Semi-autonomous Robot-assisted Cooperative Therapy Exercises for a Therapist's Interaction with a Patient

Carlos Manuel Martinez [1], Jason Fong[1], S. Farokh Atashzar[2] IEEE Member , and Mahdi Tavakoli[1] IEEE Member

Abstract— **Recent increases in demand for post-stroke motor rehabilitation services together with limited time of therapist and accessibility issues, in particular for patients living in remote areas, have created a significant burden on healthcare systems worldwide. Semi-autonomous techniques that allow for sharing the time of a therapist between multiple patients have attracted great interest. Among them Learning from Demonstration (LfD) based robots have been studied as solutions to address this growing demand. In this work, a Gaussian Mixture Model (GMM) and Gaussian Mixture Regression (GMR) based LfD approach are proposed to generate a versatile framework to deliver rehabilitation in the absence of the therapist. To collect data for training the models, a bilateral telerehabilitation system is used to enable patient-therapist collaborative task performance is one Degree of Freedom (DOF). The performance and generalizability of the trained model are demonstrated for a variety of patient actions.**

I. Introduction

Stroke is the fifth leading cause of death globally, causing approximately 6.5 million deaths each year [1]. In Canada, there are more than 62,000 stroke cases each year, and 405,000 Canadians are living with long-term stroke disability [2]. Stroke and cardiovascular disease have a major economic impact on Canada of more than $20.9B a year [3]. Motor rehabilitation is imperative for survivors in acute and chronic phases after a stroke to help patients in regaining the lost sensorimotor functionality. Research has found that by actively engaging stroke patients in repetitive exercises, it might be possible to stimulate a rewiring phenomenon of the neural pathways in the brain, which can then result in enhancement of motor functions and help patients in relearning the lost movements. This procedure is referred to as neuroplasticity in the literature [4]. Over the past two decades, there has been a growing demand for rehabilitation technologies, among which robotic systems have been widely studied for assisting recovery following a disability [5], [6], [7]. Pre-programmed robots are known for their restrictive performance in comparison to the flexibility and adaptability of the performance of a skilled human therapist.

This research was supported by the Canada Foundation for Innovation (CFI) under grant LOF 28241; the Alberta Innovation and Advanced Education Ministry under Small Equipment Grant RCP-12-021; the Natural Sciences and Engineering Research Council (NSERC) of Canada under the Collaborative Health Research Projects (CHRP) Grant #316170; and the Quanser, Inc.

[1]Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta, Canada; [2]Department of Mechanical and Aerospace Engineering, Department of Electrical and Computer Engineering, New York University, USA;

CONTACT M. Tavakoli. Email: mahdi.tavakoli@ualberta.ca

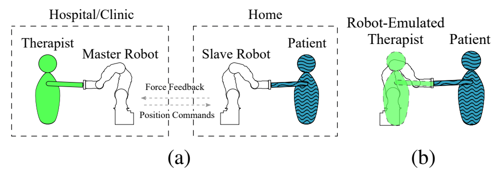


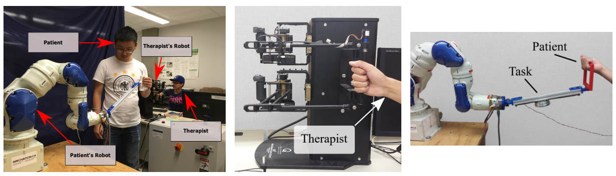
Fig. 1. Illustrations for the TIL phase (a), where the patient interacts with the therapist, and TOL phase (b), where the patient interacts with a slave robot that emulates the therapist's behavior.

In order to directly combine the therapists skills and the characteristics of a robotic system, telerobotic rehabilitation systems have been developed, using which the therapist can generate motion command through a robotic console, while perceiving the reactions of the patient, and the patient would receive the motion command of the therapist to cooperatively conduct the task in physical or virtual reality environment [8], [9], [10]. In this paper, our focus is on bilateral haptics-enabled telerobotic system, and their use to train robots in performing rehabilitation tasks. We define cooperative, upper-limb tasks as tasks that require the use of two hands to complete [11], such as holding a jar and unscrewing its lid or lifting an object with two hands. We refer to the situation where the therapist is kinesthetically interacting with the patient remotely as Therapist-In-Loop (TIL). While TIL bilateral telerehabilitation has many advantages over unilateral telerehabilitation, a therapist may not always be available to interact with the patient over the telerehabilitation medium. A solution to this problem proposed here is to first generate computational models to learn the characteristics of the administered therapy by the therapist during a live telerehabilitation session, and then imitate it, for when the therapist is not in the loop. As a result, in the absence of the therapist, the patient can continue to practice the task in cooperation with the semi-autonomous patient-side robot. We refer to this situation, when the therapist is absent, as Therapist-Out-of-Loop (TOL). The paradigm to transition from TIL to TOL will be based on learning from demonstration (LfD) techniques. Fig. 1 depicts the TIL and TOL phases.

