D3.2 – EnTimeMent platform and software libraries for multi-time analysis, entrainment, and prediction - Phase 1

Project No	GA824160
Project Acronym	EnTimeMent
Project full title	ENtrainment & synchronization at multiple TIME scales in the MENTal foundations of expressive gesture
Instrument	FET Proactive
Type of action	RIA
Start Date of project	1 January 2019
Duration	48 months

Distribution level	[PU][1]
Due date of deliverable	Month 12
Actual submission date	February 2020
Deliverable number	3.2
Deliverable title	EnTimeMent platform and software libraries for multi-time analysis, entrainment, and prediction - Phase 1
Type	ORDP (Open Research Data Pilot)
Status & version	
Number of pages	
WP contributing to the deliverable	3
WP / Task responsible	UNIGE
Other contributors	ALL
Author(s)	UNIGE, Qualisys
EC Project Officer	Teresa De Martino
Keywords	Computational models, Software libraries, Movement analysis and prediction, Machine learning

[1] **PU** = Public, **PP** = Restricted to other programme participants (including the Commission Services), **RE** = Restricted to a group specified by the consortium (including the Commission Services), **CO** = Confidential, only for members of the consortium (including the Commission Services).

Contents

Abbreviations

EU	European Union
EC	European Commission
WP	Work Package

1 Introduction

This deliverable describes the preliminary framework (Phase 1) of the project technology platform. It consists of a number of datasets, distinguished in terms of complexity (temporal scales) and scenarios, and a number of hardware and software modules constituting the architecture of the project platform. The project technology platform is available online on the project repository.

2 Activities in Phase 1

2.1 Datasets

In this section we present both novel and pre-existing datasets adopted in the first phase of EnTimeMent.

2.1.1 Feasibility studies dataset "A Tempo! First Project Workshop" (UNIGE, EuroMov, UCL, Qualisys)

The definition of research requirements produced a set of multimodal recordings. The objectives were the following:

(i) to explore scenarios and experimental setups,
(ii) to experiment, concolidate, and validate the project multimodal recording platform, and
(iii) to create interactive live demos on project objectives presented at the First Project Workshop "A Tempo!", September 2019.

This dataset includes recordings on the following:

(i) A complex improvised movement including two temporal scales (long-term quality and local movement features), and an annotated analysis of individual motor signature;
(ii) A chronic pain standard scenario (sit to stand), including sonification
(iii) Individual Vs Group motor signature, involving a human and an avatar;

Videos showing the multimodal recordings are available in the project portal: http://www.casapaganini.org/atempo/ and https://entimement.dibris.unige.it/events/22-a-tempo-first-project-workshop). The dataset is available in the online project repository.

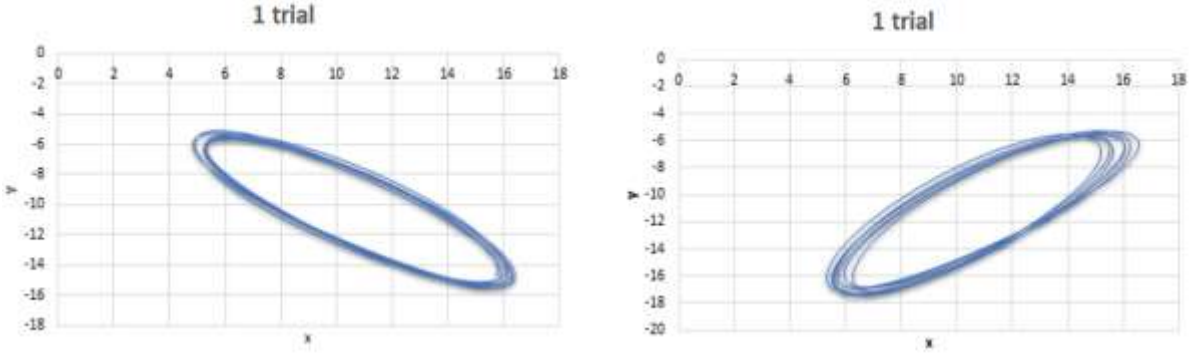
2.1.2 Singularity experiment dataset (Ellipsis) (IIT-UNIGE)

S*ingularity* is a high-level feature enabling to distinguish different people by analyzing the way they move, write, or perceive an event (for example, auditory or visual). How these actions are performed is different from individual to individual and, for this reason, we speak about singularity. An experiment was proposed to measure this high-level feature. The possibility to measure singularity can contribute to many application fields, from clinical to entertainment and consumer applications.
The experiment is grounded on the Two-Third Power Law:

$$V(t) = k * r^{\beta}$$

where *v(t)* is velocity, *k* is a constant and *r* is the ellipse radius of curvature. If β were different from each person it could be sufficient for a classification of singularity.
The hypothesis is that beta is not enough to provide a measure of singularity, and therefore we need to individuate and measure other features and apply data analysis and machine learning techniques to obtain a correct measure of this high-level feature.

The data is collected using a tablet where 8 participants perform a number of repetitions of ellipses. People try several times the same ellipse. The final outcomes will be 10 different ellipses for trial.
Each participant carries out 6x10 trials. For each trial, 10 ellipses are drawn, of which the first two and the last one are discarded in order to avoid noise due to the data acquisition phase or simply the sensitivity with which the participant draws the ellipses. The result is that for each trial, only 7 "clean" ellipses are preserved. Each trial consists of 10 executions of an ellipse at the same condition. Each participant executes the 10 trials in 6 (2x3) different conditions: 2 inclinations (+ or - 45 degrees respect to ordinate axis) and 3 drawing speeds (slow, normal, quick). In the figure below it is possible notice two different trials where inclinations change.



