Engineering Applications of Artificial Intelligence

Scoring and Assessment in Medical VR Training Simulators with Dynamic Time Series Classification

Neil Vaughan · Bogdan Gabrys

Received: 21st August 2019 / Accepted: 8th June 2020

Abstract This research proposes and evaluates scoring and assessment methods for Virtual Reality (VR) training simulators. VR simulators capture detailed n-dimensional human motion data which is useful for performance analysis. Custom made medical haptic VR training simulators were developed and used to record data from 271 trainees of multiple clinical experience levels. DTW Multivariate Prototyping (DTW-MP) is proposed. VR data was classified as Novice, Intermediate or Expert. Accuracy of algorithms applied for time-series classification were: dynamic time warping 1-nearest neighbor (DTW-1NN) 60%, nearest centroid SoftDTW classification 77.5%, Deep Learning: ResNet 85%, FCN 75%, CNN 72.5% and MCDCNN 28.5%. Expert VR data recordings can be used for guidance of novices. Assessment feedback can help trainees to improve skills and consistency. Motion analysis can identify different techniques used by individuals. Mistakes can be detected dynamically in real-time, raising alarms to prevent injuries.

Keywords Virtual reality · Simulation · Medical training · Skill assessment · Classification · Time series

1 Introduction

Virtual Reality (VR) training simulators are growing in popularity and are used by aircraft pilots [47], surgeons and clinicians [14], [12], [41], military

N.Vaughan
University of Exeter, Institute of Biomedical and Clinical Science, RILD Building, Barrack Road, Exeter, United Kingdom
Tel.: +44-7783-527-327
E-mail: n.vaughan@exeter.ac.uk

B. Gabrys
University of Technology Sydney, Advanced Analytics Institute, 15 Broadway, Ultimo NSW 2007, Australia

and defence applications [18]. Training simulators enable trainees to learn the skills required to perform skilled tasks by practicing on a virtual model in-vitro. The advantages of using simulators for surgery are well documented including the reduced risk of injury associated with practice on patients [24], ability to safely practice emergency procedures and ability to practice on various models of different patients [39]. Additionally, due to changing training structure and compliance with the European Working Time Directive, clinical experts are being required to reduce their time spent assessing novices [44]. Virtual training is useful in high risk activities such as epidural needle insertion, helping to avoid injuries [9]. Automated training with VR simulators could assist, however methods for skill assessment in VR, particularly dynamically in real-time, are still undeveloped. The following sections outline recent methods for classifying skill and define our proposed method, which is applied and tested for accuracy.

1.1 Background

Skill classification in surgical VR training has been attempted using various approaches [42], including time series clustering and classification which received considerable research attention [28], [31]. The JIGSAWS dataset [8] provides benchmark data from Da Vinci robot for testing skill classification and 100% has been achieved using deep learning [7] specifically for surgical skill assessment. Since 2018 this emerging field of Surgical Data Science rapidly accelerated with increases in Deep Learning for multivariate Time Series Classification. Recent advances producing state of art results in general time series classification include long short term memory (LSTM), a recurrent Fully Convolutional Network (FCN) for multivariate series [15] and TimeNet, a multilayered Recurrent Neural Network (RNN) [27], inspired by successful image feature extraction. Machine learning algorithms with inertial measurement units can improve the predictive power of surgeon motion analysis [43]. Support Vector Machine (SVM) classification can achieve 86% sensitivity whereas the non-machine learning Lempel–Ziv (LZ) complexity metric gave 64% sensitivity. This suggests that nonparametric supervised learning algorithm such as SVM applied to surgical skills classification can be useful for motion pattern recognition. [30] assessed skill for MIS, hand-eye bimanual coordination, spatial perception in a sample population of 4 experts, 22 residents, 16 novices, applying Linear discriminant analysis (LDA) (71%), Nonlinear support vector machine (SVM) (78.2%) and Adaptive neuro-fuzzy inference systems (ANFIS) (71.2%). Fuzzy classification and radial bias function (RBF) was used [12] with MIST-VR dataset containing 4 experts, 4 intermediate, 4 novices over 200 Epochs gaining 33%. Support vector machine (SVM) [1] gained 91.6%. Hidden Markov Models (HMMs) can be used to classify surgeon skills from surgical gestures with accuracy up to 100% and discovers rules governing task ordering [38]. Surgical skill can be classified using global measurements of the simulation, such as the distance travelled [5], the total time taken [14], force

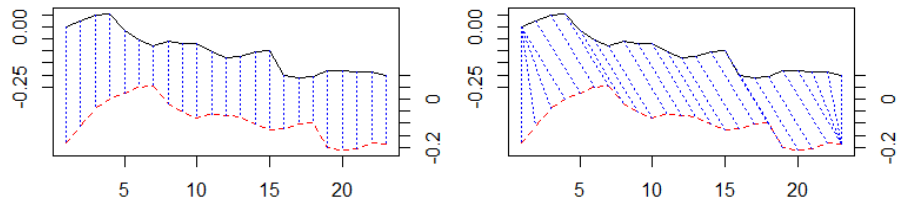

Fig. 1 Alignment by (left) Euclidean and (right) DTW distance measure. The upper time series is drawn vertically shifted to enhance visualization.

or pressure signatures [45], the number and speed of hand motions [5]. These global measurements offer the quickest methods but lack information about task structure. Dynamic time warping (DTW) could further benefit classification.

1.2 Dynamic Time Warping

Given $\{a_n\}$ and $\{b_n\}$ are two multivariate time series, various local distance functions, denoted δ, are compatible with DTW. Euclidean distance (Eq. 1) can be applied if both series are of equal length. Squared Euclidean distance uses the same formula (Eq. 1) without the square root, reducing computation. Manhattan city block is widely used and rapidly computed. Minkowski forms a generalisation of Euclidean and Manhattan distances. Others distance measures include Mahalanobis [26], Bhattacharyya [4] and Canberra [20].

$$\sum_{i=1}^{n} \sqrt{\sum_{v=1}^{5} (a_{vi} - b_{vi})^2} \tag{1}$$

where a_{vi} and b_{vi} refer to variable v from element i within $\{a_n\}$ and $\{b_n\}$, both containing n elements with 5 variables.

The dynamic programming algorithm DTW distance measure supports multivariate time series of unequal length where $n \neq m$ [3]. A comparison between Euclidean and DTW is shown in Fig. 1.

