

Strengths and Weaknesses of Recurrent Quantification Analysis in the context of Human-Humanoid Interaction

Miguel Xochicale^{1*}, Chirs Baber¹,

1 School of Engineering, University of Birmingham, Birmingham, UK

* map479@bham.ac.uk

Abstract

Movement variability is defined as the variations that occur in motor performance across multiple repetitions of a task and such behaviour is an inherent feature within and between each persons' movement. Quantifying movement variability is still an open problem, particularly when traditional methods in time domain and frequency domain fail to detect tiny modulations in frequency or phase for time series. We therefore consider methodologies from nonlinear dynamics such as reconstructed state space (RSS), uniform time-delay embedding (UTDE), recurrence plots (RPs) and recurrence quantification analysis (RQA) metrics. These algorithms require time series measured with sensors that provide well sampled data with little noise. However, for this study, we are interested in the analysis of data collected through wearable inertial sensors (IMUs) and its post-processing effects in each of the nonlinear dynamics methodologies. Twenty right-handed healthy participants (mean and standard deviation (SD) age of mean=19.8 (SD=1.39)) performed an experiment in human-humanoid imitation activities (H-H-I). Then IMUs were attached to the participants and the humanoid robot performed simple vertical and horizontal arm movements in normal and faster speed. With four window lengths and three levels of smoothed time series, we found visual differences in the patterns of RSSs and RPs. Then using metrics of RQA, we find out that the type of movements and the level of smoothness affects those metrics. In particular, we found that entropy values from RQA were well distributed and presenting variation in all the conditions for time series. Hence, we demonstrated the potential of nonlinear techniques to quantify human movement variability in the context of H-H-I can enhance the development of better diagnostic tools for various applications rehabilitation, sport science or for new forms of human-robot interaction.

Movement variability is defined as the variations that occur in motor performance across multiple repetitions of a task and such behaviour is an inherent feature within and between each persons' movement. Quantifying such movement variability is still an open problem, particularly when traditional methods in time domain and frequency domain fail to detect tiny modulations in frequency or phase for time series. For this work, we therefore consider methodologies from nonlinear dynamics such as reconstructed state space (RSS), uniform time-delay embedding (UTDE), recurrence plots (RPs) and recurrence quantification analysis (RQA) metrics (REC, DET, RATIO, and ENT). These algorithms generally require time series measured with sensors that provide well sampled data with little noise. However, for this study, we are interested in the analysis of data collected through wearable inertial sensors and its post-processing effects in each of the nonlinear dynamics methodologies. With that in mind, twenty right-handed healthy participants (mean and standard deviation (SD) age of mean=19.8 (SD=1.39)) performed an experiment in the context of human-humanoid imitation

activities. Then inertial sensors were attached to the participants and the humanoid robot that performed simple vertical and horizontal arm movements in normal and faster speed. With four window lengths and three levels of normalised smoothed of time series, we evidently found visual differences in the patterns of RSSs and RPs. Then using metrics of RQA, we find out that the type of movement (vertical or horizontal) and the level of smoothness affects those metrics. In particular, we found that ENT values were well distributed and presenting variation in all the conditions for time series which happens partially in remained metrics. We demonstrated the potential of nonlinear techniques to quantify human movement variability in the context of human-humanoid imitation activities can enhance the development of better diagnostic tools for various applications rehabilitation, sport science or for new forms of human-robot interaction.

Introduction

Human movement is a complex system where not only multiple joints and limbs are involved for a specific task in a determined environment but also how we process the external information with all of our available senses and use our experiences, which play a crucial role in the way each person moves. Recent studies in human motion recognition have revealed the possibility to estimate features from lower dimensions signals to distinguish differences between styles of pedalling motion [1, 2], to perform gait identification [3, 4] or to do pattern recognition from biological signals [5]. The lower dimension signals from biological signals are time series of 1–dimension in \mathbb{R} which commonly have high nonlinearity, complexity, and non-stationarity [5], where methods of nonlinear dynamics can objectively quantify such human movement variability [1–7]. With this in mind, [8] reviewed methods for nonlinear time series analysis based on the appropriate estimation of the embedding parameters (m embedding dimension and τ embedding delay) to reconstruct the state space, where an n -dimensional reconstructed state space using 1–dimensional time series, can preserve the topological properties of an unknown M -dimensional state space [9]. Also [8] reviewed the use of Recurrence Plots (RP), a graphical representation of a two-dimensional map which show black/white dots as recurrences in a given n -dimensional system, and Recurrence Quantification Analysis (RQA), metrics that can pick out important directions and statistics in RPs. RPs and RQA help to have a more intuitive meaning of the time series, for instance, RQA is quantitatively and qualitatively independent of embedding dimension which is also verified experimentally [10]. However, the estimation of embedding parameters and finding the right parameters to perform RQA is still an open problem. For instance, [8] pointed out that there is no general technique that can be used to compute the embedding parameters since the time series are system-dependent which means that once computing the values for embedding parameters, these may only work for one purpose (e.g., prediction) and may not work well for another purpose (e.g., computing dynamical invariants). Additionally, the methodologies of nonlinear dynamics for computing the embedding parameters e.g., autocorrelation, mutual information, and nearest neighbour require data which is well sampled and with little noise [11] or require purely deterministic signals [12]. Similarly, these methodologies for computing the embedding parameters can break down with real-world datasets which have generally different length, different values of accuracy and precision (rounding errors due to finite precision of the measurement apparatus which include frequency acquisition [4]), and data may be contaminated with different or unknown sources of noise [11]. It is surprising that even with the previous constraints with regard to the quality of data, and the problem with the estimation of embedding parameters, the results of analysis using nonlinear dynamics have proven to be helpful to understand and characterise time series [1–5, 5–8]. Another point to consider with time series

analysis using nonlinear dynamics is the appropriate use of different post-processing techniques such as interpolation, filtering or normalisation.

With that in mind, it can be said that there is little research and understanding on the effects for post-processing techniques in the interpretation of reconstructed state spaces, RPs and RQA. For this work, we are interested in investigating and exploring the effects of different features of time series (e.g. levels of smoothness, types of movements with different velocities and window length) to estimate appropriate embedded parameters for uniform time-delay embedding and for the computation for recurrence plots and recurrence quantification analysis. To explore those questions, we conducted an experiment with twenty right-handed healthy participants in the context of human-humanoid imitation activities where participants were asked to imitate simple arm movements performed by a humanoid robot. Therefore, the primary aim of the paper is to explore the following questions:

- What are the effects on the three nonlinear methods (uniform time-delay embedding, RPs, and RQA), for different embedding parameters, different recurrence thresholds and different characteristics of time series (window length, smoothness of the signal, structure)?

Materials and methods

State Space Reconstruction

The method of state space reconstruction was originally proposed by [13] and formalised by [9]. Since then, different investigations and disciplines have benefited from the use of the method of state space reconstruction [2–4, 7, 14]. The method of state space reconstruction is based on uniform time-delay embedding methodology which is a simple matrix implementation that can reconstruct an unknown d -dimensional manifold M from a scalar time series (e.g. one-dimensional time series in \mathbb{R}). A manifold, in this context, is a multidimensional curved surface within a space (e.g. a saddle) [15].

The use of a scalar time series is the main advantage of the method of state space reconstruction which in essence preserve dynamic invariants such as correlation dimension, fractal dimension, Lyaponov exponents, Kolmogorov-Sinai entropy and detrended fluctuation analysis [1, 2, 8, 16, 17]. However, selecting appropriate embedding parameters which are used to apply the state space reconstruction is still an open challenge for which we present introductions for the methodologies to compute such embedding parameters. With that in mind, in the following subsections, we describe in more detail the state space reconstruction theorem (RSSs), uniform time-delay embedding theorem (UTDE), false nearest neighbours (FNN), average mutual information (AMI) and other methodologies for state space reconstruction.

State Space Reconstruction Theorem

Following the notation employed in [9, 11, 18–21], state space reconstruction is defined by:

$$s(t) = f^t[s(0)], \quad (1)$$

where $s, s : A \rightarrow M$ given that $A \subseteq \mathbb{R}$ and $M \subseteq \mathbb{R}^d$, represents a trajectory which evolves in an unknown d -dimensional manifold M , $f : M \rightarrow M$ is an evolution function and f^t , with time evolution $t \in \mathbb{N}$, is the t -th iteration of f that corresponds to an initial position $s(0) \in M$ [9]. Then, a point of a one-dimensional time series $x(t)$ in \mathbb{R} , can be obtained with

$$x(t) = h[s(t)], \quad (2)$$

where h is a function, $h : M \rightarrow \mathbb{R}$, defined on the trajectory $s(t)$. Reconstructed state space can then be described as an n -dimensional state space defined by $y(t) = \Psi[\mathbf{X}(t)]$ where $\mathbf{X}(t) = \{x(t), x(t - \tau), \dots, x(t - (m - 1)\tau)\}$ is the uniform time-delay embedding with a dimension embedding m and delay embedding τ and $\Psi : \mathbb{R}^m \rightarrow \mathbb{R}^n$ is a further transformation of dimensionality (e.g. Principal Component Analysis, Singular Value Decomposition, etc) being $n \leq m$. With that in mind, uniform time-delay embedding, $\mathbf{X}(t)$, defines a map $\Phi : M \rightarrow \mathbb{R}^m$ such that $\mathbf{X}(t) = \Phi(s(t))$, where Φ is a diffeomorphic map [9] whenever $\tau > 0$ and $m > 2d_{box}$ and d_{box} is the box-counting dimension of M [11]. Then, if Φ is an embedding of evolving trajectories in the reconstructed space then a composition of functions represented with F^t is induced on the reconstructed state space determined:

$$\mathbf{X}(t) = F^t[\mathbf{X}(0)] = \Phi \circ f^t \circ \Phi^{-1}[\mathbf{X}(0)]. \quad (3)$$

With this in mind, an embedding is defined as "a smooth one-to-one coordinate transformation with a smooth inverse" and the total reconstruction map is defined as $\Xi = \Psi \circ \Phi$ [18]. Fig 1 illustrates the state space reconstruction.

Uniform Time-Delay Embedding (UTDE)

Frank et al. and Sama et al. refer to the state space reconstruction as "time-delay embeddings" or "delay coordinates" [3, 4]. However, we consider the term "uniform time-delay embedding" as more descriptive and appropriate terminology for our work.

The uniform time-delay embedding is represented as a matrix of uniform delayed copies of the time series $\{\mathbf{x}_n\}_{n=1}^N$ where N is the sample length of $\{\mathbf{x}_n\}$ and n is index for the samples of $\{\mathbf{x}_n\}$. $\{\mathbf{x}_n\}_{n=1}^N$ has a sample rate of T . The delayed copies of $\{\mathbf{x}_n\}$ are uniformly separated by τ and represented as $\{\tilde{\mathbf{x}}_{n-i\tau}\}$ where i goes from $0, 1, \dots, (m - 1)$ (Fig 2). Generally speaking, $\{\tilde{\mathbf{x}}_{n-i\tau}\}$ contains information of unobserved state variables and encapsulates the information of the delayed copies of the available time series in the uniform time-delay embedding matrix \mathbf{X}_τ^m , $\mathbf{X}_\tau^m \in \mathbb{R}^{m \times N}$, defined as

$$\mathbf{X}_\tau^m = \begin{pmatrix} \tilde{\mathbf{x}}_n \\ \tilde{\mathbf{x}}_{n-\tau} \\ \tilde{\mathbf{x}}_{n-2\tau} \\ \vdots \\ \tilde{\mathbf{x}}_{n-(m-1)\tau} \end{pmatrix}^\top, \quad (4)$$

where m is the embedding dimension, τ is the embedding delay and $^\top$ denotes the transpose. m and τ are known as embedding parameters. The matrix dimension of \mathbf{X}_τ^m is defined by $N - (m - 1)\tau$ rows and m columns and $N - (m - 1)\tau$ defines the length of each delayed copy of $\{\tilde{\mathbf{x}}_n\}$ in \mathbf{X}_τ^m . For further details and explicit examples of uniform time-delay embedding methodology, we refer the reader to the S1 Appendix A.

Estimation of Embedding Parameters

The estimation of the embedding parameters (m and τ) is a fundamental step for the state space reconstruction with the use of uniform time-delay embedding method. With this in mind, we review two of the most common algorithms, which will be used in our work, to compute the embedding parameters: the false nearest neighbour (FNN) and the average mutual information (AMI).

False Nearest Neighbours

To select the minimum embedding dimension m_0 , Kennel et al. [23] used the method of false neighbours which can be understood as follows: on the one hand, when the

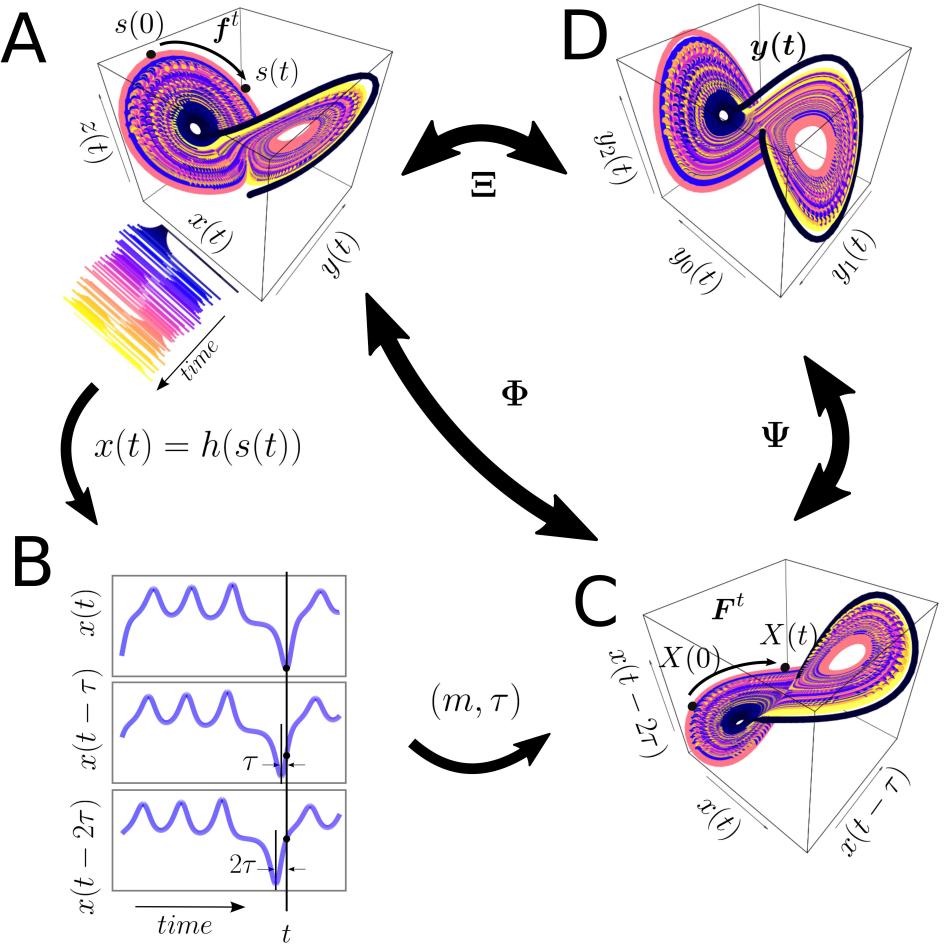


Fig 1. State space reconstruction methodology. State space reconstruction is based on $x(t) = h[s(t)] = h[f^t[s(0)]]$ where f^t is the true dynamical system, $s(t)$ indicates the state, s , at time, t , and $h[]$ the measurement function. The time-delay embedding represented as the Φ , maps the original d -dimensional state $s(t)$ into the m -dimensional uniform time-delay embedding matrix $\mathbf{X}(t)$. The transformation map Ψ then maps $\mathbf{X}(t)$ into a new state $y(t)$ of dimensions $n < m$. (A) M -dimensional manifold representing the state space (e.g. Lorenz system); (B) Delayed copies of 1-dimensional $x(t)$ from the Lorenz system; (C) m -dimensional reconstructed state space with m and τ , and (D) $y(t)$ is the n -dimensional reconstructed state space. The total reconstruction map is represented as $\Xi = \Psi \circ \Phi$ where Φ is the delay reconstruction map and Ψ is the coordinate transformation map. This figure is adapted from [1, 18, 20] and R code to reproduce it is available from [22].

embedding dimension is too small to unfold the attractor "not all points that lie close each other will be neighbours and some points appear as neighbours because of the attractor has been projected down into a smaller space", on the other hand, when increasing the embedding dimension "points that are near to each other in the sufficient embedding dimension should remain close as the dimension increase from m to $m + 1$ [17]". From a mathematical point of view, the state space reconstruction theorem is done when the attractor is unfolded with either the minimum embedding dimension,

