Int J Adv Manuf Technol (2014) 73:623–633
DOI 10.1007/s00170-014-5803-x

ORIGINAL ARTICLE

Contact-state monitoring of force-guided robotic assembly tasks using expectation maximization-based Gaussian mixtures models

Ibrahim F. Jasim · Peter W. Plapper

Received: 20 December 2013 / Accepted: 24 March 2014 / Published online: 27 April 2014
© Springer-Verlag London 2014

Abstract This article addresses the problem of contact-state (CS) monitoring for peg-in-hole force-controlled robotic assembly tasks. In order to perform such a monitoring target, the wrench (Cartesian forces and torques) and pose (Cartesian position and orientation) signals of the manipulated object are firstly captured for different CS's of the object (peg) with respect to the environment including the hole. The captured signals are employed in building a model (a recognizer) for each CS, and in the framework of pattern classification, the CS monitoring would be addressed. It will be shown that the captured signals are nonstationary, i.e., they have non-normal distribution that would result in performance degradation if using the available monitoring approaches. In this article, the concept of the Gaussian mixtures models (GMM) is used in building the likelihood of each signal and the expectation maximization (EM) algorithm is employed in finding the GMM parameters. The use of the GMM would accommodate the signals nonstationary behavior and the EM algorithm would guarantee the estimation of the optimal parameters set of the GMM for each signal, and hence the modeling accuracy would be significantly enhanced. In order to see the performance of the suggested CS monitoring scheme, we installed a test stand that is composed of a KUKA lightweight robot (LWR) doing peg-in-hole tasks. Two experiments are considered; in the first experiment, we use the EM-GMM in monitoring a typical peg-in-hole robotic assembly process, and in the second experiment, we consider the robotic assembly of camshaft caps assembly of an automotive powertrain and use the EM-GMM in monitoring its CS's. For both experiments, the excellent monitoring performance will be shown. Furthermore, we compare the performance of the EM-GMM with that obtained when using available CS monitoring approaches. Classification success rate (CSR) and computational time will be considered as comparison indices, and the EM-GMM will be shown to have a superior CSR performance with reduced a computational time.

Keywords Assembly monitoring · Expectation maximization · Gaussian mixtures · Peg-in-hole · Robotic assembly

1 Introduction

Assembly is considered one of the vital topics for both industry and research institutions, and automating the assembly for different products drew the attention of many practitioners from both academia and production sectors. Robots are considered the most important tools in automating productions and hence robotic assembly appeared to be one of the hottest research topics. An important aspect of the robotic assembly tasks is the monitoring of the process itself, that is adding the necessary skills to the robot that makes it aware of its surrounding environment using the available signals like wrench (Cartesian forces and torques) and pose (Cartesian position and orientation) of the manipulated object.

Monitoring of the assembly tasks was addressed in the framework of contact state (CS) modeling (recognition), and CS modeling of force-controlled robotic tasks was solved by different approaches. Vision-based systems can be used in building the suitable CS models for a robotic peg-in-hole assembly process. However, vision-based systems

I. F. Jasim (✉) · P. W. Plapper
Faculty of Science, Technology and Communication,
University of Luxembourg, Campus Kirchberg, Luxembourg
e-mail: ibrahim.jasim@uni.lu

would fail for occluded parts and time-varying illuminations environment that urged the researchers in this field to consider developing the CS models using the wrench and pose signals that are measured by suitable sensors. Hirai and Iwata proposed a CS recognition scheme for such robotic systems using the geometric model of the mated parts along with the sensed forces [1]. Petri net was successfully employed in modeling and planning robotic assembly tasks and promising results were obtained [2, 3]. Uncertainties were accommodated for such modeling tasks through developing a CS recognition system that relies on incorporating the sensed forces, the sensed error signals, and contact compliance [4]. The concept of discrete event systems was efficiently used in producing efficient models for robotic assembly tasks [5, 16]. In [6–8], neural networks were used in building CS classifiers for recognizing different CS's of compliant motion robots. Relying on the assembly sequence, local depart space (LDS) was successfully used in building classifiers for different CS's with accommodating possible uncertainties [9]. In [7, 8, 11, 13–15], fuzzy classifiers were used in recognizing different CS's for different objects without needing the geometrical features of the those mated parts.

Modeling of robotic peg-in-hole assembly process was successfully performed in the framework of finding analytical solutions of the contact forces for different situations between the manipulated object and the environment [17]. Cartesão et al. used a neural network and Kalman filter in building a signal diffusion system that captures human skills in robotic assembly tasks [19]. Hidden Markov models was successfully used in developing models for compliant motion robots and hence opening the door to the probabilistic modeling approaches [10, 18, 20]. In [12, 25], the authors were successful in linking the CS modeling to the geometrical parameters estimation and efficient models were obtained for each CS. Iwata et al. were successful in adding neural network reinforced tactile-based recognition skills to robots interacting with different environments [21]. In [23], force/torque mapping for each model was developed using CAD data and particle filters and enhanced CS modeling was obtained. Disturbance observer-based approach was suggested to monitor the contact of the robots without the need for using force sensors [24]. ARX modeling approach was successfully employed in adding the recognition skills to the peg-in-hole robotic assembly tasks and promising results were obtained [26].

Cabras et al. were capable of using the stochastic gradient boosting (SGB) classifier in recognizing different CS's without the need for knowing the task sequence or task graph. In [30], the authors used only the force and torque vectors in recognizing different CS's for a compliant motion robotic system. The approach computes the wrench space automatically based on the CS's graph, which describes the sequence of different CS's in a certain task. Then, a similarity index is augmented which shows the amount of overlap between wrenches that belong to different CS's. Finally, a particle filter is used to compute the likeness that a certain wrench vector belongs to a CS. The results shown in [30] are excellent for the computation time wise; however, the sequence of the CS's is still needed to be known. In [31], the authors were successful in using fuzzy clustering technique in building efficient fuzzy models. The fuzzy clusters are tuned by gravitational search algorithm (GSA) and excellent mapping capability was obtained for each model. A common feature to all of the approaches above is the lack of considering the signals nonstationary behavior, i.e., the non-normal signals distribution, which is frequently the case to many robotic assembly as will be seen throughout this article. Such signals nonstationary behavior would cause recognition performance degradation if not well-considered in developing the models.

In order to accommodate such nonstationary behavior of signals, one can consider using multiple Gaussian components instead of one and use the Gaussian mixtures models (GMM) in building the likelihood for each signal [22]. The well-known expectation maximization (EM) algorithm can be used in finding the parameters of the GMM components that maximizes the log-likelihood and hence an optimal modeling for those nonstationary signals could be resulted. Originating from such a crucial motivation, this article uses the expectation maximization-based Gaussian mixtures models (EM-GMM) in modeling the force-controlled robotic peg-in-hole assembly tasks and hence an enhanced assembly monitoring would be resulted. The captured wrench and pose signals, for the peg-in-hole assembly process, are segmented into five phases. EM-GMM is used in building models that efficiently maps the CS's to their corresponding signals. Experimental results are carried out on a KUKA lightweight robot (LWR) doing peg-in-hole assembly tasks. Two experiments are performed; in the first one, a typical peg-in-hole assembly process is studied, and in the second one, the camshaft caps assembly of an automotive powertrain, which is a multiple peg-in-hole assembly process, is considered. We considered the camshaft caps assembly so that we can show the efficiency of the suggested CS recognition scheme for distinct peg-in-hole assembly tasks with different number of pegs and geometry. Furthermore, the industrial relevance of the camshaft caps assembly is another good reason that motivated us in considering such interesting task. For both experiments, excellent CS monitoring performance will be shown. Furthermore, for comparison purpose, we develop the corresponding CS models for both experiments using the available CS recognition schemes, like the conventional fuzzy classifier (CFC) [14], SGB classifier [28], and gravitational search-fuzzy clustering algorithm (GS-FCA) classifier [31] and the superiority of the EM-GMM CS monitoring scheme, with a reduced computational time, is shown.