Summarizing, the cardinality of the dataset is given by:

$$8\ participant\ x\ 2\ angles\ x\ 3\ speed\ x\ 10\ trials\ x\ 7\ ellipses = 3360\ ellipses$$
$$of\ which\ 3227\ are\ available$$

For each ellipse, features available are *(x,y)* positions, *Velocity*, *Pressure*, *Curvature* and the acquisition time *t*. It is possible to distinguish two types of features: those deriving directly from the tablet device and those obtained from the first ones. *Velocity v* is obtained by the formula

$$v = \frac{space}{time}.$$

Curvature is extracted by building a circle inscribed in a triangle passing through three consecutive points.

2.1.3 Violinist dataset from TELMI EU ICT project (expert vs non expert experiment)

This dataset is part of the TELMI multimodal dataset (Volpe et al., 2017): it includes a selection of multimodal recordings to investigate which low-level motion features can explain the difference between the performance of a professional violinist and of a student. The original objective in TELMI was to develop real-time student assistive interactive technologies. In EnTimeMent, this is an example of dataset to investigate *individual motion signature*.
The dataset includes recordings of 7 violin players, 4 experts and 3 beginners. Three standard exercises are considered: a *scale* (G major, 4 octaves detaché, from ABRSM), one *repertoire* piece (*Salut d'amour* Op.12, Edgar), and one *right hand technique* (String crossing, Op.1 n.13, Kreutzer). 14 raw physical movement features are extracted, selected by properly analysing the state-of-art of violin pedagogy and musicians' motion and biomechanical analysis. Finally, we added further features suggested by expert violinists from the Royal College of Music of London. The features list is the following:

1) Mean Shoulders' Velocity

2) Shoulder low back asymmetry
3) Upper body kinetic energy
4) Left/Right shoulder height
5) Bow-violin incidence
6) Distance of low part of the Bow to the Violin
7) Distance of middle part of the Bow to the Violin
8) Distance of upper part of the Bow to the Violin
9) Hand-violin incidence
10) Left/Right side neck angle
11) Left/Right wrist roundness

The study was leaded computing from row mocap data, the 14 raw features using EyesWeb platform. The dataset cardinality can be summarized in the following table:

	Violinist	Total exercises	Scale exercise	Repertoire exercise	Technique exercise
Experts	4	11	3	4	4
Beginners	3	8	3	3	2
Total	7	19	6	7	6

The column "Total exercises" is the sum of the 3 columns "Scale exercise", "Repertoire exercise" and "Technique exercise". From the above table, the total number of exercises is 19, since 2 missing exercises – 1 for each class of violinists skill-level. In particular, 1 scale exercise is missing from "expert musicians" class and 1 technique exercise from the beginner one.

Volpe, G., Kolykhalova, K., Volta, E., Ghisio, S., Waddell, G., Alborno, P., ... & Ramirez-Melendez, R. (2017, September). A multimodal corpus for technology-enhanced learning of violin playing. In Proceedings of the 12th Biannual Conference on Italian SIGCHI Chapter (pp. 1-5).

2.1.4 Emo-Pain dataset: movement and EMG section (UCL)

This existing dataset was chosen to develop experiments on multiple temporal scales in EnTimeMent, following the feasibility study presented in section 2.1.1.

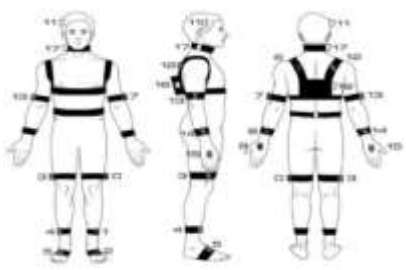
The EmoPain dataset (Aung et al., 2015) was created to support the development of physical rehabilitation technology. It comprises multimodal data: facial and vocal expressions, full body motion capture (Mocap) and electromyography (EMG) data. Data were collected from 21 people suffering from low back chronic pain (CLBP) and 18 healthy participants while they engaged in physical rehabilitation exercises that simulate typical everyday actions. For each exercise, two levels of difficulty were used to naturally elicit a wider range of pain-related behaviour. Each participant took part both trials at least once.

The easier trial consisted of: 1) standing on the preferred leg for five seconds initiated at the time of the subject's own choosing, repeated three times, 2) sitting still on a bench for thirty seconds, 3) reaching forwards with both hands as far as possible while standing, 4) standing still for thirty seconds, 5) sitting to standing initiated at the time of the subject's own choosing, repeated three times, 6) bending down to touch toes and 7) walking. In the *difficult* trial, four of the exercises were modified to increase the level of physical demand and possibly of anxiety: 1) standing on the preferred leg for five seconds initiated upon instruction repeated three times and then on the non-preferred leg in the same manner, 3) reaching forwards with both hands as far as possible while standing holding a 2 kg dumbbell, 5) sitting to standing repeated three times initiated upon instruction, and 6) walking as before while carrying one 2 kg weight in each hand, starting with bending down to pick up the weights.