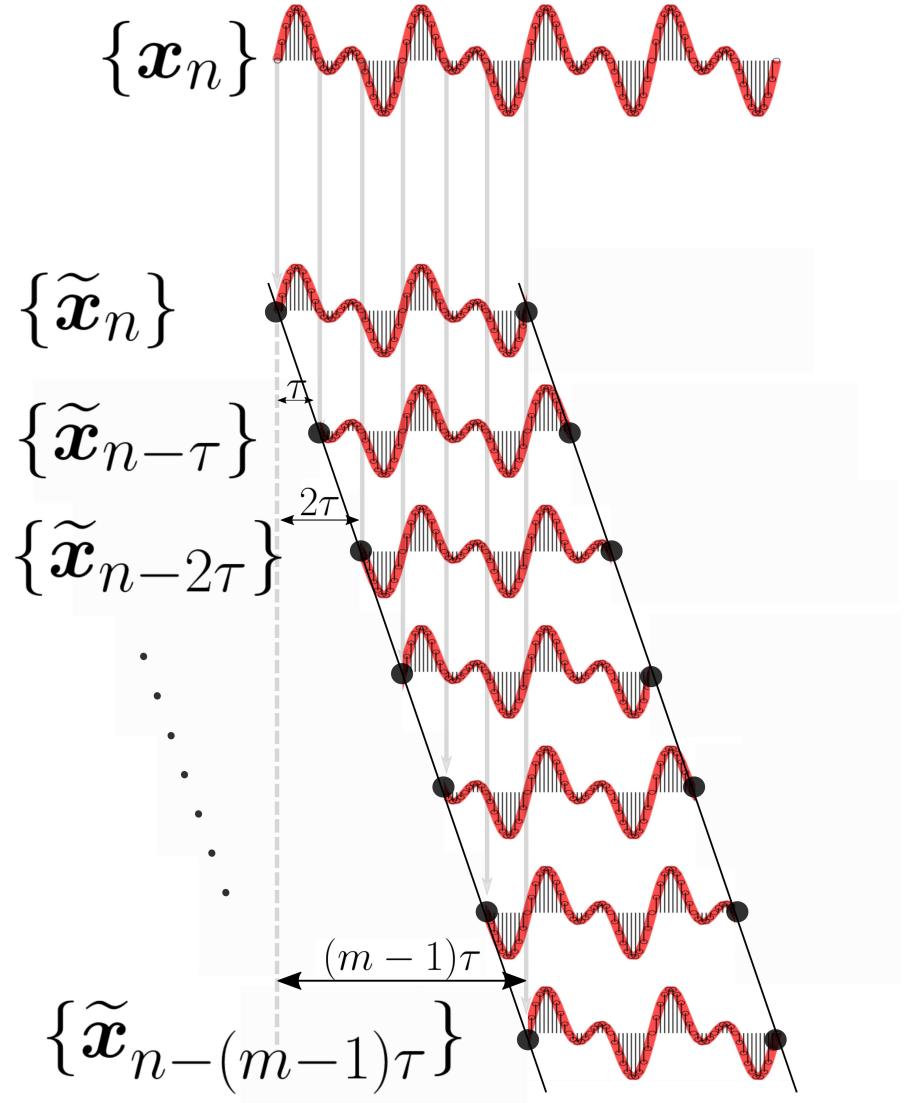


Fig 2. Uniform time-delay embedding. UTDE is illustrated as $m - 1$ delayed copies of $\{x_n\}$, uniformly separated by τ and represented as $\{\tilde{x}_n, \dots, \tilde{x}_{n-(m-1)\tau}\}$ (Eq. 4). R code to reproduce the figure is available from [22].

m_0 , or any other embedding dimension value where $m \geq m_0$ [23]. On the contrary, any large value of m_0 leads to excessive computations [8]. With this in mind, Cao [24] proposed an algorithm based on the false neighbour method where only the time-series and one delay embedding value are necessary to select the minimum embedding dimension. Cao's algorithm is based on $E(m)$ which is the mean value of all $a(i, m)$,

both defined as follows

$$\begin{aligned} E(m) &= \frac{1}{N - m\tau} \sum_{i=1}^{N-m\tau} a(i, m) \\ &= \frac{1}{N - m\tau} \sum_{i=1}^{N-m\tau} \frac{\|\mathbf{X}_i(m+1) - \mathbf{X}_{n(i,m)}(m+1)\|}{\|\mathbf{X}_i(m) - \mathbf{X}_{n(i,m)}(m)\|} \end{aligned} \quad (5)$$

where $\mathbf{X}_i(m)$ and $\mathbf{X}_{n(i,m)}(m)$ are the time-delay embeddings with $i = 1, 2, \dots, N - (m-1)\tau$ and $n(i, m) = 1 \leq n(i, m) \leq N - m\tau$. From Eq. 5, it can be seen that $E(m)$ is only dependent on m and τ for which $E_1(m)$ is defined as

$$E_1(m) = \frac{E(m+1)}{E(m)}. \quad (6)$$

$E_1(m)$ is therefore considered to investigate the variation from m to $m+1$ in order to find the minimum embedding dimension m_0 (Eq. 6). As described in [24]: " $E_1(m)$ stops changing when m is greater than some m_0 , if the time series comes from a multidimensional state space then $m_0 + 1$ is the minimum dimension". Additionally, Cao proposed $E_2(m)$ to distinguish deterministic signals from stochastic signals. $E_2(m)$ is defined as

$$E_2(m) = \frac{E^*(m+1)}{E^*(m)}, \quad (7)$$

where

$$E^*(m) = \frac{1}{N - m\tau} \sum_{i=1}^{N-m\tau} |\mathbf{X}_i(m+1) - \mathbf{X}_{n(i,m)}(m+1)|. \quad (8)$$

For instance, when the signal comes from random noise (values that are independent from each other), all $E_2(m)$ values are approximately equal to 1 (e.g. $E_2(m) \approx 1$). However, for deterministic data $E_2(m)$ is not constant for all m (e.g. $E_2(m) \neq 1$).

As an example of the use of $E_1(m)$ and $E_2(m)$ values, we consider two time series: the solution for the x variable of the Lorenz system (Fig 3E), and a Gaussian noise time series with zero mean and a variance of one (Fig 3F). We then compute $E_1(m)$ and $E_2(m)$ values for each time series. The $E_1(m)$ values for the chaotic time series appear to be constant after the dimension is equal to six. The determination of six is given that any value of m can be used as they are within the threshold of 1 ± 0.05 (Fig 3A). $E_2(m)$ values, for chaotic time series, are different to one (Fig 3C), for which, it can be concluded that for the chaotic time series the minimum embedding dimension the time series comes from a deterministic signal. With regard to the noise time series, $E_1(m)$ values appeared to be constant when m is close to thirteen, which is defined by the threshold of 1 ± 0.05 (Fig 3B). $E_1(m)$ values then indicate the minimum embedding dimension of the noisy time series is thirteen, however all of the $E_2(m)$ values are approximately equal to one (Figure 3D) for which it can be concluded that noise time series is a stochastic signal.

It is important to note that for this work not only $E_1(m)$ and $E_2(m)$ are computed but also a variation of τ from 1 to 20 is presented. The purpose of such variation for τ is to show its independence with regard to $E_1(m)$ and $E_2(m)$ values as τ is increasing (Fig 3A,B,C, and D) However, one negative of the Cao's algorithm [24] is the definition of a new threshold where m values appear to be constant in $E_1(m)$. In the case of the given examples and reported results, we defined such threshold as 0.05. Further investigation is required for the selection of the threshold in the $E_1(m)$, as the selection of the threshold in this work is base on no particular method but visual inspection.

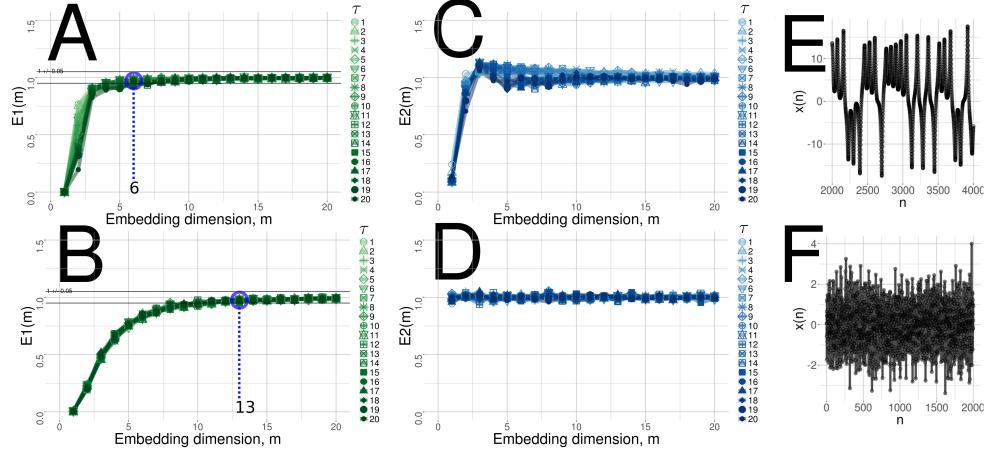


Fig 3. Minimum dimension embedding values with Cao’s method. (A, B) $E_1(m)$ values and (C, D) $E_2(m)$ values with variations of τ values from one to twenty for (E) chaotic and (F) random time series. R code to reproduce the figure is available from [22].

Average Mutual Information

When selecting the delay dimension parameter, τ , one can consider the following two cases: (i) when τ is too small, the elements of time-delay embedding will be along the bisectrix of the phase space and the reconstruction is generally not satisfactory, (ii) on the contrary, when τ is too large the elements of the uniform time-delay embedding will become spread and uncorrelated which makes recovering the underlying attractor more difficult if not impossible [18, 25, 26]. With regard to the algorithms to compute τ , Emrani et al. [25], for instance, used the autocorrelation function in which the first zero crossing is considered as the minimum delay embedding parameter. However, the autocorrelation function is a linear statistic over which the Average Mutual Information (AMI) algorithm is preferred because the AMI takes into account the nonlinear dynamical correlations [17, 27]. With this in mind, the AMI algorithm is described below to estimate the minimum delay embedding parameter, τ_0 .

To compute the AMI, an histogram of $x(n)$ using n bins is calculated and then a probability distribution of data is computed [12]. AMI is therefore denoted by $I(\tau)$ which is the average mutual information between the original time series, $x(n)$, and the delayed time series, $x(n - \tau)$, delayed by τ [28]. AMI is defined by

$$I(\tau) = \sum_{i,j}^N p_{ij} \log_2 \frac{p_{ij}}{p_i p_j}. \quad (9)$$

Probabilities are defined as follows: p_i is the probability that $x(n)$ has a value inside the i -th bin of the histogram, p_j is the probability that $x(n + \tau)$ has a value inside the j -th bin of the histogram and $p_{ij}(\tau)$ the probability that $x(n)$ is in bin i and $x(n + \tau)$ is in bin j . The AMI is measured in bits (base 2, also called shannons) [12, 29]. For small τ , AMI will be large and it will then decrease more or less rapidly. As τ increase and goes to a large limit, $x(n)$ and $x(n + \tau)$ have nothing to do with each other and p_{ij} is factorised as $p_i p_j$ for which AMI is close to zero. Then, in order to obtain τ_0 , “it has to be found the first minimum of $I(\tau)$ where $x(n + \tau)$ adds maximal information to the knowledge from $x(n)$, or, where the redundancy is the least” [12].

For example, we compute the AMI for two time series: A) the x solution of the

Lorenz system, and B) a noise time series using a normal distribution with mean zero and standard deviation equal to one. From Fig 4, it can then be concluded that the amount of knowledge for any noise time series is zero for which the first minimum embedding parameter is $\tau_0 = 1$. On the contrary, the first minimum of the AMI for the chaotic time series is $\tau_0 = 17$ which is the value that maximize the independence between $x(n)$ and $x(n + \tau)$ in the reconstructed state space [8]. Similarly as Cao's

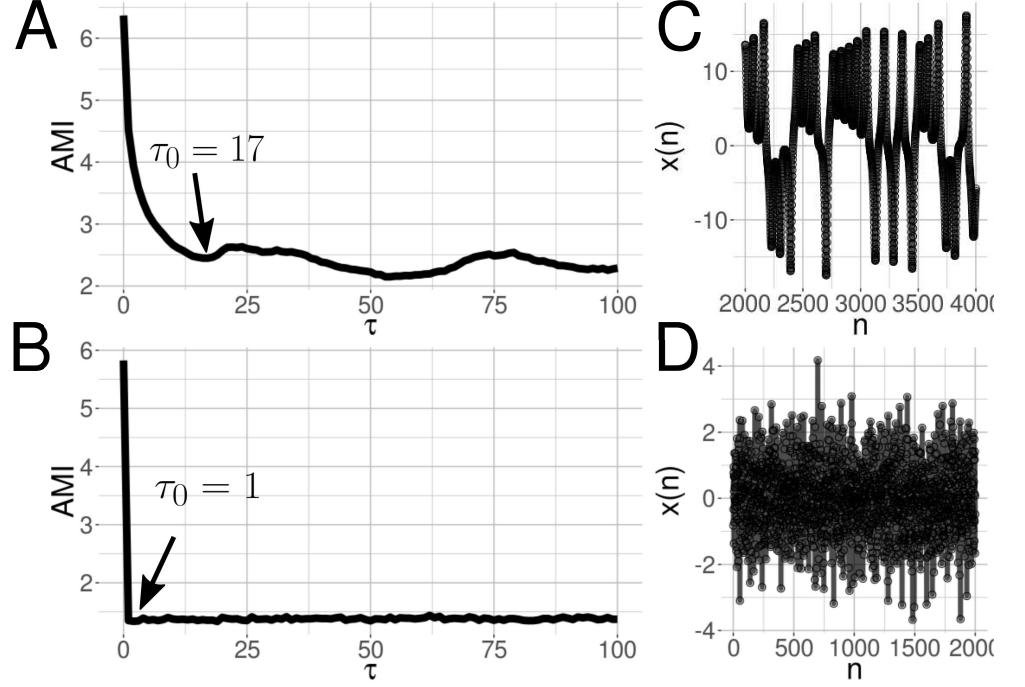


Fig 4. Minimum delay embedding values with AMI's method. (A, B) AMI values where its first minimum value in the curve is the minimum time delay embedding (τ_0), for (C) a chaotic and (D) noise time series. R code to reproduce the figure is available from [22].

algorithm negatives, AMI's algorithm is not an exception for negatives, which are worthwhile to mention for further investigations. For instance, (i) is not clear why the choose of the first minimum of the AMI is the minimum delay embedding parameter [12] and (ii) the probability distribution of the AMI function is computed with the use of histograms which depends on a heuristic choice of number of bins for which AMI depends on partitioning [26].

Other methodologies for state space reconstruction.

It is important to note that other methods for state space reconstruction have been recently investigated, for instance: (i) the nonuniform time-delay embedding methodology where the consecutive delayed copies of $\{x_n\}$ are not equidistant. Such method has been proben to create better representations of the dynamics of the state space to analyse, for instance, quasiperiodic and multiple time-scale time series over the conventional uniform time-delay embedding algorithm [1, 2, 16, 20, 30], and (ii) Uniform 2 time-delay embedding method which takes advantage of finding an embedding window instead of the traditional method of finding the embedding parameters separately [5]. In general, Uniform 2 time-delay embedding method computes m with False Nearest

Neighbour (FNN) algorithm and τ is computed as $\tau = d_w/(m - 1)$, where d_w is given by the minimisation of the Minimum Description Length [31]. However, the previous methods are out of the scope of the paper but we think is important to refer readers to those references for further investigations in this regard.

Recurrence Quantification

Recurrence Plots

Originally, Henri Poincaré in 1890 introduced the concept of recurrences in conservative systems, however such discovery was not put into practice until the development of faster computers [32], for which Eckmann et al. [33] in 1987 introduced a method where recurrences in the dynamics of a system can be visualised using Recurrence Plots. The intention of Eckmann et al. [33] was to propose a tool, called Recurrence Plot (RP), that provides insights into high-dimensional dynamical systems where trajectories are very difficult to visualise. Therefore, "RP helps us to investigate the m -dimensional phase space trajectories through a two-dimensional representation of its recurrences" [34]. Similarly, Marwan et al. [34] pointed out that additionally to the methodologies of the state space reconstruction and other dynamic invariants such as Lyapunov exponent, Kolmogorov-Sinai entropy, the recurrences of the trajectories in the phase space can provide important clues to characterise natural process that present, for instance, periodicities (as Milankovitch cycles) or irregular cycles (as El Niño Southern Oscillation). Such recurrences can not only be presented visually using Recurrence Plots (RP) but also be quantified with Recurrence Quantification metrics, which leads to applications of these in various areas such as economy, physiology, neuroscience, earth science, astrophysics and engineering [32].