Visualization of the DTW with warping path $\{w_n\}$ is shown in Fig. 2, left. The Global constraints Itakura Parallelogram [37] applies (Fig. 2, middle) (Eq. 2).

$$w_i = w_{i-1} + min[(a_{n+1}, b_{m+1}), (a_{n+2}, b_{m+3}), (a_{n+3}, b_{m+2})] \tag{2}$$

1.3 Time Series Prototyping

Several time series clustering methods require combination of several time series into a single prototype time series, representing characteristics of all time series within a cluster [10]. Prototyping is an essential tool for many clustering

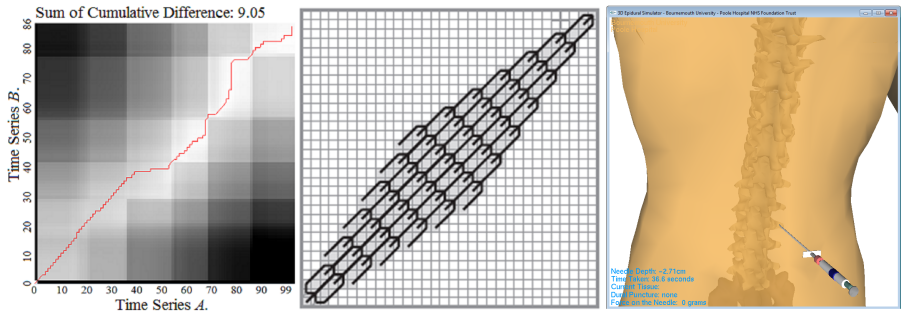

Fig. 2 (left) Local Cost Matrix (LCM) with warping path W shown as red line, (middle) Itakura Parallelogram, (right) developed epidural simulator.

algorithms like K-Means, or Ascendant Hierarchical Clustering to reposition cluster centroids and describe each cluster. Mean and median prototyping does not perform accurately and perturbs convergence of clustering algorithms, producing a non-representative prototype [32], [42]. Partition Around Medoids (PAM) prototyping has a benefit that it avoids modifying any time series by calculating for each time series $\{a_n\}$ the sum of distances to all other series, and selecting the prototype to be the one time series which has lowest total distance (Eq. 3).

$$PAM(\{a_n\}) = DTW(\{a_n\},\{b_n\}) + DTW(\{a_n\},\{c_n\}) + DTW(\{a_n\},\{d_n\}) + ... \quad (3)$$

DTW barycenter averaging (DBA) [32] is an iterative global method. The global nature of DBA enables the avoidance of iterative pairwise averaging, so results are unaffected by ordering. Shape based extraction [31] or fuzzy-based prototypes use fuzzy clustering such as fuzzy c-medoids (FCMdd) [13]. For this work, we propose a new method of time series prototyping: DTW Multivariate Prototyping (DTW-MP).

1.4 Dynamic monitoring with Upper and Lower Envelopes

Upper and lower envelopes are generated in this work to create a tunnel of acceptable motion using lower bounding so that an alert can be triggered if a VR object exits from the normal path of motion. Existing lower bounding methods have been proposed including: LB_Yi [48], LB_Kim [19], LB_Keogh [17], $LB\_Improved$ [22]. For this work there are three purposes of applying lower bounds: (1) To speed up the classification of new insertions by reducing the complexity of the similarity search required for new insertions, (2) To enable an alarm to be raised if the new time series does not stay within the tunnel of acceptable motion, between the upper and lower envelope around the cluster's prototype insertion. (3) The upper and lower envelopes define an area

which will contain all of the expert insertions, and our proposed $DTW - MP$ prototype.

1.5 Dynamic Assessment of Incomplete Series

Skill classification taking place during an insertion only has access to a partial time series, which is the first part of a surgical procedure, but not the end, because the remainder of the procedure has not yet been completed. This problem is related to time series sub-sequences [34]. The given time series $\{a_n\}$ of length n represents a sequence, $\{a_n\} = a_1, a_2, ..., a_n$. A subsequence $\{q_m\} = q_1, q_2, ..., q_m$, is a shorter region of length m from within $\{a_n\}$ which starts at any position i within $\{a_n\}$ (whereby i is restricted such that $i \leq (n - m)$, whereby $n \geq m \geq 1$. There are requirements which we aim to achieve when dealing with dynamic data: (1) Estimate during a procedure what proportion of the procedure has been completed. (2) Identify if the procedure is running fast or slow in comparison to the training insertions. (3) Compute the distance between a new partial trajectory and the cluster prototype. Recent research on detection of unusual time series events refer to discord subsequences [46], outliers, unusual, abnormal [29], [33], novel, deviant or anomalous time series subsequences [16], [21]. Our previous research has outlined methods for prediction of time series [23], [36], which are particularly relevant in the context of the incomplete time series and could be applied to predict events which are likely to arise in the time series, which is a future work. We apply comparison of incomplete time series in this work to enable dynamic monitoring of insertions in real-time.

2 Methods

2.1 Development of an Epidural Simulator

We developed a virtual reality epidural training simulator using 3 Degrees Of Freedom (DOF) haptic input with force feedback and epidural pressure measurements (Fig. 2, right) [41]. The 3D graphics model contains vertebrae with software and haptic based biomechanical models of soft tissues based on measurements from our clinical trial with obstetric patients of various Body Mass Index (BMI). The data generated from the VR simulator consists of multivariate time series recording position, force and pressure over time. The 3D motion of the tool in x, y and z planes is recorded over time by the haptic device. The fourth dimension is force applied by the user, measured within the haptic device. The fifth dimension is pressure, measured within the syringe plunger using a custom wireless microcontroller system [40]. Each epidural procedure tends to last around 20 seconds, during which the measurements were recorded at 500MHz, with 2 millisecond intervals. The resulting time series lengths are approximately 10,000 for each of the 5 measurements.

2.2 Data Collection Trial

We recorded simulator training data using our VR epidural training simulator [40]. Seven participants were in two groups: Group-C contained 3 medical trained NHS clinicians, each with varying experience of performing epidurals on real patients: ClinicianA had performed around 1000 real epidurals, ClinicianB had performed around 100 insertions and ClinicianC had performed around 20 epidurals. Group-N contained 4 non-clinicians who were not medically trained. Within Group-N, NonClinicianA had performed over 300 simulated epidurals, and the other 3 participants (NonClinicianB, NonClinicianC, NonClinicianD) had never performed real or simulated epidurals before. Skill labels (N=Novice, I=Intermediate, E=Expert) were assigned to each insertion based on the experience level, clinical background and number of clinical epidurals each clinician had previously completed. In total 271 needle insertions were recorded. NonClinicianA recorded 101 epidurals. NonClinicianB, NonClinicianC and NonClinicianD recorded 95, 31 and 20 epidurals. ClinicianA recorded 10, ClinicianB recorded 9 and ClinicianC recorded 5. To avoid data skew, epidural-40 subset of the 271 epidural dataset was created containing 40 insertions which exactly matches the class distribution and size of JIGSAWS dataset N, I, E, E, I, N, N, N with 8 participants performing 5 insertions each. This 40 epidural dataset was created at both lengths of 500 and 5000. All of the data recordings were of simulated epidural insertion, using the same haptic device, the same build version of the VR software and on the same computer. Recordings from 271 insertions were stored as multivariate time series $\{a_n\}$ containing 5 variables: $x, y, z, pressure, force$. Each series has different length, relative to the time taken.

2.3 Normalization of the Data

Due to collection methods, standardization of the raw data recorded is necessary to set microcontroller sensor data onto the same scale as the haptic device data. A standard normalization method is applied which scales each element a_n within the time series $\{a_n\}$ by subtracting the population mean from a_n and diving by the standard deviation σ. After normalization if one time series contains very subtle movement on z axis and another has large movement on z axis, these will be normalized into the same scale.