The rest of the paper is organized as follows: Section 2 explains the robotic peg-in-hole assembly tasks and Section 3 details the EM-GMM modeling process. Experimental validation is presented in Section 4, and Section 5 gives the concluding remarks and recommendation for future works.

2 Peg-in-hole robotic assembly

Consider the robotic peg-in-hole assembly task shown in Fig. 1. Such a task is composed of a robot inserting an object into a certain hole and such a task is considered the backbone to many assembly tasks. In order to model the peg-in-hole task, the overall motion is segmented into different phases according to the status of the manipulated object with respect to the environment. For each segment, different signals are collected and models are developed. Consider the peg-in-hole process shown in Fig. 1. The wrench signals, of the manipulated object, are described as follows:

$$w = [f_x, f_y, f_z, \tau_x, \tau_y, \tau_z] \tag{1}$$

Where f_x, f_y, and f_z are the Cartesian forces and τ_x, τ_y, and τ_z are the torques around the Cartesian axes both measured for the manipulated object. Likewise, to the pose of the manipulated object, it can be written as follows:

$$p = [x, y, z, \Psi_x, \Psi_y, \Psi_z] \tag{2}$$

Where x, y, and z are the Cartesian position and Ψ_x, Ψ_y, and Ψ_z are the orientation around the Cartesian axes of the manipulated object. Hence, one would have 12 input signals for the classifier, say $x_k = [x_{1,k}, x_{2,k}, ..., x_{12,k}]$ with k as the sample index. The CS classification problem can be formulated as follows:

$$y_k = \begin{cases} 1 & \text{if } (x_k \in \text{current CS}) \\ 0 & \text{Otherwise} \end{cases} \tag{3}$$

y_k is the output of the CS classifier. It can be seen that (3) represents a nonlinear mapping between x_k and y_k and the goal of almost all modeling and classification researches is to approximate or realize this mapping as accurate as possible. The next section explains the methodology that will be used throughout this paper in realizing (3).

3 Expectation maximization-based Gaussian mixtures models

Before explaining the EM-based GMM process, the principles of the Bayesian modeling (or classification) will be clarified.

3.1 Bayesian classification

Suppose that one is given a vector set $\boldsymbol{x}_k = [x_{k,1}, x_{k,2}, ..., x_{k,D}]^T$ where D is the width of the vector (in the CS recognition addressed in this paper, it is clear that $D = 12$ as each model has 12 inputs). Suppose that we have the classes $y_k = \{c_1, c_2, ..., c_C\}$ that one of them the vector $\boldsymbol{x}_k$ corresponds to. Then the vector $\boldsymbol{x}_k$ belongs to a class c_i, implying that [22]:

$$p(c_i|\boldsymbol{x}_k) \geq p(c_j|\boldsymbol{x}_k) \tag{4}$$

for $i \neq j$. $p(c_k|\boldsymbol{x}_k)$ is called as the posterior probability of class c_k given the vector $\boldsymbol{x}_k$ and can be computed using:

$$p(c_i|\boldsymbol{x}_k) = \frac{p(\boldsymbol{x}_k|c_i)p(c_i)}{p(\boldsymbol{x}_k)} \tag{5}$$

where $p(\boldsymbol{x}_k|c_i)$ is the probability density function (pdf) of class c_i in the vector space of $\boldsymbol{x}_k$, $p(c_i)$ is the a priori probability that represents the probability of class c_i, and $p(\boldsymbol{x}_k)$ is the probability of the vector space $\boldsymbol{x}_k$ that can be computed as follows:

$$p(\boldsymbol{x}_k) = \sum_{i=1}^{C} p(\boldsymbol{x}_k|c_i)p(c_i) \tag{6}$$

From Eq. 6, one can notice that for equal class a priori $p(c_i)$, the term $p(\boldsymbol{x}_k)$ of Eq. 5 would be merely a scaling factor. Therefore, it can be deduced that the vector $\boldsymbol{x}_k$ belongs to a class c_i, implying that:

$$p(\boldsymbol{x}_k|c_i)p(c_i) \geq p(\boldsymbol{x}_k|c_j)p(c_j) \tag{7}$$

for $i \neq j$. According to Eq. 7, the best approximation of the term $p(\boldsymbol{x}_k|c_j)$ results in the best classification for the pattern $\boldsymbol{x}_k$. In the conventional Bayesian classifier, a Gaussian

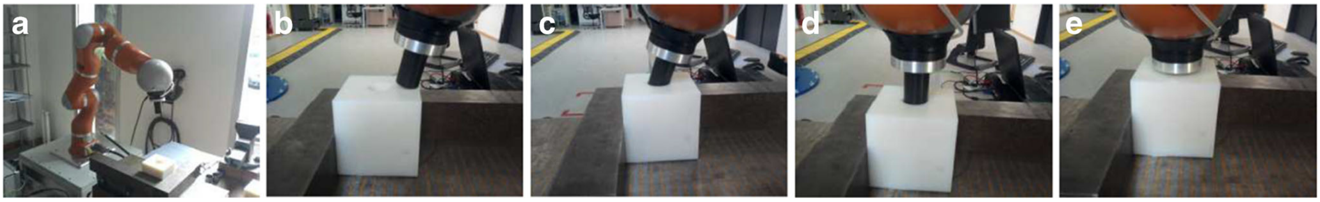

Fig. 1 Experiment 1: robotic peg-in-hole assembly phases: **a** phase 1 (free space), **b** phase 2, **c** phase 3, **d** phase 4, and **e** phase 5

distribution is used in approximating the term $p(\boldsymbol{x}_k|c_j)$, that is:

$$p(\boldsymbol{x}_k|c_i) = \frac{1}{|2\pi|^{D/2}|\Sigma|^{\frac{1}{2}}} exp\left(-\frac{1}{2}(\boldsymbol{x}_k - \propto)^T \Sigma^{-1}(\boldsymbol{x}_k - \propto)\right) \tag{8}$$

where $\propto \in R^D$ is the mean, $\Sigma \in R^{D\times D}$ is the covariance matrix, and $|\Sigma|$ is the determinant of Σ. It was shown that the approximation (8) performs well in the case of normal distribution. However, in many cases, one may face situations in which the vector space signals, or the several signals of the vector space, have non-normal distribution and consequently the use of Eq. 5 would result in increased modeling errors.