For the creation of a recurrence plot based on time series $\{\mathbf{x}_n\}$, it is first computed the state space reconstruction with uniform time-delay embedding $X(i) = \{\tilde{\mathbf{x}}_n, \dots, \tilde{\mathbf{x}}_{n-(m-1)\tau}\}$ where $i = 1, \dots, N$, N is the number of considered states of $X(i)$ and $X(i) \in \mathbb{R}^m$ [33]. The recurrence plot is therefore a two-dimensional $N \times N$ square matrix, \mathbf{R} , where a black dot is placed at (i, j) whenever $X(i)$ is sufficiently close to $X(j)$:

$$\mathbf{R}_{i,j}^m(\epsilon) = \Theta(\epsilon_i - \|X(i) - X(j)\|) \quad (10)$$

where $i, j = 1, \dots, N$, ϵ is a threshold distance, $\|\cdot\|$ a norm, and $\Theta(\cdot)$ is the Heaviside function (i.e. $\Theta(x) = 0$, if $x < 0$, and $\Theta(x) = 1$ otherwise) (Fig 5) [32–34]. RP is also characterised with a line of identity (LOI) which is a diagonal line due to $R_{i,j} = 1$ ($i, j = 1, \dots, N$).

Structures of Recurrence Plots

Pattern formations in the RPs can be designated either as topology for large-scale patterns or texture for small-scale patterns. In the case of topology, the following pattern formations are commonly presented: (i) homogeneous where uniform recurrence points are spread in the RP e.g., uniformly distributed noise (Fig 6A), (ii) periodic and quasi-periodic systems where diagonal lines and checkerboard structures represent oscillating systems, e.g., sinusoidal signals (Fig 6B), (iii) drift where paling or darkening recurrence points away from the LOI is caused by drifting systems, e.g., logistic map (Fig 6C), and (iv) disrupted where recurrence points are presented white areas or bands that indicate abrupt changes in the dynamics, e.g. Brownian motion (Fig 6D) [33, 34]. Texture patterns in RPs can be categorised as: (i) single or isolated recurrence points that represent rare occurring states, and do not persist for any time or fluctuate heavily, (ii) dots forming diagonal lines where the length of the small-scale parallel lines in the

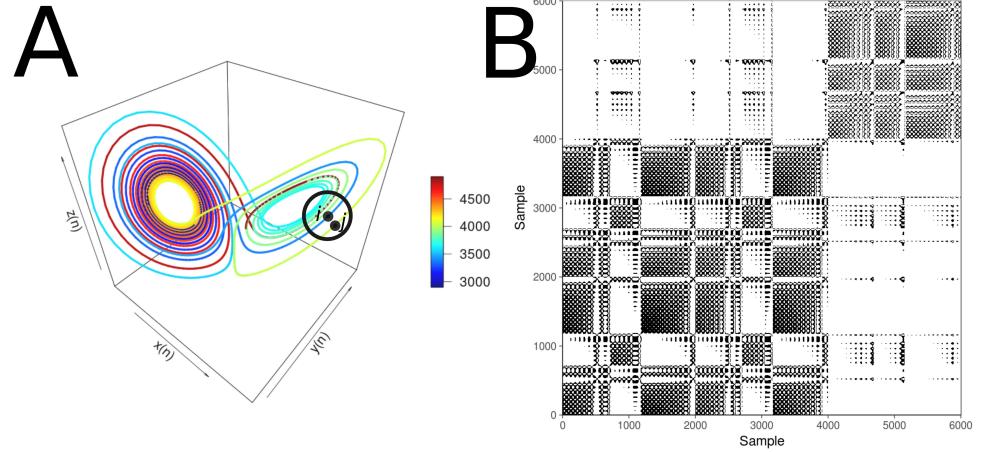


Fig 5. Recurrence Plots. (A) State space of the Lorenz system with controlling parameters ($\rho = 28$, $\sigma = 10$, $\beta = 8/3$). A point, j , in trajectory $X()$ which falls into the neighborhood (black circle) of a given point at i is a recurrent point and is represented as a black dot in the recurrence plot at location (i, j) or white otherwise. (B) Recurrence plot using the three components of the Lorenz system and the RP with no embeddings and threshold $\epsilon = 5$. This figure is adapted from [34] and R code to reproduce it is available from [22].

diagonal are related to the ratio of determinism or predictability in the dynamics of the system, and (iii) dots forming vertical and horizontal lines where the length of the lines represent a time length where a state does not change or change very slowly and these patterns formation represent discontinuities in the signal, and (iv) dots clustering to inscribe rectangular regions which are related to laminar states or singularities [34].

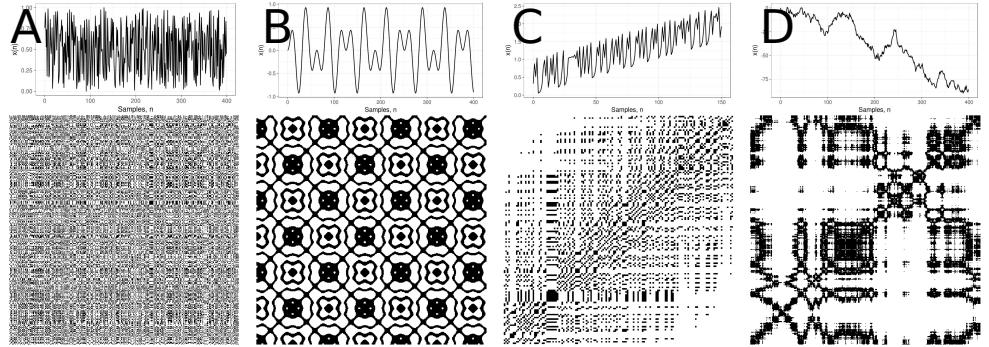


Fig 6. Patterns in Recurrence Plots. Time-series with its respective recurrence plot for: (A) uniformly distributed noise, (B) super-positionet harmonic oscillation ($\sin(\frac{1}{5} * t) * \sin(\frac{5}{100} * t)$), (C) drift logistic map ($x_{i+1} = 4x_i(1 - x_i)$) corrupted with a linearly increase term ($0.01 * i$), and (D) disrupted brownian motion ($x_{i+1} = x_i + 2 * rnorm(1)$). This figure is adapted from [34] and R code to reproduce it is available from [22].

Although, each of the previous pattern descriptions of the structures in the RP offer an idea of the characteristics of dynamical systems, these might be misinterpreted and conclusions might tend to be subjective as these require the interpretation of a particular researcher(s). Because of that, recurrence quantification analysis (RQA) offer

objective methodologies to quantify such visual characteristics of previous recurrent pattern structures in the RP [35].

Recurrence Quantifications Analysis (RQA)

Originally, Zbilut et al. [35] proposed metrics to investigate the density of recurrence points in RPs, then histograms of lengths for diagonal lines in RPs were studied by [36] which were the introduction to the term recurrence quantification analysis (RQA) [37]. RQA has been applied in many fields such as life science, engineering, physics, and others [37]. Particularly in human movement to investigate noise and complexity of postural control [40], postural control [38] or interpersonal coordination [39]. The success of RQA is not only due to its simple algorithmic implementation but also to its capacity to detect tiny modulations in frequency or phase which are not detectable using standard methods e.g. spectral or wavelet analysis [6], and that RQA's metrics are quantitatively and qualitatively independent of embedding dimension which is verified experimentally by [10]. RQA metrics comprehend percentage of recurrence, percentage of determinism, ratio, Shannon entropy of the frequency distributions of the line lengths, maximal line length and divergence, trend and laminarity [32, 34]. For this work, we considered only four RQA metrics, due to its consistency with our preliminary experiments, which are described below. Such metrics are computed the nonlinearTseries R package [29].

REC values

The percentage of recurrence (REC) is defined as

$$REC(\epsilon, N) = \frac{1}{N^2 - N} \sum_{i \neq j=1}^N \mathbf{R}_{i,j}^m(\epsilon), \quad (11)$$

which enumerates the black dots in the RP excluding the line of identity. REC is a measure of the relative density of recurrence points in the sparse matrix [34].

DET values

The percent determinism (DET) is defined as the fraction of recurrence points that form diagonal lines and it is determined by

$$DET = \frac{\sum_{l=d_{min}}^N l H_D l}{\sum_{i,j=1}^N \mathbf{R}_{i,j}(\epsilon)}, \quad (12)$$

where

$$H_D(l) = \sum_{i,j=1}^N (1 - \mathbf{R}_{i-1,j-1}(\epsilon))(1 - \mathbf{R}_{i+l,j+l}(\epsilon)) \prod_{k=0}^{l-1} \mathbf{R}_{i+k,j+k}(\epsilon) \quad (13)$$

is the histogram of the lengths of the diagonal structures in the RP. DET can be interpreted as the predictability of the system for periodic signals which, in essence, have longer diagonal lines than the short diagonals lines for chaotic signals or absent diagonal lines for stochastic signals [32, 34]. Similarly, DET is considered as a measurement for the organisation of points in RPs [10].

RATIO values

RATIO is defined as the ratio between DET and REC and it is calculated from the frequency distributions of the lengths of the diagonal lines. RATIO is useful to discover dynamic transitions [34].

ENT values

ENT is the Shannon entropy of the frequency distribution of the diagonal line lengths and it is defined as

$$ENT = - \sum_{l=d_{min}}^N p(l) \ln p(l) \quad \text{with} \quad p(l) = \frac{H_D(l)}{\sum_{l=d_{min}}^N H_D(l)}. \quad (14)$$

ENT reflects the complexity of the deterministic structure in the system. For instance, for uncorrelated noise or oscillations, the value of ENT is rather small and indicates low complexity of the system, therefore "the higher the ENT is the more complex the dynamics are" [32, 34].

Sensitivity and robustness of RPs and RQA.

RP and RQA are a very young field in nonlinear dynamics and many questions are still open, for instance, different parameters for window length size of the time series, embedding parameters or recurrence threshold can generate different results in RQA's metrics [6, 33].

The selection of recurrence threshold, ϵ , can depend on the system that is analysed. For instance, when studying dynamical invariants ϵ require to be very small, for trajectory reconstruction ϵ requires to have a large thresholds or when studying dynamical transition there is little importance about the selection of the threshold [6]. Other criteria for the selection of ϵ is that the recurrence threshold should be five times larger than the standard deviation of the observational noise or the use of diagonal structures within the RP is suggested in order to find the optimal recurrence threshold for (quasi-)periodic process [6].

Similarly, Iwanski et al. [10] highlighted the importance of choosing the right embedding parameters to perform RQA for which many experiments have to be performed using different parameters in order to have a better intuition of the nature of the time series and how this is represented by using RQA.

With that in mind, this work explores the sensitivity and robustness of the window size of time series, embedding parameters for RSS with UTDE and recurrence threshold for RP and RQA in order to gain a better insight into the underlying time series collected from inertial sensors in the context of human-humanoid imitation activities.

Experiment

We conducted an experiment in the context of human-humanoid imitation (HHI) activities where participants were asked to imitate simple horizontal and vertical arm movements performed by NAO, a humanoid robot [41]. Such simple movements were repeated ten times for the participant who copied NAO's arm movements in a face-to-face imitation activity. Also, wearable inertial measurement unit (IMU) sensors were attached to the right hand of the participant and to the left hand of the robot (Figure 7 A,C). Data were then collected with four NeMEMSi IMU sensors with sampling rate of 50Hz providing tri-axial data of the accelerometer, gyroscope and magnetometer sensors and quaternions [42]. A further description of the NeMEMSi IMU sensors is given in S1 Appendix B.

Participants

Twenty-three participants, from now on defined as pN where N is the number of participant, were invited to do the experiment. However, data for three participants

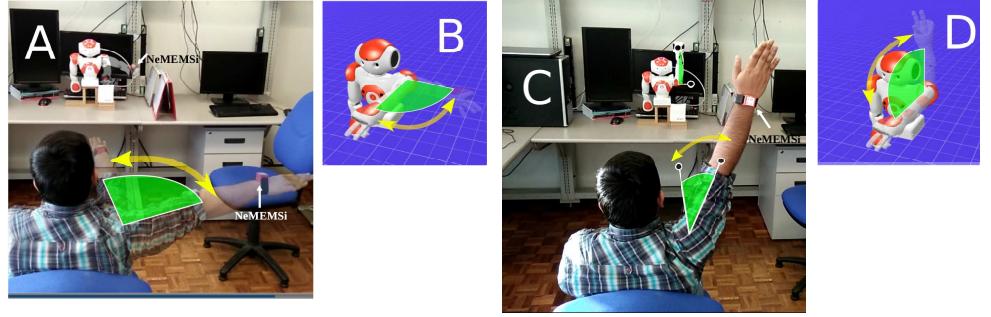


Fig 7. Human-humanoid imitation activities. Face-to-face human-humanoid imitation (HHI) activities for (A) HHI of horizontal arm movement, (B) Humanoid horizontal arm movement, (C) HHI of vertical arm movement, and (D) Humanoid vertical arm movement.

were not used because the instructions for *p01*, who was the only left-handed, were mistakenly given in a way that movements were performed different from what had been planned, and for participants *p13* and *p16* data were corrupted because bluetooth communications problems with the sensors. With that in mind, data for twenty participants were analysed in this work.

Of the twenty participants, all of them are right-handed healthy participants of whom four are females and sixteen are males with a mean and standard deviation (SD) age of mean=19.8 (SD=1.39). All participants provided informed consent forms prior to participation in the experiment.

Human-humanoid imitation activities

For human-humanoid imitation (HHI) activities four neMEMSi sensors were used, two of which were attached to the right hand of the participant and the other two to the left hand of the humanoid robot. Then, each participant was asked to imitate repetitions of simple horizontal and vertical arm movements performed by the humanoid robot in the following conditions: (i) ten repetitions of horizontal arm movement at normal (HN) and faster (HF) speed (Figure 7 A), and (ii) ten repetitions of vertical arm movement at normal (VN) and faster (VF) speed (Figure 7 C). The normal and faster speed of arm movements is defined by the duration in number of samples of one repetition of NAO's arm movements. We select NAO's arm movements duration to distinguish between normal and faster arm movements as NAO's movements have less variation between repetition to repetition. The duration for one repetition of the horizontal arm movement at normal speed, HN, is about 5 seconds considering that each repetition last around 250 samples. For horizontal arm movement at faster speed, HF, each repetition were performed in around 2 seconds which correspond to 90 samples of data. The vertical arm movement at normal speed, VN, were performed in 6 seconds which is around 300 samples of data. For vertical arm movement at faster speed, VF, each repetition lasts about 2.4 seconds which correspond to 120 samples of data. To visualise the distinction between normal and faster speed for horizontal and vertical arm movements, Fig 8 shows smoothed time series for axes Z and Y of the gyroscope sensors with four window lengths: 2-sec (100-samples), 5-sec (250-samples), 10-sec (500-samples) and 15-sec (750-samples).

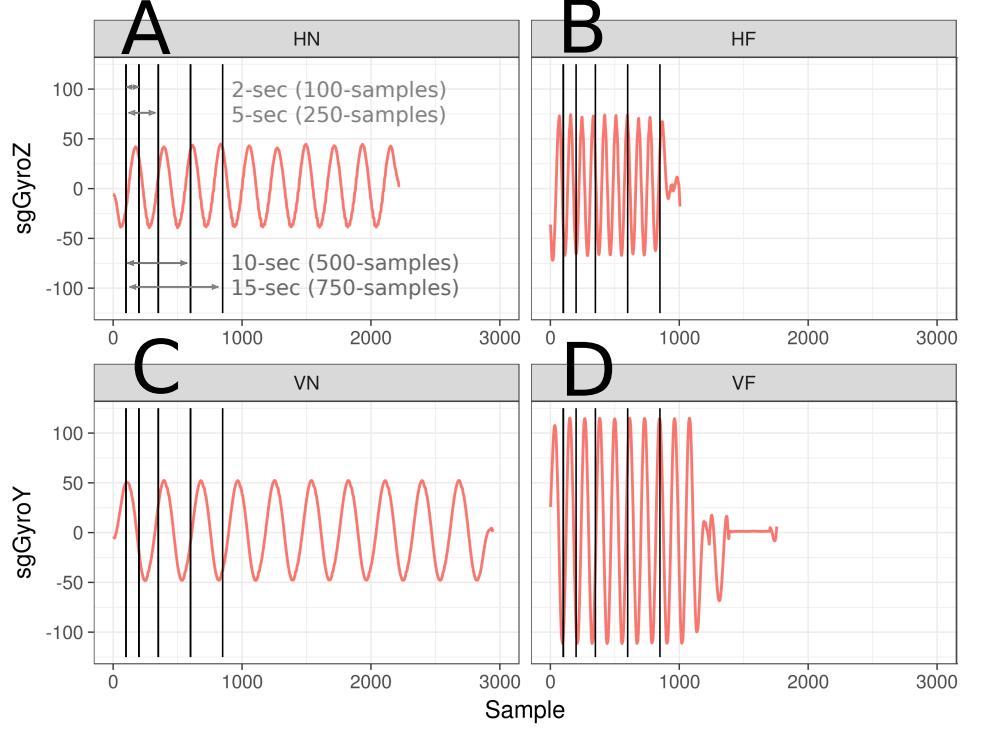


Fig 8. Time series duration of horizontal and vertical arm movements.