2.4 Skill classification Method 1: DTW 1-NN

The DTW-k-NN classifier was used to classify insertion skill, making use of all recorded insertions. Each of the training examples is checked to identify which most closely resembles the new insertion, by calculating the DTW distance between the new insertion and all previous recorded examples. The new insertion is labelled the same class as the closest training example (N, I or E).

2.5 Skill classification Method 2: Nearest Centroid

Nearest centroid classifier applies the k-NN classifier by measuring DTW distance between the new insertions and the cluster prototypes, reducing the number of DTW computations required compared to DTW-k-NN which requires all insertions. A range of 7 state-of-art prototype methods were applied: Mean, SoftDTW, DBA, PAM, Shape Extraction, and we propose 2 new prototype methods: DTW-MP$_D$ and DTW-MP$_I$. The prototype of each skill level (N,I,E) is calculated to represent a prototypical insertion for each cluster. The prototypical insertions are built from all insertions in each cluster. We propose the DTW Multivariate Prototyping (DTW-MP) algorithm to produce prototypical time series using DTW [42]. Our proposed prototyping method (DTW-MP) has advantages over Mean and Median: (1) DTW-MP retains features which occur in two time series at different times which would not be aligned by Mean. (2) DTW-MP can handle two series of different length unlike Mean. (3) The DTW-MP prototype is guaranteed to stay within the summative-envelope. Our proposed prototyping method (DTW-MP) starts by calculating the warping path $\{w_n\}$ between two time series (Fig. 2, left). The new DTW-MP prototype $\{p_n\}$ is created with the same length as the warping path $\{w_n\}$. Each element p_n in $\{p_n\}$ is set to the mean of the two elements from $\{a_n\}$ and $\{b_m\}$ which were aligned in the equivalent element w_n of the warping path $\{w_n\}$. Therefore, the length k of the new DTW-MP prototype $\{p_n\}$, will be the same length as the warping path $\{w_n\}$, which is $max(m, n) \leq k \leq m + n$.

2.6 Skill classification Method 3: Deep learning

Four deep learning techniques were used for time series classification: (1) a relatively deep Residual Network (ResNet) with 9 convolutional layers and a Global Average Pooling (GAP) layer [11]. (2) Fully Convolutional Network (FCN) [25] with a final layer of Global Average Pooling (GAP). (3) Convolutional Neural Network (CNN) [49] with final discriminative layer taking the result of the convolutions to give probability distribution over class variables. (4) Multi-Channel Deep Convolutional Neural Network (MCDCNN) [50] with architecture of a traditional deep CNN, plus the convolutions are applied in parallel on each dimension of the input MTS. These four DL architectures were chosen to provide a range of frameworks due to their previous successful applications to time series classification tasks. The DL implementation architectures were matching with the open source time series classification framework [6]. Each method was applied to the 40 epidural subset for skill classification.

Details of each deep learning architecture are in Fig. 3. Our method harnesses transfer learning, within FCN architecture, as one advantage of the utilized FCN method is the invariance which enables the use of a transfer learning approaches to train models on one dataset and further tune it on other target datasets. All the convolutions in the framework have a stride of 1

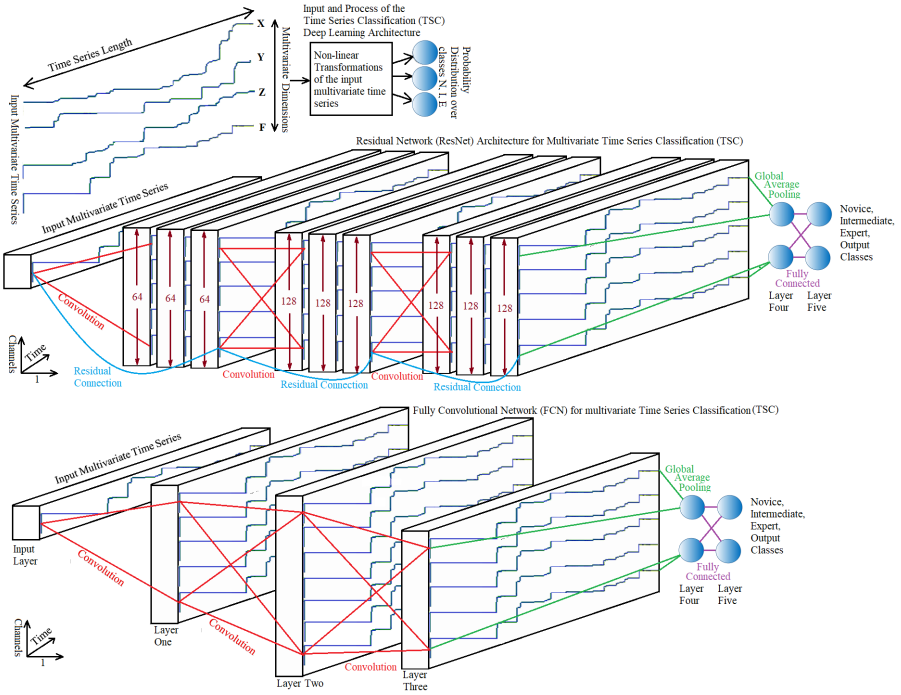

Fig. 3 Deep Learning Architecture for Multivariate Time Series Classification. (Top) input formatting process, (middle) ResNet Architecture, (lower) FCN Architecture.

which preserves the length of the time series after convolution. The methods we used to incorporate the time series data into the deep learning architectures is shown in Fig. 3 (top). The architectures for ResNet (Fig. 3, middle) and FCN (Fig. 3, lower) are shown. Architectures were based on adopted highest performing frameworks [6]. This illustrates that deep learning with multivariate time series is a challenging problem.

3 Methods for Dynamic Monitoring

3.1 Phase estimation to detect proportion of time series

It is first required to dynamically estimate how much of the procedure has been completed at any given time. Phase estimation is achieved by producing 10 subsequences of the prototype increasing in size from 10% to 100% at 10% intervals. The DTW distance is computed between the incomplete new insertion and each of the 10 prototype subsequences. The best match identifies the proportion of the procedure which is completed, irrespective of whether one time series occurred faster than the other.

3.2 Cluster Prototyping

Hierarchical clustering is applied to group similar good or bad insertions techniques together. In order to perform clustering, a distance matrix comparing each time series to all others is generated. The DTW Euclidean normalised distance is pre-calculated between all pairs of insertions to produce a distance matrix of size 271^2, which requires $\frac{271^2-271}{2}$ DTW computations. Within each time series $\{a_n\}$, the number of elements a_n is approximately 1000, so each DTW comparison requires approximately 1000^2 element alignments which equates to 36×10^9 Euclidean distance calculations. The distance matrix is used as input to hierarchical clustering.