3.2 Gaussian mixtures models

In order to accommodate the possible non-normal distribution of the signals, Gaussian mixtures is employed in modeling the features (input signals), i.e., assigning more than a Gaussian component for each feature. Suppose that a single Gaussian distribution is represented as follows:

$$N(\boldsymbol{x}_k, \propto, \Sigma) = \frac{1}{|2\pi|^{D/2}|\Sigma|^{\frac{1}{2}}} exp\left(-\frac{1}{2}(\boldsymbol{x}_k - \propto)^T \Sigma^{-1}(\boldsymbol{x}_k - \propto)\right) \tag{9}$$

Then a Gaussian mixtures model (GMM) can be described as follows:

$$p(\boldsymbol{x}_k|c_i) = \sum_{q=1}^{M} \omega_q N_q(\boldsymbol{x}_k, \propto_q, \Sigma_q) \tag{10}$$

M is the total number of the Gaussian mixtures, ω_q, $\propto_q$, and Σ_q are the weight, mean, and covariance of the q^{th} Gaussian component. Suppose that $\theta_q = (\omega_q, \propto_q, \Sigma_q)$ and consider the parameter vector $\theta = [\theta_1, \theta_2, ..., \theta_M]^T$. It is clear that finding the values of the parameters is very important in having a precise modeling of the given features. Therefore, one can write the model (10) in terms of the parameters θ as follows:

$$p(\boldsymbol{x}_k|c_i;\theta) = \sum_{q=1}^{M} \omega_q N_q(\boldsymbol{x}_k, \propto_q, \Sigma_q) \tag{11}$$

Finding the parameter vector θ that optimizes the models from the available measurements would optimize the performance of the classification process.

3.3 Expectation maximization

One of the most efficient approaches in finding those parameters is the expectation maximization (EM) algorithm. The EM algorithm is considered one of the simplest and computationally effective iterative scheme in finding the GMM parameters. It is composed of two steps; the E-step in which the log-likelihood is estimated for the current parameters, and the M-step in which the parameter θ is updated such that a maximized log-likelihood would be resulted. In order to explain the EM algorithm, let's consider the overall data $X = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N]^T$. Then the likelihood function for the data X given the parameters θ can be defined as follows [27]:

$$\ell(X;\theta) = \prod_{n=1}^{N} p(x_n;\theta) \tag{12}$$

Define the logarithm of $\ell(X;\theta)$ to be $L(X;\theta)$ which is called log-likelihood. Taking the logarithm for both sides of Eq. 12, then the log-likelihood can be expressed as follows:

$$L(X;\theta) = \sum_{n=1}^{N} \ln(p(x_n;\theta)) \tag{13}$$

The parameter θ that maximizes (13) can be described as follows:

$$\theta(t) = \arg\max{}_{\theta} L(X;\theta(t)) \tag{14}$$

subject to:

$$\sum_{q=1}^{M} \omega_q = 1$$

Equation 13 is a constrained optimization problem and the analytical solutions can be intractable. Therefore, iterative solutions like the EM algorithm were suggested to solve such a problem. An important quantity that plays a vital role in the EM algorithm is the conditional probability of y given $\mathbf{x}$ and let's denote $p(c_i = 1|x_k)$ as $\gamma(c_{ik})$. The value of $\gamma(c_{ik})$ can be computed using Bayes rule as follows:

$$\gamma(c_{ik}) = \frac{p(c_i = 1)p(x_k|c_i = 1)}{\sum_{j=1}^{M} p(c_j = 1)p(x_k|z_j = 1)} \tag{15}$$

which leads to:

$$\gamma(c_{ik}) = \frac{w_i N_i(x_k, \propto_i, \Sigma_i)}{\sum_{j=1}^{M} w_j N_j(x_k, \propto_j, \Sigma_j)} \tag{16}$$

$\gamma(c_{ik})$ is called the responsibility that the ith component takes for explaining x_k [22]. The following steps summarizes the EM algorithm:

Step 1: Initialize the parameter vector $\theta_i = (\omega_i, \propto_i, \Sigma_i)$ randomly. Initialize the convergence parameters ε and ϵ.

Step 2: (E-Step) For the current parameter vector θ_i compute the responsibilities using (16).

Step 3: (M-Step) Re-estimate the parameters using the current responsibilities:

$$\propto_i^{new} = \frac{1}{N_i}\sum_{n=1}^{N}\gamma(c_{in})x_n \tag{17}$$

$$\Sigma_i^{new} = \frac{1}{N_i}\sum_{n=1}^{N}\gamma(c_{in})(x_n - \propto_i^{new})(x_n - \propto_i^{new})^T \tag{18}$$

$$\omega_i^{new} = \frac{N_i}{N} \tag{19}$$

with:

$$N_i = \sum_{n=1}^{N}\gamma(c_{in}) \tag{20}$$

Step 4: Compute the log-likelihood:

$$\ln p(X;\theta) = \sum_{n=1}^{N}\ln\left\{\sum_{i=1}^{M}\omega_i N(x_n,\theta)\right\} \tag{21}$$

Step 5: Check for the convergence: If $|\theta^{new} - \theta| \leq \varepsilon$ or $|\ln p(X;\theta^{new}) - \ln p(X;\theta)| \leq \epsilon$ then stop. Otherwise, go to Step 2.

See ([22]: chapter 9) for more details on the EM-GMM algorithm and the derivations of the equations above. In the next section, experimental results will be shown when using the EM-GMM based modeling in monitoring the CS of robotic peg-in-hole assembly processes.

4 Experimental results

In order to see the performance of the EM-GMM CS monitoring process, a test stand was built that is composed of a KUKA lightweight robot (LWR) 4+ doing peg-in-hole assembly tasks. The key features of the KUKA LWR 4+ is detailed in [29]. The KUKA LWR 4+ is equipped with appropriate sensors that enable researchers in capturing the wrench and pose signals of the manipulated object through a fast research interface (FRI) port which is installed within the robot. The FRI port is connected to a remote PC that performs the computational aspects of the modeling process. The features of the PC that we used in our experiments are as follows: Intel (R) Core (TM) i5-2540 CPU with 2.6 GHz speed and 4 GB RAM running under a Linux environment. The rate of the communication between the remote PC and the robot, through the FRI, is 100 Hz. The programming is done through a C++ platform. Two experiments will be considered; in the first one, monitoring a typical robotic peg-in-hole assembly process is targeted. In the second experiment, a camshaft caps assembly of an automotive powertrain (which is multiple pegs-in-holes assembly process) is considered. For both of the experiments, the EM-GMM scheme is used in monitoring the processes.

