pin Task: 3D motion of the master manipulator. Manual performance (left) versus HMC performance (right) displayed side-to-side using the same scale.



(b) Suturing Task: 3D motions of the left and right master manipulators. Manual (left) versus HMC (right) performance displayed side-to-side using the same scale. The left ellipse always corresponds to the left manipulator.

Fig. 7. Workspace analysis of the master manipulators.

The benefits of the approach for easing tele-operation control lie in the fact that the motions of the master-manipulators are greatly reduced. Indeed, the large transportation motions are executed automatically and do not require any operator movements. This has two consequences: the need for clutching to readjust the master manipulators becomes almost inexistent and master-manipulator collisions are less likely to occur.

An analysis of the master manipulators workspace is shown in Fig. 7. It compares the master manipulators motions when the HMC system is used or not. For the pin task, Fig. 7(a) shows that the master manipulator workspace is reduced to a much smaller volume when the HMC system is used, since the movement to be accomplished is now restrained to only grasping or pinning down a pin. For the suturing task, only the left instrument is automated. Fig. 7(b) shows that the right manipulator uses a similar workspace when the HMC system is used. The left manipulator, however, uses a much smaller workspace with the HMC framework, as its role is reduced to two fine motions, namely grasping and handing in the needle. The compared

standard deviations of the master manipulator positions, in each direction and averaged over 5 sequences, are given in Table III.

We observe a similar result for the distances traveled by the manipulators: the traveled distance is in average reduced by a factor of 4 during our experiments with the pin task, and by a factor of 1.3 for the suture task. We are planning a wider study that should confirm these results on a larger population of users. The times for the task completion are similar with and without the use of the HMC approach: in average 48 seconds for the pin task and 64 seconds for the suturing task. It would be straightforward to reduce the times by accelerating the automatic transportation motions. We have not explored this direction yet, for safety concerns.

VII. DISCUSSION AND CONCLUSION

In this paper, we have proposed a novel human-machine collaborative approach for tele-surgery based on learning from demonstration. Fine motions are performed by the operator, while real-time recognition of their termination triggers the automatic execution of previously learned motions

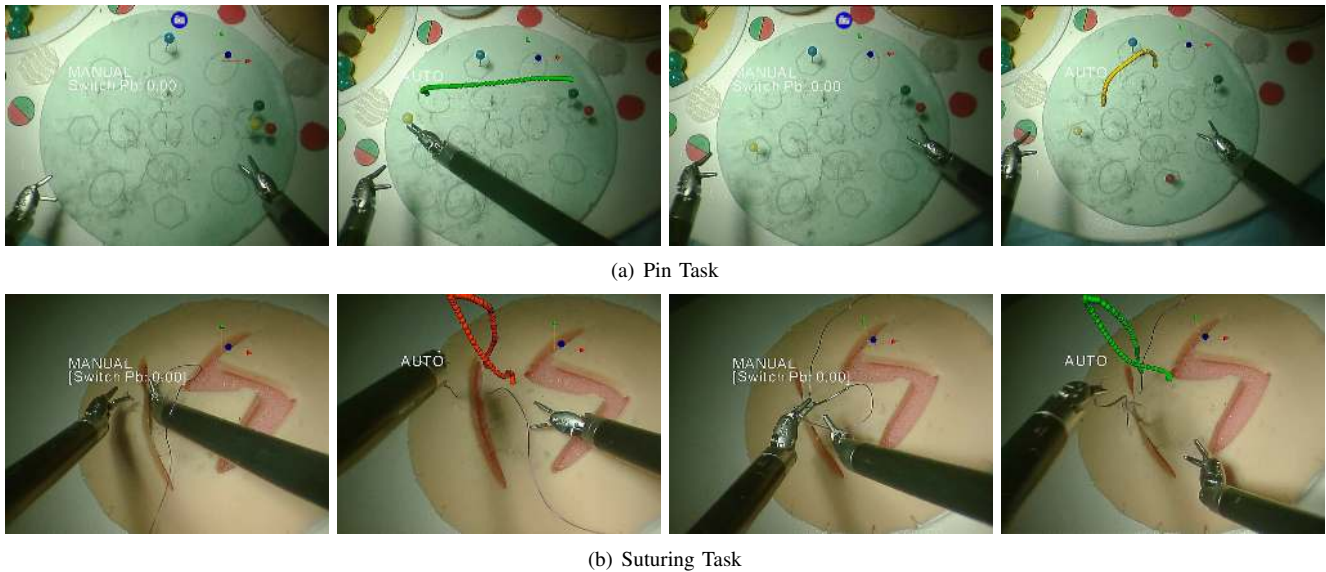


Fig. 8. Illustration of the tasks performed with the HMC framework. A complete performance can be seen in the illustrative video.

