Segmentation and generalisation for writing skills transfer from humans to robots

Chunxu Li[1], Chenguang Yang[1,2] ✉, Cinzia Giannetti[1]

[1]*Zienkiewicz Centre for Computational Engineering, Swansea University, Swansea, SA1 8EN, UK*
[2]*Bristol Robotics Laboratory, University of the West of England, Bristol, BS16 1QY, UK*
✉ *E-mail: cyang@ieee.org*

Abstract: In this study, the authors present an enhanced generalised teaching by demonstration technique for a KUKA iiwa robot. Movements are recorded from a human operator, and then the recorded data are sent to be segmented via MATLAB by using the difference method (DV). The outputted trajectories data are used to model a non-linear system named dynamic movement primitive (DMP). For the purpose of learning from multiple demonstrations correctly and accurately, the Gaussian mixture model is employed for the evaluation of the DMP in order to modelling multiple trajectories by the teaching of demonstrator. Furthermore, a synthesised trajectory with smaller position errors in 3D space has been successfully generated by the usage of the Gaussian mixture regression algorithm. The proposed approach has been tested and demonstrated by performing a Chinese characters writing task with a KUKA iiwa robot.

1 Introduction

The realisation of natural and friendly human–robot interaction (HRI) is an important prerequisite for the service of robot technology into human daily life. HRI is a sub-area of human–computer interaction, which is the embodiment of interaction between human and robotics. It is based on the interaction between humans and computers, and is more intelligent and anthropomorphic. HRI is widely used in the search of dangerous scenes and the production/cleaning process of hazard goods, at this time, it is necessary to remotely operate the robot [1]. It can also be used to care for the elderly or disabled and to provide entertainment to humans [2]. There are various ways for human to interact with robots. Commonly, there are two main methods widely used to interact with robots, which are physical interaction and teleoperated interaction, respectively. The physical HRI (PHRI) field studies the design, control and planning issues that arise from the close physical interaction between humans and robotics in a shared workspace. Previous research in the field of PHRI has developed a safe and responsive control method to cope with the physical reactions that occur when a robot performs a task. Hogan *et al.* suggest that impedance control is one of the most commonly used methods of moving a robot along a given orbit when there is someone in the workspace [3]. Using this control method, the robot acts like a spring: it allows humans to be

propelled, but after the humans stop applying force, they return to their original position.

The field of research with respect to machine learning has also attracted great attention during the past decades. Machine learning has been used in many fields, such as autopilot filtering system of email, the recommendation system of e-shopping, handwriting recognition in the post office and automatic driving system of the vehicle [4]. The potential industrial value of machine learning has attracted a growing number of schools, companies as well as researchers devoting themselves into the field. Google announced its driverless car project in 2010, and released a video, in which its one of the employees Steve Mahan who lost 95% of the vision, safely drove 12 miles [5]. Ding *et al.* proposed optimisation extreme learning machine method of the SMO, however, owing to the large data set optimal parameters *C*, the algorithm requires more iterations to converge to the optimal solution of the optimisation problem [6]. In addition, teaching by demonstration (TbD) technique, the conception of which is demonstrated in Fig. 1, plays an important role in robot learning. TbD technique has been already applied widely not only to imitate the demonstration of human, but also to learn skills from human and optimise a specific task. The field of research with respect to understanding of human movement based on TbD has attracted great attention during the past decades [7].

The motion data can be modelled with some special characteristic methods, such as dynamic movement primitive [9] and hidden Markov model (HMM). The trajectory level characterisation based on the probability model is to model the movement as a stochastic model, such as Gaussian mixture model (GMM), which has a coding and noise processing capabilities, and this leads to dealing with high-dimensional problem more efficiently. Behavioural reproduction includes the movement trajectory reproduction and the movement control. Trajectory reproduction is the process by which the encoded data is passed through regression techniques, such as Gaussian process regression (GPR) or Gaussian mixed regression (GMR) [10]. Control reproduction is the generalised output mapped to the robot movement control, to achieve action reproduction, that is, the playback process after learning from a demonstrator.