In this paper, we are interested in creating cooperative tasks with variable difficulty levels. The tasks will be cooperatively performed, with the therapist generating the motion command, to deliver intervention, using a master robot, while

978-1-7281-2723-1/19/$31.00 ©2019 IEEE

the slave robot and the patient directly interact with the physical tasks. The experimental platform is shown Fig. 2. In this paper, LfD based on Gaussian Mixture Models (GMM) and Gaussian Mixture Regression (GMR) will then be implemented for the purpose of learning and computationally modeling the way that the therapist interacts with the patient. We hypothesize that the combination of these techniques can provide a suitable middle ground between hand-over-hand and fully semi-autonomous therapy.



(a) (b) (c)

Fig. 2. (a) Experiment setup and demonstration; (b) HD^2 High Definition Haptic Device used as the master robot by the therapist; (c) a Motoman SIA-5F industrial robot.

A. Prior Art

The concept of semi-autonomous systems and LfD has attracted a great deal of research interest in the past decade. Calinon et al. [12] propsed LfD models to train a robot for cooperatively lifting a beam in a setup similar to what we propose in this work. Gribovskaya et al. [13] extended the work to ensure global asymptotic stability (GAS) of the system. Peternel et al. [14] created a variant to learn motion and compliance during a highly dynamic cooperative sawing task. In addition, few groups have applied LfD techniques towards the practice of physical therapy in rehabilitation medicine. Maaref et al. [15] described the use of LfD as the underlying mechanism for an assist-as-needed paradigm. Lydakis et al. [16] learned and classified demonstrations of therapy tasks through EMG measurements. Lauretti et al. [17] optimized a system built on dynamic motor primitives for learning therapist-demonstrated paths for activities of daily living. Najafi et al. [18] learned the trajectory and interaction impedance provided by a therapist and provided user experiment evaluations. Based on our previous research, in this present work, we have developed a new approach for LfD with direct application in autonomous robotic rehabilitation.

II. Learning from Demonstration

LfD is a paradigm focused on allowing a human user to program a robot through demonstration of desired behaviors. In other words, a trainer (which can be a human or even another machine) physically demonstrates the behaviors to be imitated by the robot, thereby programming the robot without needing to change computer code (which is called the demonstration phase) [19], [20], [21]. In general, the behaviors are actions or movements to be later imitated by the robot (which is called the imitation phase).

In this paper, GMM and GMR are used as the underlying learning and imitation algorithms for the LfD paradigm. The GMM algorithm takes multiple demonstrations and extracts the necessary parameters to describe the data with Gaussian functions. This process avoids redundancy of data in memory. The GMR algorithm uses the stored data and, based on the regression input, retrieves the general form of the output.

A. Gaussian Mixture Models

GMM is a probability density function widely used for generatively modeling data [22], [23]. The model parameterizes a set of datapoints and its underlying function as weighted sums of Gaussian component densities, with each Gaussian having its own mean and covariance. Because of the simplistic, adaptable nature of Gaussian functions and the advantages that come with generative modeling, GMM is widely used for LfD. GMM is a weighted sum of K component Gaussian densities given by the equation,

$$p(\xi_j) = \sum_{k=1}^{K} p(k) p(\xi_j|k) \qquad (1)$$

where $p(k)$ are the prior probabilities, $p(\xi_j|k)$ is the conditional density function, and ξ_j is the D-dimensional continuous-valued data vector.

The Expectation-Maximization (EM) algorithm is used to train the GMM parameters. It is widely used to iterate the parameters until convergence of an optimization factor. EM has a simple local search technique that guarantees increase of the likelihood.

