

Trajectory Creation Towards Fast Skill Deployment in Plug-and-Produce Assembly Systems: A Gaussian-Mixture Model Approach

Melanie Zimmer, Ali Al-Yacoub, Pedro Ferreira, Niels Lohse

Abstract—In this paper, a technique that reduces the changeover time in industrial workstations is presented. A Learning from Demonstration-based algorithm is used to acquire a new skill through a series of real-world human demonstrations in which the human shows the desired task. Initially, the collected data are filtered and aligned applying Fast Dynamic Time Warping (FastDTW). Then the aligned trajectories are modelled with a Gaussian Mixture Model (GMM), which is used as an input to generate a generalisation of the motion through a Gaussian Mixture Regression (GMR). The proposed approach is set into the context of the openMOS framework to efficiently add new skills that can be performed on different workstations. The main benefit of this work in progress is providing an intuitive, simple technique to add new robotics skills to an industrial platform which accelerates the changeover phase in manufacturing scenarios.

Keywords— *Plug-and-Produce; Changeover; Learning by Demonstration; Fast Dynamic Time Warping; Gaussian Mixture Regression; Trajectory Learning.*

I. INTRODUCTION

In today's manufacturing, the introduction of a new product or product variants is becoming more and more common. A change in product requirements, however, often leads to a modification to the manufacturing system. In order to do these changes, a certain amount of system downtime will result, which manufacturers are keen to keep as limited as possible. One way how manufacturers and research approach the problem of product variety is through the implementation of flexible manufacturing systems that allow handling various product types [1]. As an example of a project that falls into the category of flexible manufacturing systems and in which the presented work is also set is the European project 'open Dynamic Manufacturing Operating System for Smart 'Plug-and-Produce' automation components' (openMOS) [2]. The goal of the openMOS project is the development of an innovative, openly accessible plug-and-produce system platform, which facilitates a rapid and smooth ramp-up and changeover of equipment such as industrial robots.

Enhancements in robot technology increased the use of robots in industry with one area of interest in robot skill learning [3]. Learning from Demonstration (LfD) facilitates the acquisition of new skills for the robot as the motion can simply be demonstrated using motion sensors involving less programming [4]–[6]. An important field in LfD is trajectory learning. Here, advantage is taken from the strength of human motion planning. Humans will exemplify the trajectory

considering special limitations and speed implications. However for humans, it is often difficult to keep a steady hand, and thus the produced trajectories might most certainly not be optimal so that a way of optimising the outcome is required.

This paper proposes an approach to reduce the changeover time for workstations in plug-and-produce assembly systems by using an LfD method. As part of openMOS, this work aims to contribute to the growing area of research in plug-and-produce assembly systems by proposing an intuitive changeover approach. This is envisioned to be achieved by allowing operators to build a library of desired robot skills through human demonstration that can be deployed using different types of equipment that share the same atomic skills. It is expected that this will lead to efficient learning of new skills and will ultimately decrease the delay time in changeover scenarios.

II. RELATED WORK

Within the domain of flexible manufacturing systems, the plug-and-produce paradigm facilitates the introduction, replacement or removal of a manufacturing device into the system, similar to the idea of plug-and-play in computing [7]. These capabilities of plug-and-produce systems allow for the quick introduction of a new product type.

In literature, there is no clear definition of the term changeover. Generally, changeover can be described as the total of activities of making a production line or machine ready from one production run to another, which comprises the machine set-up and clean-up process [8]. The generation of a robot skill, for example, falls under the set-up phase.

In robot skill learning, the optimisation of certain task execution is enabled through learning and imitating human motions [3]. LfD has been highly studied (e.g. [5]). One field of LfD that has been studied in more detail recently is the learning of a statistical model of a trajectory [9]. Learning a trajectory allows to instinctively add new necessary manipulation skills which are essential to handle products. Previous to being able to learn a trajectory from human motion, however, means of tracking this motion are required. Tracking physical motions of humans has received great attention over the past years [10].