3.3 Upper and Lower Bounds tunnel of acceptable motion

To enable dynamic real-time monitoring a tunnel or envelope of acceptable motion is created. We propose summative-envelope algorithm which creates an envelope pathway guided in shape by the closest insertions to the expert cluster prototype. It generates a new pair of time series which contain a summative upper $\{su_n\}$ (Eq. 4) and summative lower $\{sl_n\}$ (Eq. 5) envelopes from the expert insertion envelopes. Each element of the summative upper envelope $\{su_n\}$ is set to the maximum of any expert upper envelope at that time (Eq. 5), denoted as $\{e1_n\}$, $\{e2_n\}$, $\{e3_n\}$, ... The summative-envelope is generated for all 5 variables in the multivariate time series, so the summative-envelope could be visualised as a 5D tunnel of acceptable motion.

$$su_{vi} = max(e1_{vi}, e2_{vi}, e3_{vi}, ...) \tag{4}$$

$$sl_{vi} = min(e1_{vi}, e2_{vi}, e3_{vi}, ...) \tag{5}$$

where su_{vi} refers to variable v from element i within time series $\{su_n\}$.

This has benefit that new insertions will fit within the summative-envelope if they are within the envelope of any previously expert insertion, but not if it contains an event unlike any previously seen event in an insertion. The strength of our proposed summative-envelope technique is that the summative-envelope combines data from all insertions, whereas the LB_Keogh lower bounding upper and lower envelopes only contain information from one time series. The summative-envelope only needs to be computed once from the training data. It can subsequently be used to raise an alarm each time a new insertion goes outside of the summative-envelope. During a simulation, summative-envelopes could enable the allowable motion to be visualised by the user in real-time, providing more clarity of procedural requirements. When reviewing completed simulations, the summative-envelope can enable a visualisation showing where and when it went wrong. If a new insertion is one of the closest to the prototype, the envelope is then re-generated when the insertion completes, adapting to the new data. This was previously a problem with black-box skill classification.

3.4 Scoring a procedure based on tunnel of acceptable motion

The next step is to score each new insertion by checking that it remains inside a tunnel or envelope of acceptable motion which is created using prototypical insertion data. In order to score a new insertion dynamically during the insertion, the summative envelope from the identified cluster is used. The new insertion is monitored dynamically to identify whether it stays inside the envelope. If an insertion goes out of the envelope, an alarm is raised. The score is updated dynamically using Eq. 6, during the insertion taking into account what proportion of the insertion was outside of the envelope and the distance outside the envelope.

$$D(\{a_n\}) = \sum_{i=1}^{n} \sqrt{\sum_{v=1}^{5} max(a_{vi} - su_{vi}, sl_{vi} - a_{vi}, 0)^2} \tag{6}$$

Where $D(\{a_n\})$ is the total sum of distances the new insertion $\{a_n\}$ was outside the summative upper $\{su_n\}$ and summative lower $\{sl_n\}$ envelopes, which have 5 dimensions: x, y, z, pressure and force.

4 Results for Skill Classification

This section describes the application of the proposed methods to our VR simulator for epidural needle insertion.

4.1 Skill Classification 1: DTW-1-NN

DTW-1-NN was applied to classify skill of the Epidural-40 subset into classes N, I, E. Accuracy of DTW-1-NN was 60%. 1-NN outperformed k-NN with k = 2 to 9. Results were verified by LOOCV and 5 fold CV. Accuracy wasn't affected by length reduction from 5000 to 500, which doesn't largely affect DTW distances [35], [42]. The DTW-k-NN classification was also applied to the full dataset of 271 epidural insertions giving 90.03% accuracy.

4.2 Skill Classification 2: Nearest Centroid

Results from the nearest centroid classifier applied to the 40-epidural subset for skill classification showed that accuracy depends largely on the algorithm used for generating the centroid. Of the 7 prototype algorithms applied, SoftDTW gave highest accuracy (77.5%) for skill classification, results for each are in Table I. The cluster prototypes are shown in Fig. 4. SoftDTW (red), Partition Around Medoids (PAM) (pink), DTW Barycenter Averaging (DBA)(Yellow), Shape Extraction (SE) (cyan), DTW-MP$_I$ (green), DTW-MP$_D$ (navy), Mean (black), and individual time series (grey) in each class (N, I, E).

Table 1 Nearest Centroid - Skill Classification Accuracy

SoftDTW	Mean	DBA	SE	PAM	DTW-MPD	DTW-MPI
77.5%	70%	47.5%	60%	60%	50%	52.5%

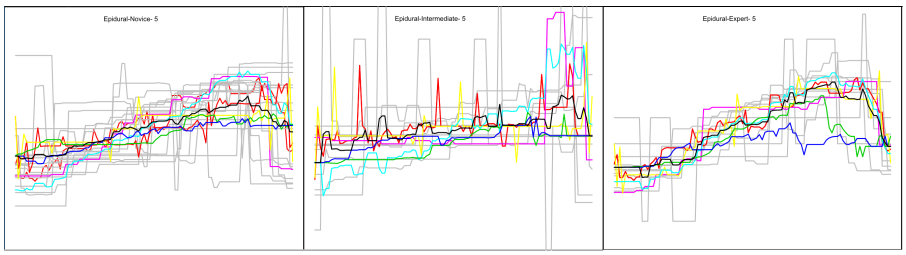

Fig. 4 All centroids - dimension 5 (Pressure) of Epidural-40 left-right: N, I, E

Table 2 Deep Learning - Skill Classification Accuracy

ResNet	FCN	CNN	MCDCNN
85% (60.2%)	75% (82.5%)	72.5% (72.5%)	28.5% (23.6%)

4.3 Skill Classification 3: Deep Learning

Four deep learning techniques for time series classification were applied for skill classification. Time series length of 5000 and 500 were tested and produced similar results. Results are shown in Table II validated with both LOOCV (and 5-fold CV in brackets).

5 Results From Dynamic Assessment

5.1 Clustering epidural insertions

Clustering was applied to the 271 insertions which produced seven clusters, as shown in the dendrogram in Fig. 5. This identifies the optimum separation between clusters and the optimum similarity within clusters according to the Cluster Validity Indices (CVIs). Most of the clusters contain a mixture of skill levels and individuals (Fig. 6). One individual can use several techniques or one technique can be used by several individuals. Clusters may represent numerous valid techniques of performing a good insertion, or common mistakes repeated by different people. For these reasons, when clustering was applied to produce 3 clusters representing each skill level (N, I, E), insertions from each skill level were not all grouped together.

Assessment of cluster validity was performed using seven CVIs including Sil, Dunn, COP, DB, DBStar, SF and CH [2]. The CVIs showed that highest validity was achieved by hierarchical clustering. Generating seven clusters pro-