4.1 Experiment 1: Typical peg-in-hole assembly

In this experiment, the EM-GMM is used in recognizing different phases of the peg-in-hole process shown in Fig. 1. The overall task is divided into five segments and the goal is to use the sensed wrench and pose signals in building a model for each phase. Figure 2 shows the captured wrench and

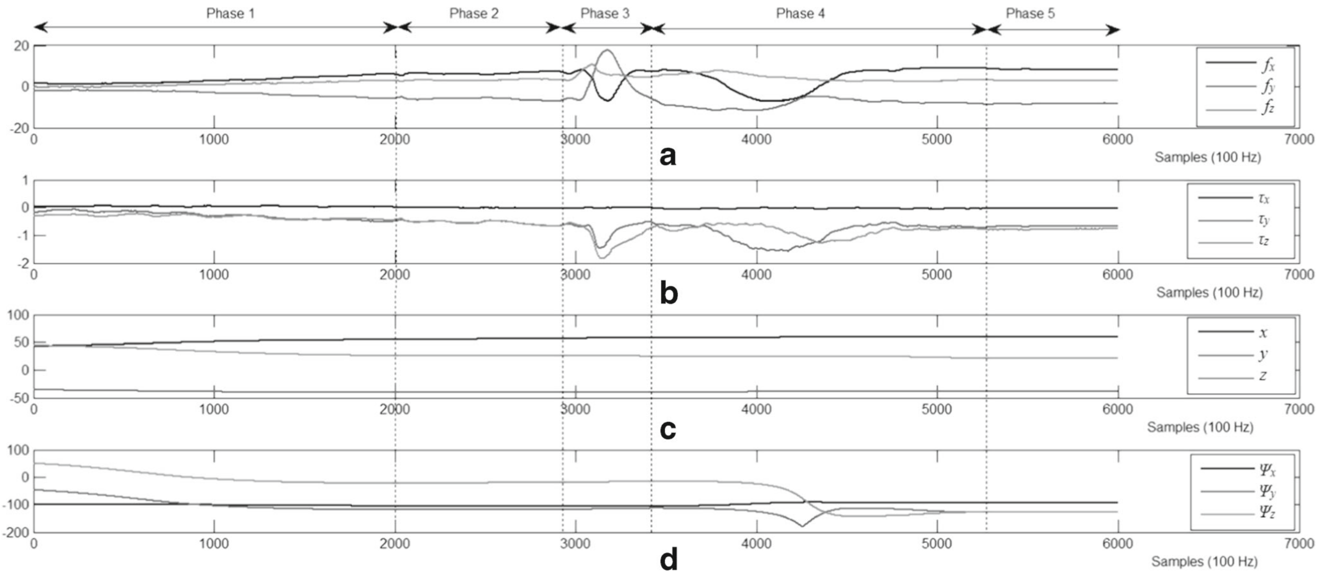

Fig. 2 Experiment 1: manipulated object wrench and pose measurements: **a** Cartesian forces (in N), **b** Torques around the Cartesian axes (in N m), **c** Cartesian position (in cm), and **d** orientation around the Cartesian axes (in degree)

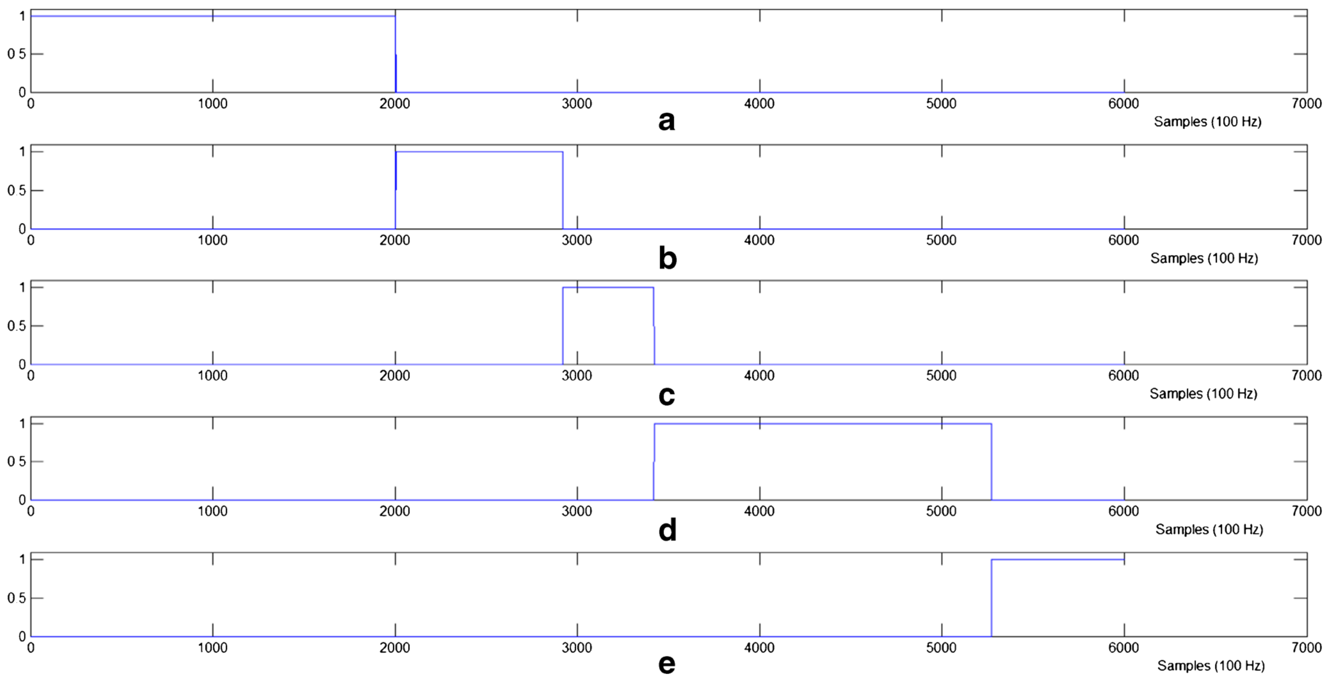

Fig. 3 Experiment 1: EM-GMM models outputs: **a** phase 1 (free space), **b** phase 2, **c** phase 3, **d** phase 4, and **e** phase 5

pose signals for each phase of this task. The signals of Fig. 2 were segmented into five segments according to the phases shown in Fig. 1. We can notice that as the task is being executed, different signals are varied within a phase and between different phases. For instance, if we examine the captured signals for phase 1 and phase 3 shown in Fig. 2, we can see the significant variations in those signals that would lead to the possibility of building efficient models for phase 1 and phase 3 through those signals. Similar fact could be deduced to the other phases of the assembly. Such inter-model signals variation is the key factor that enables us of developing a model for each phase. Two hundred fifty samples from each phase were taken out as test samples, so that one can check the performance of the developed models. For each signal, three mixtures were used and the models were developed using the EM-GMM scheme. Figure 3 shows the EM-GMM models outputs for each phase (for both training and test samples). The models outputs shown in Fig. 3 were generated on the basis that for the ith model output is considered as 1 if its EM-GMM likelihood (11) is more than the rest, otherwise it is 0. That is to say, the firing of the model is 1 or 0 depending on the value of the model output for a certain sample. If a model is having a maximum output value in a sample, then the firing is 1 (which means this sample belongs to the class of that model); otherwise, the firing would be 0 (which means the sample does not belong to the class of that model). From the segments intervals shown in Fig. 2 and models outputs shown in Fig. 3, the excellent monitoring accuracy for the EM-GMM based approach can be readily noticed. Hence, as a comparison between Figs. 2 and 3, the excellent recognition performance of the suggested EM-GMM CS recognition scheme can be clearly seen from those two figures. For instance, if we examine the periods that belong to phase 1 in Fig. 2, we can see that the samples of this phase were exhilaratingly detected through the suggested scheme and as shown in Fig. 3a. The same can be inferred for the other phases. We have five phases and

Table 1 Experiment 1: classification success rate (CSR) for EM-GMM, GS-FCA, SGB, and CFC modeling approaches

Modeling type	CSR (%)
EM-GMM	95.1
GS-FCA	71.4
SGB	67.5
CFC	30.8

Table 2 Experiment 1: computational time for EM-GMM, GS-FCA, SGB, and CFC modeling approaches