B. Gaussian Mixture Regression

The GMR model uses the Gaussian conditioning theorem and linear combination properties of Gaussian distributions to retrieve the desired output values from a GMM [23]. GMR traditionally uses temporal values (ξ_t) as query points to estimate the corresponding spatial values ($\hat{\xi}_s$) through regression. Given a set of temporal and spatial values for a kth component of a GMM, the representations of the mean and covariance matrices are given as

$$\mu_k = \{\mu_{t,k}, \mu_{s,k}\}, \Sigma_k = \begin{pmatrix} \Sigma_{t,k} \Sigma_{ts,k} \Sigma_{st,k} \Sigma_{s,k} \end{pmatrix} \qquad (2)$$

Conditional expectation ($\hat{\xi}_s$) and conditional covariance ($\hat{\Sigma}_s$) of the output ξ_s given ξ_t are then calculated for a mixture of all GMM k components.

Note that while the query points are described as temporal points, these inputs to the GMM and GMR can be any type of data. As is the case in our work, the learned system behaviors can be time-independent. Spatial coordinates, as an example, can be used as the query points.

III. Experiments, materials, and methods

A. Materials

The teleoperation system has two robots: a master robot (Quanser High Definition Haptic Device, or HD^2) used directly by the therapist, and a slave robot (Yaskawa-Motoman SIA5F) handled by the patient. Even though both robots have upwards of seven DOF, the movements of the users and

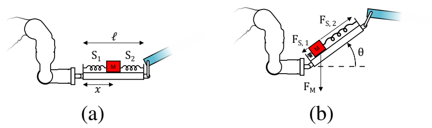


Fig. 3. Design of the cooperative task. The slave robot holds one side of the bar, while the patient holds the bar from its other side.

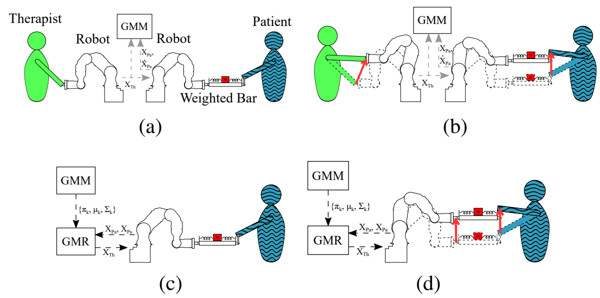


Fig. 4. The two phases of LfD are shown separately through (a)-(b) and (c)-(d), respectively. In (a) and (b), the therapist is present (making this the TIL phase). Then in (c) and (d), the patient is practicing in the absence of the therapist (the TOL phase).

robots are constrained to only one DOF due to the nature of the cooperative task. A potentiometer is used to measure the angle θ that a bar attached to the Motoman makes with the horizontal axis. A mass is placed on the bar, and allowed to slide along the length of the bar. Two identical springs attached to opposite sides of the bar pull the sliding mass towards their respective sides. Fig. 3 shows the design of the bar.

B. Methods

1) Task: We choose a task that requires the therapist and the patient to collaborate to lift a bar. The spring-mass system on attached to the bar will allow the mass to slide towards one end of the bar in a manner directly proportional to θ, similar to if a box was being lifted by the participants with objects inside of it that slide back and forth freely. The therapist can thereby adjust the amount of force the patient must exert to lift the bar by either lowering or raising his/her own end, effectively resulting in an assistive/resistive therapy provided in the context of a functional task.

2) GMM and GMR design: The demonstration phase uses GMM to create K Gaussian distributions of dimensionality D. In this paper, K has a value of 12 (decided experimentally) and D has a value of 3. D has as many dimensions as inputs to the GMM. These inputs are:

- Therapist position in vertical axis (X_{Th})
- Patient position in vertical axis (X_{Pa})
- Patient velocity in vertical axis $(\dot{X}_{Pa})$

The imitation phase uses GMR to retrieve the trajectory of the movements. The GMR algorithm takes X_{Pa} and $\dot{X}_{Pa}$ as inputs, and based on the GMM distributions, retrieves an appropriate value for X_{Th} as an output. Fig. 4 shows the process of learning and reproducing the therapist's behavior with the given GMM inputs.