Time series of smoothed data from gyroscope sensor for different speed arm movements performed by NAO: (A) Horizontal Normal arm movement (HN), (B) Horizontal Faster arm movement (HF), (C) Vertical Normal arm movement (VN) and (D) Vertical Faster arm movement (VF). Additionally, (A) shows window sizes for 2-seconds (100 samples), 5-seconds (250 samples), 10-seconds (500 samples) and 15-seconds (750 samples) which are also presented in (B), (C) and (D). R code to reproduce figure is available [22].

Data from Inertial Measurement Units

Raw data

Considering the work of [43] which provided evidence of an improvement in recognition activities when combining data from accelerometer and gyroscope. We focus our analysis from data of the accelerometer and gyroscope of the NeMEMSi sensors [42] and leave the data of the magnetometer and quaternions for further investigation because of their possible variations with regard to magnetic disturbances.

Data from the accelerometer is defined by triaxial time series $A_x(n)$, $A_y(n)$, $A_z(n)$ which forms the matrix \mathbf{A} (Eq. 15), and the same for data from the gyroscope which is defined by triaxial time-series of $G_x(n)$, $G_y(n)$, $G_z(n)$ representing the matrix \mathbf{G} (Eq. 16). Both triaxial time series of each sensor, a and g , are denoted with its respective axes subscripts x, y, z , where n is the sample index and N is the same maximum length of all axes for the time series. Matrices \mathbf{A} and \mathbf{G} are represented as follows

$$\mathbf{A} = \begin{pmatrix} A_x(n) \\ A_y(n) \\ A_z(n) \end{pmatrix} = \begin{pmatrix} a_x(1), a_x(2), \dots, a_x(N) \\ a_y(1), a_y(2), \dots, a_y(N) \\ a_z(1), a_z(2), \dots, a_z(N) \end{pmatrix}, \quad (15)$$

and

$$\mathbf{G} = \begin{pmatrix} G_x(n) \\ G_y(n) \\ G_z(n) \end{pmatrix} = \begin{pmatrix} g_x(1), g_x(2), \dots, g_x(N) \\ g_y(1), g_y(2), \dots, g_y(N) \\ g_z(1), g_z(2), \dots, g_z(N) \end{pmatrix}. \quad (16)$$

Postprocessing data

After the collection of raw data from four NeMEMsi sensors, time synchronisation alignment and interpolation were performed in order to create time series with the same length and synchronised time. We refer the reader to [42] for further details about the time synchronisation process.

Data normalization

Data is normalised to have zero mean and unit variance using sample mean and sample standard deviation. The sample mean and sample standard deviation using $x(n)$ is given by

$$\mu_{x(n)} = \frac{1}{N} \left(\sum_{i=1}^N x(i) \right), \quad \sigma_{x(n)} = \sqrt{\frac{\sum_{i=1}^N (x(i) - \mu_{x(n)})^2}{N-1}}, \quad (17)$$

and the normalised data, $\hat{x}(n)$, is computed as follows

$$\hat{x}(n) = \frac{x(n) - \mu_{x(n)}}{\sigma_{x(n)}}. \quad (18)$$

Smoothing data

Commonly, a low-pass filter is the method either to capture the low frequencies that represent %99 of the human body energy or to get the gravitational and body motion components of accelerations [44]. However, for this work the elimination of certain range of frequencies is not the main focus but the conservation of the structure in the time series in terms of the width and heights where, for instance, Savitzky-Golay filter can help to accomplish such task [45]. Savitzky-Golay filter is based on the principle of moving window averaging which preserves the area under the curve (the zeroth moment) and its mean position in time (the first moment) but the line width (the second moment) is violated and that results, for example, in the case of spectrometric data where a narrow spectral line is presented with reduced height and width. With that in mind, the aim of Savitzky-Golay filtering is to find filter coefficients c_n that preserve higher momentums which are based on local least-square polynomial approximations [45–47]. Therefore, Savitzky-Golay coefficients are therefore computed using an R function `sgolay(p,n,m)` where p is the filter order, n is the filter length (must be odd) and m is the m -th derivative of the filter coefficients [48]. Smoothed signal is represented with a tilde over the original signal: $\tilde{x}(n)$.

Window size data

With regard to the window size, [43] investigated its effects using seven window lengths (2, 5, 10, 15, 20, 25, 30 seconds) and combination of inertial sensors (accelerometer, gyroscope and linear acceleration sensor) to improve the activity recognition performance for repetitive activities (walking, jogging and biking) and less repetitive activities (smoking, eating, giving a talk or drinking a coffee). With that in mind, Shoaib et al. [43] concluded that the increase of window length improve the recognition of complex activities because these requires a large window length to learn the repetitive motion patterns. Particularly, one of the recommendations is to use large window size

to recognise less repetitive activities which mainly involve random hand gestures. Therefore, for the four activities (HN, HF, VN, and VF) in this work, which are mainly repetitive, we select only four window sizes for analysis: 2-s window (100 samples), 5-s window (250 samples), 10-s (500 samples) and 15-s window (750 samples) (Fig 8).

Results

Once time series were collected, we investigated the robustness and weaknesses of the reconstructed state spaces (RRSs) using the uniform time-delay embedding technique (UTDE) and recurrence plots (RPs) for recurrent quantification analysis (RQA) methodologies in the following conditions:

- Three levels of smoothness for the normalised data ($sg0zmuv$, $sg1zmuv$ and $sg2zmuv$), computed from two different filter lengths (29 and 159) with the same polynomial degree of 5 using the function `sgolay(p,n,m)` [48],
- Four velocities arm movement activities: horizontal normal (HN), horizontal faster (HF), vertical normal (VN), and vertical faster (VF), and
- Four window lengths: {2-sec (100 samples), 5-sec (250 samples), 10-sec (500 samples) and 15-sec (750 samples) }.

Time series

After the data collection, raw time series were windowed, normalised and smoothed. We only present 10-sec (500 samples) window length time series, due to space limitations, for three participants (p01, p01 and p03) performing horizontal arm movements (axis GyroZ) and vertical arm movements (axis GyroY) (Figs 9 and 10).

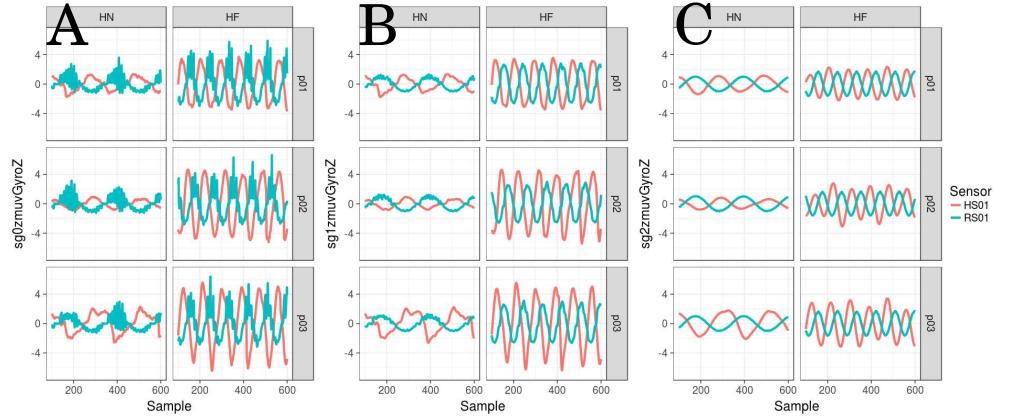


Fig 9. Time series for horizontal arm movements. (A) raw-normalised ($sg0zmuvGyroZ$), (B) normalised-smoothed 1 ($sg1zmuvGyroZ$) and (C) normalised-smoothed 2 ($sg2zmuvGyroZ$). Time series are only for three participants (p01, p02, and p03) for horizontal movements in normal and faster velocity (HN, HF) with the normalised GyroZ axis ($zmuvGyroZ$) and with one sensor attached to the participant (HS01) and other sensor attached to the robot (RS01). R code to reproduce the figure is available from [22].

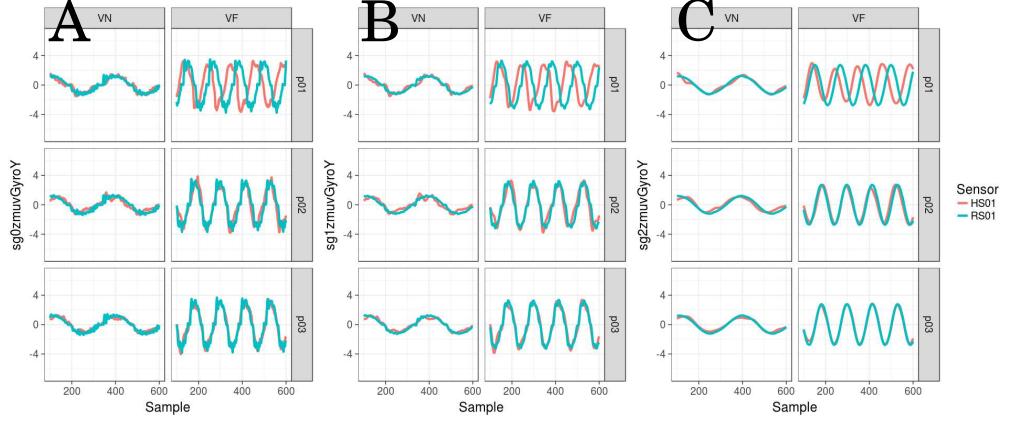


Fig 10. Time series for vertical arm movements. (A) raw-normalised ($sg0zmuvGyroY$), (B) normalised-smoothed 1 ($sg1zmuvGyroY$) and (C) normalised-smoothed 2 ($sg2zmuvGyroY$). Time series are only for three participants (p01, p02, and p03) for vertical movements in normal and faster velocity (VN, VF) with the normalised GyroY axis ($zmuvGyroY$) and with one sensor attached to the participant (HS01) and other sensor attached to the robot (RS01). R code to reproduce the figure is available from [22].

Minimum embedding parameters

Minimum embedding parameters were firstly computed to create the reconstructed state spaces with UTDE and RQAs with RPs.

False Nearest Neighbour

Considering the time series for twenty participants, minimum embedding dimensions were computed using False Nearest Neighbour for horizontal and vertical arm movements. Figs 11 and 12 show that minimum embedding values appear to be more constant for sensor RS01 than the slightly variations of such values for sensor HS01. It can also be seen that there is a minor decrease of minimum embedding values as smoothness of time series increase. To have an overall minimum dimension value that represent participants, sensors and activities, a sample mean were computed over all the minimum values in Figs 11 and 12 which results in $\bar{m}_0 = 6$.

Average Mutual Information

Similarly, considering the time series for twenty participants, minimum delay values were computed as the first minimum values of the Average Mutual Information (AMI) for horizontal and vertical arm movements (Figs 13 and 14).

For horizontal arm movements, Fig 13A shows that values tend to be more spread as the smoothness is increased which is different for Fig 13C where values show no effect as the smoothness of time series increase. In contrast, Fig 13B shows the values are less spread as smoothness is increased which we believe the reason for that is due to the high frequencies on robots movements in the horizontal normal movement. However, values in Fig 13D tend to be spread as smoothness is increasing which are due to very different curves in the AMI. With regard to vertical arm movements, values in Figs 14A and 14C show an slightly increase of the spread values as the smoothness increase and values in Fig 14B appear to have less variation as the smoothness of the signals is increasing. However, that do not happen for the second smoothed values

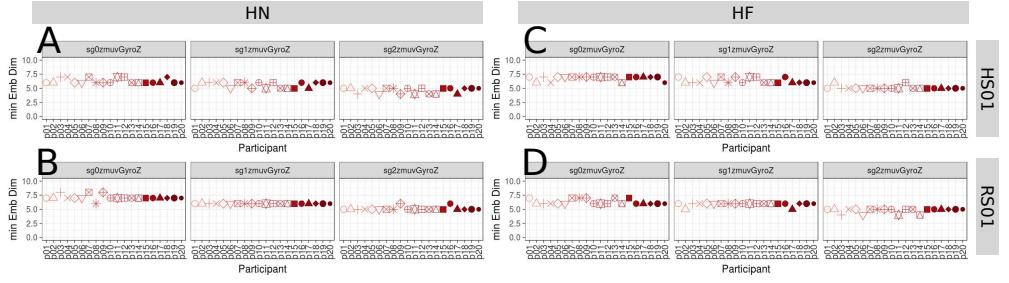


Fig 11. Minimum embedding dimensions for horizontal arm movements. (A, B) Horizontal Normal (HN), (C, D) Horizontal Faster (HF) movements, (A, C) sensor attached to participants (HS01), and (B, D) sensor attached to robot (RS01). Minimum embedding dimensions are for twenty participants (p01 to p20) with three smoothed signals (sg0zmuvGyroZ, sg1zmuvGyroZ and sg2zmuvGyroZ) and window lenght of 10-sec (500 samples). R code to reproduce the figure is available from [22].

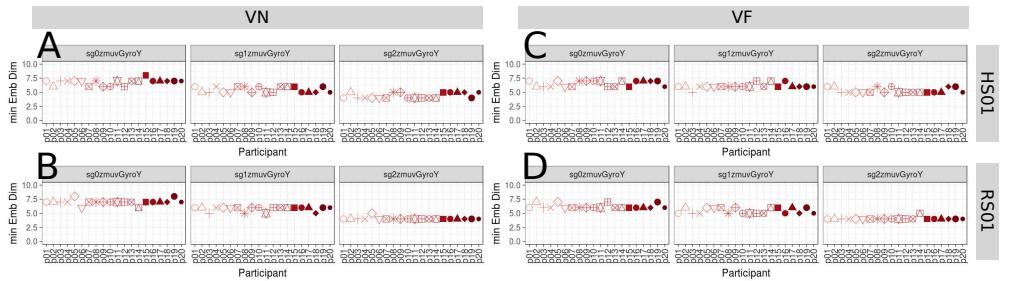


Fig 12. Minimum embedding dimensions for vertical arm movements. (A, B) Vertical Normal (VN), (C, D) Vertical Faster (VF) movements, (A, C) sensor attached to participants (HS01), and (B, D) sensor attached to robot (RS01). Minimum embedding dimensions are for twenty participants (p01 to p20) with three smoothed signals (sg0zmuvGyroY, sg1zmuvGyroY and sg2zmuvGyroY) and window length of 10-sec (500 samples). R code to reproduce the figure is available from [22].

(sg2zmuvGyroY) in Fig 14D. We also computed an overall minimum delay value that represent participants, sensors and activities, using a sample mean of all values in Figs 13 and 14 which results in $\bar{\tau}_0 = 8$.

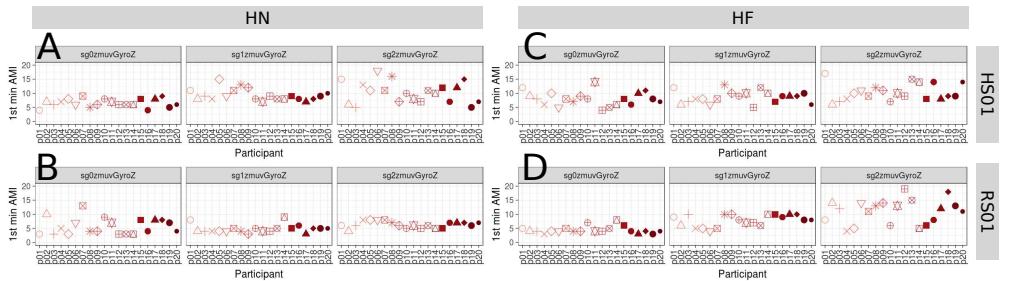


Fig 13. First minimum AMI values for horizontal arm movements. (A, B) Horizontal Normal (HN), (C, D) Horizontal Faster (HF) movements, (A, C) sensor attached to participants (HS01), and (B, D) sensor attached to robot (RS01). First minimum AMI values are for twenty participants (p01 to p20) with three smoothed signals (sg0zmuvGyroZ, sg1zmuvGyroZ and sg2zmuvGyroZ) and window lenght of 10-sec (500 samples). R code to reproduce the figure is available from [22].

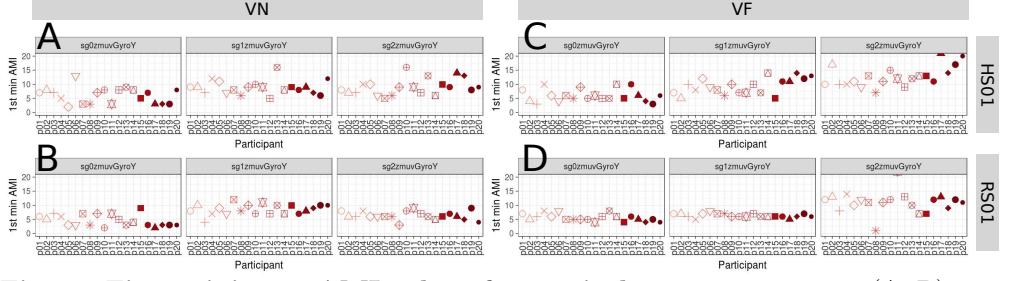


Fig 14. First minimum AMI values for vertical arm movements. (A, B) Vertical Normal (VN), (C, D) Vertical Faster (VF) movements, (A, C) sensor attached to participants (HS01), and (B, D) sensor attached to robot (RS01). First minimum AMI values are for twenty participants (p01 to p20) with three smoothed signals ($sg0zmuvGyroZ$, $sg1zmuvGyroZ$ and $sg2zmuvGyroZ$) and window lenght of 10-sec (500 samples). R code to reproduce the figure is available from [22].