Modeling type	Computational time (in s)
CFC	0.001
EM-GMM	15.823
SGB	77.168
GS-FCA	197.931

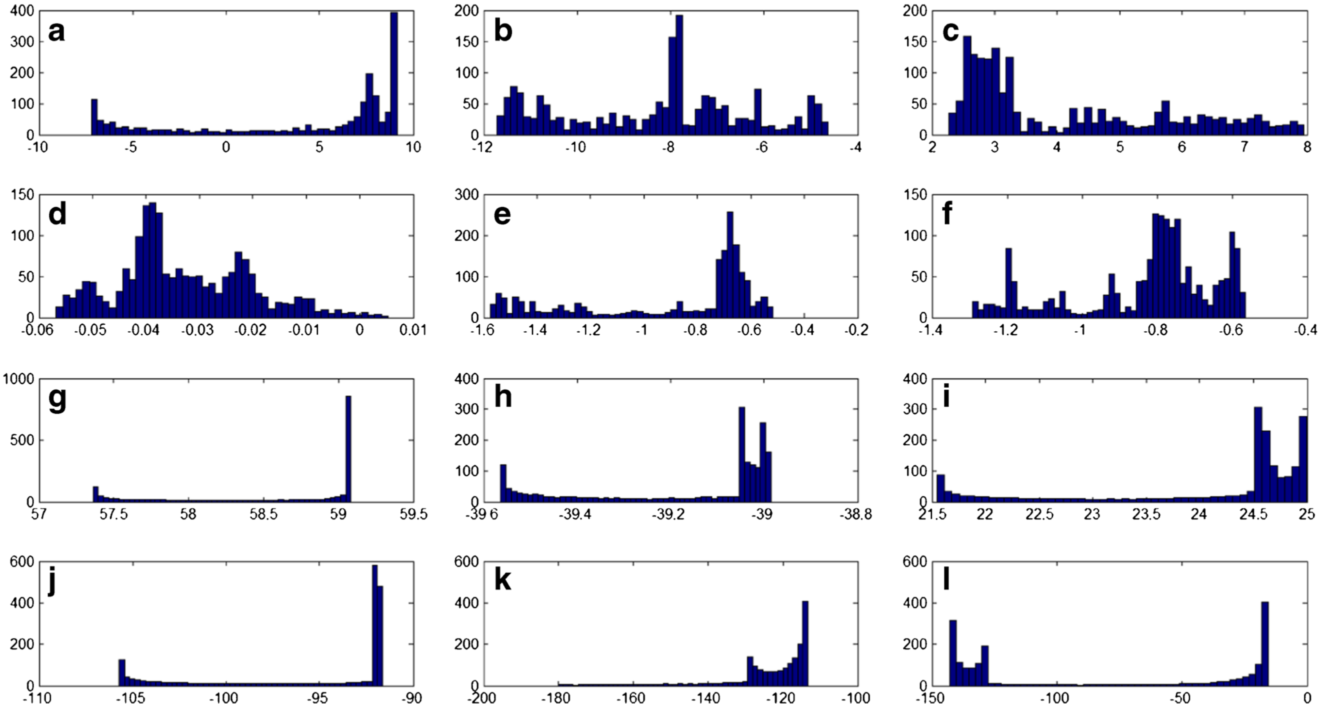

Fig. 4 Experiment 1: histograms of phase 4: **a** f_x, **b** f_y, **c** f_z, **d** τ_x, **e** τ_y, **f** τ_z, **g** x, **h** y, **i** z, **j** Ψ_x, **k** Ψ_y, and **l** Ψ_z

as mentioned above for each phase, we have 250 test samples. So totally, we have $5 \times 250 = 1,250$ test samples. Using the EM-GMM scheme, 1,189 samples were correctly classified and 61 samples were misclassified. Hence, the classification success rate (CSR) was computed to be 95.1 % when using the EM-GMM CS recognition scheme. In order to have a comparison study, the available CS modeling approaches were considered in modeling the same task above so that a comparison can be performed with the suggested approach. The approaches that were considered in the comparison are conventional fuzzy classifier (CFC) [14], the stochastic gradient boosting (SGB) classifier [28], and the gravitational search-fuzzy clustering algorithm (GS-FCA) [31] schemes in modeling the same task. For the GS-FCA, 893 samples were correctly classified and 357 samples were misclassified. For the SGB CS recognition

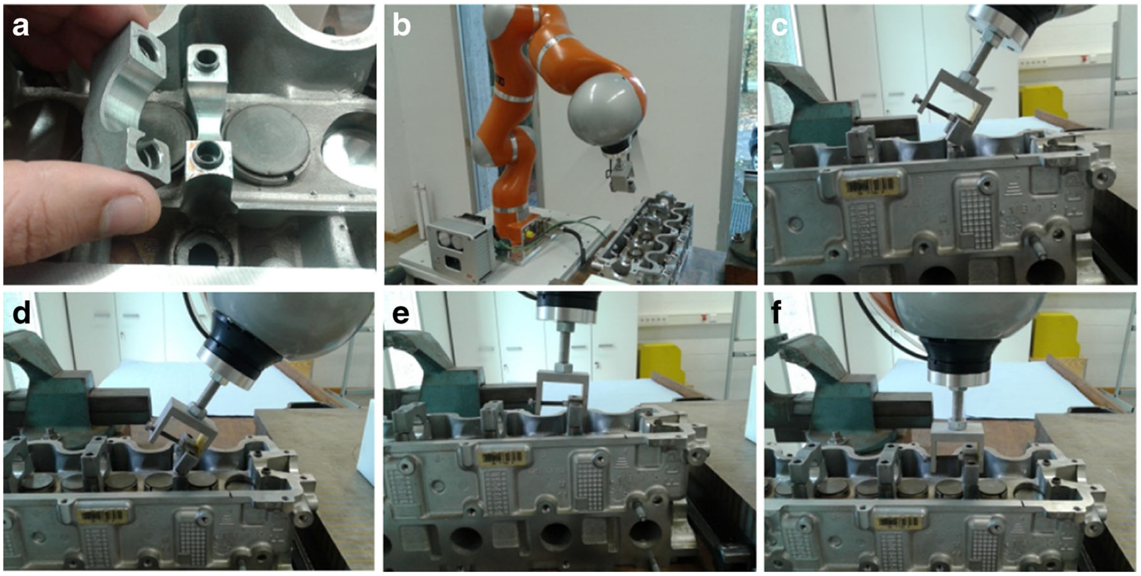

Fig. 5 Experiment 2: camshaft caps assembly phases: **a** camshaft caps assembly as double pegs-in-holes, **b** phase 1 (free space), **c** phase 2, **d** phase 3, **e** phase 4, and **f** phase 5