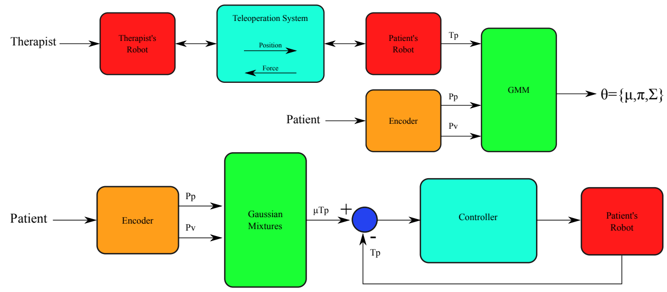


Fig. 5. The top shows a block diagram for the demonstration phase while the bottom shows a block diagram for the reproduction phase.

3) Experiments: The system is trained with different scenarios. The idea is to create an adaptive system capable of assisting the patient, resisting the patient, or keeping a neutral behavior with the patient, based on the patient's performance. To do so, the GMM is trained with three different demonstrations of these scenarios. Later, during the demonstration phase, the GMR takes the patient's behavior as input to reproduce the therapist behavior.

The system measures the patient's position and velocity to learn the patient's behavior. Based on the patient's velocity, the therapist can make a decision on how much assistance or resistance to apply. We selected four different general scenarios for the system to learn, described as follows. A positive and fast velocity means that the patient is able to easily perform the task and a resistance can be applied to challenge the patient during the therapy. A positive and medium velocity means that the patient can perform the task and the therapist only has to keep/maintain a neutral behavior. A positive and slow patient's velocity means that the patient has some problems/difficulties in performing the task so assistance is provided by the therapist. Finally, a negative patient's velocity means the patient is experiencing significant difficulty and is unable to perform the task. These scenarios will also be referred to as ”fast”, ”medium”, ”slow”, and ”back”, respectively. Two block diagrams of the system are shown in Fig. 5.

IV. Results and Discussion

Two able-bodied participants played the roles of the therapist and the patient. During the experiments, the therapist's position, patient's velocity and the patient's position data were recorded and then used in the GMM algorithm to train the robot. Three demonstrations of each scenario were used to train the GMM. After the demonstration phase, the system's imitation performance was tested. The GMR model takes the patient position and patient velocity as inputs and returns the estimated therapist position as an output. This estimated therapist position was used to move the slave robot in the imitation phase. Therapist positions and patient positions are mainly used in analysis of the results. We present our results in two parts: first a qualitative examination of the system's imitation results , and second an evaluation of the training data's efficacy.

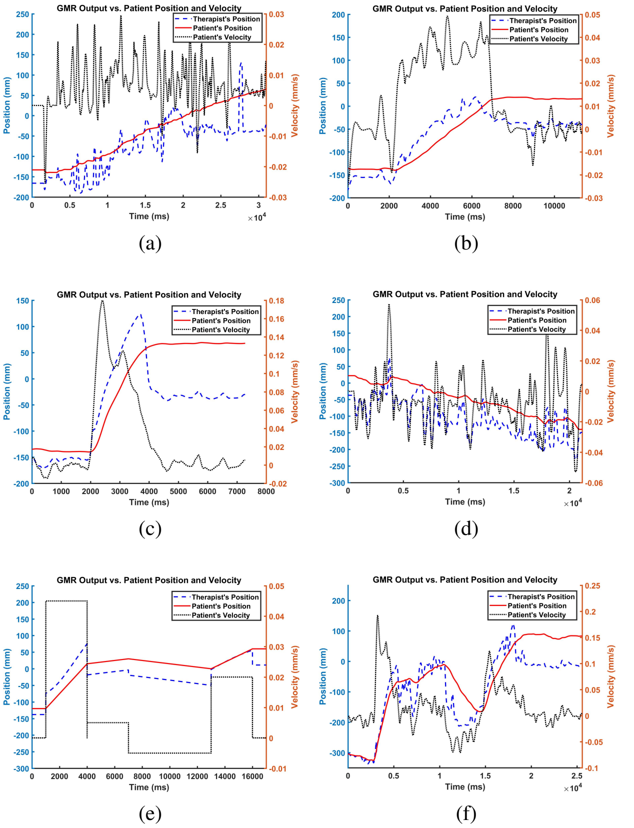


Fig. 6. The plots shows the GMR output (blue and dashed line), the patient's position (red and solid line) and the patient's velocity (black and dotted line). (a) shows the slow scenario, (b) the medium scenario, (c) the fast scenario, (d) the back scenario, (e) the simulated trajectory, and finally the multi-behavioral trajectory in (f).