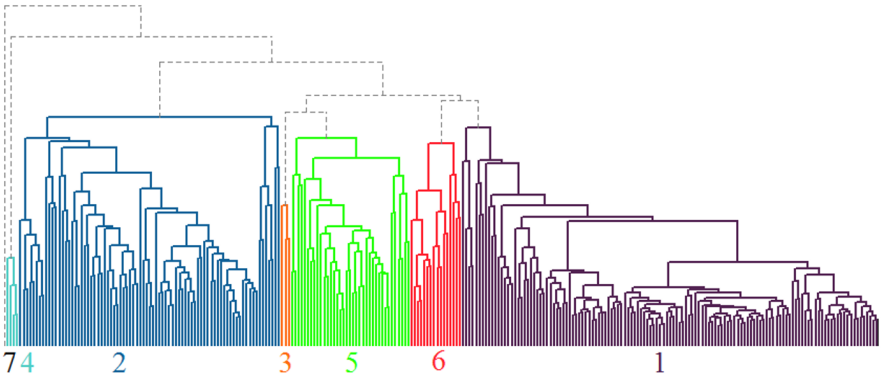

Fig. 5 Dendrogram of hierarchical clustering for 271 time series in 7 clusters

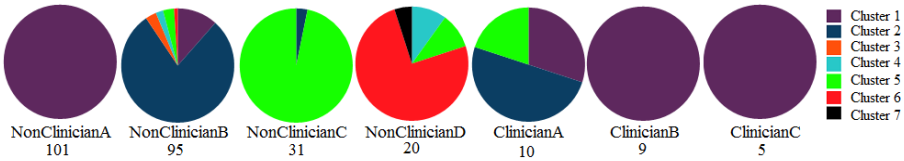

Fig. 6 The proportion of each cluster made up by each individual

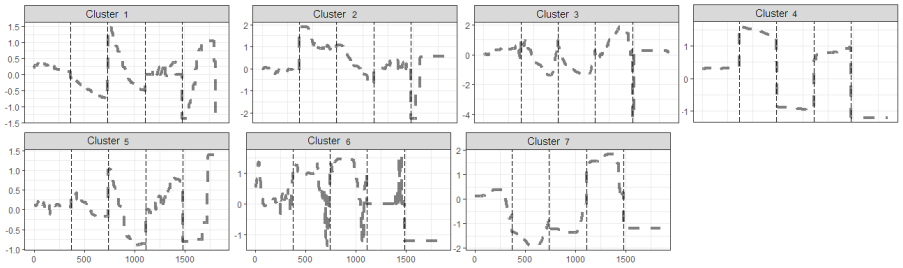

Fig. 7 The prototype of the x axis for each of the 7 clusters found using hierarchical clustering using DTW distance

duced various numbers of insertions in each cluster: 129, 81, 3, 4, 37, 16, 1 with hierarchical clustering or 45, 83, 11, 24, 76, 26, 6 with partitional clustering.

5.2 Prototypes of the 7 clusters

A prototype was generated for each of the 7 clusters identified. Fig. 7 shows a comparison between the 7 prototypes on the x axis. This reveals why certain insertions were clustered separately. Our proposed prototyping method (DTW-MP) was used.

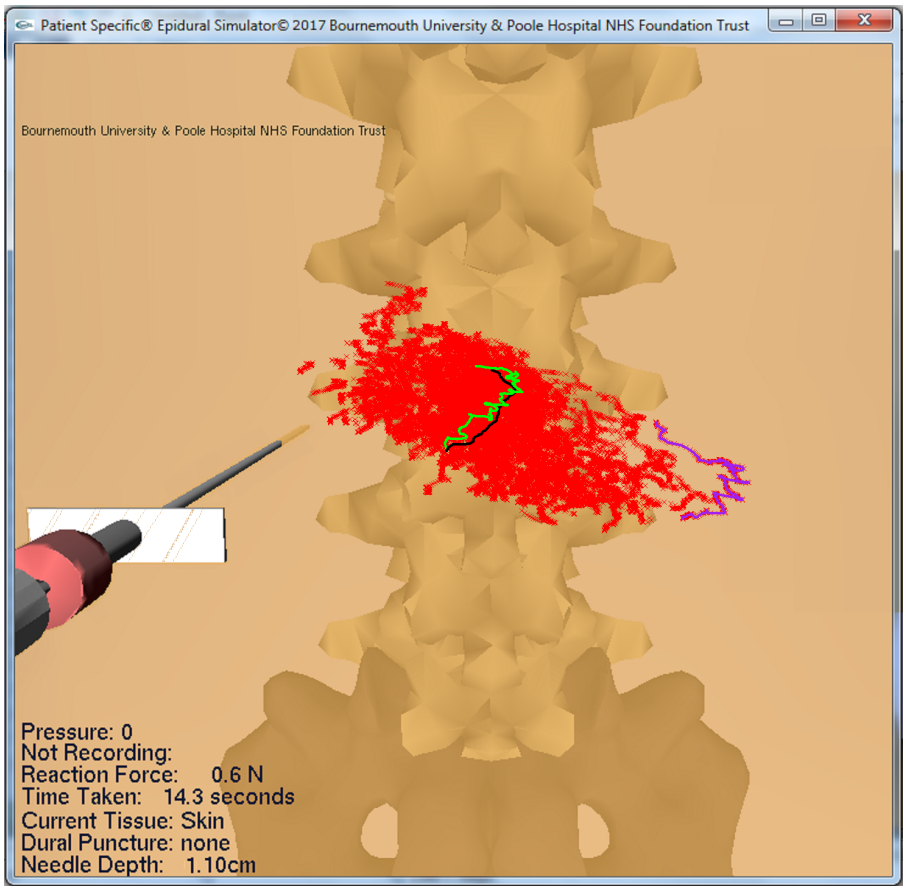

Fig. 8 Plot of all movement points from insertions in cluster 1. Black: The ideal prototypical trajectory. Green: The best insertion matching closely to the prototype. Purple: The worst insertion furthest from the prototype.

5.3 Scoring by distance from cluster prototype

All insertions from cluster 1 are shown in Fig. 8. Highlighted black is the cluster 1 prototype insertion. Highlighted in Purple is the number one discord found in all trajectories which was furthest away from the prototype and highlighted in green is the sequence closest to the prototype, based on DTW with Euclidean distance. The recorded trajectories and centroid was plotted into the VR simulator as shown in Fig 7, showing all trajectories in cluster 1. This is useful for the VR trainee who can visualise their performance in comparison to previous insertions.

Each insertion can be dynamically scored according to the DTW distance from the prototype, generating the cumulative error cost matrix (Fig. 9). Cumulative error is much lower in the best trajectory (left) and higher error in the worst trajectory (right). This graph could be useful for clinicians to visu-

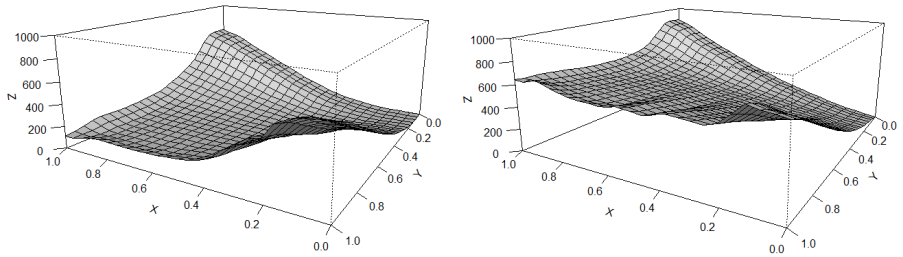

Fig. 9 Cumulative cost matrix of (left) best trajectory (right) worst trajectory.