Participants were asked to wear four wireless surface electromyographic (EMG) probes, and a motion capture suit consisting of eighteen Inertial Measuring Units (IMU). The data

collection included also a camera rig supporting five face level cameras and two extra cameras for different perspectives and a wireless microphone. In EnTimeMent, we focus on the motion capture and EMG data. Twelve IMU sensors were placed on rigid limb segments; one on the hip, centre of the torso, and one on each shoulder, neck and on the head totalling eighteen sensors (see figure 2). The IMUs were connected in parallel and each returned 3-D Euler angles sampled at 60 Hz. Two wireless EMG adhesive probes (BTS FREEEMG 300) (see figure 2) were placed on the trapezius muscles bilaterally. Two further EMG probes were placed on the lumbar paraspinal muscles. The EMG data was recorded at a rate of 1 kHz.

Two physiotherapists expert in CLBP rehabilitation and two psychologists with expertise in pain related behaviour labelled frame by frame the full body videos gathered during the session. They labelled each framed according to the presence of any of the six protective behaviours listed in Table 2. These behaviour categories are based on the Keefe & Block pain protetctive behavior framework (Keefe FJ, Block., 1982) The data were also labelled using self-reported levels of pain and anxiety at the end of each individual exercise.



(a)

(b)

Fig. 2 (a) Mocap IMU and (b) EMG sensor positions (Taken from (Aung et al., 2015)

Table 2: Protective behaviour definitions (refined from (Keefe et al., 1982)

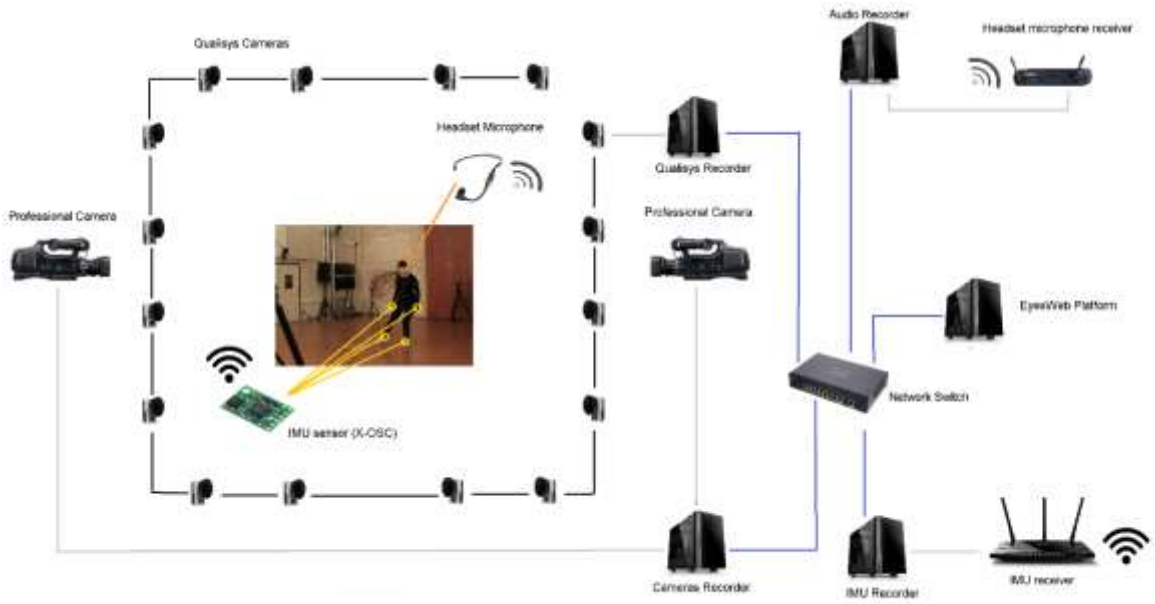
Type	Definition
Guarding or stiffness	Stiff, interrupted or rigid movement. It cannot occur while motionless
Hesitation	Stopping part way through a continuous movement with the movement appearing broken into stages
Bracing or support	Position in which a limb supports and maintains an abnormal distribution of weight during a movement which could be done without support.
Abrupt action	Any sudden movement extraneous to the intended motion; not a pause as in hesitation.
Limping	Asymmetric cadence, stride, timing and inequality of weight-bearing during movements.
Rubbing or stimulating	Massaging touching an affected body part with another body part, or shaking hands or legs.

Keefe FJ, Block AR. "Development of an observation method for assessing pain behaviour in chronic low back pain patients". Behaviour Therapy, 13(4), 1982.

M. S. Aung, S. Kaltwang, B. Romera-Paredes, B. Martinez, A. Singh, M. Cella, M. Valstar, H. Meng, A. Kemp, M. Shafizadeh, et al. The automatic detection of chronic pain-related expression: requirements, challenges and the multimodal emopain dataset. IEEE transactions on affective computing, 7(4):435–451, 2015.