Reconstructed State Spaces using Uniform Time-Delay Embedding

Although the implementation of Uniform Time-Delay Embedding for the reconstructed state space, one of the main challenges of the latter is the selection of embedding parameters because each time series is unique in terms of its structure (modulation of amplitude, frequency and phase) [3, 4, 8]. With that in mind, the problem is not to compute individual embedding parameters for each of the time series but to deal with a selecting of two parameters that can represent all the time series. Our solution for that was to compute a sample mean over all values in each of the conditions of the time series of Figs 11, 12 for minimum dimension values and Figs 13 and 14 for minimum delay values, resulting in an average minimum embedding parameters of ($\bar{m}_0 = 6$, $\bar{\tau}_0 = 8$). Then, the reconstructed state spaces were computed with ($\bar{m}_0 = 6$, $\bar{\tau}_0 = 8$) and the first three axis of the rotated data of the PCA are shown in Figs 15 for horizontal arm movements and Figs 16 for vertical arm movements.

Evidently, it is easy to observe by eye the differences in each of the trajectories in the reconstructed state spaces (Figs 15, 16), however one might be not objective when quantifying those differences since those observation might vary from person to person. With that in mind, we tried to objectively quantify those differences using euclidean distances between the origin to each of the points in the trajectories, however these created suspicious metric, specially for trajectories which looked very messy. With that in mind, we computed Recurrence Quantification Analysis to objectively quantify the differences in each of the cases of the time series.

Recurrences Plots

Considering the time series of Figs 9 and 10, we computed its Recurrence Plots for horizontal arm movements (Fig 17) and for vertical arm movements (Fig 18) using the average embedding parameters ($m = 6$, $\tau = 8$) and an recurrence threshold of $\epsilon = 1$. With regard to the selection of recurrence threshold, Marwan et al. [6] pointed out that choosing an appropriate recurrence threshold is crucial to get meaningful representations in RPs, however, for our work where quantifying movement variability is our aim, we give little importance to the selection of the recurrence threshold as as long as it is able to represent the dynamical transitions in each of the time series.

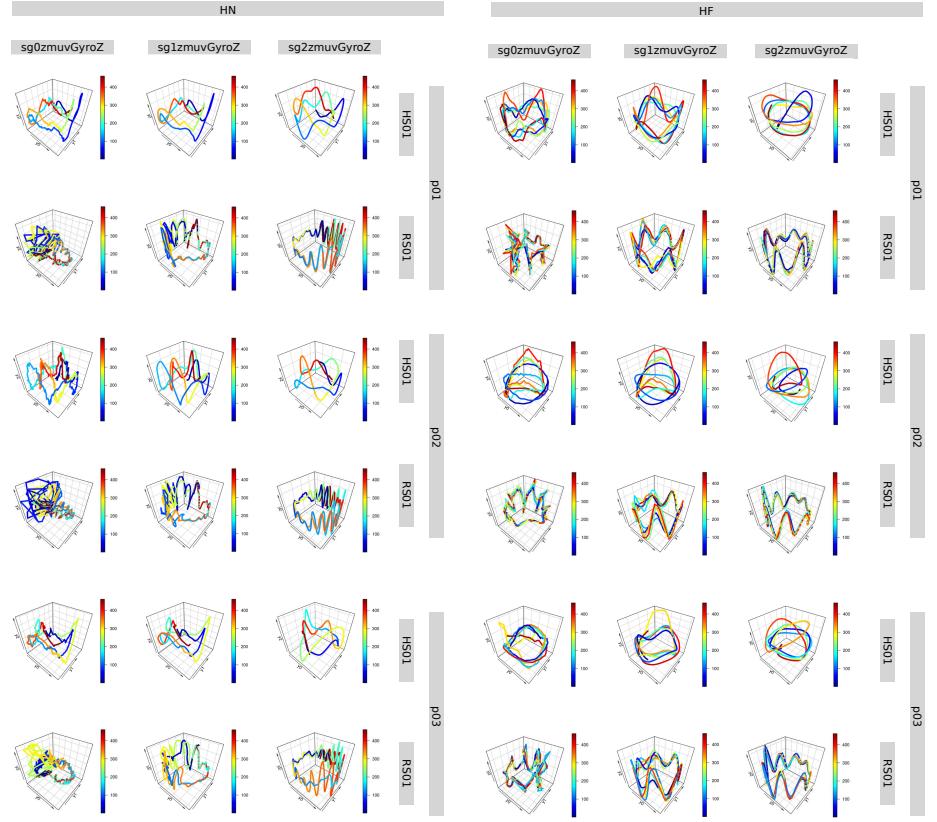


Fig 15. RSSs for horizontal arm movements. Reconstructed state spaces for time series of Figure 9. Reconstructed state spaces were computed with embedding parameters $m = 6$, $\tau = 8$. R code to reproduce the figure is available from [22].

As similar as with the Reconstructed State Spaces, the differences in the RPs can be easily noticed by eye for different conditions of the time series (Figs 18, Fig 17), which lead us to apply Recurrence Quantification Analysis to have an objective quantification of each of the time series.

Recurrence Quantification Analysis

RPs provide pattern formations for each of the time series conditions (Figs 17 and 18) which are quantified with four metrics of RQA: REC, DET, RATIO and ENTR.

REC values

In Figs 19 and 20 can be seen that REC values are more spread for HN than HF movements with data coming from HS01 sensor. In contrast, REC values appear to be constant and present little variation for both HN and HF movements with data from the sensor attached to the humanoid robot RS01. With regard to the increase of smoothness of data ($sg0zmuvGyroZ$, $sg1zmuvGyroZ$ and $sg2zmuvGyroZ$), REC values present little variation as the smoothness is increasing for data from HS01 and REC values more similar as the smoothness is increasing for data from RS01.

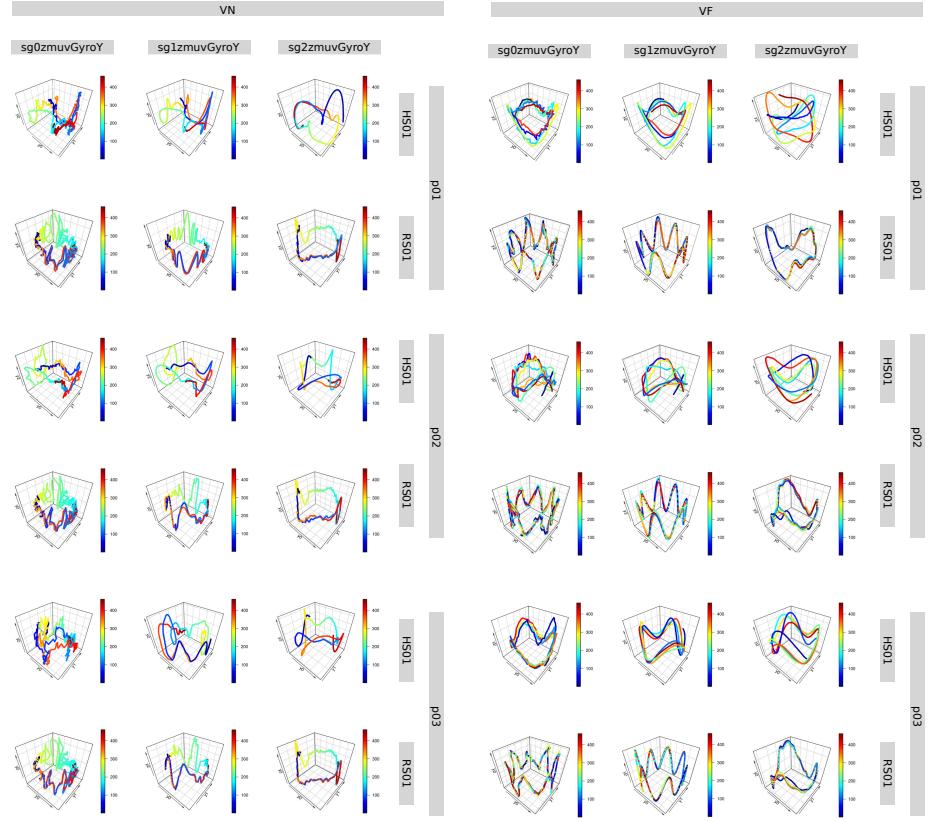


Fig 16. RSSs for vertical arm movements. Reconstructed state spaces for time series of Figure 10. Reconstructed state spaces were computed with embedding parameters $m = 6$, $\tau = 8$. R code to reproduce the figure is available from [22].

DET values

Little can be said with regard to the variation of DET values as these change very little even for type of movement or type of sensor (Figs 21 and 22). With regard to the smoothness of time series, DET values appear to be more similar as the smoothness of the data is increasing.

RATIO values

RATIO values for HN movements vary less than HF movements for HS01 sensor which is similar behaviour of RATIO values for RS01 sensors in both vertical and horizontal movements (Figs 23 and 24). It can also noticed a decrease of variation in RATIO values as the smoothness of the signal is increasing.

ENTR values

ENTR values show more variation for HS01 sensor than ENTR values for RS01 sensor which appear to be more constant and the smoothness of data affects little to the variation of ENTR values (Figs 25 and 26).

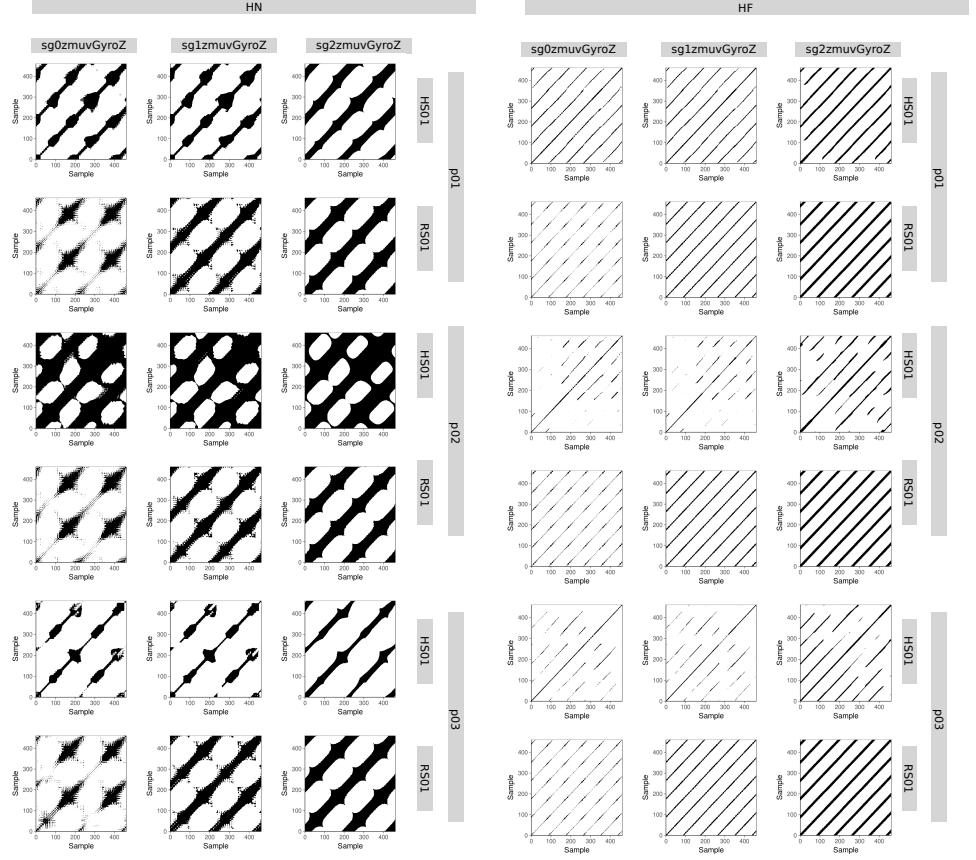


Fig 17. RPs for horizontal arm movements. Recurrence plots were computed with embedding parameters $m = 6$, $\tau = 8$ and $\epsilon = 1$. R code to reproduce the figure is available from [22].

RQA metrics with different embedding parameters, recurrence thresholds, window lengths, levels of smoothness, and time series structures.

Zbilut et al. [35] established RQA metrics with the aim of determining embedding parameters, their method consisted on creating 3D surfaces with RQA metrics with an increase of embedding parameters (m and τ), then Zbilut et al. [35] explored fluctuations and gradual changes in the 3D surfaces that provide information about the embeddings. Much recently, Marwan et al. [34] created 3D surfaces for visual selection of not only embedding parameters but also recurrence thresholds. Following same methodologies, we explored the stability and robustness of RQA metrics (REC, DET, RATIO and ENTR) using 3D surfaces by an unitary increase of the pair embedding parameters ($0 \leq m \leq 10$, $0 \leq \tau \leq 10$) and a decimal increase of 0.1 for recurrence thresholds ($0.2 \geq \epsilon \leq 3$) (Fig. 27). We also computed 3D surfaces of RQA metrics for different sensors and different activities (Fig. 29). RQA metrics are also affected by the window length where for example four window lengths of 100, 250, 500 and 750 samples (Fig. 29). Three level of smoothness were computed for RQA metrics showing smoothed 3D surfaces ad the level of smoothness increase (Fig. 30). Similarly, 3D surfaces of RQA metrics were also computed for three participants (Fig. 31).



Fig 18. RPs for vertical arm movements. Recurrence plots were computed with embedding parameters $m = 6$, $\tau = 8$ and $\epsilon = 1$. R code to reproduce the figure is available from [22].

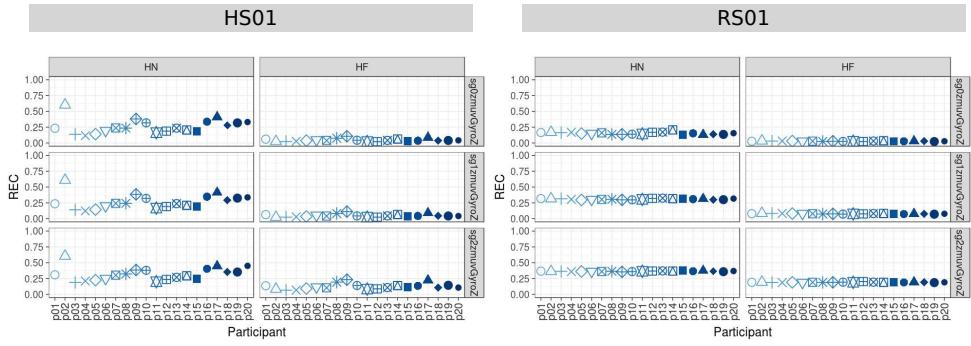


Fig 19. REC values for horizontal arm movements. REC values (representing % of black dots in the RPs) for 20 participants performing HN and HF movements with sensors HS01, RS01 and three smoothed-normalised axis of GyroZ (sg0zmuvGyroZ, sg1zmuvGyroZ and sg2zmuvGyroZ). REC values were computed with embedding parameters $m = 6$, $\tau = 8$ and $\epsilon = 1$ R code to reproduce the figure is available from [22].

Discussion

It is evidently that time series from different sources (participants, movements, axis type, window length or levels of smoothness) presents visual differences for embedding

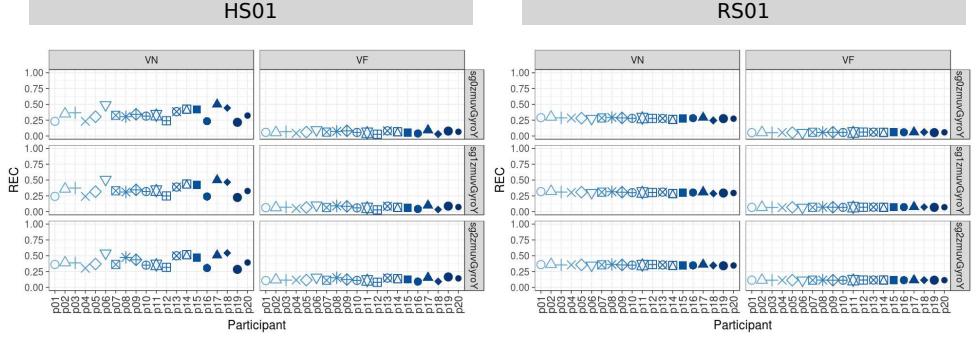


Fig 20. REC values for vertical arm movements. REC values (representing % of black dots in the RPs) for 20 participants performing VN and VF movements with sensors HS01, RS01 and three smoothed-normalised axis of GyroY (sg0zmuvGyroY, sg1zmuvGyroY and sg2zmuvGyroY). REC values were computed with embedding parameters $m = 6$, $\tau = 8$ and $\epsilon = 1$. R code to reproduce the figure is available from [22].