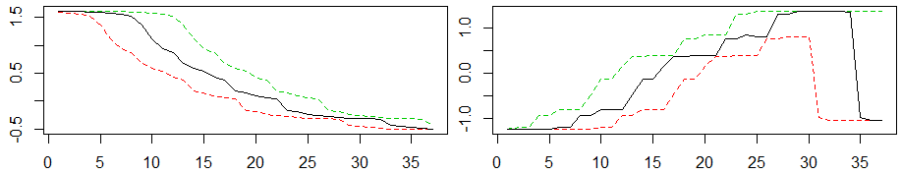

Fig. 10 The upper $\{u_n\}$ (green) and lower $\{l_n\}$ (red) envelopes for (left) z axis and (right) pressure from the prototype of cluster 1 using LB_Keogh lower bounds.

alise which stage of the procedure contained most error. In the cost matrices (Fig. 9), the worst insertion has total cost of 656 whereas the best insertion has lower cost of 112 (shown on z axis in leftmost corner). This indicates that the worst insertion was approximately 5.8 times further from the ideal trajectory.

5.4 Generating envelopes for dynamic score monitoring

The upper and lower bounds were generated from the prototypes of each cluster. LB_Keogh is not compatible with multivariate data, so the upper and lower envelopes $\{u_n\}$ and $\{l_n\}$ were generated for each axis individually (Fig. 10).

The summative-envelopes were computed around the prototypes of each cluster, including the 4 best insertions in each cluster, according to their DTW distance from the cluster prototype. In Fig. 11, $\{su_n\}$ and $\{sl_n\}$ were created by taking the four insertions (solid black lines) which were closest to the prototype of cluster 1. Then LB_Keogh lower bounding algorithm was applied to generate the 4 upper envelopes (dashed red lines) and 4 lower envelopes (dashed green lines). The envelopes were combined by creating the summative-envelope for cluster 1 shown in Fig. 11 as bold lines. Two unseen insertions (solid purple lines) are shown, which both fit within the summative envelope, so they would be classified as good insertions. Where subsections of the insertion require higher accuracy than others, the summative-envelope intuitively produces a thinner tunnel in those areas.

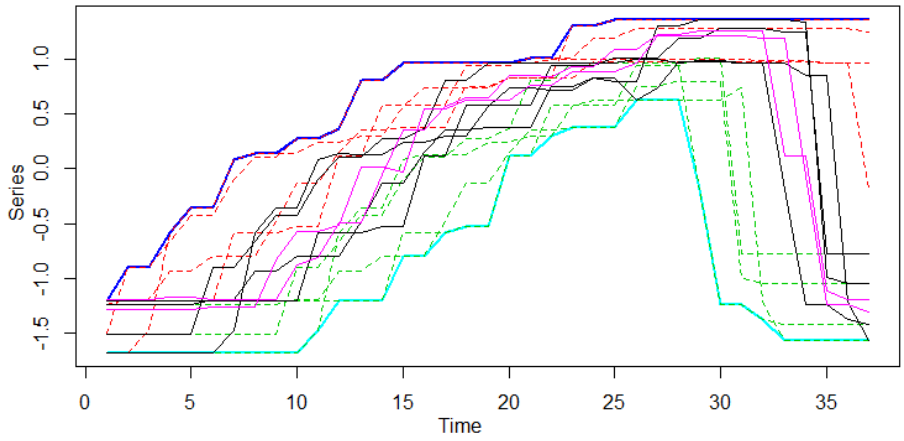

Fig. 11 Proposed Summative-envelope algorithm (bold lines) for syringe pressure around 4 insertions closest to the prototype of cluster 1.

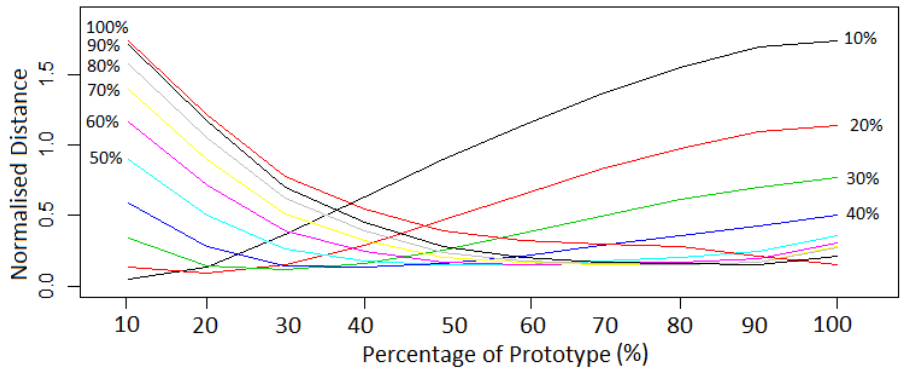

Fig. 12 Dynamic comparison between a partial time series during insertion and partial prototypes of various window length.

5.5 Phase estimation with incomplete time series

The phase estimation algorithm predicted correctly 90% of the time which proportion of the insertion had taken place. Even with a bad insertion, this method can detect that it's a bad insertion even with only the first 10% of the data, as the normalised distance is already much greater than that of the best insertions. During insertion, Fig. 12 shows that the completed part of an incomplete insertion has low normalised distance when compared to a similar percentage of the prototype, but a high distance when compared to a different percentage of the prototype. The percentages within Fig. 12 indicate the current percentage that has been completed of an incomplete insertion.

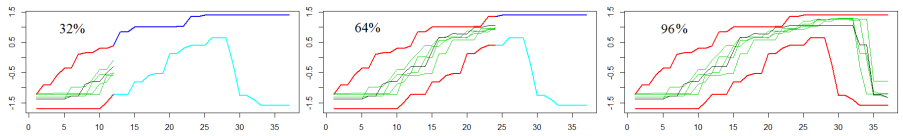

Fig. 13 Dynamic scoring of syringe pressure: partial new unseen time series shown in green, with the corresponding proportion of the Cluster 1 prototype in black. The Cluster 1 summative-envelope upper in dark blue, lower in light blue, turning red at the corresponding proportion.

5.6 Dynamic scoring

The dynamic scoring begins by computing the distance between a new incomplete insertion and equal proportion of the prototype predicted by phase estimation. This results in a dynamic system able to score a partial insertion in real-time during insertion. Fig. 13 shows the dynamic scoring process applied to four new unseen partial insertions (green). All four new insertions stayed within the summative-envelope upper (dark blue) and summative-envelope lower (light blue) but some were closer to the cluster prototype (black) than others. The envelope has differing width in some parts, which represents the variation in accuracy required in some stages of the procedure.

The system can offer dynamic adaptive learning by adding new insertions into the training set. When a new insertion is combined with an existing cluster, it is possible either to re-compute the prototype including the new insertions or combine the new insertion with the existing prototype with weighting. Unusual new insertions can be identified if they are allocated into existing clusters with known bad insertions. These can then be added into the training set as bad insertions.

5.7 Identifying the individual

The DTW-k-NN classifier was applied to identify who performed each insertion. Of the 247 non-clinician insertions 95.4% were classified as non-clinicians. Some individuals were easier to classify than others, NonClinicianA was correctly classified 100% of the time due to high consistency (Fig. 14), whereas ClinicianA only 20%. Overall, the individual was classified correctly in 81.2% of insertions.