2.2 Hardware and Software platform modules

2.2.1 Overall architecture of the project platform



The current recording architecture is represented in the figure, and it is composed by:

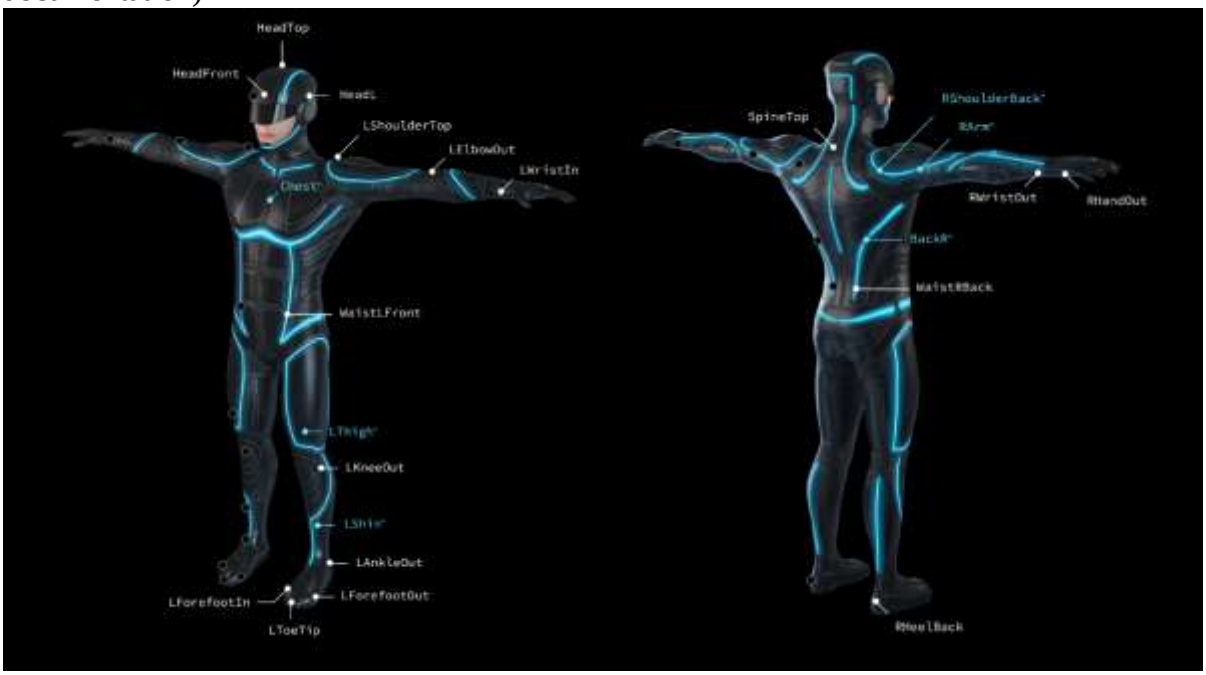
- 16 OQUS Camera (mixed setup with 700+, 700, 300)
- 2 Professional Cameras
- 1 Respiration Microphone
- 2-4 Accelerometers

The system allows to add biometric sensors or other devices. The architecture components are described in the following.

Synchronization system
Synchronization is guaranteed by the EyesWeb software platform. EyesWeb is used to generate the reference clock used by all other devices. The generated reference clock is sent to the recorders in a format compatible with each specific device.
The synchronization signal is called SMPTE. **SMPTE timecode** is a set of cooperating standards to label individual frames of video or film with a time code defined by the Society of Motion Picture and Television Engineers.
The Qualisys Motion Capture system receives SMPTE encoded in an audio stream. Also the two broadcast video-cameras and the *Audio Recorder* use SMPTE encoded as an audio signal. The *IMU Recorder* receives the reference clock via network, through the OSC protocol.
To guarantee synchronization EyesWeb keeps track, for every recorded frame or sample, of the SMPTE when the data was received. As a matter of facts, not all streams can be hardware-synchronized (e.g., with a genlock signal), thus, a software synchronization is performed by EyesWeb by keeping track of the time at which the data was received in a separate file, and using such information when playing back the data. IMU sensors is an example of a device which is synchronized in this way.

Qualisys Recorder
This Recorder is dedicated to the processing and the recording of data stream from Qualisys cameras and computes the motion tracking by Qualisys Track Manager (QTM). In the last

version of the software, QTM provides both markers and skeleton informations, in real time and after the recording, exporting the session, in C3D or TSV format.
In order to improve the skeleton solver potential, is suggested to use one of the dedicated marker sets for Animation (Animation Marker Set or Sports Marker Set – from Qualisys documentation)



This recorder module streams synchronized data to the EyesWeb Platform, using the SMPTE signal sent by EyesWeb software to the Qualisys SyncBox .

Video Recorder
2 Professional cameras are recorded synchronized with the SMPTE signal. The ambient audio is recorded in the left audio channel of the video. The SMPTE signal is recorded in the right audio channel.

Audio Recorder
A radio headset microphone sends the audio signal to this recorder. The track contains the respiration informations and the voice of the user.

IMU Recorder
The user has 2 IMU on the wrists and 2 on the ankles. The recorder stores all the synchronized signals from IMUs:

- 3 axis acceleration
- 3 axis gyroscope
- 3 axis magnetometer

EyesWeb Platform
The EyesWeb platform is used to:

- Synchronize all the devices
- Manage the recordings
- Playback the synchronized signals
- Analyse in real time and post-recording the signals

A new feature developed in EyesWeb for EnTimeMent is the skeleton receiver. EyesWeb allows to receive the skeleton both in real time and from .qtm exported files. The skeleton stream is composed by a hierarchical set of segment positions (in mm) and the segment rotation data expressed as quaternions.

2.2.2 Overview of Sensor System for Chronic Pain Data Collection in Participant Homes

The system is composed of four main components:

Body Movement Sensors
The body movement sensors are wearable, and easily portable IMUs that record multi-dimensional angle data for full-body (or partial-body, depending on how many units are worn) joints over the course of a given movement or activity.