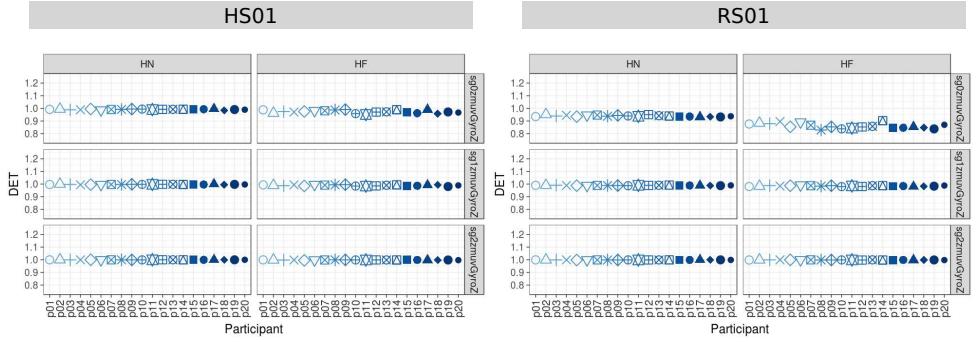


Fig 21. DET values for horizontal arm movements. DET values (representing predictability and organisation of the RPs) for 20 participants performing HN and HF movements with sensors HS01, RS01 and three smoothed-normalised axis of GyroZ (sg0zmuvGyroZ, sg1zmuvGyroZ and sg2zmuvGyroZ). DET values were computed with embedding parameters $m = 6$, $\tau = 8$ and $\epsilon = 1$. R code to reproduce the figure is available from [22].

parameters and therefore for RRSs. For which, the selection of embedding parameters was our first challenge where we computed embedding parameters for each time series and then computed a sample mean over all time series in order to get two embedding parameters to compute all RSSs with its corresponded type of movement. Then we found that the quantification of variability with regard to the shape of the trajectories in RSSs requires more investigation since our original proposed method base on euclidean metric failed to quantify those trajectories. Specially, for trajectories which were not well unfolded. With that in mind, we proceed to take advantage of four RQA metrics (REC, DET, RATIO and ENTR) in order to avoid any subjective interpretations or personal bias with regard to the evolution of the trajectories in RSSs.

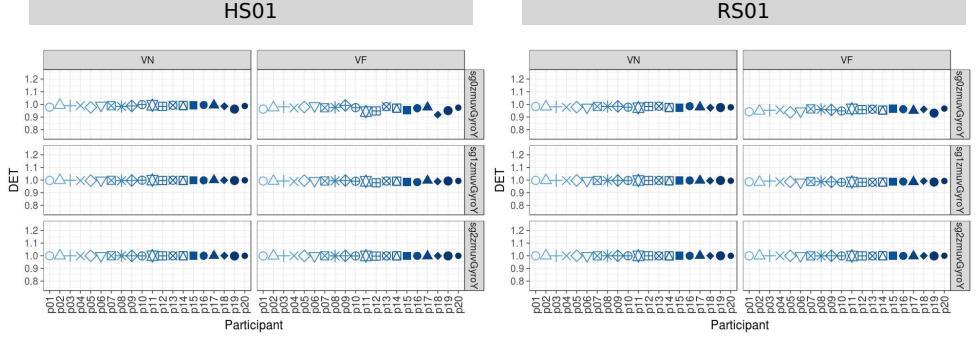


Fig 22. DET values for vertical arm movements. DET values (representing predictability and organisation of the RPs) for 20 participants performing VN and VF movements with sensors HS01, RS01 and three smoothed-normalised axis of GyroY ($sg0zmuvGyroY$, $sg1zmuvGyroY$ and $sg2zmuvGyroY$). DET values were computed with embedding parameters $m = 6$, $\tau = 8$ and $\epsilon = 1$. R code to reproduce the figure is available from [22].

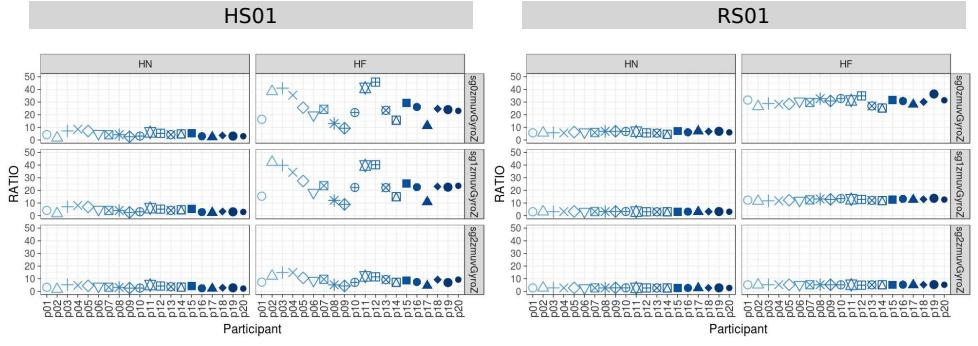


Fig 23. RATIO values for horizontal arm movements. RATIO (representing dynamic transitions) for 20 participants performing HN and HF movements with sensors HS01, RS01 and three smoothed-normalised axis of GyroZ ($sg0zmuvGyroZ$, $sg1zmuvGyroZ$ and $sg2zmuvGyroZ$). RATIO values were computed with embedding parameters $m = 6$, $\tau = 8$ and $\epsilon = 1$. R code to reproduce the figure is available from [22].

RQA metrics with fixed parameters

Considering that RQA metrics were computed with fixed embedding parameters ($m = 6$ and $\tau = 8$) and recurrence thresholds ($\epsilon = 1$), we found the following. REC values, which represents the % of black points in the RPs, were more affected with an increase in normal speed movements (HN and VN) than faster movements (HF and VF) for the sensor attached to the participants (HS01). Such decrease of REC values from normal speed to faster speed movements is also presented in data from sensor attached to the robot (RS01), and little can be said with regard to the dynamics of the time series coming from RS01. Similarly, DET values, representing predictability and organisation in the RPs, present little variation in any of the time series where little can be said. In contrast, RATIO values, which represent dynamic transitions, were more variable for faster movements (HF and VF) than normal speed movements (HN and VN) with sensors attached to the participants (HS01). For data coming from sensors attached to

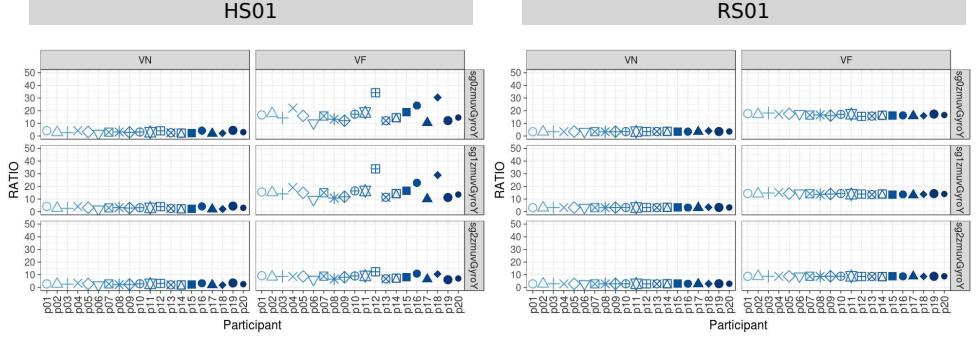


Fig 24. RATIO values for vertical arm movements. RATIO (representing dynamic transitions) for 20 participants performing VN and VF movements with sensors HS01, RS01 and three smoothed-normalised axis of GyroY (sg0zmuvGyroY, sg1zmuvGyroY and sg2zmuvGyroY). RATIO values were computed with embedding parameters $m = 6$, $\tau = 8$ and $\epsilon = 1$. R code to reproduce the figure is available from [22].

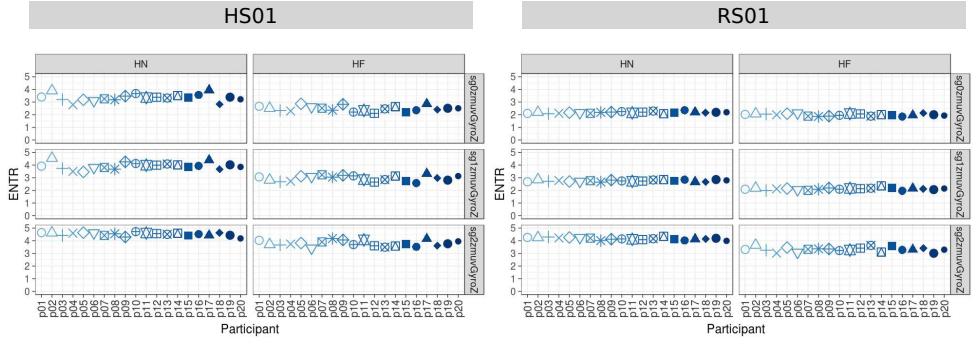


Fig 25. ENTR values for horizontal arm movements. ENTR values (representing the complexity of the deterministic structure in time series) for 20 participants performing HN and HF movements with sensors HS01, RS01 and three smoothed-normalised axis of GyroZ (sg0zmuvGyroZ, sg1zmuvGyroZ and sg2zmuvGyroZ). ENTR values were computed with embedding parameters $m = 6$, $\tau = 8$ and $\epsilon = 1$. R code to reproduce the figure is available from [22].

the robot (RS01), RATIO values from horizontal movements (HN, HF) appear to vary more than values coming from vertical movements (VN, VF). With that, it can be said that RATIO values can represent a bit better than REC or DET metrics for the variability of imitation activities in each of the conditions for time series. In the same way, ENTR values for HN were higher than values for HF and ENTR values varied more for sensor attached to participants than ENTR values for sensors of the robot. It is evidently that the higher the entropy the more complex the dynamics are, however, ENTR values for HN appear a bit higher than HF values, for which we believe this happens because of the structure the time series which appear more complex for HN than HF movements which presented a more consistence repetition.

We observed that some RQA metrics are affected by the smoothness of data. For which, we also explored the effect of smoothness of raw-normalised data where, for example, REC and DET values were not affected by the smoothness of data since these seemed to be constants. However, for RATIO values, the effect of smoothness can be

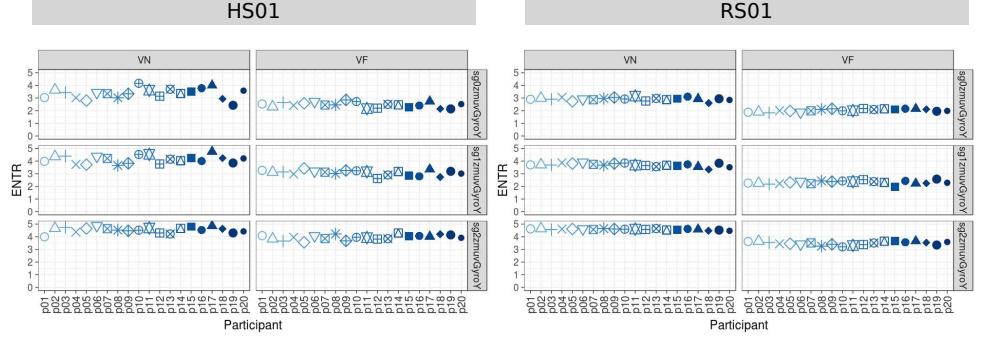


Fig 26. ENTR values for vertical arm movements. ENTR values (representing the complexity of the deterministic structure in time series) for 20 participants performing VN and VF movements with sensors HS01, RS01 and three smoothed-normalised axis of GyroY (sg0zmuvGyroY, sg1zmuvGyroY and sg2zmuvGyroY). ENTR values were computed with embedding parameters $m = 6$, $\tau = 8$ and $\epsilon = 1$. R code to reproduce the figure is available from [22].

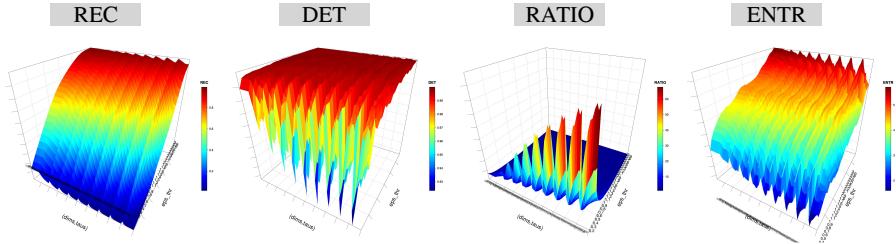


Fig 27. 3D surfaces for RQA metrics. 3D surfaces for REC, DET, RATIO and ENTR values with increasing pair embedding parameters ($0 \geq m \leq 10$, $0 \geq \tau \leq 10$) and recurrence thresholds ($0.2 \geq \epsilon \leq 3$). RQA metrics values for time series of participant p01 using HS01 sensor, HN activity and sg0zmuvGyroZ axis and 500 samples window length. R code to reproduce the figure is available from [22].

noticed with a slight decrease of amplitude in any of the time series conditions which is also presented with ENTR values.

RQA metrics with different parameters

Patterns in RPs and metrics for RQA are independent of embedding dimension parameters [10], however, that is not the case when using different recurrence thresholds. Such changes of recurrence threshold values can modify the patterns in RPs and therefore the values of RQA metrics. We therefore computed 3D surfaces to explore the sensibility and robustness of embedding parameters and recurrence threshold in RQA metrics. Following the same methodology of computing 3D surfaces, we also considered variation of window length size to present RQA metrics dependencies with embedding parameters, recurrence thresholds and window length size.

Conclusions

We conclude that using a different level of smoothness for time series help us to visualise and to quantify the variation of movements between participants using RSSs, RPs and RQA. Also, it is important to mention that some RQA's metrics (e.g. DET and ENTR) are more robust to the effect of smoothness of time series.

Although our experiment is limited to twenty healthy right-handed participants of a range age of mean 19.8 SD=1.39, RQA metrics can quantify human movement variability. With that in mind, we conclude that quantification of human-humanoid imitation activities is possible for participants of different ages, state of health and anthropomorphic features.

In general, activity type, window length and structure of the time series affects the values of the metrics of RQA for which certain RQA metrics are better to describe determined type of movement. Using determined RQA metrics depends on what one wants to quantify, for instance, one can find predictability, organisation of the RPs, dynamics transitions, or complexity and determinism.

Similarly, such differences in time series created differences in each of the RQA metrics, for instance, RATIO and ENTR are helpful to distinguish differences in any of the categories of the time series (sensor, activity, level of smoothness and number of participant), however for certain time series (data from the sensor attached to the robot) seemed to have little variations between each of the participants. The latter phenomena was in a way evidently as robot degrees of freedom did not allow it to move with a wide range of variability.

Future Work

Inertial Sensors

To have more fundamental understanding of nature of signals collected through inertial sensors in the context of human-robot interaction, we are considering to apply derivatives to the acceleration data. We can then explore the jerkiness of movements and therefore the nature of arm movements which typically have minimum jerk [51], its relationship with different body parts [49, 50] or the application of higher derivatives of displacement with respect time such as snap, crackle and pop [52].

RQA

Having presented our results with RQA metrics, we believe that further investigation is required to have more robust metrics. For example, Marwan et al. [32, 34] reviewed different aspects to compute RPs using different criteria for neighbours, different norms (L_1 -norm, L_2 -norm, or L_∞ -norm) or different methods to select the recurrence threshold ϵ , such as using certain percentage of the signal [53], the amount of noise or using a factor based on the standard deviation of the observational noise among many others [32].

Supporting information

S1 Appendix A. Examples of Uniform Time-Delay Embedding Two examples are presented in this appendix: (A.1) using a 20 sample length vector, and (A.2) using a time series from an horizontal movement of a triaxial accelerometer.

A.1. Vector of sample length of $N = 20$.