	Pin Task			Sut Task		
	x	y	z	x	y	z
Manual	55.4	50.4	22.1	17.7	21.4	38.3
HMC	10.6	19.9	23.1	12.1	16.7	23.2

TABLE III

STANDARD DEVIATIONS OF THE MASTER MANIPULATOR POSITIONS (IN MILLIMETERS), FOR PIN-TASK (RIGHT MASTER MANIPULATOR) AND SUT-TASK (LEFT MASTER MANIPULATOR). COMPARISON WITH AND WITHOUT THE USE OF THE HMC SYSTEM.

that do not involve any interaction with the environment. We have shown, using the da Vinci tele-surgical robot, that when such motions are large transportation motions, this form of collaboration improves the usage the master-manipulators workspace. Moreover, experiments show that such human-machine collaboration permits seamless and intuitive switching between manual operation and automatic execution.

Our approach neither requires a complete description of the environment nor motion-preprogramming: the recognition system, the executed motion and the ordering of the sub-tasks are directly inferred from demonstrated sequences in which the automated parts are labeled. Furthermore, by displaying the planned 3D trajectory in the field of view of the operator, he/she can supervise the automated motions. Since the executed motion is superimposed onto the master-manipulator movements, he/she can adjust the trajectory if needed. Finally, the automatic execution can be safely stopped either by clutching the robot or by asking an assistant to stop the control.

In the future, we plan to extend our approach to fine manipulation subtasks, such as needle insertion by taking advantage of visual information provided by the stereo endoscope. We will investigate how to incorporate cues from

the environment in the learning framework, like contacts between tissues and instruments. We would also like to address longer tasks, in which the succession of the subtasks may not be sequential but could contain options.

Finally, in addition to learnt task-specific motions, it would be interesting to enhance the HMC system with the capability to automate simple generic motion, for instance triggered by voice command. Generic motions, like the automatic displacement of the camera to focus on a specific tool, could also improve the ergonomic usage of a tele-operated robot.

VIII. ACKNOWLEDGMENTS

The authors would like to thank Brandon Itkowitz, Simon DiMaio from Intuitive Surgical™ and Anton Deguet from the Johns Hopkins University for their support with the Intuitive Research API. This work is funded by NSF grants CPS-0931805 and CDI-0941362. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

- [1] H. C. Lin, I. Shafran, D. Yuh, and G. D. Hager, "Towards automatic skill evaluation: Detection and segmentation of robot-assisted surgical motions," *Computer Aided Surgery*, vol. 11, no. 5, pp. 220–230, 2006.
- [2] J. van den Berg, S. Miller, D. Duckworth, H. Hu, A. Wan, X.-Y. Fu, K. Goldberg, and P. Abbeel, "Superhuman performance of surgical tasks by robots using iterative learning from human-guided demonstrations," in *ICRA*, 2010, pp. 2074–2081.
- [3] H. G. Mayer, F. J. Gomez, D. Wierstra, I. Nagy, A. Knoll, and J. Schmidhuber, "A system for robotic heart surgery that learns to tie knots using recurrent neural networks," in *IROS*, 2006, pp. 543–548.
- [4] F. Nageotte, P. Zanne, C. Doignon, and M. de Mathelin, "Stitching planning in laparoscopic surgery: Towards robot-assisted suturing," *I. J. Robotic Res.*, vol. 28, no. 10, pp. 1303–1321, 2009.
- [5] G. Guthart and J. K. S. Jr., "The intuitive™ telesurgery system: Overview and application," in *ICRA*, 2000, pp. 618–621.
- [6] J. Rosen, J. Brown, L. Chang, M. Sinanan, and B. Hannaford, "Generalized approach for modeling minimally invasive surgery as a stochastic process using a discrete markov model," *IEEE Trans. on Biomedical Engineering*, vol. 53, no. 3, pp. 399–413, 2006.