A. *GMR output for different patient behaviors*

GMR output results are shown in Fig. 6. We show results for assisting, resisting or keeping a neutral behavior given velocity-specific patient trajectories.

The obtained plots show how the system is able to respond similarly to how a reference therapist would. Based on the input data, the GMR output demonstrates a reasonable accuracy through most of the different scenarios. For patient trajectory data with higher velocities, GMR returns accurate trajectories with low variance, as in Fig. 6 (c). For slower velocities however, velocity measurements are heavily affected by noise from hand tremor, muscle fatigue, etc.; the ”slow” and ”back” scenarios seen in Fig. 6 (a) and (d) exhibit this problem. Results for these scenarios are less accurate, often switching between behaviors. A simulated data trajectory is also used to show the system's response through different scenarios given an ideal patient motion trajectory with minimal velocity fluctuation. In this situation the system produces very accurate results. Finally, a real complex demonstration that combines multiple behaviors is used to show the robustness of the system, including the transitions between scenarios. The system responds quite accurately throughout the task, but transitions are made too quickly to be safely implemented in clinical settings. Designing a different motion controller, for example based on impedance control principles, is a possible solution.

B. *Evaluation of Training Data Quality*

Motivated by the previous results, we examine the efficacy of the dataset used to train the system. In the first experiment, a total of 12 demonstrations were recorded for training the GMM, with N = 3 demonstrations for each of the four behaviors. We now remove a single demonstration and use it instead as the input for the GMR process; this is performed for every demonstration dataset used for training. By operating under this assumption, we can find demonstrations that are less useful if their trajectories are already included in the system. We quantify this as the error between the reference therapist trajectory, used to train the system, and the GMR output.

Table 1 shows the average error between the GMR output and the recorded therapist position for every removed demonstration. By extension, we can infer that the system is able to better fit the Gaussian components to higher velocity data. Results for overall average error of the slow and medium cases in particular are very similar. This may indicate that the GMM may not be able to distinguish between the velocities of the two cases well, resulting in equal sensitivities when demonstrations from either case is removed.

TABLE I

ERROR BETWEEN GMR OUTPUT AND RECORDED THERAPIST POSITION, AVERAGED OVER THE DURATION OF THE DEMONSTRATION.

Removed demonstration	Average error between GMR output and recorded therapist position (mm)			
	Slow	Medium	Fast	Back
1	34.576	18.088	17.661	40.135
2	24.320	35.364	15.404	55.013
3	30.739	36.326	8.549	53.515
Overall average	29.878	29.926	13.871	49.554

A resultant suggestion for works in the field of rehabilitation medicine looking to incorporate LfD principles would be to provide more demonstrations when aiming to imitate motions with large inherent variation, such as the slower movements seen in this work.

V. Conclusion

In this paper, semi-autonomous imitations of therapist movements were performed. The results showed that demonstrations provided by the therapist in response to faster patient movements were better learned by the algorithm. Slower patient movements had larger variations in velocity and produced less accurate imitations of the therapists behavior. Examining the sensitivity of the system to the number of demonstrations showed that the difference between the GMR-produced interactions and those provided by a therapist was relatively small (between 8.549 and 17.661 mm) at a higher patient velocity. However, the mentioned difference was increased substantially (up to 55.515 mm) for finer patient motions that involve lower velocities.

References

[1] E. J. e. a. Benjamin, "Heart disease and stroke statistics—2017 update: A report from the american heart association," *Circulation*, 2017.

[2] Heart and S. F. of Canada, *Mind the connection: Preventing stroke and dementia*. Stroke Report, 2016.

[3] Heart and S. F. of Canada, *Getting to the heart of the matter: solving cardiovascular disease through research*. Report on the Health of Canadians, 2015.

[4] P. Lum, D. Reinkensmeyer, R. Mahoney, W. Z. Rymer, and C. Burgar, "Robotic devices for movement therapy after stroke: Current status and challenges to clinical acceptance," *Topics in Stroke Rehabilitation*, vol. 8, no. 4, pp. 40–53, 2002. PMID: 14523729.

[5] H. M. Van der Loos, D. J. Reinkensmeyer, and E. Guglielmelli, *Rehabilitation and Health Care Robotics*, pp. 1685–1728. Cham: Springer International Publishing, 2016.