Video cameras
Video cameras will be used to additionally capture body movements of participants for the purpose of observer labelling of the data and/or to enable the researcher better interpret other sensor data.

Physiological sensors
In addition to the behavioural sensors described above, where possible, physiological signals (e.g. electrodermal activity) of the participant will be additionally recorded.

Speaker and Microphone
We plan to use a pair of wireless earbuds with a microphone embedded in them, to provide regular pre-recorded self-report prompts to and capture self-report response from the participant while they complete a given activity.

2.3 Software Libraries

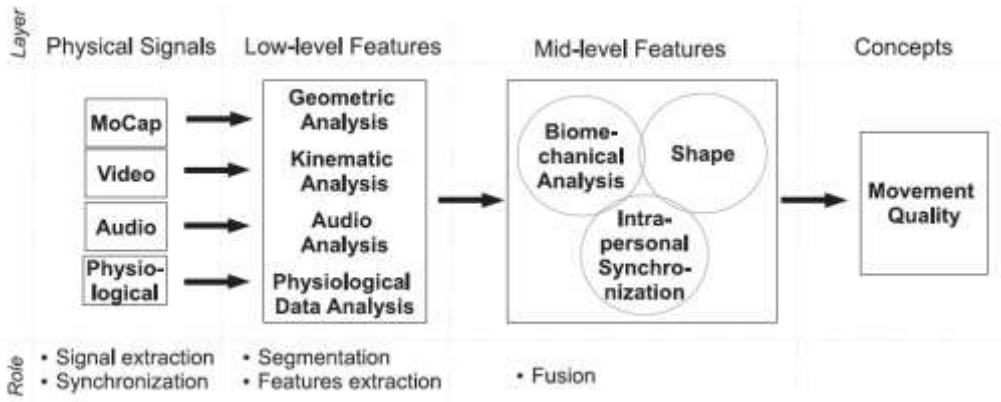
Different data analytic and machine learning technologies can be employed exploiting information captured by the current generation of motion capture and movement analysis systems and empower them, with these computational models, to achieve a novel generation of time-aware multisensory motion perception and prediction systems. In particular, EnTimeMent will investigate two main families of method.
The first families considered are the traditional Machine Learning models (Ensemble Methods, Kernel Based Methods, Shallow Neural Network, Bayesian Methods, etc.) [1] where feature are extracted from data based on the knowledge of the problem, during the feature extraction phase, in order to create a rich and expressive description in order to even detect and characterize the body movements even discover the multiple time scales that need to be detected in order to well characterize them. In order to reach this goal features must be engineered so to capture information at multiple resolutions of time and space.
The second family of methods, called deep learning models [2], removes the features engineering phase and replaces it with another Machine Learning model, usually a (deep)-neural network, able to automatically learn the rich and expressive representation of the data automatically from the data itself.
In order to reach this goal, Recurrent Neural Network families are the most suited architectures able to automatically learn a rich and expressive representation able to express and describe the movement at multiple time scales. In fact, RNN can, with different techniques (Long Short

Term Methods, Clockwork Networks, Attention Mechanism, Convolution in Time, etc) are able, in principle, to capture the multiple time scales of a phenomena.
With respect to the first family the deep learning models, in general, require much more data to be trained and are more prone to overfitting. For this reason it is always necessary to exploit both families of methods to ensure the good quality of the data, the meaningfulness of the extracted information, avoiding to capture spurious correlations, the statistical robustness and consistency of the methodology. In this sense, rigorous statistical procedures for model selection and error estimation [3] together with a posteriori analysis of the learned models with feature ranking/selection, attention maps or, more generally, some techniques to make model explainable [4], should be employed.

[1] Shalev-Shwartz, S. and Ben-David, S. Understanding machine learning: From theory to algorithms. Cambridge University Press. 2014.
[2] Goodfellow, I. and Bengio, Y. and Courville, A. Deep learning. MIT Press. 2016.
[3] Oneto, L. Model Selection and Error Estimation in a Nutshell. Springer. 2020.
[4] Molnar, C. Interpretable machine learning. Lulu. com. 2019.

2.3.1 Software libraries for analysis of qualities of movement (first version)

Our computational framework for the analysis of movement quality in full-body physical activities is shown in the following figure



Biomechanical efficiency means whether movement is efficient according to biomechanical laws (e.g., minimum jerk (Flash and Hogans 1985) or two-thirds power law (Viviani and Schneider 1991)). Biomechanical efficiency helps avoid injuries and waste of energy. In sport, biomechanically efficient movements approach theoretical maximal effectiveness (in the sense of velocity or force) and minimalize energy effort.
Shape concerns postural aspects of the movement performance, for example, whether the appropriate posture is maintained. It focuses on the static shapes (postures) that the body takes as well as how the body changes from one shape to another during movement.
Intrapersonal synchronization focuses on body limb coordination and on the time relationship between the movement of different parts of the body, for example, whether arms move synchronously.
Our definition of movement quality focuses on the technical aspects of a movement performance. We exclude subjective factors that may influence the perception of movement quality at the individual level, for example, cultural background of the observer. Moreover, the correct performance of a movement (i.e., a high-quality movement) is usually related to the