- [7] B. Varadarajan, C. E. Reiley, H. Lin, S. Khudanpur, and G. D. Hager, "Data-derived models for segmentation with application to surgical assessment and training," in *MICCAI (1)*, 2009, pp. 426–434.
- [8] C. Staub, T. Osa, A. Knoll, and R. Bauernschmitt, "Automation of tissue piercing using circular needles and vision guidance for computer aided laparoscopic surgery," in *ICRA*, 2010, pp. 4585–4590.
- [9] D. Kragic and G. Hager, "Task modeling and specification for modular sensory based human-machine cooperative systems," in *IROS*, 2003, pp. 3192–3197.
- [10] B. P. L. Lo, A. Darzi, and G.-Z. Yang, "Episode classification for the analysis of tissue/instrument interaction with multiple visual cues," in *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2003, pp. 230–237.
- [11] K. Yoshimitsu, F. Miyawaki, T. Sadahiro, K. Ohnuma, Y. Fukui, D. Hashimoto, and K. Masamune, "Development and evaluation of the second version of scrub nurse robot (snr) for endoscopic and laparoscopic surgery," in *IROS*, 2007, pp. 2288–2294.
- [12] S. Speidel, G. Sudra, J. Senemaud, M. Drentschew, B. P. Miller-Stich, C. Gutt, and R. Dillmann, "Recognition of risk situations based on endoscopic instrument tracking and knowledge based situation modeling," in *Med. Imaging. SPIE*, 2008.
- [13] N. Padoy, T. Blum, A. Ahmadi, H. Feussner, M.-O. Berger, and N. Navab, "Statistical modeling and recognition of surgical workflow," *Medical Image Analysis*, doi:10.1016/j.media.2010.10.001, 2010.
- [14] N. Padoy, D. Mateus, D. Weinland, M.-O. Berger, and N. Navab, "Workflow monitoring based on 3d motion features," in *Proceedings of the International Conference on Computer Vision Workshops, IEEE Workshop on Video-oriented Object and Event Classification*, 2009.
- [15] F. Devernay, F. Mourgues, and É. Coste-Manière, "Towards endoscopic augmented reality for robotically assisted minimally invasive cardiac surgery," in *MIAR*, 2001, pp. 16–20.
- [16] J. Leven, D. Burschka, R. Kumar, G. Zhang, S. Blumenkranz, X. Dai, M. Awad, G. Hager, M. Marohn, M. Choti, C. Hasser, and R. Taylor, "Davinci canvas: A telerobotic surgical system with integrated, robot-assisted, laparoscopic ultrasound capability," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2005, Lecture Notes In Computer Science (LNCS)*, vol. 4190, 2005, pp. 811–818.
- [17] B. Vagvolgyi, S. P. DiMaio, A. Deguet, P. Kazanzides, R. Kumar, C. Hasser, and R. H. Taylor, "The surgical assistant workstation," *Insight Journal*. [Online]. Available: <http://hdl.handle.net/10380/1466>
- [18] J.-B. Gómez, A. Ceballos, F. Prieto, and T. Redarce, "Mouth gesture and voice command based robot command interface," in *ICRA*, 2009, pp. 333–338.
- [19] S. DiMaio and C. Hasser, "The da vinci research interface," *Insight Journal*. [Online]. Available: <http://hdl.handle.net/10380/1464>
- [20] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
- [21] K. Wang and T. Gasser, "Alignment of curves by dynamic time warping," *Annals of Statistics*, vol. 25, no. 3, pp. 1251–1276, 1997.
- [22] S.-A. Ahmadi, T. Sielhorst, R. Stauder, M. Horn, H. Feussner, and N. Navab, "Recovery of surgical workflow without explicit models," in *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2006, pp. 420–428.
- [23] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 26, no. 1, pp. 43–49, 1978.
- [24] E. B. Dam, M. Koch, and M. Lillholm, "Quaternions, interpolation and animation," University of Copenhagen, Tech. Rep. DIKU-TR-98/5, 1998.
- [25] F. L. Markley, Y. Cheng, J. L. Crassidis, and Y. Oshman, "Averaging quaternions," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 4, pp. 1193–1196, 2007.
- [26] F. Zhou and F. De la Torre, "Canonical time warping for alignment of human behavior," in *Advances in Neural Information Processing Systems Conference (NIPS)*, December 2009.
- [27] A. Deguet, R. Kumar, R. Taylor, and P. Kazanzides, "The cisst libraries for computer assisted intervention systems," *Insight Journal*. [Online]. Available: <http://hdl.handle.net/10380/1465a>