Cognitive computing is a generalisation of the characteristics of a new generation of intelligent systems. This includes some systems that have some cognitive capabilities at the functional level and can perform certain cognitive tasks well. They also
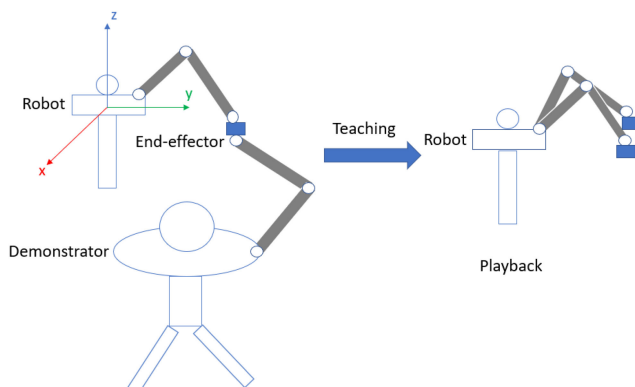

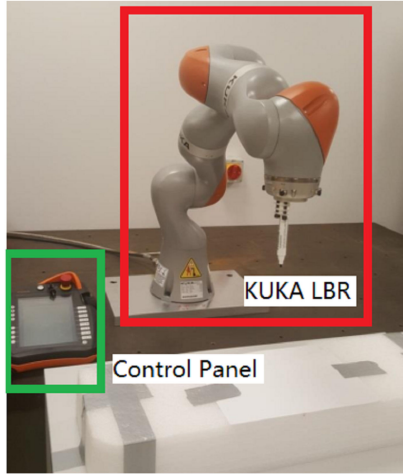**Fig. 1** *Image for the conception of TbD, modified from [8]*
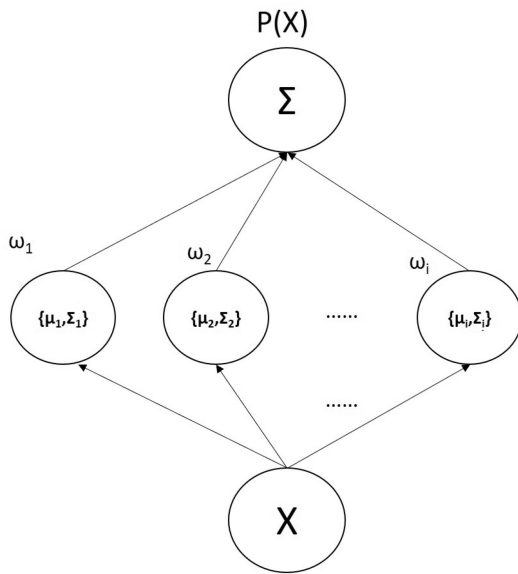
Fig. 2 *Image of KUKA LBR robot*

Fig. 3 *Calculation structure of the GMM, modified from [22]*

include the construction of the human brain at the structural level and redesign the non-von-Neumann system [11]. The architecture of the computer is more efficient in performing the operations required for cognition. One representative of the latter is the SyNAPSE project [12]. Cognitive systems need the same ability to receive, process and understand sound, images and language as humans. The ability of natural language processing represented by Watson and Siri needs to be further deepened, and future systems can be more intelligent in dialogue with humans [13]. Speech recognition and synthesis will be more accurate and more convenient to use. The understanding of images and videos will recognise objects, humans and even anomalies in the future.

The research focus of this paper is artificial intelligence algorithms which teach robots how to mimic human actions for writing, adapt and adjust themselves to replace human operators or cooperate with human operators in the performance of their tasks. In the traditional industrial production line, the production line was long and repetitive with no real chance of skill expansion or adjustments [14]. When encountering new research and development tasks, it is often necessary to re-record the point-to-point trajectory, and the machine equipment reads the cycle again [15]. This process usually takes a lot of time and money, which provides new challenges for research work. With this in mind, the researchers set out to achieve the following objectives: through programming based on DMP, GMM and GMR algorithm, the robot is able to self-adapt and generalise spatially/temporally, which greatly reduces the amount of time it takes robots to learn. This will, in turn, improve work efficiency; all the strokes are extracted

automatically by the upgraded codes employed with DM, thus the teaching process can be written continuously. In this paper, a KUKA iiwa robot has been used to test the method of programming by recording a series of movements which are taught by a human demonstrator, and then applying the DTW and GMR to process the recorded movements.

2 Preliminary

2.1 Difference method

DM is a numerical method for differential equations; it approximates the derivative by finite difference and seeks the approximate solution of the differential equation [16]. It is an approximate numerical solution of differential equations. Specifically, the difference method is to replace the differential with finite difference and to replace the derivative with a finite difference quotient, so that the basic equation and the boundary condition (generally the differential equation) are approximately changed to the difference equation (algebraic equation) [17]. The problem of solving differential equations is changed to solve the problem of algebraic equations.