6 Discussions

This research proposed three methods for skill classification by analyzing multivariate time series recorded during a VR epidural training simulator procedure. Our data collection of 271 virtual reality epidural procedures included three trained clinicians from the NHS. (1) DTW-1-NN achieved 60% classification accuracy. (2) Nearest centroid classifier SoftDTW achieved 77.5%. (3)

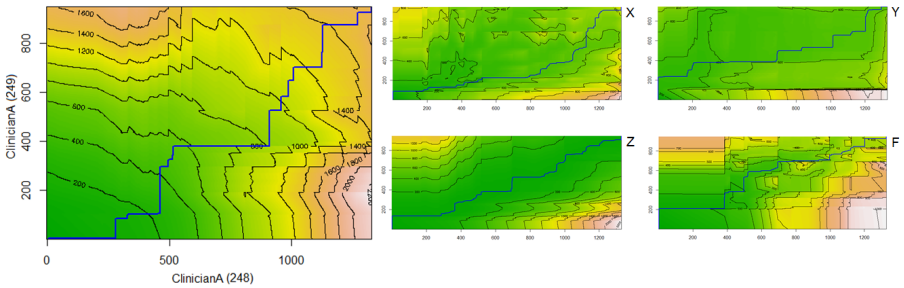

Fig. 14 Left: DTW comparison between two of ClinicianA's insertions. Right: DTW comparison between the same two insertions using univariate data for x, y, z and $force$ dimensions individually.

Within deep learning methods, ResNet achieved 85%, FCN 75%, CNN 72.5% and MCDCNN was inaccurate with 28.5%. Therefore the nearest centroid approach is competitive and only outperformed by the ResNet deep learning architecture. High performance of ResNet matches with previous TSC results applying DL methods (Fawaz et al., 2019). Insights into why SoftDTW was the optimum prototyping method for nearest centroid include that SoftDTW is differentiable and both its value and gradient can be computed with quadratic time/space complexity making it more suited to cluster time series under the DTW geometry. In this case ResNet outperformed SoftDTW leveraging favorable features of our 5-dimensional multivariate time series dataset. Our dataset originated from epidural needle procedures which are relatively short compared to DaVinci surgeries which commonly produce longer, higher dimensional time series and future work could investigate whether SoftDTW or ResNet would perform similarly on those additional datasets. We proposed a new time series prototyping algorithm, DTW-MP which is applied to create a prototype insertion for each cluster. New insertions are classed according to their DTW distance from the cluster prototype. The research developed dynamic methods to assess the score of a virtual reality task while the task is being completed. (1) Clustering is performed to divide the time series training set into groups according to the different techniques. (2) We propose the Summative-envelopes algorithm, which takes the best time series from each cluster including the prototype to create a combined envelope using lower bounds. This can raise alarms in real-time if a time series exits the envelope tunnel. Our experiment showed that DTW-1-NN can recognise which trainee performed a virtual reality task in 81% of the cases. The developed methods enable trainees to view their score, clustering and summative-envelope in real-time during insertion. The summative-envelope reveals which parts of the procedure were abnormal. After reviewing the performance, the trainee's technique can improve by repetitive practice until their motion becomes closer to the cluster prototype representing an expert, improving performance and skill of trainees. Over time, trajectory clustering algorithms can enable the measurement of consistency within a single trainee's performances, and identify

the trainee's improvement of consistency over time. Future work can use these results for several purposes including: (i) Adaptation and automation of VR training based on the recorded data to customise VR training for individual requirements. (ii) Detecting the type of motion or hand gestures using classification. (iii) Recognising actions which increase the risk of injury to raise an alert. In future, the developed methods could be applied to in-vivo data, tracking devices or cameras monitoring surgeons with hospital patients as well as being applied to trajectories from VR training simulators.

7 Ethics

The Bournemouth University Ethics service has reviewed the study plan prior to initiation. The dataset does not contain identifying information such as names or personal information.

Acknowledgements This project was supported by the Royal Academy of Engineering (RAEng) under the Research Fellowship scheme awarded to Professor Neil Vaughan, also support from University of Exeter, University of Technology Sydney and Bournemouth University during the time of the research.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. Allen, B., Nistor, V., Dutson, E., Carman, G., Lewis, C., Faloutsos, P.: Support vector machines improve the accuracy of evaluation for the performance of laparoscopic training tasks. Surgical Endoscopy **11**(1), 170 (2009)
2. Arbelaitz, O., Gurrutxaga, I., Muguerza, J., PéRez, J.M., Perona, I.: An extensive comparative study of cluster validity indices. Pattern Recognition **46**(1), 243–256 (2013)
3. Berndt, D.J., Clifford, J.: Using dynamic time warping to find patterns in time series. In: KDD workshop, vol. 10(16), pp. 359–370. Seattle, WA (1994)
4. Bhattacharyya, A.: On a measure of divergence between two statistical populations defined by their probability distributions. Bull. Calcutta Math. Soc. **35**, 99–109 (1943)
5. Datta, V., Mackay, S., Mandalia, M., Darzi, A.: The use of electromagnetic motion tracking analysis to objectively measure open surgical skill in the laboratory-based model. Journal of the American College of Surgeons **193**(5), 479–485 (2001)
6. Fawaz, H.I., Forestier, G., Weber, J., Idoumghar, L., Muller, P.A.: Deep learning for time series classification: a review. Data Mining and Knowledge Discovery **33**(4), 917–963 (2019)
7. Forestier, G., Petitjean, F., Senin, P., Despinoy, F., Huaulmé, A., Fawaz, H.I., Weber, J., Idoumghar, L., Muller, P.A., Jannin, P.: Surgical motion analysis using discriminative interpretable patterns. Artificial intelligence in medicine **91**, 3–11 (2018)
8. Gao, Y., Vedula, S.S., Reiley, C.E., Ahmidi, N., Varadarajan, B., Lin, H.C., Tao, L., Zappella, L., Béjar, B., Yuh, D.D., et al.: Jhu-isi gesture and skill assessment working set (jigsaws): A surgical activity dataset for human motion modeling. In: MICCAI Workshop: M2CAI, vol. 3, p. 3 (2014)