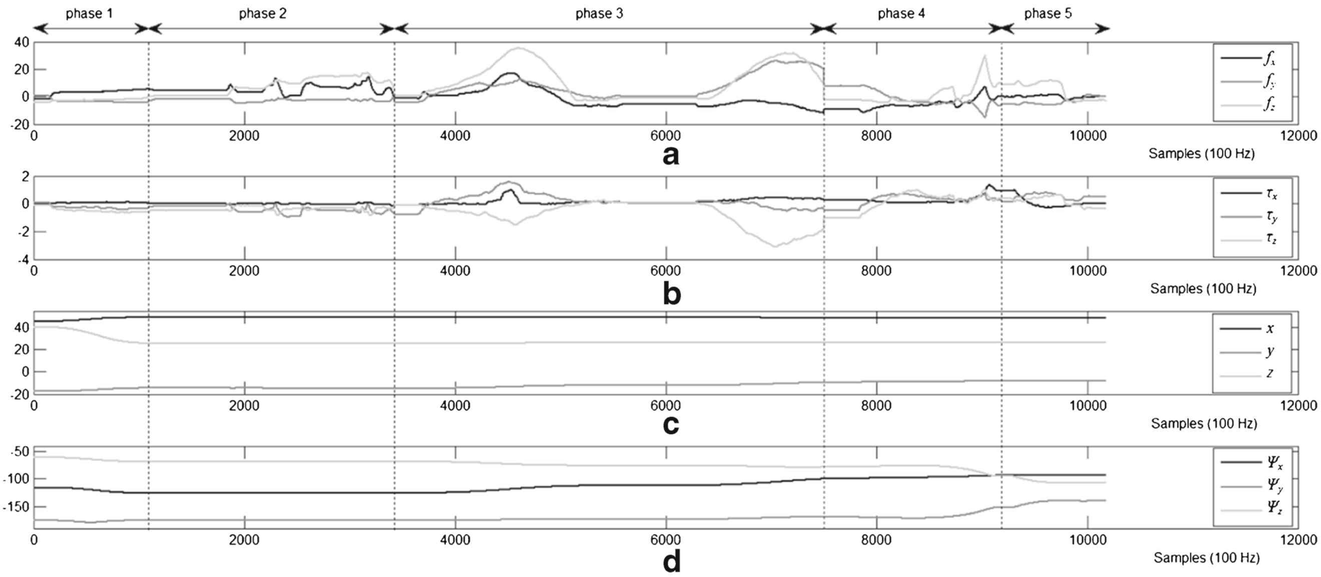

Fig. 6 Experiment 2: the manipulated object wrench and pose measurements: **a** Cartesian forces (in N), **b** torques around the Cartesian axes (in N m), **c** Cartesian position (in cm), and **d** orientation around the Cartesian axes (in degree)

scheme, we got 844 samples correctly classified with 406 samples misclassified. The CFC approach had 385 samples correctly classified with 865 misclassified. Hence, the CSRs were computed to be 71.4, 67.5, and 30.8 % for the GS-FCA, SGB, and CFC, respectively. Table 1 summarizes the success rate for the approaches above including the EM-GMM.

We also measured the computational time for building the models of each approach. The CFC modeling scheme was shown to have the least computational time of 0.001 s. The CFC computation time is very small since it involves only the computations of the mean and standard deviation of the captured signals. For the EM-GMM modeling scheme, the computational time was measured to be 15.823 s and that of the SGB and GS-FCA were computed to be 77.168 and 197.931 s, respectively. Table 2 summarizes the computational cost for all of the approaches considered in the experiment.

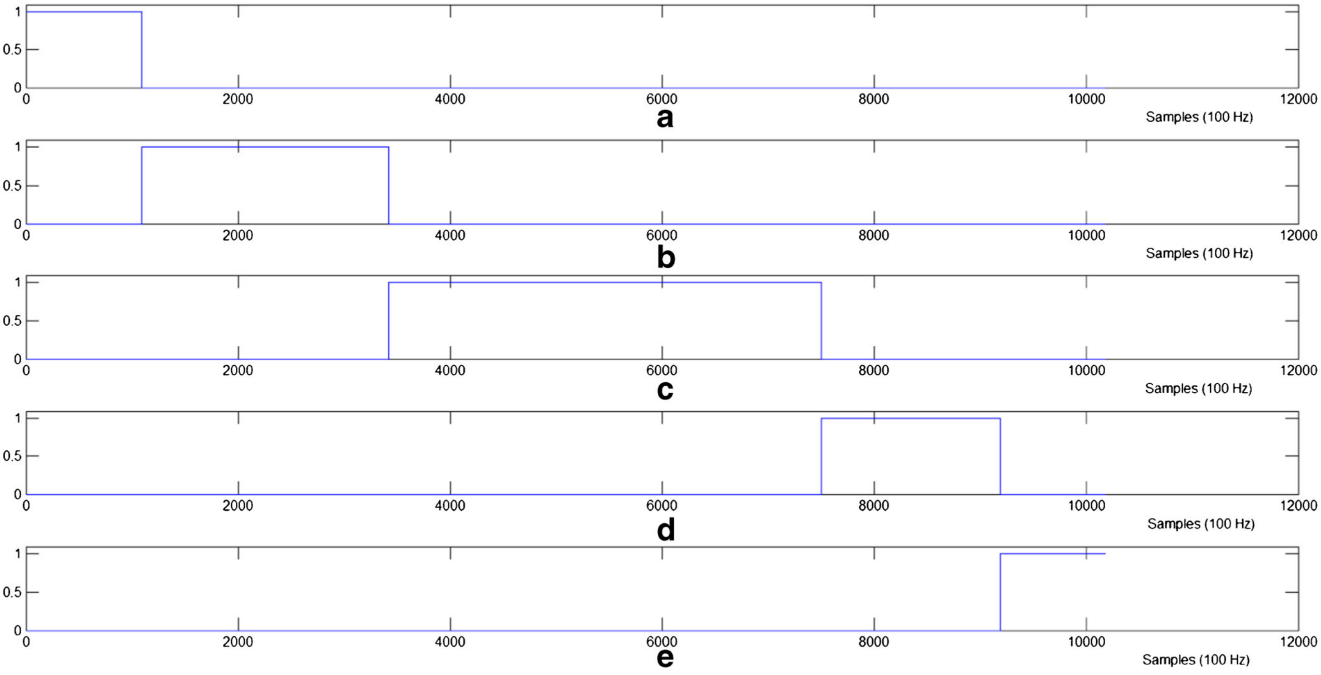

Fig. 7 Experiment 2: EM-GMM based models outputs: **a** phase 1 (free space), **b** phase 2, **c** phase 3, **d** phase 4, and **e** phase 5

Table 3 Experiment 2: classification success rate (CSR) for EM-GMM, GS-FCA, SGB, and CFC modeling approaches

Modeling type	CSR (%)
EM-GMM	97.4
GS-FCA	73.6
SGB	68.7
CFC	32.5

Despite the efficient computational time obtained with the CFC modeling scheme, the degraded CSR of 30.8 % makes it to be undesirable in modeling such signals. From the other hand, the EM-GMM is having a moderate computational time (better than that of the SGB and GS-FCA schemes) with excellent CSR of 95.1 % that make it more suitable for capturing such signals with reduced computational cost. One can notice that compared with the available CS recognition approaches, the EM-GMM is outperforming the rest. The main reason of such superiority comes from the fact that the EM-GMM can accommodate the nonstationary distribution of the signals and hence improve the CS recognition performance. As a sample, phase 4 was considered and the histograms of its signals were sketched and as shown in Fig. 4. It can be seen that all signals are nonstationary that gives the privilege to the EM-GMM in monitoring such a phase. All other phases are also nonstationary and they were not sketched to save space. Furthermore, using the EM algorithm in computing the parameters of the mixtures enhanced the performance of the CS monitoring process since it already results in maximizing the log-likelihood.