2.2 KUKA iiwa robot

KUKA series robots have been used mainly in the industrial fields. Wherein, the KUKA iiwa robot (Fig. 2) is the first widely produced sensitive robots, at the same time, it is a robot with human–robot collaboration capabilities [18]. KUKA iiwa robots can set up the direct cooperation with human operators to complete the tasks with high accuracy requirements [19]. Moreover, the formation of a new work area can improve economic efficiency and achieve the highest efficiency.

2.3 Gaussian mixture model

GMM is a model composed of multiple Gaussian probability density functions. In fact, it is a multidimensional probability density function. The Gaussian model is completely determined by the two parameters of mean and variance [20]. Different learning mechanisms are adopted for the learning of mean and variance, which are able to directly result in influence for the model's accuracy, convergence and stability. Assuming an *M*-order GMM is weighted and summed with *M* Gaussian probability density functions to obtain [21]

$$P(X|\lambda) = \sum_{i=1}^{M} \omega_i b_i(X), \quad i = 1, 2, \ldots, M \qquad (1)$$

where X is a *D*-dimensional random vector, *M* is the order of the model and ω_i is the weight of each Gaussian component and satisfies $\sum_{i=1}^{M} \omega_i = 1$. $b_i(X)$ is each Gaussian component, and each Gaussian component is a Gaussian probability density function of *D* dimension, which can be expressed as follows [21]:

$$b_i(X) = \frac{1}{\sqrt{(2\pi)^{D/2}|\Sigma_i|^{1/2}}} \exp\left\{-0.5(x - \mu_i)^{\mathrm{T}} \sum_{i=1}^{M} (X - \mu_i)\right\} \qquad (2)$$

where μ_i is the mean vector and $\sum_i$ is the covariance matrix. Then a GMM can be represented by three parameters: mean vector, covariance matrix and mixed weight, hence a GMM can be expressed as

$$\lambda = \left\{\mu_i, \sum_i, \omega_i\right\}, \quad i = 1, 2, \ldots, M \qquad (3)$$

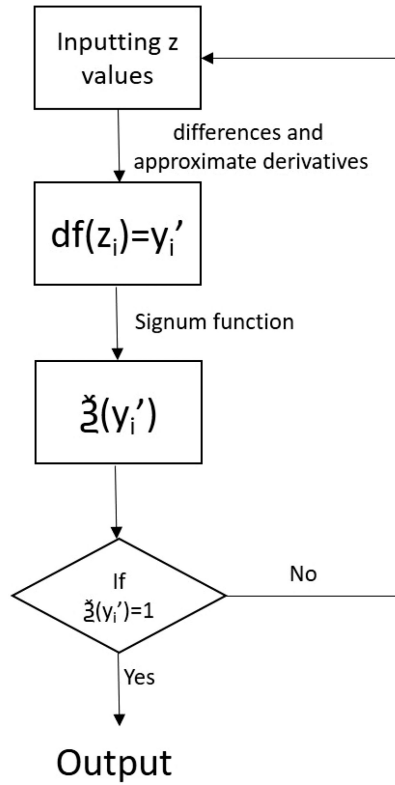In addition, the calculation structure of the GMM is shown in Fig. 3.

Cogn. Comput. Syst., 2019, Vol. 1 Iss. 1, pp. 20-25

21

Fig. 4 *Flowchart of the segmentation*

3 Methodology

In this section, we demonstrate the methods used to segment data, represent the conception of DMP and introduce the procedure of trajectory generation using multiple demonstrations.

3.1 Extraction of strokes of Chinese character

In this paper, we use the DM to segment the experimental data. Considering the variable y_i depends on the independent variable z_i. When z_i changes to $z_i + 1$, the amount of change of the dependent variable $y_i = f(z_i)$, $df(z_i) = f(z_i + 1) - f(z_i)$ is called the difference of the function $f(z_i)$ with a step length of 1 at point z_i, often referred to as the difference of the function $f(z_i)$, and called d as the difference operator. Difference has an arithmetic property similar to differential. The equation is shown as follows [16]:

$$f'(z_i) = df(z_i) = y'_i \simeq \frac{y_{i+1} - y_i}{z_{i+1} - z_i} \qquad (4)$$

where one significant factor of writing is that when every single stroke is completed, the pen would be picked up for once. Hence, the 'z' coordinate values from the experimental data are treated as the reference of the segmentation. $f(z_i)$ is the set of z_i values. After the DM, we have

$$\xi(y'_i) = \text{sign}(|y'_i| - \theta) \qquad (5)$$

where ξ is the gaping factor; θ is a constant, by giving different values of θ we can adapt the segmentation characters, such as the size of the segmented data set, here $\theta = 0.5$; sign is the Signum function, for each element of ξ, the formulation can be defined as follows:

$$\text{sign}(\xi) = \begin{cases} -1 & \text{if } \xi < 0, \\ 0 & \text{if } \xi = 0, \\ 1 & \text{if } \xi > 0. \end{cases} \qquad (6)$$