9. Gupta, S., Collis, R., Harries, S.: Increasing dural tap rate: is this a national trend. Int J Obstet Anesth **16**, S17 (2007)
10. Gusfield, D.: Algorithms on stings, trees, and sequences: Computer science and computational biology. Acm Sigact News **28**(4), 41–60 (1997)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778 (2016)
12. Huang, J., Payandeh, S., Doris, P., Hajshirmohammadi, I.: Fuzzy classification: towards evaluating performance on a surgical simulator. Studies in health technology and informatics **111**, 194–200 (2005)
13. Izakian, H., Pedrycz, W., Jamal, I.: Fuzzy clustering of time series data using dynamic time warping distance. Engineering Applications of Artificial Intelligence **39**, 235–244 (2015)
14. Judkins, T.N., Oleynikov, D., Stergiou, N.: Objective evaluation of expert and novice performance during robotic surgical training tasks. Surgical endoscopy **23**(3), 590 (2009)
15. Karim, F., Majumdar, S., Darabi, H., Chen, S.: Lstm fully convolutional networks for time series classification. IEEE access **6**, 1662–1669 (2017)
16. Keogh, E., Lin, J., Fu, A.: Hot sax: Efficiently finding the most unusual time series subsequence. In: Fifth IEEE International Conference on Data Mining (ICDM'05), pp. 8–pp. Ieee (2005)
17. Keogh, E., Ratanamahatana, C.A.: Exact indexing of dynamic time warping. Knowledge and information systems **7**(3), 358–386 (2005)
18. Khooshabeh, P., Choromanski, I., Neubauer, C., Krum, D.M., Spicer, R., Campbell, J.: Mixed reality training for tank platoon leader communication skills. In: 2017 IEEE Virtual Reality (VR), pp. 333–334. IEEE (2017)
19. Kim, S.W., Park, S., Chu, W.W.: An index-based approach for similarity search supporting time warping in large sequence databases. In: Proceedings 17th International Conference on Data Engineering, pp. 607–614. IEEE (2001)
20. Lance, G.N., Williams, W.T.: Computer programs for hierarchical polythetic classification ("similarity analyses"). The Computer Journal **9**(1), 60–64 (1966)
21. Laptev, N., Amizadeh, S., Flint, I.: Generic and scalable framework for automated time-series anomaly detection. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1939–1947 (2015)
22. Lemire, D.: Faster retrieval with a two-pass dynamic-time-warping lower bound. Pattern recognition **42**(9), 2169–2180 (2009)
23. Lemke, C., Gabrys, B.: Meta-learning for time series forecasting and forecast combination. Neurocomputing **73**(10-12), 2006–2016 (2010)
24. Lendvay, T.S., Brand, T.C., White, L., Kowalewski, T., Jonnadula, S., Mercer, L.D., Khorsand, D., Andros, J., Hannaford, B., Satava, R.M.: Virtual reality robotic surgery warm-up improves task performance in a dry laboratory environment: a prospective randomized controlled study. Journal of the American College of Surgeons **216**(6), 1181–1192 (2013)
25. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440 (2015)
26. Mahalanobis, P.C.: On test and measures of group divergence: theoretical formulae. Journal and Proceedings of Asiatic Society of Bengal **26**, 541–588 (1930)
27. Malhotra, P., TV, V., Vig, L., Agarwal, P., Shroff, G.: Timenet: Pre-trained deep recurrent neural network for time series classification. arXiv preprint arXiv:1706.08838 (2017)
28. Montero, P., Vilar, J.A., et al.: Tsclust: An r package for time series clustering. Journal of Statistical Software **62**(1), 1–43 (2014)
29. Naftel, A., Khalid, S.: Classifying spatiotemporal object trajectories using unsupervised learning in the coefficient feature space. Multimedia Systems **12**(3), 227–238 (2006)
30. Oropesa, I., Sánchez-González, P., Chmarra, M.K., Lamata, P., Pérez-Rodríguez, R., Jansen, F.W., Dankelman, J., Gómez, E.J.: Supervised classification of psychomotor competence in minimally invasive surgery based on instruments motion analysis. Surgical endoscopy **28**(2), 657–670 (2014)

31. Paparrizos, J., Gravano, L.: k-shape: Efficient and accurate clustering of time series. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, pp. 1855–1870 (2015)
32. Petitjean, F., Ketterlin, A., Gançarski, P.: A global averaging method for dynamic time warping, with applications to clustering. Pattern Recognition **44**(3), 678–693 (2011)
33. Pokrajac, D., Lazarevic, A., Latecki, L.J.: Incremental local outlier detection for data streams. In: 2007 IEEE symposium on computational intelligence and data mining, pp. 504–515. IEEE (2007)
34. Rakthanmanon, T., Campana, B., Mueen, A., Batista, G., Westover, B., Zhu, Q., Zakaria, J., Keogh, E.: Searching and mining trillions of time series subsequences under dynamic time warping. In: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 262–270 (2012)
35. Ratanamahatana, C.A., Keogh, E.: Everything you know about dynamic time warping is wrong. In: Third workshop on mining temporal and sequential data, vol. 32. Citeseer (2004)
36. Ruta, D., Gabrys, B., Lemke, C.: A generic multilevel architecture for time series prediction. IEEE Transactions on Knowledge and Data Engineering **23**(3), 350–359 (2010)
37. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. IEEE transactions on acoustics, speech, and signal processing **26**(1), 43–49 (1978)
38. Tao, L., Elhamifar, E., Khudanpur, S., Hager, G.D., Vidal, R.: Sparse hidden markov models for surgical gesture classification and skill evaluation. In: International conference on information processing in computer-assisted interventions, pp. 167–177. Springer (2012)
39. Vaughan, N., Dubey, V.N., Wainwright, T.W., Middleton, R.G.: Does virtual-reality training on orthopaedic simulators improve performance in the operating room? In: 2015 Science and Information Conference (SAI), pp. 51–54. IEEE (2015)
40. Vaughan, N., Dubey, V.N., Wee, M.Y., Isaacs, R.: Epidural pressure measurements from various bmi obstetric patients. Journal of Medical Devices **8**(3) (2014)
41. Vaughan, N., Dubey, V.N., Wee, M.Y., Isaacs, R.: Parametric model of human body shape and ligaments for patient-specific epidural simulation. Artificial intelligence in medicine **62**(2), 129–140 (2014)
42. Vaughan, N., Gabrys, B.: Comparing and combining time series trajectories using dynamic time warping. Procedia Computer Science **96**, 465–474 (2016)
43. Watson, R.A.: Use of a machine learning algorithm to classify expertise: Analysis of hand motion patterns during a simulated surgical task. Academic Medicine **89**(8), 1163–1167 (2014)
44. Williams, N.: The implementation of the working time directive and its impact on the nhs and health professionals, taskforce, independent working time regulations. London: The Royal College of Surgeons of England (2014)
45. Yamauchi, Y., Yamashita, J., Morikawa, O., Hashimoto, R., Mochimaru, M., Fukui, Y., Uno, H., Yokoyama, K.: Surgical skill evaluation by force data for endoscopic sinus surgery training system. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 44–51. Springer (2002)
46. Yankov, D., Keogh, E., Rebbapragada, U.: Disk aware discord discovery: finding unusual time series in terabyte sized datasets. Knowledge and Information Systems **17**(2), 241–262 (2008)
47. Yavrucuk, I., Kubali, E., Tarimci, O.: A low cost flight simulator using virtual reality tools. IEEE Aerospace and Electronic Systems Magazine **26**(4), 10–14 (2011)
48. Yi, B.K., Jagadish, H.V., Faloutsos, C.: Efficient retrieval of similar time sequences under time warping. In: Proceedings 14th International Conference on Data Engineering, pp. 201–208. IEEE (1998)
49. Zhao, B., Lu, H., Chen, S., Liu, J., Wu, D.: Convolutional neural networks for time series classification. Journal of Systems Engineering and Electronics **28**(1), 162–169 (2017)
50. Zheng, Y., Liu, Q., Chen, E., Ge, Y., Zhao, J.L.: Exploiting multi-channels deep convolutional neural networks for multivariate time series classification. Frontiers of Computer Science **10**(1), 96–112 (2016)

Please cite this article as: N. Vaughan and B. Gabrys, Scoring and assessment in medical VR training simulators with dynamic time series classification. Engineering Applications of Artificial Intelligence (2020) 103760, https://doi.org/10.1016/j.engappai.2020.103760