[6] R. Voelker, "Rehabilitation medicine welcomes a robotic revolution," *JAMA*, vol. 294, no. 10, pp. 1191–1195, 2005.

[7] S. F. Atashzar, M. Shahbazi, and R. Patel, "Haptics-enabled interactive neurorehabilitation mechatronics: Classification, functionality, challenges and ongoing research," *Mechatronics*, vol. 57, pp. 1 – 19, 2019.

[8] C. R. Carignan and H. I. Krebs, "Telerehabilitation robotics: Bright lights, big future?," *Journal of Rehabilitation Research and Development*, vol. 43, pp. 695–710, August 2006. Copyright - Copyright Superintendent of Documents Aug/Sep 2006; Document feature - Diagrams; Photographs; Illustrations; Last updated - 2017-11-09; CODEN - JRRDDB.

[9] S. F. Atashzar, I. G. Polushin, and R. V. Patel, "A small-gain approach for nonpassive bilateral telerobotic rehabilitation: Stability analysis and controller synthesis," *IEEE Transactions on Robotics*, vol. 33, pp. 49–66, Feb 2017.

[10] S. F. Atashzar, M. Shahbazi, M. Tavakoli, and R. V. Patel, "A grasp-based passivity signature for haptics-enabled human-robot interaction: Application to design of a new safety mechanism for robotic rehabilitation," *The International Journal of Robotics Research*, vol. 36, no. 5-7, pp. 778–799, 2017.

[11] R. Sainburg, D. Good, and A. Przybyla, "Bilateral synergy: A framework for post-stroke rehabilitation," *Journal of neurology & translational neuroscience*, vol. 1, October 2013.

[12] S. Calinon, P. Evrard, E. Gribovskaya, A. Billard, and A. Kheddar, "Learning collaborative manipulation tasks by demonstration using a haptic interface," in *Advanced Robotics, 2009. ICAR 2009. International Conference on*, pp. 1–6, IEEE, 2009.

[13] E. Gribovskaya, S. M. Khansari-Zadeh, and A. Billard, "Learning non-linear multivariate dynamics of motion in robotic manipulators," *The International Journal of Robotics Research*, vol. 30, no. 1, pp. 80–117, 2011.

[14] L. Peternel, T. Petrič, E. Oztop, and J. Babič, "Teaching robots to cooperate with humans in dynamic manipulation tasks based on multi-modal human-in-the-loop approach," *Autonomous Robots*, vol. 36, pp. 123–136, January 2014.

[15] M. Maaref, A. Rezazadeh, K. Shamaei, R. Ocampo, and T. Mahdi, "A bicycle cranking model for assist-as-needed robotic rehabilitation therapy using learning from demonstration," *IEEE Robotics and Automation Letters*, vol. 1, pp. 653–660, July 2016.

[16] A. Lydakis, Y. Meng, C. Munroe, Y.-N. Wu, and M. Begum, "A learning-based agent for home neurorehabilitation," in *Rehabilitation Robotics (ICORR), 2017 International Conference on*, pp. 1233–1238, IEEE, July 2017.

[17] C. Lauretti, F. Cordella, E. Guglielmelli, and L. Zollo, "Learning by demonstration for planning activities of daily living in rehabilitation and assistive robotics," *IEEE Robotics and Automation Letters*, vol. 2, pp. 1375–1382, July 2017.

[18] M. Najafi, K. Adams, and M. Tavakoli, "Robotic learning from demonstration of therapist's time-varying assistance to a patient in trajectory-following tasks," in *Rehabilitation Robotics (ICORR), 2017 International Conference on*, pp. 888–894, IEEE, July 2017.

[19] S. Schaal, A. Ijspeert, and A. Billard, "Computational approaches to motor learning by imitation," *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, vol. 358, no. 1431, pp. 537–547, 2003.

[20] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne, "Imitation learning: A survey of learning methods," *ACM Computing Surveys*, vol. 50, pp. 21:1–21:35, Apr. 2017.

[21] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 469 – 483, 2009.

[22] D. Reynolds, "Gaussian mixture models," *Encyclopedia of Biometrics*, pp. 827–832, 2015.

[23] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, *Robot Programming by Demonstration*, pp. 1371–1394. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008.