Up to now, we could output all the values of $\text{sign}(\xi)$, that are equal to 1, to different local text files for the usage of GMM and DMP

generalisation, which are corresponding those 'z' coordinate data increasing sharply, where it means the every single time's stroke writing has been completed. Fig. 4 demonstrates the flowchart of the segmentation.

3.2 Dynamic movement primitive

DMP is a dynamical system studied from biological studies that learns from motor primitives to produce an advanced prototype [23]. The concept of dynamic primitive can be divided into two categories, where one is to use different formulas based on dynamic system to represent state; the second one is to generate the track through the interpolation through the interpolating points [24]. DMP consists of two parts: the transformed system r and the canonical system h. The formula is shown as follows:

$$\dot{s} = h(s) \qquad (7)$$

$$\dot{t} = r(t, s, w) \qquad (8)$$

where t and s are the states of the transformed system and the canonical system, respectively, the transforming parameter for the output of the canonical system h is denoted as w.

There is an exponential differential equation representing the canonical system, which is given by

$$\tau \dot{s} = -\alpha_f s \qquad (9)$$

where s is the phase value varied between 0 and 1, $\tau > 0$, α_f are the temporal scaling factor and the stable factor, respectively.

The transformed system consists of two parts, which are non-linear term and a spring damping system in Cartesian space; the equations are described as follows [23]:

$$\tau \dot{v} = k(g - p) - cv + X(g - p_0) \qquad (10)$$

$$\tau \dot{p} = v \qquad (11)$$

where p_0 is the starting position, $p \in R$ is the position of Cartesian, $v \in R$ denotes the velocity of the end-effector of the robot, g represents the target, k is the spring factor and c denotes the damping coefficient. X is the transformation function presenting complex non-linear systems, which can transform the results of the canonical system, which is given by the following formula:

$$X = \sum_{i=1}^{N} w_i l(s) \qquad (12)$$

where the number of GMM is denoted as N, $w_i \in R$ is the weights, l represents the normalised radial's variable value, which can be given as follows:

$$l(s) = \frac{\exp(-h_i(s - c_i)^2)}{\sum_{m=1}^{N} \exp(-h_m(s - c_m)^2)} \qquad (13)$$

where $c_i > 0$ are the centres and $h_i > 0$ are the widths of the Gaussian basis functions. N is the number of the Gaussian functions.

Furthermore, we can use the weight parameter to generate movements by choosing the canonical system's ($s = 0$) starting point x_0 and target g, which are the integration of the canonical system. Calculating the non-linear transformation function X by learning the motions of the presenter is the principle of DMP. However, there are limitations to creating a transformation system with multiple demonstrations; hence the GMM is applied to overcome the above problems.

3.3 Trajectory generation

The parameter estimation of GMM is the process of acquiring the model parameters under certain criteria. It is actually the process of learning the model parameters, namely the process of solving
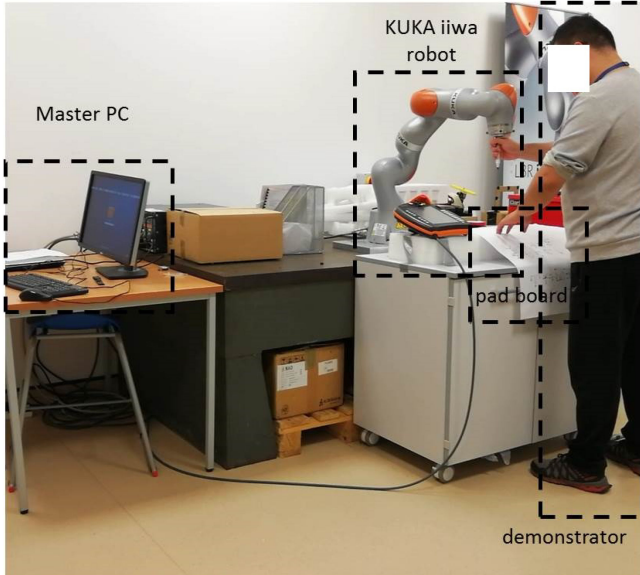
Fig. 5 *Illustration of experimental setup*