Before a motion can be recognised, the use of sensor technologies is required [11]. For the data collection, image- or non-image-based types of sensors can be used. The first type includes markers with an optical camera, single and stereo cameras, or depth sensors, whereas for the second one gloves,

The authors are with the Intelligent Automation Centre, Loughborough University, Loughborough, LE11 3TU, UK (+44 (0)1509 227625; e-mail: {m.zimmer2, a.al-yacoub, p.ferreira, n.lohse}@lboro.ac.uk).

bands and even non-wearables, such as a radio frequency-based system, are considered.

As a final step, gesture classification is undertaken [5]. As this is a machine learning problem, various machine learning approaches can be found in literature. Liu and Wang [11] provide a review of the most common approaches for gesture classification, amongst others K-Nearest Neighbour, Hidden Markov Model, Support Vector Machine and Artificial Neural Networks. The authors provide a comparison of the different approaches and conclude that in order to build on their advantages, a combination of different algorithms can be deployed. Another common approach is the application of the Gaussian Mixture Model (GMM) in combination with Gaussian Mixture Regression (GMR) [3].

III. METHODOLOGY

This section describes the approach that has been taken as an initial step of providing fast skill deployment in plug-and-produce assembly systems. The central advantage of this approach is that it reduces the required time to design and program a new skill as it does not need extensive technical knowledge to programme robots. The proposed methodology follows the procedure as set out below.

Firstly, a human will demonstrate the hand movement to be learned. This movement is tracked using a motion tracker system, that allows estimating an object's position in a workspace through passive inferred markers. This tracking system has been chosen as it is non-invasive and can be attached to any object, in the case of this work, a handheld gripper. It is assumed that more than one trajectory of the same movement will be captured and it must, therefore, be ensured next that these individual trajectories are aligned for further processing. Then, using the Gaussian Mixture Model and Regression generates a new approximated trajectory. Although using these algorithms to cluster the dataset and generate a trajectory is not new, it is considered to bring advances in plug-and-produce environments such as the openMOS framework. Both, the captured trajectories and the outcome of the GMR will be displayed to the user through a Graphical User Interface (GUI). Here, the user has the possibility to make small changes to the trajectory by, for example, removing unwanted points on the line via a mouse click. Through the GUI the execution of the trajectory on the robot side can be triggered as the human will ultimately have to provide approval before deploying and testing the skill on the robot. The final coordinates of the trajectory will then be provided to the openMOS framework as an input for this new skill. An overview of these stages is depicted in **Figure 1**.

For this paper, the focus is on learning the trajectories from a dispensing task that can be used on an industrial robot. This is only the first step in the above presented methodology and is explained in more technical terms in the next subsections (see also **Figure 2**).

A. Collected Trajectories

The collection of the demonstrated trajectories is denoted by the set $\{T_m = (p_{m,1}, p_{m,2}, \dots, p_{m,n_m})\}_{m=1}^M$, where M is the number of demonstrations, m is used for indexing these demonstrations, and n_m indicates the number of points of

demonstration m . Each point of a trajectory is considered as a 2-dimensional vector $p_m = ((x_1, y_1)_m, \dots, (x_{n_m}, y_{n_m})_m)$, for $n = 1, \dots, n_m$ as for the dispensing process a flat surface is considered. The collected data will have to be pre-processed before they can be further used for the actual problem of generating a new trajectory. This process is explained in more detail in the following sections.