Given $\{\mathbf{x}_n\}_{n=1}^{20}$ with a sample length $N = 20$, we implement an uniform time-delay embedding matrix, \mathbf{X}_τ^m , with embedding dimension of $m = 5$ and delay dimension of $\tau = 3$ (Eq. (4)). The representation of the uniform time-delay embedding matrix \mathbf{X}_3^5 is as follows

$$\mathbf{X}_3^5 = \begin{pmatrix} \tilde{\mathbf{x}}_n \\ \tilde{\mathbf{x}}_{n-3} \\ \tilde{\mathbf{x}}_{n-6} \\ \tilde{\mathbf{x}}_{n-9} \\ \tilde{\mathbf{x}}_{n-12} \end{pmatrix}^\top \quad (19)$$

The dimension of the uniform time-delay embedding matrix is defined by $N - (m - 1)\tau$ rows and m columns. $N - (m - 1)\tau$ is also the sample length of the delayed copies of \mathbf{x}_n which is equal to eight ($20 - ((5 - 1) * 3) = 8$). Therefore, \mathbf{X}_3^5 can be explicitly represented as

$$\mathbf{X}_3^5 = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 \\ x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} & x_{11} \\ x_7 & x_8 & x_9 & x_{10} & x_{11} & x_{12} & x_{13} & x_{14} \\ x_{10} & x_{11} & x_{12} & x_{13} & x_{14} & x_{15} & x_{16} & x_{17} \\ x_{13} & x_{14} & x_{15} & x_{16} & x_{17} & x_{18} & x_{19} & x_{20} \end{pmatrix}^\top. \quad (20)$$

After transposing \mathbf{X}_3^5 , one can see that the ranges of values of the uniform time-delay embedded matrix are between $((m - 1)\tau) + 1$ to N (for this example from 13 to 20):

$$\mathbf{X}_3^5 = \begin{pmatrix} x_1 & x_4 & x_7 & x_{10} & x_{13} \\ x_2 & x_5 & x_8 & x_{11} & x_{14} \\ x_3 & x_6 & x_9 & x_{12} & x_{15} \\ x_4 & x_7 & x_{10} & x_{13} & x_{16} \\ x_5 & x_8 & x_{11} & x_{14} & x_{17} \\ x_6 & x_9 & x_{12} & x_{15} & x_{18} \\ x_7 & x_{10} & x_{13} & x_{16} & x_{19} \\ x_8 & x_{11} & x_{14} & x_{17} & x_{20} \end{pmatrix} = \begin{pmatrix} \mathbf{X}[13] \\ \mathbf{X}[14] \\ \mathbf{X}[15] \\ \mathbf{X}[16] \\ \mathbf{X}[17] \\ \mathbf{X}[18] \\ \mathbf{X}[19] \\ \mathbf{X}[20] \end{pmatrix}. \quad (21)$$

A.2 Time series from a triaxial accelerometer.

For this example, it is considered a time series of a triaxial accelerometer, (Fig 32(C)), captured from repetitions of a horizontal movements trajectory (Fig 32(A, B)). From Fig 32(C)) is evidently that the $A_y(n)$ is the most affected axis of the accelerometer due to horizontal movement trajectory. With that in mind, we select $A_y(n)$ as the input time series for the uniform time-delay embedding theorem. Therefore, considering that the sample rate of the data is 50 Hz, we have a sample length of $N = 1000$. We then selected $m = 7$ and $\tau = 11$ as the minimum embedding parameters for $A_y(n)$. The uniform time-delay embedding matrix, \mathbf{A}_{y11}^7 , has a dimension of $N - (m - 1)\tau$ (934) rows and m (7) columns and is represented as follows:

$$\mathbf{A}_{y11}^7 = \begin{pmatrix} A_y(n) \\ A_y(n - 11) \\ A_y(n - 22) \\ A_y(n - 33) \\ A_y(n - 44) \\ A_y(n - 55) \\ A_y(n - 66) \end{pmatrix}^\top = \begin{pmatrix} a_y(1) & \dots & a_y(934) \\ a_y(12) & \dots & a_y(945) \\ a_y(23) & \dots & a_y(956) \\ a_y(34) & \dots & a_y(967) \\ a_y(45) & \dots & a_y(978) \\ a_y(56) & \dots & a_y(989) \\ a_y(67) & \dots & a_y(1000) \end{pmatrix}^\top, \quad (22)$$

$$\mathbf{A}_{y_{11}}^7 = \begin{pmatrix} a_y(1) & a_y(12) & a_y(23) & a_y(34) & a_y(45) & a_y(56) & a_y(67) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_y(934) & a_y(945) & a_y(956) & a_y(967) & a_y(978) & a_y(989) & a_y(1000) \end{pmatrix}, \quad (23)$$

$$\mathbf{A}_{y_{11}}^7 = \begin{pmatrix} \mathbf{A}_{y_{11}}^7[67] \\ \vdots \\ \mathbf{A}_{y_{11}}^7[1000] \end{pmatrix}. \quad (24)$$

S1 Appendix B. Inertial Sensors For this work, data were collected using NeMEMSi sensors, which provide triaxial accelerometer, triaxial magnetometer, triaxial gyroscope and quaternions (Fig 33). NeMEMSi sensors were tested against the state-of-the-art device MTi-30 IMU from Xsense. The comparison values between NeMEMSi and MTi-30 in was in terms of standard deviation of noise of each component of Euler angles at a static state were lower than 0.1 degrees. Additionally, the NeMEMSi provides not only a lower-power consumption but also the smaller dimensions against other state-of-the-art brands of IMUs [42]. We refer the reader to check [42] for further details with regard to the IMU such as sample rate, power consumption, orientation, accelerometer, gyroscope, magnetometer, microprocessor, connectivity and form factor.

Acknowledgments

This work was funded by the Mexican National Council of Science and Technology (CONACyT) as part of Miguel P Xochicale's PhD degree at University of Birmingham UK. I would like to thank to Dr. Dolores Columba Perez Flores and Prof. Martin J Russell for their helpful comments to polish the use of the language of mathematics and to Constantino Antonio Garcia Martinez et al. for developing the R package nonlinearTseries that significantly help to accelerate the analysis of the nonlinear time series in this work.

Author Contributions

Conceptualisation MX, CB

Data Curation MX

Formal Analysis MX

Funding Acquisition MX, CB

Investigation MX

Methodology MX

Project Administration MX

Resources CB

Software MX

Supervision CB

Validation MX

Verification MX

Writing - Original Draft Preparation MX

Writing - Review CB

Writing - Editing MX

References

1. Quintana Duque JC. Non-linear Dynamic Invariants based on Embedding Reconstruction of Systems for Pedaling Motion. In: Byshko R, editor. Sportinformatik 2012 : 9. vom 12.- 14. Sept. 2012 in Konstanz : Beiträge. Universität; 2012. p. 28–38.
2. Quintana Duque JC. Non-linear methods for the quantification of cyclic motion [PhD dissertation]. University of Konstanz. Konstanz, Germany; 2016.
3. Samà A, Ruiz FJ, Agell N, Pérez-López C, Català A, Cabestany J. Gait identification by means of box approximation geometry of reconstructed attractors in latent space. Neurocomputing. 2013;121:79–88.
doi:10.1016/j.neucom.2012.12.042.
4. Frank J, Mannor S, Precup D. Activity and Gait Recognition with Time-Delay Embeddings. AAAI Conference on Artificial Intelligence. 2010; p. 407–408.
5. Gómez-García JA, Godino-Llorente JI, Castellanos-Dominguez G. Non uniform Embedding based on Relevance Analysis with reduced computational complexity: Application to the detection of pathologies from biosignal recordings. Neurocomputing. 2014;132(Supplement C):148 – 158.
6. Marwan N. How to Avoid Potential Pitfalls in Recurrence Plot Based Data Analysis. International Journal of Bifurcation and Chaos. 2011;21:1003.
doi:10.1142/S0218127411029008.
7. Stergiou N, Decker LM. Human movement variability, nonlinear dynamics, and pathology: Is there a connection? Human Movement Science. 2011;30(5):869 – 888.
8. Bradley E, Kantz H. Nonlinear time-series analysis revisited. Chaos: An Interdisciplinary Journal of Nonlinear Science. 2015;25(9):097610.
doi:10.1063/1.4917289.
9. Takens F. Detecting strange attractors in turbulence. Dynamical Systems and Turbulence, Warwick 1980: Proceedings of a Symposium Held at the University of Warwick 1979/80. 1981; p. 366–381.
10. Iwanski JS, Bradley E. Recurrence plots of experimental data: To embed or not to embed? Chaos: An Interdisciplinary Journal of Nonlinear Science. 1998;8(4):861–871. doi:10.1063/1.166372.
11. Garland J, Bradley E, Meiss JD. Exploring the topology of dynamical reconstructions. Physica D: Nonlinear Phenomena. 2016;334:49 – 59.
12. Kantz H, Schreiber T. Nonlinear Time Series Analysis. New York, NY, USA: Cambridge University Press; 2003.

13. Packard NH, Crutchfield JP, Farmer JD, Shaw RS. Geometry from a Time Series. *Phys Rev Lett.* 1980;45:712–716. doi:10.1103/PhysRevLett.45.712.
14. Aguirre LA, Letellier C. Modeling Nonlinear Dynamics and Chaos: A Review. *Mathematical Problems in Engineering.* 2009;11(12):1–22.
15. Guastello SJ, Gregson RAM. Nonlinear Dynamical Systems Analysis for the Behavioral Science Using Real Data. CRC Press; 2011.
16. Quintana-Duque J, Saupe D. Evidence of Chaos in Indoor Pedaling Motion using Non-linear Methods. July. Routledge; 2013.
17. Krakovská A, Mezeiová K, I HB. Use of False Nearest Neighbours for Selecting Variables and Embedding Parameters for State Space Reconstruction. *Journal of Complex Systems.* 2015;2015.
18. Casdagli M, Eubank S, Farmer JD, Gibson J. State space reconstruction in the presence of noise. *Physica D: Nonlinear Phenomena.* 1991;51(1):52 – 98.
19. Gibson JF, Farmer JD, Casdagli M, Eubank S. An analytic approach to practical state space reconstruction. *Physica D: Nonlinear Phenomena.* 1992;57(1):1 – 30.
20. Uzal LC, Grinblat GL, Verdes PF. Optimal reconstruction of dynamical systems: A noise amplification approach. *Phys Rev E.* 2011;84:016223. doi:10.1103/PhysRevE.84.016223.
21. Uzal LC, Verdes PF. Optimal Irregular Delay Embeddings. In: Dynamics Days South America 2010. National Institute for Space Research; 2010. Available from: <http://urlib.net/8JMKD3MGP7W/3824342>.
22. Xochicale MP. How Well You Move Dataset; 2018. Available from: <https://github.com/mxochicale/hwum-dataset>.
23. Kennel MB, Brown R, Abarbanel HDI. Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Phys Rev A.* 1992;45:3403–3411. doi:10.1103/PhysRevA.45.3403.
24. Cao L. Practical method for determining the minimum embedding dimension of a scalar time series. *Physica D: Nonlinear Phenomena.* 1997;110:43–50.
25. Emrani S, Gentimis T, Krim H. Persistent Homology of Delay Embeddings and its Application to Wheeze Detection. *IEEE Signal Processing Letters.* 2014;21(4):459–463. doi:10.1109/LSP.2014.2305700.
26. Garcia SP, Almeida JS. Nearest neighbor embedding with different time delays. *Phys Rev E.* 2005;71:037204. doi:10.1103/PhysRevE.71.037204.
27. Fraser AM, Swinney HL. Independent coordinates for strange attractors from mutual information. *Phys Rev A.* 1986;33:1134–1140.
28. Kabiraj L, Saurabh A, Wahi P, Sujith RI. Route to chaos for combustion instability in ducted laminar premixed flames. *Chaos: An Interdisciplinary Journal of Nonlinear Science.* 2012;22(2):023129. doi:10.1063/1.4718725.
29. Garcia CA, Sawitzki G. nonlinearTseries: Nonlinear Time Series Analysis; 2016. Available from: <https://github.com/constantino-garcia/nonlinearTseries>.

30. Pecora LM, Moniz L, Nichols J, Carroll TL. A unified approach to attractor reconstruction. *Chaos: An Interdisciplinary Journal of Nonlinear Science*. 2007;17(1):013110. doi:10.1063/1.2430294.
31. Small M, Tse CK. Optimal embedding parameters: a modelling paradigm. *Physica D: Nonlinear Phenomena*. 2004;194(3):283 – 296. doi:<https://doi.org/10.1016/j.physd.2004.03.006>.
32. Marwan N, Romano MC, Thiel M, Kurths J. Recurrence plots for the analysis of complex systems. *Physics Reports*. 2007;438(5):237 – 329. doi:<https://doi.org/10.1016/j.physrep.2006.11.001>.
33. Eckmann JP, Kamphorst SO, Ruelle D. Recurrence Plots of Dynamical Systems. *EPL (Europhysics Letters)*. 1987;4(9):973.
34. Marwan N, Webber CL. Mathematical and Computational Foundations of Recurrence Quantifications. In: Webber CL, Marwan N, editors. *Recurrence Quantification Analysis: Theory and Best Practices*. 1st ed. Springer, Cham; 2015. p. 3–43.
35. Zbilut JP, Webber CL. Embeddings and delays as derived from quantification of recurrence plots. *Physics Letters A*. 1992;171(3):199 – 203. doi:[https://doi.org/10.1016/0375-9601\(92\)90426-M](https://doi.org/10.1016/0375-9601(92)90426-M).
36. Trulla LL, Giuliani A, Zbilut JP, Webber CL. Recurrence quantification analysis of the logistic equation with transients. *Physics Letters A*. 1996;223(4):255 – 260. doi:[https://doi.org/10.1016/S0375-9601\(96\)00741-4](https://doi.org/10.1016/S0375-9601(96)00741-4).
37. Marwan N. A historical review of recurrence plots. *The European Physical Journal Special Topics*. 2008;164(1):3–12. doi:10.1140/epjst/e2008-00829-1.
38. Apthorp D, Nagle F, Palmisano S. Chaos in Balance: Non-Linear Measures of Postural Control Predict Individual Variations in Visual Illusions of Motion. *PLOS ONE*. 2014;9(12):1–22. doi:10.1371/journal.pone.0113897.
39. Duran ND, Fusaroli R. Conversing with a devil's advocate: Interpersonal coordination in deception and disagreement. *PLOS ONE*. 2017;12(6):1–25. doi:10.1371/journal.pone.0178140.
40. Rhea CK, Silver TA, Hong SL, Ryu JH, Studenka BE, Hughes CML, et al. Noise and Complexity in Human Postural Control: Interpreting the Different Estimations of Entropy. *PLOS ONE*. 2011;6(3):1–9. doi:10.1371/journal.pone.0017696.
41. Gouaillier D, Hugel V, Blazevic P, Kilner C, Monceaux J, Lafourcade P, et al. Mechatronic Design of NAO Humanoid. In: *Proceedings of the 2009 IEEE International Conference on Robotics and Automation. ICRA'09*. Piscataway, NJ, USA: IEEE Press; 2009. p. 2124–2129. Available from: <http://dl.acm.org/citation.cfm?id=1703775.1703795>.
42. Comotti D, Galizzi M, Vitali A. neMEMSi: One step forward in wireless attitude and heading reference systems. In: *2014 International Symposium on Inertial Sensors and Systems (ISISS)*; 2014. p. 1–4.
43. Shoaib M, Bosch S, Incel OD, Scholten H, Havinga PJM. Complex Human Activity Recognition Using Smartphone and Wrist-Worn Motion Sensors. *Sensors*. 2016;16(4). doi:10.3390/s16040426.

44. Anguita D, Ghio A, Oneto L, Parra X, Reyes-Ortiz JL. A Public Domain Dataset for Human Activity Recognition Using Smartphones. In: 21th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2013; 2013.
45. Press WH, Teukolsky SA, Vetterling WT, Flannery BP. Numerical Recipes in C (2Nd Ed.): The Art of Scientific Computing. New York, NY, USA: Cambridge University Press; 1992. Available from: https://www2.units.it/ipl/students_area/imm2/files/Numerical_Recipes.pdf.
46. Savitzky A, Golay MJE. Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*. 1964;36:1627–1639.
47. Schafer RW. On the frequency-domain properties of Savitzky-Golay filters. In: 2011 Digital Signal Processing and Signal Processing Education Meeting (DSP/SPE); 2011. p. 54–59.
48. signal R developers. signal: Signal processing; 2014. Available from: <http://r-forge.r-project.org/projects/signal/>.
49. de Vries JJP, Visser GHA, Prechtl HFR. The emergence of fetal behaviour. I. Qualitative aspects. *Early Human Development*. 1982;7(4):301 – 322. doi:[https://doi.org/10.1016/0378-3782\(82\)90033-0](https://doi.org/10.1016/0378-3782(82)90033-0).
50. Mori H, Kuniyoshi Y. Is the developmental order of fetal behaviors self-organized in an uterine environment? In: 2012 IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL); 2012. p. 1–2.
51. Flash T, Hogan N. The coordination of arm movements: an experimentally confirmed mathematical model. *Journal of Neuroscience*. 1985;5(7):1688–1703. doi:[10.1523/JNEUROSCI.05-07-01688.1985](https://doi.org/10.1523/JNEUROSCI.05-07-01688.1985).
52. Eager D, Pendrill AM, Reistad N. Beyond velocity and acceleration: jerk, snap and higher derivatives. *European Journal of Physics*. 2016;37(6):065008.
53. Letellier C. Estimating the Shannon Entropy: Recurrence Plots versus Symbolic Dynamics. *Phys Rev Lett*. 2006;96:254102. doi:[10.1103/PhysRevLett.96.254102](https://doi.org/10.1103/PhysRevLett.96.254102).