goals of the movement (e.g., an excellent performance in terms of intrabody synchronization may evoke positive aesthetic feelings in spectators), but movement quality is not evaluated in goal-oriented terms.
Modeling goal-oriented effectiveness of a physical activity (e.g., whether a particular movement conveys an intended affective meaning or not (Pasch et al. 2009)) is out of the scope of this work.
In order to measure movement quality, we designed a computational framework consisting of four layers and several modules (see previous Figure). This is grounded on a conceptual framework conceived for analysis of expressive content conveyed by full-body movement and gesture (Camurri et al. 2004, 2016b).
The first layer of our framework–the *Physical Signals Layer*–consists of modules that capture and preprocess sources of data from different modalities. Below are more details about these modules.
—*MoCap Module*: It retrieves the 3D positions of body joints from a motion capture system, applies basic processing techniques (e.g., signal filtering), and computes basic kinematic features, such as velocity or acceleration.
—*Video Module*: It receives video streams from one or more video cameras and/or RGB-D sensors and possibly runs basic video-processing techniques (e.g., background subtraction and motion tracking).
—*AudioModule*: It captures audio streams from one ormore environmental or on-bodymicrophones and possibly runs basic audio-processing techniques (e.g., denoising). In our system, we focus on nonverbal audio.
—*Physiological Signals Module*: It retrieves data from physiological sensors — such as respiration or skin conductance response sensors — and applies basic processing techniques (e.g., signal filtering).
The *Low-Level Features Layer* consists of modules that compute basic features describing movement quality. Such modules are regrouped into different sets performing different analyses:
—*Geometric Analysis*, for example, computes distances and angles between joints.
—*Kinematic Analysis* computes movement trajectories as well as basic kinematic information (e.g., acceleration peaks and kinetic energy).
—*Audio Analysis* performs extraction and analysis of acoustic features, for example, Mel Frequency Cepstral Coefficients (MFCC) (Davis and Mermelstein 1980; Zheng et al. 2001), main frequency F0, or volume.
—*Physiological Data Analysis* performs physiological signal processing and analysis, for example, signal peak detection and signal periodicity.
At the *Mid-Level Features Layer*, quality is analyzed with respect to its three major components discussed in the previous section: *Biomechanical Analysis*, *Shape*, and *Intrapersonal Synchronization*.
Finally, on the top, the *Concepts Layer* is composed of one module that computes overall movement quality. The aim of this level is to merge the different facets of quality into one meaningful value. Since overall movement quality is related to the goals of each specific activity, different fusion models should be used for different activities. For example, Intrapersonal Synchronization may weight more in classic ballet, whereas Biomechanical Efficiency may weight more in sport activities that require a lot of physical effort.

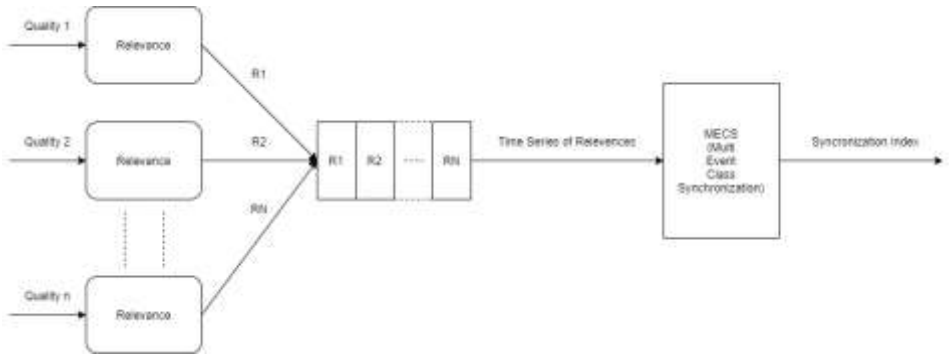
For each quality, a specific temporal scale was selected, drawing from psychophysical studies on time perception and mental simulation (Fraisse 84).
After the extraction of the quality in a specific temporal scale, we evaluate the Relevance of the quality.

Relevance is an analysis primitive that can be computed on any movement quality *X*. The idea is to consider the histogram of *X* and to estimate the "distance" between the bin in which lies the current value of *X* and the bin corresponding to the most frequently occurring values of *X* in the "past".

Given the time series x=x1,…,xn of *n* observations of movement quality *X* (xn is the latest observation), Relevance is computed as follows:

- we compute HistX, the histogram of *X*, considering $\sqrt{n}$ equally spaced intervals; we call occ_i the number of occurrences in interval *i* ($i=1,\ldots,\sqrt{n}$) of the elements of *x*,
- let iMAX be the interval corresponding to the highest bin (i.e., the bin of highest number of occurrences), and let occMAX be the number of occurrences in interval iMAX,
- let i_n be the interval to which x_n belongs to, and let occ_n be the number of occurrences in i_n,
- we compute $D1=|iMAX - i_n|$,
- we compute $D2=occMAX-occ_n$,
- we compute Relevance as $D1*D2*$, where α is a constant positive real normalization factor.