$\lambda = \{\mu_i, \sum_i, \omega_i\}$ [25] which is to give the observation sequence to the GMM. The most used parameter estimation is the method of maximum likelihood estimation. Its basic idea is to find the model parameter λ when the maximum likelihood of GMM is obtained by giving the observation sequence X, which has been obtained previously by DMP, hence that λ is the model [10]. The optimal parameter, λ, describes the distribution of the observed sequence to the greatest extent possible.

After giving the training data, the ultimate goal of the maximum likelihood estimation is to find a model parameter that maximises the likelihood of the GMM. For a training vector sequence $X = \{x_1, x_2, \ldots, x_D\}$ of length D, the likelihood of GMM can be expressed as

$$P(X|\lambda) = \prod_{t=1}^{D} P(X_t|\lambda) \tag{14}$$

Then, the parameter λ keeps continuously updated until a set of parameters λ is found to maximise $P(X|\lambda)$, which is

$$\hat{\lambda} = \arg \max_{\lambda} P(X|\lambda) \tag{15}$$

For the convenience of analysis, $P(X|\lambda)$ usually takes its log likelihood, then we have

$$\log(P(X|\lambda)) = \sum_{t=1}^{D} \log P(X_t|\lambda) \tag{16}$$

Considering that there is a relatively complex non-linear relationship between the likelihood function and the model parameters, and the maximum value cannot be calculated according to the simple likelihood estimation method, the expectation-maximisation algorithm can be used for parameter estimation. The EM algorithm is actually an iterative algorithm for the maximum likelihood estimation of the probability model. The process of each iteration is to estimate the unknown data distribution based on the parameters that have been acquired, and then calculate the new model parameters under the maximum likelihood condition. Let the initial model parameter be λ, which satisfies

$$P(X|\lambda') \geq P(X|\lambda) \tag{17}$$

First we estimate the new model parameter λ' according to the above formula, and then use the parameter λ' as the initial parameter of the next iteration. This iteratively iterates until the

convergence condition is met. Here we assume a Q function representing the E step of EM method, which is shown below:

$$Q(\lambda, \lambda') = \sum_{i=1}^{M} P(X, i|\lambda) \log P(X, i|\lambda) \tag{18}$$

where i is a hidden state and is unknown and random. $Q(\lambda, \lambda')$ refers to the expectation of the log-likelihood of all observed data. Calculating the maximum value of $Q(\lambda, \lambda')$ can obtain the maximum log-likelihood of the observed data, which represents the M step of EM method. Substituting (14) and (15) into (18), we can obtain

$$Q(\lambda, \lambda') = \sum_{i=1}^{M} \sum_{t=1}^{T} r_t(i) \log \omega_i b_i(x) \tag{19}$$

$$r_t(i) = P(X_t, i|\lambda) \tag{20}$$

Then, according to E and M, the estimated values of each parameter are obtained. Step E is calculating the posterior probability of the t_{th} sample X_t of the training data in the i_{th} state according to the Bayesian formula; M step is first using the Q function to derive the three parameters separately, and then calculating their corresponding estimates. E and M steps are iteratively repeated to re-evaluate the parameters. When the maximum value of the likelihood function is found, the iteration is stopped.

When using the EM algorithm to estimate the parameters of the GMM, the first step is to determine the number of Gaussian components in the GMM, such as the order of the model M and the initial parameter λ of the model [21]. The determination of the order M of the model needs to be selected according to the actual situation, such as the amount of training data. For the initial parameter λ of the model, the most commonly used method is the K-means algorithm. The K-means algorithm is currently the most simple and effective classification algorithm, which is widely used in various models [26]. The GMM employed in this paper uses the K-means algorithm to select the initial parameters. The K-means algorithm divides the data into K clusters according to the principle of the smallest the within-cluster sum of squares in the class. After K-means algorithm is used to cluster the feature vectors, the mean and variance of each class are calculated, and the percentage of the feature vectors of each class is calculated as the blending weight [27]. The mean, variance and mixed weight are obtained as the initial values of the parameter estimation.