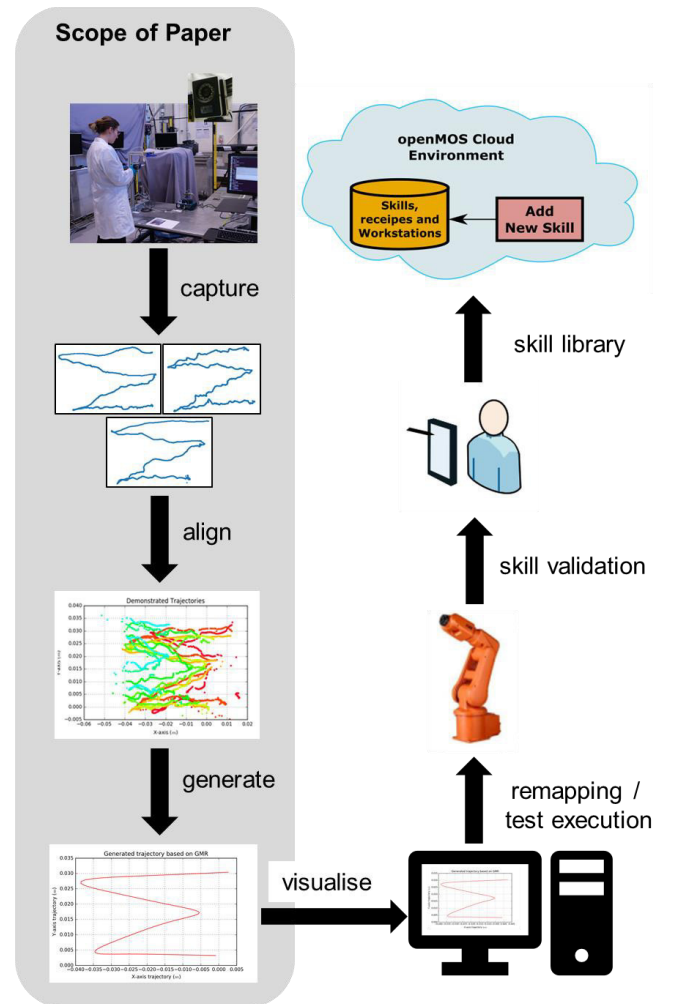


Figure 1. Overall methodology and scope of this work.

B. Pre-processing

Due to the nature of human demonstration, the sensory data collected during each trial are noisy and differ in length depending on the participant's speed. It is therefore essential to filter and align the collected trajectories prior to the learning process. To remove the noise, the Moving Average Filter was used, which averages subsets of the data [12]. In the next step, the individual trajectories were aligned in the time domain using the multi-dimensional Dynamic Time Warping (DTW) algorithm [13]. This algorithm is often used to determine time series similarity, classification, and to find corresponding regions between two-time series. DTW has a quadratic time and space complexity that limits its use to only small time series data sets. In this paper, FastDTW [14] was applied, which approximates DTW and has a linear time and space complexity.

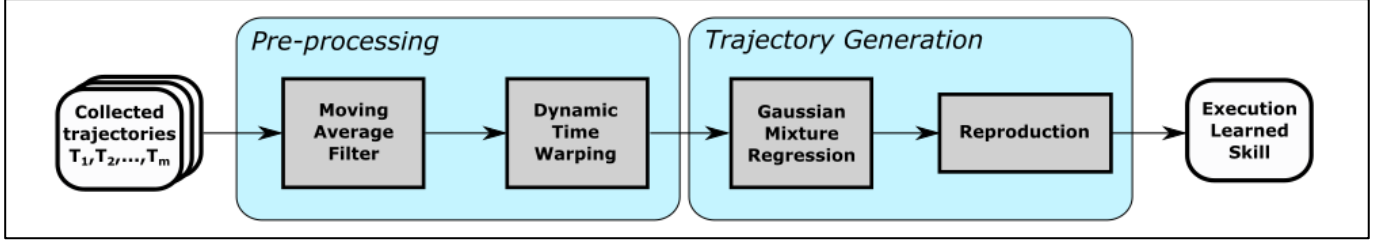


Figure 2. More detailed overview of the technical work done within the scope of this paper.