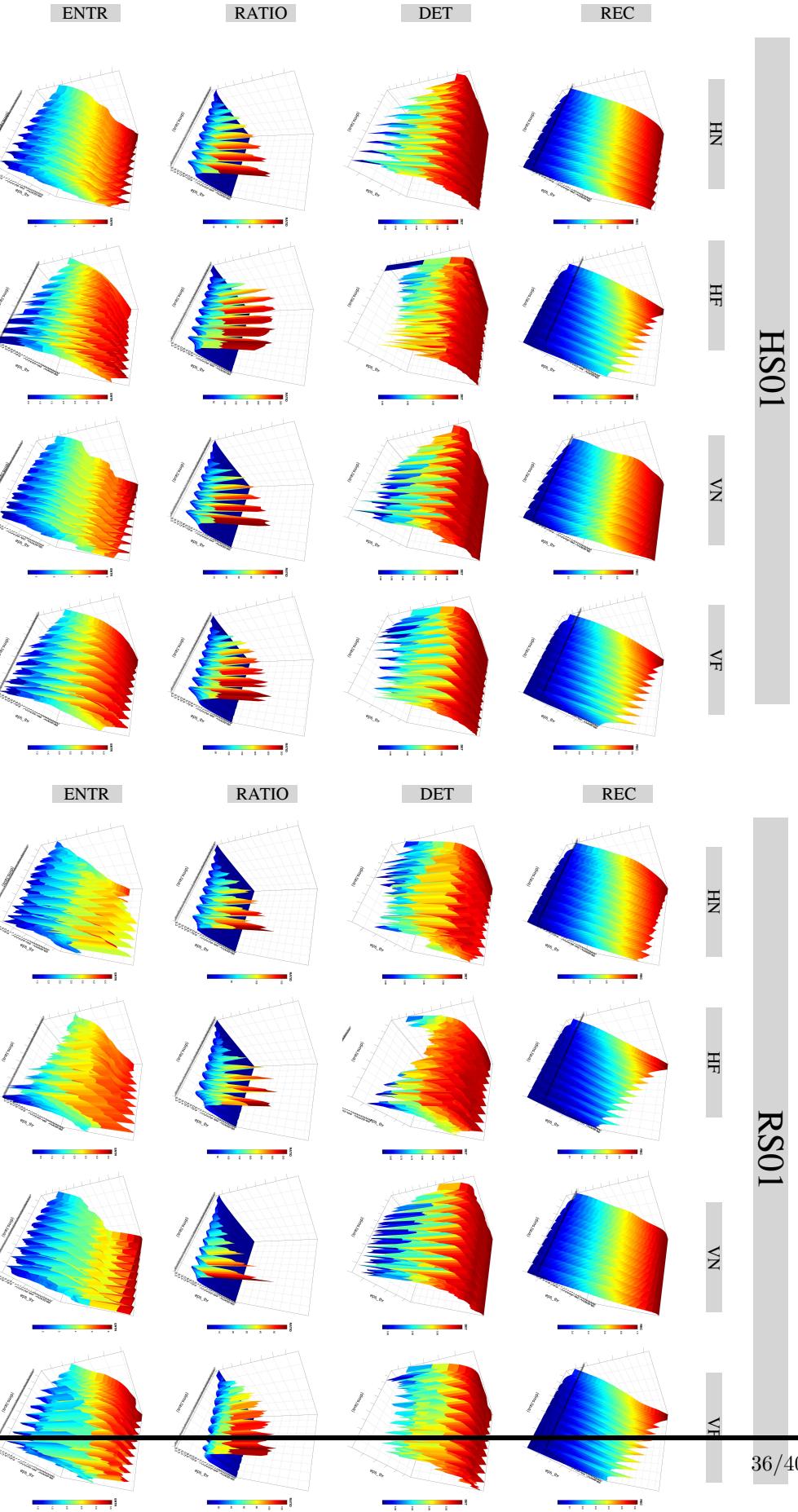


Fig 28. 3D surfaces of RQA metrics for sensors and activities. 3D surfaces with four rows and four columns. The first row contains the 3D surfaces for HS01, and the second row contains the 3D surfaces for RS01. The columns represent the RQA metrics: ENTR, RATIO, DET, and REC. The rows represent the activities: HN, HF, VN, and VF. The vertical axis for each plot represents the RQA metric value, ranging from blue (low) to red (high). The horizontal axes represent the activity categories.

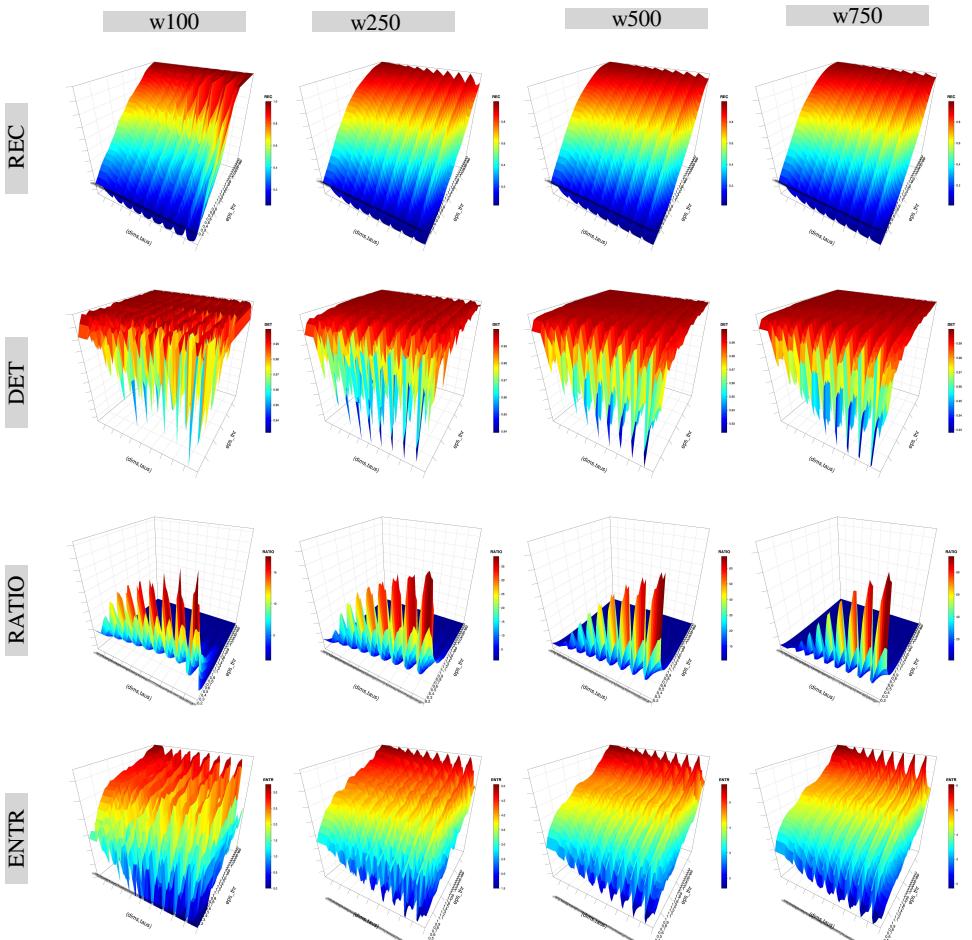


Fig 29. 3D surfaces for RQAs metrics with four window lengths. 3D surfaces of RQAs metric values with increasing embedding parameters and recurrence thresholds are for four window lengths (w100, w250, w500 and w750). RQA metrics values are for time series of participant p01 using HS01 sensor, HN activity and sg0zmuvGyroZ axis. R code to reproduce the figure is available from [22].

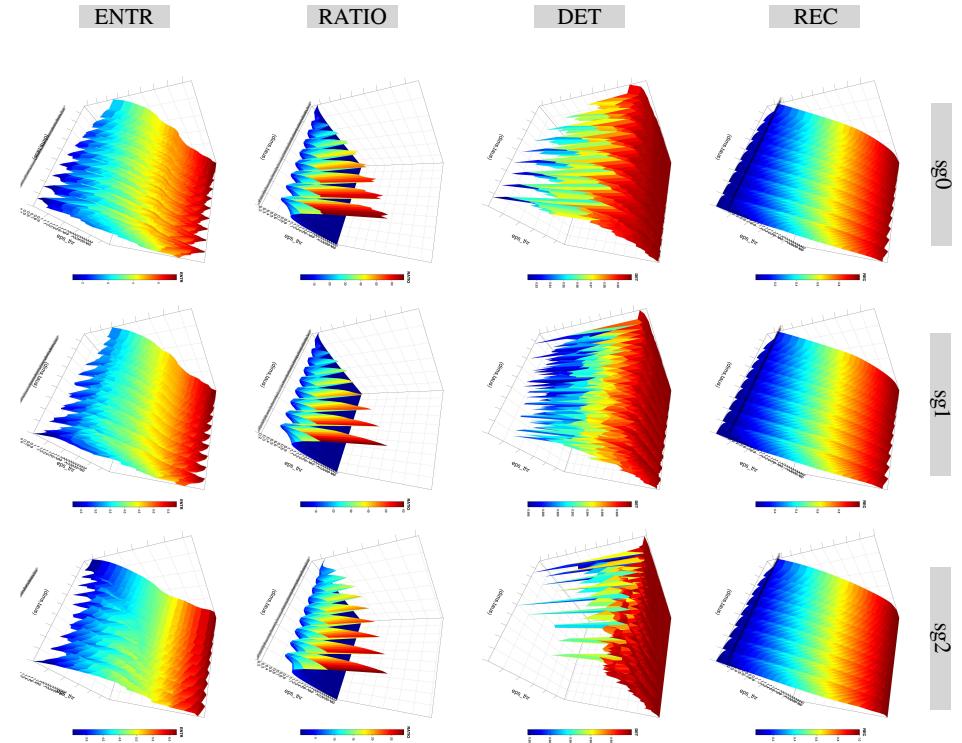


Fig 30. 3D surfaces for RQA metrics with three levels of smoothness. 3D surfaces of RQAs metric values with increasing embedding parameters and recurrence thresholds are for three levels of smoothness (sg0zmuvGyroZ, sg1zmuvGyroZ and sg1zmuvGyroZ). RQA metrics values are for time series of participant p01 using HS01 sensor, HN activity and 500 samples window length. R code to reproduce the figure is available from [22].

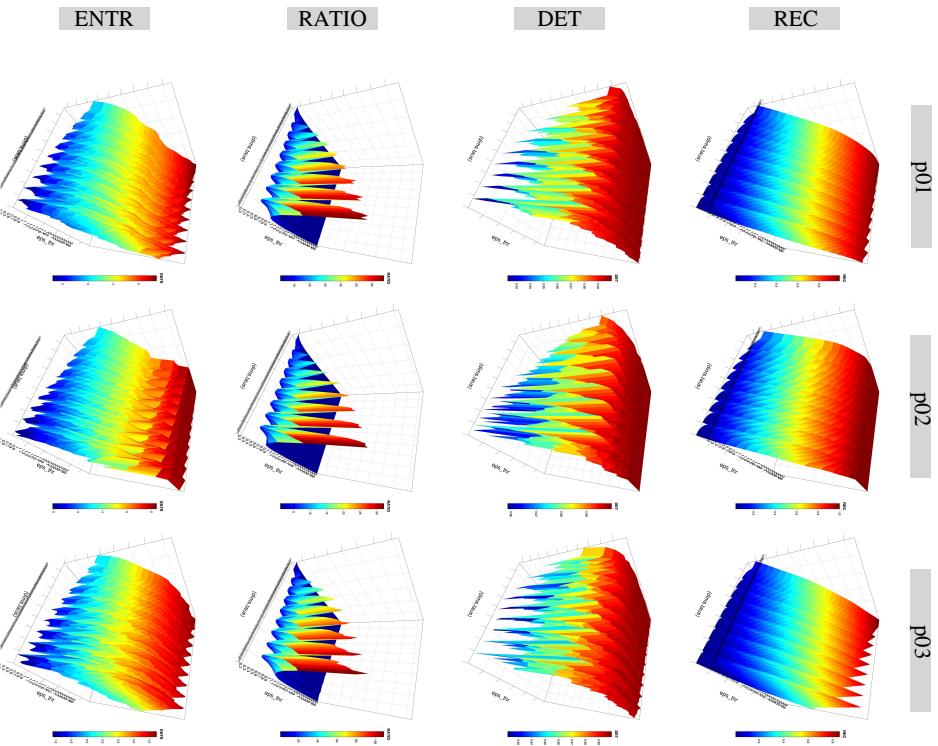


Fig 31. 3D surfaces for RQA metrics with three participants. 3D surfaces of RQAs metric values for participants p01, p02 and p03 with increasing embedding parameters and recurrence thresholds. RQA metrics values are for time series of HS01 sensor, HN activity and 500 samples window length. R code to reproduce the figure is available from [22].

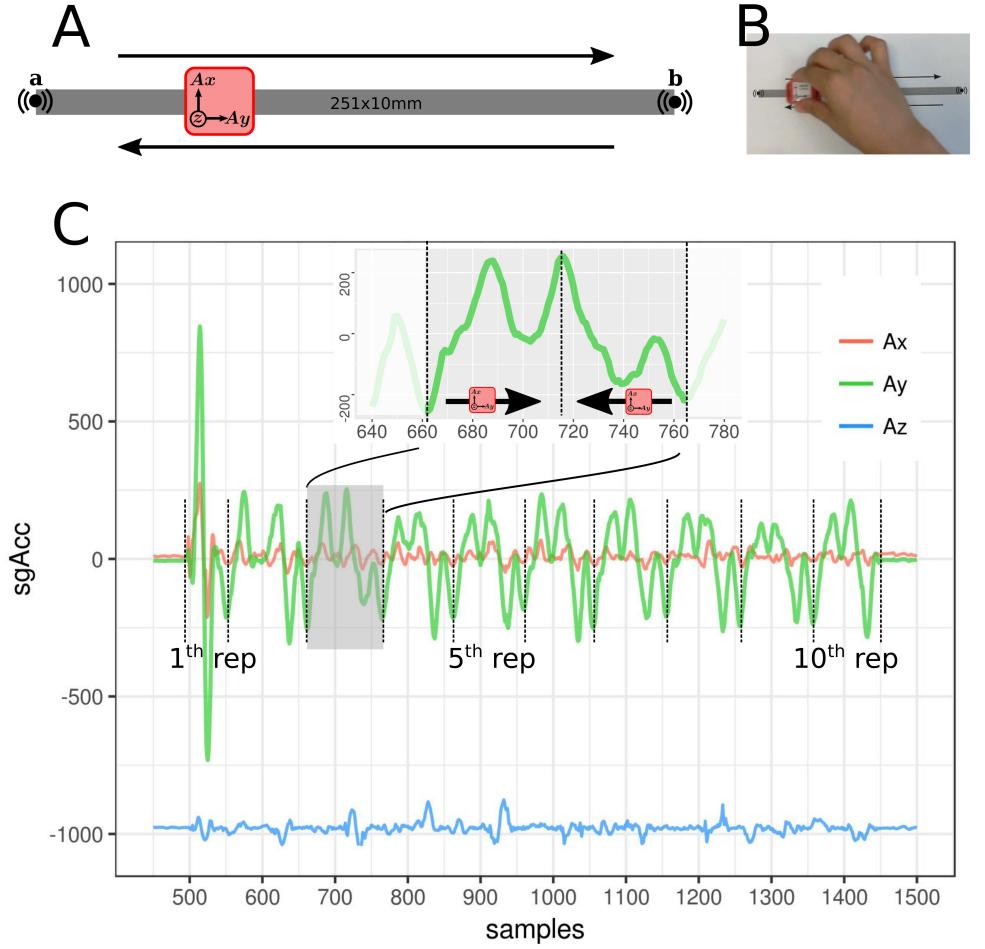


Fig 32. Time series for horizontal movement trajectory. (A). Triaxial accelerometer (in red) is moved repetitively across a line of 251 mm from point **a** to **b** and then from **b** to **a**. The points **a** and **b** indicate when a click sound is produced. (B). Person's hand holding and moving the sensor horizontally across the line. (C). Time series for the triaxial accelerometer ($A_x(n)$, $A_y(n)$, $A_z(n)$) for ten repetitive horizontal movements across a line. The top time series only shows A_y axis which corresponds to one cycle of the horizontal movement and the black arrows represent the movement's direction of the accelerometer with respect to the produced time series. R code to reproduce the figure is available from [22].

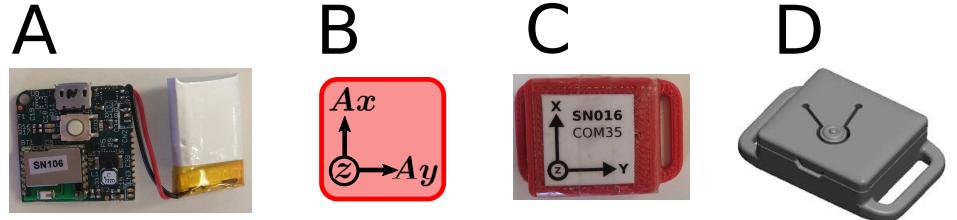


Fig 33. neMEMSi IMU sensor. (A) Printed Circuit Board (PCB) with 165mAh battery, (B) axis orientation, (C) real case, and (D) 3D model for the case.