4 Experimental study

A KUKA iiwa robot is used in our experiment to verify the effectiveness of the proposed method, which is the first widely produced sensitive robots. The KUKA iiwa robot is controlled utilising the KUKA SmartPad shown in Fig. 5. It illustrates the positions of the human demonstrator, the master computer for programming offline, the pad board for writing and the KUKA robotic system. The demonstrator stands just in front of the KUKA robot to physically guide for teaching to write. A desktop computer running the codes to program the experimental data offline is placed opposite to the robot. There is a pad board between the robot and the demonstrator.

4.1 Experiment procedure

The experimental procedure is as follows: the demonstrator teaches the robot to write the first half of a Chinese poem by holding the marker pen attached in the end-effector of KUKA robot. Meanwhile, the trajectories data are outputted locally for the usage of programming. The trajectories data are then sent to the remote computer to be segmented using MATLAB program. The principle is that the demonstrator needs to pick up the marker pen after the writing of the every single stroke. The data outputted by the robot are a set of the coordinates of the end-effector, including the values of X, Y, Z, A, B, C. Wherein the A, B, C values are the Euler angles

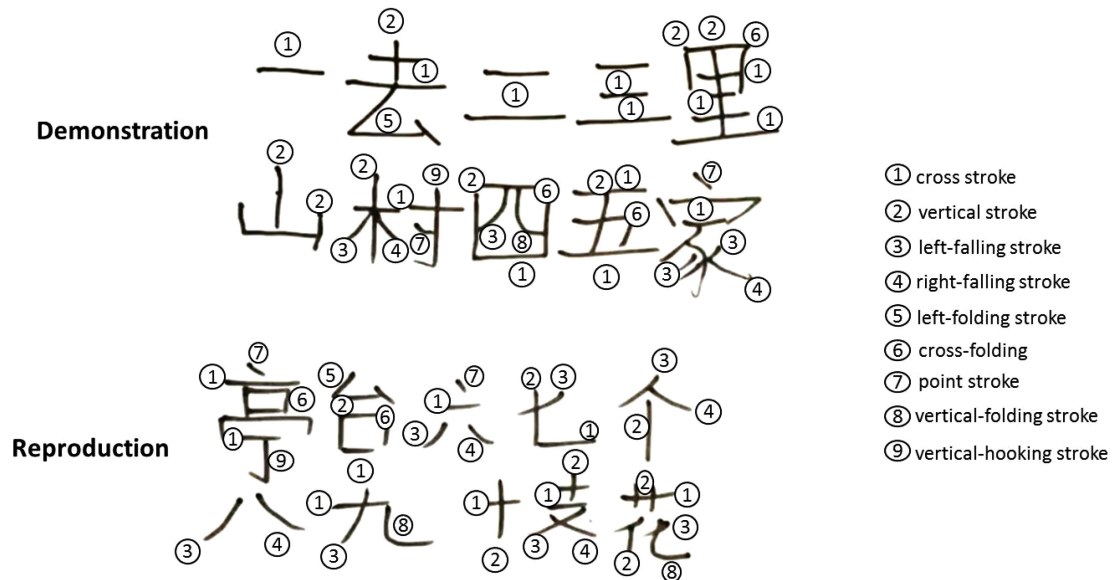*Cogn. Comput. Syst.*, 2019, Vol. 1 Iss. 1, pp. 20-25

23

Fig. 6 *Demonstrated Chinese characters and the DMP generalized Chinese poem characters with their strokes contents*

of the end-effector. Hence, we can find the various range of *Z* value, if it increases suddenly, it we expect a stroke has been complete.

After the segmentation of experimental data, the split groups of data are sent for optimisation based on GMR. By applying GMR, all the repetitive strokes can be optimised into a single stroke. Although there are only ten Chinese characters in the first half of poem, almost all the strokes are exported without any repetition after the application of GMR. Then, those data are encoded with the DMP model to generalise. By doing this, all the strokes can be adapted accordingly, which are relied to re-group the new Chinese characters.