4.2 Experiment 2: Powertrain camshaft caps assembly

In this experiment, the EM-GMM is used in monitoring the KUKA LWR doing the assembly of powertrain camshaft caps. We considered such interesting task so that we can show the applicability and superiority of the suggested CS recognition scheme in a real-world assembly task of industrial relevance. A camshaft cap assembly is a multiple peg-in-hole assembly process as shown in Fig. 5a. We can see that the shape and dimensions of the camshaft caps assembly is different from the typical peg-in-hole assembly process shown in Fig. 1. Different phases of this assembly process are shown in Fig. 5b through Fig. 5f. Figure 6 shows the captured wrench and pose signals for this process. Likewise to experiment 1, the task signals are segmented into five segments and according to the phases shown in Fig. 5. Two hundred fifty samples were selected out as test samples for each phase, and we used the rest for training. Using the EM-GMM CS recognition system with three mixtures for each signal, a model is developed for each phase. Figure 7 shows the output of each model for all (training + test) signals of this process. By examining Figs. 6 and 7, one can notice the excellent monitoring capability of the EM-GMM for all phases of the given assembly task. We have 5 × 250 = 1, 250 test samples and when using the EM-GMM

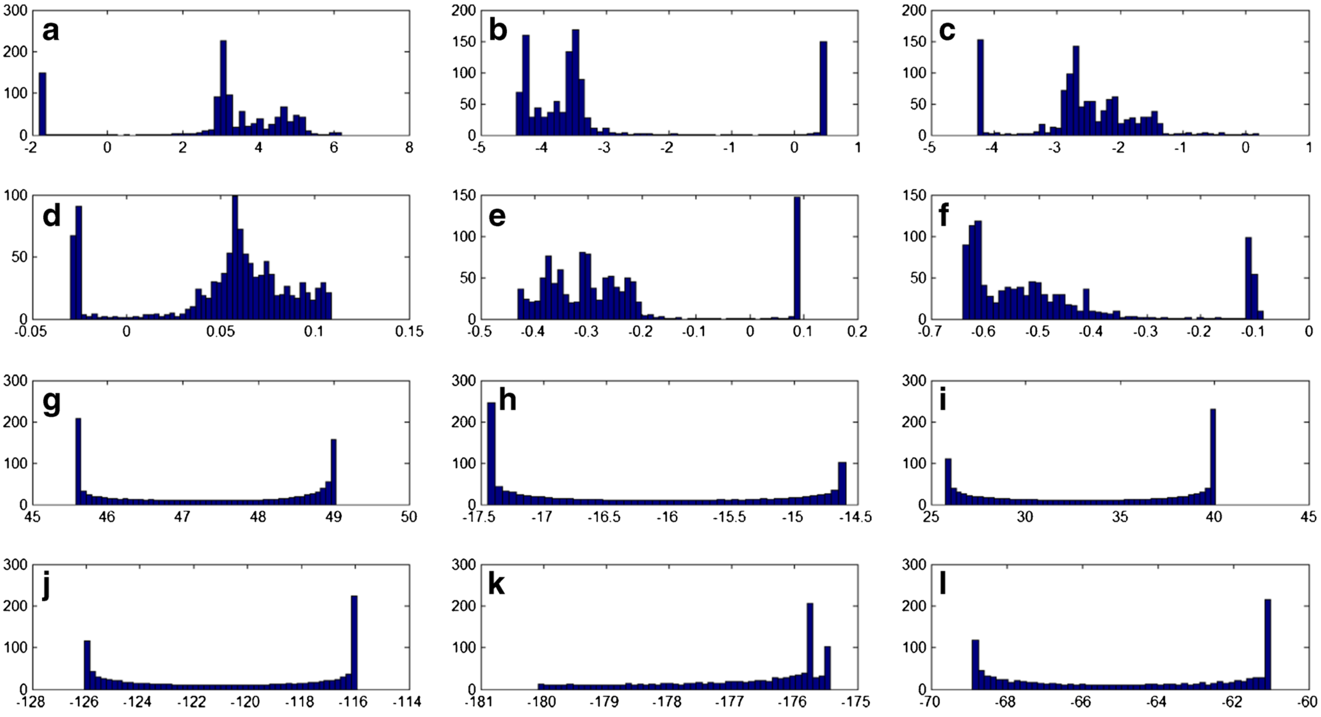

Fig. 8 Experiment 2: histogram of phase 5: **a** f_x, **b** f_y, **c** f_z, **d** τ_x, **e** τ_y, **f** τ_z, **g** x, **h** y, **i** z, **j** Ψ_x, **k** Ψ_y, and **l** Ψ_z

Table 4 Experiment 2: computational time for EM-GMM, GS-FCA, SGB, and CFC modeling approaches

Modeling type	Computational time (in s)
CFC	0.0014
EM-GMM	26.635
SGB	129.899
GS-FCA	333.184

CS modeling scheme, 1,218 samples were correctly classified and 32 were misclassified. This would result in a CSR of 97.4 % when using the EM-GMM CS modeling scheme. For the GS-FCA approach, 920 samples were correctly classified and 330 samples were misclassified. When using the SGB classifier, we had 859 samples successfully classified and 391 samples were misclassified. The CFC resulted with 406 correctly classified samples and 844 misclassified samples. Hence, the CSR for the EM-GMM, SGB, and CFC schemes were computed to be 73.6, 68.7, and 32.5 %, respectively. Table 3 summarizes the CSR for approaches, and it can be seen that the EM-GMM is outperforming the rest. In order to see the cause of such superiority, the histograms of all signals of phase 5 were sketched and as shown in Fig. 8. It can be noticed that almost all signals are nonstationary (see Fig. 8). The other phases are also nonstationary; however, they were not sketched to save space. Since the EM-GMM can accommodates such nonstationary distribution, then it would result in a more accurate monitoring scheme. Furthermore, the use of the EM algorithm in computing the parameters of the mixtures components would enhance the EM-GMM in the modeling process.

Likewise to experiment 1, we measured the computational time for the modeling schemes considered in experiment 2. The computational time for the CFC, EM-GMM, SGB, and GS-FCA modeling schemes were found to be 0.0014, 26.635, 129.899, and 333.184 s, respectively. The computational time for the mentioned approaches is summarized in Table 4. Compared with the GS-FCA and SGB modeling scheme, the EM-GMM is of superior performance and reduced computational cost. The degraded performance of the CFC modeling scheme makes it undesirable even though it has the least computational time.