C. Gaussian Mixture Model

In general, mixture modelling is a simplified approach for density approximation of continuous or discrete data [15]. Such models allow for an appropriate flexible trade-off between model complexity and variations of the available demonstration data. From a mathematical perspective, mixture models are represented as a K component density function as shown in **Equation**

$$(1). \quad P(p_i) = \sum_{k=1}^K P(k)P(p_i | k) \quad (1)$$

where p_i is the i^{th} data point, $P(k)$ is the prior, and $P(p_i | k)$ is the conditional probability density function.

Given the demonstrated trajectories, which have been aligned previously, the dataset consists of 2D data points. These data points symbolise the operator 2D hand path during the dispensing process. This dataset is modelled by a mixture of K mixture of Gaussians. Hence, **Equation**

(1) can be re-written as seen in **Equation** (2).

$$P(p) = \sum_{k=1}^K \pi_k \frac{1}{\sqrt{(2\pi)^D |\Sigma_k|}} e^{-\frac{1}{2}(p-\mu_k)^T \Sigma_k^{-1} (p-\mu_k)} \quad (2)$$

where $\{\pi_k, \mu_k, \Sigma_k\}$ are the k^{th} Gaussian parameters, which denote the prior, mean, and covariance respectively. In this paper, we assume that the number of Gaussians is equal to the straight lines in the reference trajectory which is four.

C. Trajectory Generation

To create a general 2D trajectory from the mixture of Gaussian, GMR is applied [16]. In such a case, the sequential time steps (temporal data) are used as input data to generate the corresponding 2D trajectory values through regression. For each component in the GMM, the input (temporal data) and output parameters are separated such that the mean and covariance matrix of the k^{th} Gaussian component is defined by **Equation** (3).

$$\mu_k = \begin{bmatrix} \mu_t^k \\ \mu_s^k \end{bmatrix} \quad (3)$$

$$\Sigma_k = \begin{bmatrix} \Sigma_{t,k} & \Sigma_{ts,k} \\ \Sigma_{st,k} & \Sigma_{s,k} \end{bmatrix}$$

The conditional expectation of $p_{\{s,k\}}$, given p_k and the estimated conditional covariance of $p_{\{s,k\}}$, given p_t , are presented in **Equation** (4).

$$\hat{p}_{\{s,k\}} = \mu_s^k + \Sigma_{st,k} (\Sigma_{t,k})^{-1} (p_t - \mu_t^k) \quad (4)$$

$$\hat{\Sigma}_{\{s,k\}} = \Sigma_{s,k} - \Sigma_{st,k} (\Sigma_{t,k})^{-1} \Sigma_{ts,k}$$

According to the Gaussian distribution parameters of the k^{th} component, the estimated variable $\hat{p}_{\{s,k\}}$ and covariance $\hat{\Sigma}_{\{s,k\}}$ are mixed. Therefore, $\hat{p}_{\{s,k\}}$ can be used to estimate the mixing weighting β_k of component k as shown in **Equation** (5).

$$\beta_k = \frac{P(k)P(p_t | k)}{\sum_{i=1}^K P(i)P(p_t | i)} \quad (5)$$

Successively, the conditional expectation of p_s , given p_t , and the conditional covariance of p_s , given p_t , can be estimated by using **Equations** (4) and (5) as given in **Equation** (6).

$$\hat{p}_s = \sum_{k=1}^K \beta_k \hat{p}_{\{s,k\}} \quad (6)$$

$$\hat{\Sigma}_s = \sum_{k=1}^K \beta_k^2 \hat{\Sigma}_{\{s,k\}}$$

Evaluating $\{\hat{p}_s, \hat{\Sigma}_s\}$ at different time steps p_t with the associated covariance matrices consequently gives an estimated trajectory $\hat{T} = \{\hat{p}_t, \hat{p}_s\}$. The time step between two consecutive points depends on the process and controller requirements of the robot and the dispensing process. It is worth mentioning that only the means and covariance matrices of the modelled Gaussians are needed to reproduce a new trajectory.