4.2 Experiment results

As shown in Fig. 6, 20 separate Chinese characters were written, wherein there are a half characters written during teaching and half during the DMP generalisation based playback process. There are totally nine strokes cross in the first half of the Chinese poem written by the KUKA robot under the teaching of demonstrator, which are vertical stroke, left-falling stroke, right-falling stroke, cross-folding stroke, left-folding stroke, point stroke, vertical-folding stroke and vertical-hooking stroke, respectively. It can be observed from Fig. 6 that the strokes of generated Chinese characters from the DMP-based generalisation are same as those that taught by the demonstrator and they are corresponding one by one with each other. To be more precise, all the strokes segmented from the recorded trajectories of the teaching process are able to re-group those characters written during the playback process of KUKA robot after the usage of GMR and DMP generalisation. For example, the upper part of the second character 'tai' in the second half of the poem is made by the lower part of the second character 'qu' in the first line of the poem; the lower part of the second character 'tai' in the second half of the poem is made by the outer part of the third character 'si' in the second line of the poem, where the size and the position of the character parts are adjusted with the DMP model.

GMR is used to optimise all the repetitive strokes into a single one, such as the horizontal stroke, which is the most repetitive stroke. As a consequence, after some experiments, the GMM number is able to result in the influence for the forming of optimal trajectories. Furthermore, we choose the GMM number as 20 in this work, which is obtained after the huge amount of training, to achieve great performances for our experimental section. Through the employment of the GMR, the restoration of data during the TbD has been converted into a process, where we use GMR algorithm to estimate the joint distribution, which can be approximated by a mixture of Gaussian functions. During the calculation, the certain correlation of the points in sample set of linear data is significant for the learning procedure of robots. As mentioned above, the GMM number decides the prediction process. The generated trajectories (the last two lines) are then drawn by the KUKA LBR robot arm over the recorded patterns to provide a visual result (shown in Fig. 6), which can be easily compared with the pre-written Chinese characters (the first two lines).

5 Conclusion

A GMM and DMP enhanced TbD technology has been developed in this paper, which is an effective and advanced approach for human–robot skill transfer. By applying the two above methods, the complication of the robot learning is reduced. Data of the trajectories are segmented via MATLAB automatically into single set by applying DM, where each group represent a single stroke. After that, DMP has been employed to establish the modelling for the data obtained from learning process. Then we combine all above methods with GMM to encode multiple trajectories for the repetitive strokes. By performing procedure above, the data can be managed, and GMR is an effective method to decrease errors for movement paths of a KUKA iiwa robot. This study shows that by reducing errors on complex movements learned from demonstrator, the synthesisely generated movement paths calculated from the recorded movements obtained a great result. After the application of DMP based segmentation and generalisation, the reproduction of regrouping work using separate strokes for the Chinese characters can be achieved. Future work will focus on improving the accuracy of the resultant generated path by making the recording independent of time.

6 Acknowledgments

7 References

[1] Arkin, R.C., Fujita, M., Takagi, T.*, et al.*: 'An ethological and emotional basis for human–robot interaction', *Robot. Auton. Syst.*, 2003, **42**, (3–4), pp. 191–201

[2] Bartneck, C., Forlizzi, J.: 'Shaping human-robot interaction: understanding the social aspects of intelligent robotic products'. CHI'04 Extended Abstracts on Human Factors in Computing Systems, Vienna, Austria, 2004, pp. 1731–1732

[3] Hogan, N.: 'Controlling impedance at the man/machine interface'. Proc. 1989 IEEE Int. Conf. on Robotics and Automation, Scottsdale, Arizona, USA, 1989, pp. 1626–1631

[4] Libbrecht, M.W., Noble, W.S.: 'Machine learning applications in genetics and genomics', *Nat. Rev. Genetics*, 2015, **16**, (6), p. 321

[5] Poczter, S.L., Jankovic, L.M.: 'The google car: driving toward a better future?', *J. Bus. Case Stud. (Online)*, 2014, **10**, (1), p. 7

[6] Cao, L., Sun, F., Li, H., *et al.*: 'Advancing the incremental fusion of robotic sensory features using online multi-kernel extreme learning machine', *Front. Comput. Sci.*, 2017, **11**, (2), pp. 276–289

[7] Li, C., Yang, C., Wan, J., *et al.*: 'Teleoperation control of baxter robot using kalman filter-based sensor fusion', *Syst. Sci. Control Eng.*, 2017, **5**, (1), pp. 156–167

[8] Oksanen, T., Kostamo, J., Tamminen, P., *et al.*: 'Robot competition as a teaching and learning platform'. Proc. the 18th IFAC World Congress, Milano, August 2011, pp. 5176–5181

[9] Schaal, S., Peters, J., Nakanishi, J., *et al.*: 'Learning movement primitives'. Robotics Research, The Eleventh Int. Symp., 2005, pp. 561–572