5 Conclusion

Expectation maximization-based Gaussian mixtures models (EM-GMM) was successfully employed in monitoring the force-controlled robotic assembly tasks. The wrench (Cartesian forces and torques) and pose (Cartesian position and orientation) were captured for peg-in-hole robotic assembly tasks. The captured signals were segmented into different phases and using the EM-GMM, a model was developed for each phase. The developed models were used in monitoring the contact state (CS) of those assembly tasks. In order to see the performance of the suggested CS monitoring scheme, a KUKA lightweight robot (LWR) was installed and used in doing two assembly experiments. In the first experiment, a typical peg-in-hole assembly process was considered, and the EM-GMM was efficiently used in monitoring the process. In the second experiment, the task of camshaft caps assembly was studied, and the EM-GMM was successfully used in monitoring this process also. For both experiments, comparisons with the results of the available CS monitoring approaches like conventional fuzzy classifier (CFC), stochastic gradient boosting (SGB) classifier, and the gravitational search-fuzzy clustering (GS-FCA) classifier were considered. Through considering the classification success rate (CSR) and the computational time as comparison indices, the superiority of the EM-GMM modeling scheme was shown with a reduced computational time. Such excellent performance resulted from accommodating the nonstationary behavior for the captured signals and the use of the EM algorithm in computing the parameters of the Gaussian components. Despite the excellent recognition ability of the EM-GMM, it requires the number of the mixtures of the Gaussian components for each signal that would add more burden to the realization of this approach. Therefore, future researches should focus on relaxing the need of knowing the number of the components for each signal. A possible approach of realizing that is through the use of variational Bayesian inference for computing the Gaussian mixtures. Another possible improvement is to make the models developed more parsimonious through selecting only important features when developing the models. However, this is left to future research efforts.

Acknowledgements This research is supported by the Fonds National de la Recherche (FNR) in Luxembourg under grant no. AFR 2955286. The authors would like to thank the teams of KUKA Roboter GmbH in Houthalen, Belgium and Augsburg, Germany for their support on the technical issues of the KUKA Lightweight Robot 4+.

References

1. Hirai S, Iwata K (1992) Recognition of contact state based on geometric model. In: Proceedings 1992 IEEE international conference robotics automation, Nice-France, pp 1507–1512
2. McCarragher BJ (1994) Petri net modeling for robotic assembly and trajectory planning. IEEE Trans Indust Elect 41(6):631–640
3. Cao T, Sanderson AC (1994) Task decomposition and analysis of robotic assembly task plans using Petri nets. IEEE Trans Indust Elect 41(6):620–630
4. Farahat AO, Graves BS, Trinkle JC (1995) Identifying contact formations in the presence of uncertainty. In: Proceedings 1995

IEEE/RSJ international conference intelligent robotics systems, Pittsburgh-PA, pp 59–64

5. McCarragher BJ (1996) Task primitives for the discrete event modeling and control of 6-DOF assembly tasks. IEEE Trans Robot Autom 12(2):280–289
6. Hovland GE, McCarragher BJ (1997) Combining force and position measurements for the monitoring of robotic assembly. In: Proceedings 1997 IEEE/RSJ international conference intelligent robotics systems, Grenoble-France, pp 654–660
7. Skubic M, Volz RA (1997) Learning force sensory patterns and skills from human demonstration. In: Proceedings 1997 IEEE international conference robotics automation, NM-USA, pp 284–290
8. Skubic M, Castrianni SP, Volz RA (1997) Identifying contact formations from force signals: a comparison of fuzzy and neural network classifiers. In: Proceedings 1997 international conference neur. net., Houston-TX, pp 1623–1628
9. Mosemann H, Raue A, Wahl F (1998) Classification and recognition of contact states for force guided assembly. In: Proceedings IEEE international conference system manual, cybernetics, San Diego-CA, pp 3400–3405
10. Hovland GE, McCarragher BJ (1998) Hidden Markov models as a process monitor in robotic assembly. In: International journal of robotics research, vol. 17, no. 2, pp 153–168
11. Xiao J, Liu L (1998) Contact states: representation and recognizability in the presence of uncertainties. In: Proceedings IEEE international conference robotics automation, Victoria-Canada, pp 1151–1156
12. De Schutter J, Bruyninckx H, Dutré S, De Geeter J, Katupitiya J, Demey S, Lefenvre T (1999) Estimating first-order geometric parameters and monitoring contact transitions during force controlled compliant motions. Int J Robot Res 18(12):1161–1184
13. Everett LJ, Ravuri R, Volz RA, Skubic M (1999) Generalized recognition of single-ended contact formations. IEEE Trans Robot Autom 15(5):829–836
14. Skubic M, Volz RA (2000) Identifying single-ended contact formations from force sensor patterns. IEEE Trans Robot Autom 16(5):597–603
15. Son C (2001) A neural/fuzzy optimal process model for robotic part assembly. Int J Mach Manuf 41(12):1783–1794
16. Chung S-Y, Lee DY (2001) Discrete event systems approach to fixtureless peg-in-hole assembly. In: Proceedings 2001 American control conference, Arlington-VA, pp 4962–4967
17. Fei Y, Zhao X (2003) An assembly process modeling and analysis for robotic multiple peg-in-hole. J Intell Robot Syst 36:175–189
18. Lau HY (2003) A hidden Markov model-based assembly contact recognition system. Mechatronics 13(8–9):1001–1023
19. Cortesão R, Koeppem R, Nunes U, Hirzenger G (2004) Data fusion for robotic assembly tasks based on human skills. IEEE Trans Robot Autom 20(6):941–952
20. Debus TJ, Dupot PE, Howe RD (2004) Contact state estimation using multiple model estimation and Markov models. Int J Robot Res 23(4–5):399–413
21. Iwata H, Sugano S (2005) Human-robot-contact-state identification based on tactile recognition. IEEE Trans Ind Elect 52(6):1468–1477
22. Bishop CM (2006) Pattern recognition and machine learning. Springer, New York, pp 423–455
23. Thomas U, Molkenstruck S, Iser R, Wahl FM (2007) Multi sensor fusion in robot assembly using particle filters. In: Proceedings IEEE international conference robotics automation, Rome-Italy, pp 3837–3843
24. Katsura S, Matsumoto Y, Ohnishi K (2007) Modeling of force sensing and validation of disturbance observer for force control. IEEE Trans Ind Elect 54(1):530–538
25. Meeussen W, Rutgeerts J, Gadeyne K, Bruyninckx H, De Schutter J (2007) Contact-state segmentation using particle filters for programming by human demonstration in compliant-motion tasks. IEEE Trans Robot 23(2):218–231
26. Okuda H, Takeuchi H, Inagaki S, Suzuki T (2008) Modeling and analysis of peg-in-hole task based on mode segmentation. In: Proceedings 2008 SICE annual conference, Tokyo-Japan, 20-22 Aug, pp 1595–1600
27. Hastie T, Tibshirani R, Friedman J (2009) The elements of statistical learning: data mining, inference, and prediction. Springer, New York, pp 265–267
28. Cabras S, Castellanos ME, Staffetti E (2010) Contact-state classification in human-demonstrated robot compliant motion tasks using the boosting algorithm. IEEE Trans Syst Man Cyber B 40(5):1372–1386
29. Lightweight Robot 4+ Specification, Version: Spez LBR 4+ V5 en, KUKA Roboter GmbH, 22 Dec 2011
30. Hertkorn K, Rao MA, Preusche C, Borst C, Hirzinger G (2012) Identification of contact formations: resolving ambiguous force torque information. In: Proceedings of the 2012 IEEE international conference robotics automation, St. Paul-MN, 14–18 May, pp 3278–3284
31. Jasim IF, Plapper PW (2013) T-S fuzzy contact state recognition for compliant motion robotic tasks using gravitational search-based clustering algorithm. In: Proceedings IEEE International Conference Fuzzy Systems, Hyderabad-India, 7–10 Jul, pp 1–8