A time series of the resulting different time scale relevances is analyzed by the MECS algorithm, to evaluate if exist a synchronization between the detected relevances



2.3.2 Synchronization among temporal scales: the MECS algorithm

Multi-Event-Class Synchronization (MECS, paper in preparation) is a technique to measure synchronization between events detected in multiple time series. Synchronization is computed in terms of temporal alignment (within a time window) of the events occurring in the time series. After grouping events into classes, synchronization is computed within a class, i.e., between events belonging to the same class (*intra-class synchronization*) and between classes, i.e., between events belonging to different classes (*inter-class synchronization*). Additionally, events can be combined into *macro-events* on which synchronization is measured. A *macro-event* is an aggregation of events that satisfy some constraints. A relevant example of macro-event is a specific sequence of events. Events and macro-events can be grouped into *macro-classes* and synchronization can be computed within and between them.

With respect to other existing techniques, MECS brings substantial extensions, which enable modeling a broader collection of real-life phenomena. Differently from the well-known Event

Synchronization (ES) algorithm (Quian Quiroga et al, 2002), MECS always provides a normalized output (thus fixing a relevant drawback of ES). It can also compute synchronization of more than two multivariate time series, but unlike the technique proposed by Kreuz and colleagues (2009), which considers only one single event class, it can manage multiple classes of events detected in the time series. With respect to the work by Iqpal and Riek (2016), MECS additionally manages computation of synchronization between events belonging to different classes (i.e., inter-class synchronization). Finally, differently from all the algorithms mentioned above, MECS introduces the computation of synchronization between multiple event classes over multiple (more than two) time series, handling both macro-events and macro classes.

MECS was specifically created with the purpose of studying multimodal human-human and human-machine interaction. MECS can be applied to a large variety of problems and, in particular, it can be used by human centered systems, multi-modal interfaces for human-machine interaction, or to study multi-modal expressive behaviors of individuals as well as social signals in groups. As an example, suppose that we are interested in measuring the level of motor coordination of the members of a group consisting of some users performing a motor task (e.g., a fitness exercise). Synchronization between the movements of the participants in the group can be taken as a cue for coordination. In this case, the time series may describe the motor activity of the users, events are instances of movements the users perform, and classes may identify specific kinds of movements (for example “step performed”, “object grabbed”, “object released”, and so on). Then, MECS allows us to compute intra- and inter-class synchronization between instances of movements of such classes. If motor behavior is instead characterized by movement qualities, a higher-level analysis can be performed by identifying instances of expressed movement qualities as events (for example “a hesitant step”, “a fluid arm movement”, “an energic jump”, and so on). Then, MECS can compute synchronization between displays of expressed movement qualities both at intra-personal and at inter-personal (i.e., between the participants) level, thus enabling to study coordination both at the level of motor activity and at the level of expressed movement quality.

In the framework of EnTimeMent, we expect to use MECS for measuring synchronization of events in time series of data characterizing movement at different time scales both at intra-personal and at inter-personal level. Moreover, the capability of MECS of handling macro-events can provide us with a useful tool to investigate synchronization at multiple time scales since a macro-event at one time scale (e.g., a sequence of events at a low-level time scale) may correspond to one single event at another (higher-level) time scale.

P. Alborno, M. Mancini, R. Niewiadomski, S. Piana., A. Camurri, G. Volpe, “The Multi-Event-Class Synchronization (MECS) Algorithm”, in preparation.

R. Quian Quiroga, T. Kreuz, and P. Grassberger, “Event synchronization: a simple and fast method to measure synchronicity and time delay patterns,” Physical review E, vol. 66, no. 4, article 041904, 2002.

T. Kreuz, D. Chicharro, R. G. Andrzejak, J. S. Haas, and H. DI. Abarbanel, “Measuring multiple spike train synchrony,” Journal of neuroscience methods, vol. 183, no. 2, pp. 287–299, 2009.

T. Iqbal and L. D. Riek, “A Method for Automatic Detection of Psychomotor Entrainment,” IEEE Transactions on Affective Computing, vol. 7, no. 1, pp. 3–16, 2016.

2.3.3 Automated measure of the origin of movement

In (Kolykhalova et al., 2020) we proposed an approach to perform an automated analysis of the perceived origin of full-body human movement, i.e., the point at which such movement appears to be originated from the point of view of an observer.

The approach, which is grounded on both cooperative game theory and graph theory, consists of the following steps:

The human body is modeled as an undirected graph, in which the vertices are the joints, whereas the edges are both physical and non-physical connections between such body joints.The edges are associated with weights, whose values depend on a suitable feature, extracted from motion capture data.

Physical links represent connections of consecutive physical body joints, such as the forearm. Instead, non-physical links model dependencies between joints that are not physically connected. More specifically, they are derived by correlations observed in the chosen movement feature between such joints. For example, a hand moving towards the head followed by a sort of movement response of the head in the same direction determines, in the proposed approach, the presence a non-physical link between the hand and the head. Therefore, non-physical links play the role of potential bridges, joining body parts that are not directly connected within the skeletal structure, but exhibit correlated dynamics during the movement performed.

Starting from the graph representing the skeletal structure augmented by non-physical links, in (Kolykhalova et al., 2020) a mathematical game (Maschler et al., 2013) is defined. IN such a game, the vertices (i.e., the body joints) are the players, whereas the edges model communication channels (over which movement can propagate) between such players. Body movement is therefore studied via such a game constructed on the body graph. Since both the vertices and the edges contribute to the overall movement, a cooperative game is chosen. Then, the Shapley value is exploited, which is a classical solution concept from cooperative game theory, able to provide a ranking of the players according to their relevance in the game. Such a value is computed for all the body joints and is adopted as a measure its relevance. In such a way, one estimates how much each joint contributes to the way in which a specific movement-related feature is transferred among the joints themselves. It is worth noting that the use of cooperative game theory to provide relevance measures for players does not limit to situations in which the players are modelled as rational/intelligent entities. For instance, the Shapley value was used in network analysis as a measure of network centrality (Michalak et al., 2013), and in the machine learning literature to assess the importance of different features (Cohen et al., 2007).