[10] Li, C., Yang, C., Ju, Z., *et al.*: 'An enhanced teaching interface for a robot using DMP and GMR', *Int. J. Intell. Robot. Appl.*, 2018, **2**, (1), pp. 110–121

[11] Modha, D.S., Ananthanarayanan, R., Esser, S.K., *et al.*: 'Cognitive computing', *Commun. ACM*, 2011, **54**, (8), pp. 62–71

[12] Ananthanarayanan, R., Esser, S.K., Simon, H.D., *et al.*: 'The cat is out of the bag: cortical simulations with 10 9 neurons, 10 13 synapses'. Proc. of the Conf. on High Performance Computing Networking, Storage and Analysis, Portland, OR, USA, 2009, p. 63

[13] Ahmed, M.N., Toor, A.S., O'Neil, K., *et al.*: 'Cognitive computing and the future of health care cognitive computing and the future of healthcare: the cognitive power of IBM watson has the potential to transform global personalized medicine', *IEEE. Pulse.*, 2017, **8**, (3), pp. 4–9

[14] Green, S.A., Billinghurst, M., Chen, X.Q., *et al.*: 'Human-robot collaboration: A literature review and augmented reality approach in design', *Int. J. Adv. Robot. Syst.*, 2008, **5**, (1), p. 1

[15] Brogårdh, T.: 'Present and future robot control development an industrial perspective', *Annu. Rev. Control*, 2007, **31**, (1), pp. 69–79

[16] Curtis, A.R., Reid, J.K.: 'The choice of step lengths when using differences to approximate jacobian matrices', *IMA J. Appl. Math.*, 1974, **13**, (1), pp. 121–126

[17] Kwong, M.K., Zettl, A.: '*Norm inequalities for derivatives and differences*' (Springer, Springer-Verlag Berlin Heidelberg New York, US, 2006)

[18] Bischoff, R., Kurth, J., Schreiber, G., *et al.*: 'The kuka-dlr lightweight robot arm-a new reference platform for robotics research and manufacturing'. Robotics (ISR), 2010 41st Int. Symp. on and 2010 6th German Conf. on robotics (ROBOTIK), Munich, Germany, 2010, pp. 1–8

[19] Son, H., Kim, C., Kim, H., *et al.*: 'Trend analysis of research and development on automation and robotics technology in the construction industry', *Korean Soc. Civ. Eng. KSCE J. Civ. Eng.*, 2010, **14**, (2), pp. 131–139

[20] Li, M.-Z., Ma, Z.-Q., Shan, Y., *et al.*: 'Adaptive background update based on Gaussian mixture model under complex condition', *Jisuanji Yingyong/J. Comput. Appl.*, 2011, **31**, (7), pp. 1831–1834

[21] Xuan, G., Zhang, W., Chai, P.: 'Em algorithms of Gaussian mixture model and hidden markov model'. Proc. 2001 Int. Conf. on Image Processing (Cat. No. 01CH37205), Thessaloniki, Greece, 2001, vol. 1, pp. 145–148

[22] Boyd, S., Parikh, N., Chu, E., *et al.*: '*Distributed optimization and statistical learning via the alternating direction method of multipliers*' (Foundations and Trends® in Machine Learning, Hanover, MA, USA, 2011), vol. 3, no. 1, pp. 1–122

[23] Schaal, S.: 'Dynamic movement primitives-a framework for motor control in humans and humanoid robotics', in Kimura, H., Tsuchiya, K., Ishiguro, A., *et al.* (Eds.): '*Adaptive motion of animals and machines*' (Springer, Springer, Tokyo, Japan, 2006), pp. 261–280

[24] Ijspeert, A.J., Nakanishi, J., Hoffmann, H., *et al.*: 'Dynamical movement primitives: learning attractor models for motor behaviors', *Neural Comput.*, 2013, **25**, (2), pp. 328–373

[25] Huang, Z.-K., Chau, K.-W.: 'A new image thresholding method based on Gaussian mixture model', *Appl. Math. Comput.*, 2008, **205**, (2), pp. 899–907

[26] Abdul Nazeer, K.A., Sebastian, M.P.: 'Improving the accuracy and efficiency of the k-means clustering algorithm'. Proc. of the World Congress on Engineering, London, U.K, 2009, vol. 1, pp. 1–3

[27] Tatiraju, S., Mehta, A.: 'Image segmentation using k-means clustering, EM and normalized cuts'. University of California Irvine, 2008

Cogn. Comput. Syst., 2019, Vol. 1 Iss. 1, pp. 20-25

25