IV. EXPERIMENT

As a first step towards achieving fast skill deployment in plug-and-produce assembly systems, a manual dispensing experiment was conducted for an initial data collection. For this ethics-approved experiment (Ethics Approvals (Human Participants) Sub-Committee at Loughborough University), a total of twenty participants was recruited and each participant was asked to produce one zigzag pattern on a prepared aluminum plate (6.5x9cm). The main physical components of the setup were a time-pressure dispensing unit (Fisnar JB1113N), which was connected to a syringe and a Schunk gripper with 3D-printed handle. Vicon markers were attached to this handle to capture the trajectories the participants performed. Additionally, a push button was fixed to the handle

allowing to trigger the dispensing process through a Raspberry Pi 3. A picture of a participant doing the task and exemplary outcomes can be found on the left-hand side of **Figure 1**.

from the European Union's Horizon 2020 research and innovation programme under grant agreement No 680735, project openMOS (Open Dynamic Manufacturing Operating

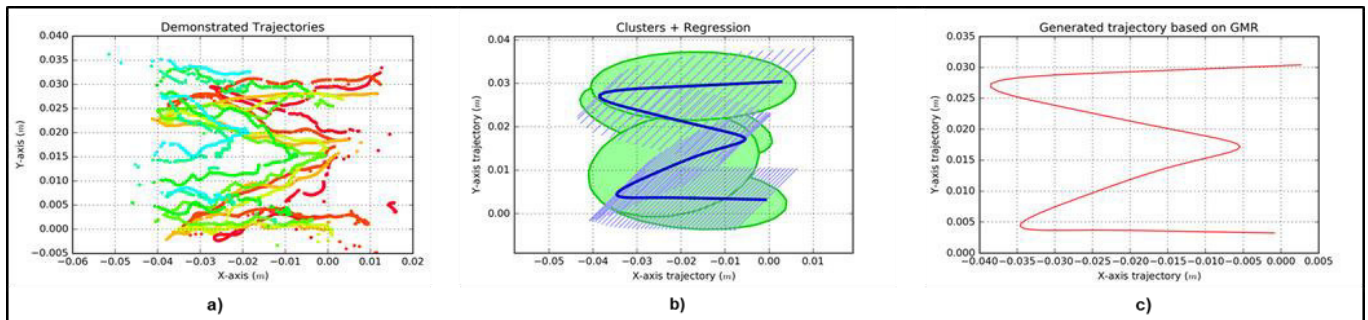


Figure 3. Results of the presented approach: a) filtered and aligned trajectories, b) four clusters found by GMM and the resulting trajectory using GMR, c) generated trajectory on its own.

V. RESULTS AND DISCUSSION

The outcomes of applying the previously proposed methodology to the collected data are illustrated in **Figure 3**. **Figure 3a)** shows the collected trajectories after they have been aligned with the multi-dimensional DTW algorithm. As can be seen, the collected trajectories contain several outliers and noise. These are due to the measurement errors of the Vicon tracking system as well as reflective objects within the workspace that were mistakenly considered as markers. **Figure 3b)** visualises the outcome of applying GMM and GMR to the pre-processed data. The GMM was instantiated to find four clusters, each resembling an edge of the pattern. Based on this, the learned GMR was used to generate a new trajectory. The trajectory can be seen in better detail in **Figure 3c)**. Compared to the trajectories produced by humans, the generated trajectory is relatively smooth.

One limitation of the experimental setup that was noticed is that the motion was captured as long as the push button on the gripper was pressed to dispense the material. Due to the viscosity of the dispensed material, some participants released the button before the nozzle was above the last point of the pattern to let the material drop. This problem can, however, be easily solved in using a different dispensing material with higher viscosity or programming of the motion tracking.

VI. CONCLUSION AND FUTURE WORK

In this paper, the first stage towards the transfer of a human skill into a manufacturing cloud system is presented to reduce the need for using conventional robotics programming approaches in similar context. The following has been shown: 1) Collection of trajectories while skill is demonstrated by human, 2) filtration of collected data and alignment of different trials, 3) modelling of demonstrated trials using Gaussian Mixture Regression. Future work will include the reproduction of the learned skill on the robot side ensuring to meet the semantic description of the openMOS skill. This will require to validate the reproduced trajectory and add the GMR model to the openMOS cloud system.

ACKNOWLEDGEMENT

The research leading to these results has received funding

System for Smart Plug-and-Produce Automation Components). Funding from the Engineering and Physical Science Research Council Centre for Doctoral Training in Embedded Intelligence (grant no. EP/L014998/1) is also acknowledged.

REFERENCES

- [1] B. Esmailian, S. Behdad, and B. Wang, "The evolution and future of manufacturing: A review," *J. Manuf. Syst.*, vol. 39, pp. 79–100, 2016.
- [2] "“open Dynamic Manufacturing Operating System for Smart “Plug-and-Produce” automation components’ (openMOS).” [Online]. Available: www.openmos.eu.
- [3] C. Li, C. Yang, Z. Ju, and A. S. K. Annamalai, "An enhanced teaching interface for a robot using DMP and GMR," *Int. J. Intell. Robot. Appl.*, vol. 2, no. 1, pp. 110–121, 2018.
- [4] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, "Robot programming by demonstration," in *Handbook of Robotics*, B. Siciliano and K. Oussam, Eds. New York: Springer-Verlag, 2008.
- [5] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Rob. Auton. Syst.*, vol. 57, pp. 1–37, 2009.
- [6] N. Vuković, M. Mitić, and Z. Miljković, "Trajectory learning and reproduction for differential drive mobile robots based on GMM/HMM and dynamic time warping using learning from demonstration framework," *Eng. Appl. Artif. Intell.*, vol. 45, pp. 1–12, 2015.
- [7] T. Arai, Y. Aiyama, Y. Maeda, M. Sugi, and J. Ota, "Agile Assembly System by “Plug and Produce”," *Ann. CIRP*, vol. 49, 2000.
- [8] K. Mustafa and K. Cheng, "Improving Production Changeovers and the Optimization: A Simulation Based Virtual Process Approach and Its Application Perspectives," *Procedia Manuf.*, 2017.
- [9] M. Field, D. Stirling, Z. Pan, and F. Naghdy, "Learning Trajectories for Robot Programming by Demonstration Using a Coordinated Mixture of Factor Analyzers," *IEEE Trans. Cybern.*, vol. 46, no. 3, 2016.
- [10] C. Li, H. Ma, C. Yang, and M. Fu, "Teleoperation of a virtual iCub robot under framework of parallel system via hand gesture recognition," in *2014 IEEE International Conference on Fuzzy System*.
- [11] H. Liu and L. Wang, "Gesture recognition for human-robot collaboration: A review," *Int. J. Ind. Ergon.*, vol. 68, pp. 355–367, 2018.
- [12] S. Hiroaki and C. Seibi, "Dynamic programming algorithm optimization for spoken word recognition," *Ieee Trans. Acoust. Speech, Signal Process.*, vol. ASSP-26, no. February, pp. 43–49, 1978.
- [13] L. Biagiotti and C. Melchiorri, "Online trajectory planning and filtering for robotic applications via B-spline smoothing filters," *IEEE Int. Conf. Intell. Robot. Syst.*, vol. 9, pp. 5668–5673, 2013.
- [14] S. Salvador and P. Chan, "Toward Accurate Dynamic Time Warping in Linear Time and Space."
- [15] S. Calinon, *Robot programming by demonstration: a probabilistic approach*, 1st ed. Lausanne, Switzerland: EPFL Press, 2009.
- [16] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active Learning with Statistical Models," *J. Artif. Intell. Res.*, vol. 4, 1996.