In the approach proposed in (Kolykhalova et al., 2020), we search for joints that separate clusters, where each cluster is characterized by similar values of a movement feature. In order to clarify this, let us consider a situation in which one moves an arm, whereas all the other body parts are at rest. In this case, the shoulder corresponding to that arm may be interpreted as a quite relevant joint because, although being at rest (in one cluster), it plays, in a sense, a relevant

role in the control of the arm movement (being the arm in another cluster). Hence, the proposed method tends to attribute large relevance to a vertex that connects two clusters of joints, or even a larger number of such clusters. Here, each cluster represents a subset of connected joints associated with similar values of a movement feature, and is identified by applying a suitable clustering technique (spectral clustering, in the specific case). Finally, the output of cluster analysis is used to construct an auxiliary graph containing only edges joining different clusters, on which the final cooperative game model (and then the computation of the Shapley value) is based. An additional filtering step is used to record joints that have been evaluated as the most relevant ones for a prespecified number of repetitions of the procedure.

The possibility to know, moment by moment, which joint(s) are the most representative in the ongoing full-body movement represents a precious information for the automated analysis of expressiveness in movement. For example, the joints with the highest Shapley values are candidate to be the perceived origin of movement propagating in the body. They can also provide useful cues to detect which parts of the body are most relevant for the analysis of expressive movement and worth to be observed in details by means of further analysis techniques (possibly at a finer scale), as well as to inform automated techniques of movement prediction.

Finally, the method of movement analysis proposed in (Kolykhalova et al., 2020) is evaluated therein against a data set of about one hundred fragments of motion capture recordings, which constitute a repository of stimuli of expressive movement useful also for further research studies on movement analysis. Validation of the proposed approach includes an online survey (based on the data repository) in which participants with different levels of expertise in dance took part.

Exploiting this approach to investigate automatically movement features associated with expressive gesture communication can allow the development of multimodal interfaces based on full-body human interaction, which are also capable to support non-verbal expressive, affective, and social communication.

Possible developments, able to overcome current limitations of the method, include its application to a more complex skeletal structure (for which each cluster of joints is associated to a specific joint in the simpler 20-joint skeletal structure used in that work), making it possible to analyze movement in parallel at a finer interacting spatio-temporal scale in a multiple-scale approach (in line with the objectives of EnTimeMent). In this way, one could compare the Shapley value of a joint in the simpler structure with the sum of the Shapley values of the associated joints in the more complex structure (a smaller Shapley value would be expected for each of the latter joints).

S. Cohen, G. Dror, and E. Ruppin, "Feature selection via coalitional game theory," Neural Computation, vol. 19, no. 7, pp. 1939-1961, 2007.

K. Kolykhalova, G. Gnecco, M. Sanguineti, G. Volpe, and A. Camurri, "Automated analysis of the origin of movement: An approach based on cooperative games on graphs," submitted, 2020.

M. Maschler, E. Solan, and S. Zamir, Game Theory. Cambridge, UK: Cambridge University Press, 2013.

T. P. Michalak, K. W. Aadithya, P. L. Szczepa´nski, B. Ravindran, and N. R. Jennings, "Efficient computation of the Shapley value for game-theoretic network centrality," Journal of Artificial Intelligence Research, vol. 46, pp. 607-650, 2013.

2.3.4 Software application for the sonification and visualisation of neural network attention scores

The BANet neural network (Wang et al. 2019) provides weights that indicate the joint groups to which it has paid most attention in reaching its conclusion of protective or non-protective behavior. In order to expose this information in a multi-modal fashion for exploration by a physiotherapist and patient, an application is under development to provide multi-temporal sonification of the movement attention over time. Multi-temporality is supported through the use of musically coherent works divided into multiple channels. The gain of each channel is controlled by the relative score for attention, thus permitting musical time to continue separately from movement time. Movement can be played forward or backward or freely manipulated, and scaled in time to permit exploration. The sonification is complemented by a visual representation of a figure decorated with Bezier curves whose line weight is proportional to the attention score of the respective joint group. A proof of concept has shown that changes in the attention score can be seen and heard appropriately and ongoing work is beginning to create the capability for sequential and parallel dyadic sonification and visualization, additionally incorporating 3d animation.

C. Wang, M. Peng, T. A. Olugbade, N. D. Lane, A. C. de C. Williams, and N. Bianchi-Berthouze. "Learning Temporal and Bodily Attention in Protective Movement Behavior Detection." In 2019 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW), pp. 324-330. IEEE, 2019.

2.3.5 Model selection and Error Estimation

Model selection (MS) and Error Estimation (EE) deal with the problem of tuning and assessing the performance of a learning algorithm (Oneto et al., 2019).
One of the possible way is though resampling techniques which rely on a simple idea: the original dataset D_n is resampled once or (n_r) many times, with or without replacement, to build three independent datasets called learning, validation and test sets, L, V, T. Note that $L \cap V = \emptyset, L \cap T = \emptyset \; and \; V \cap T = \emptyset$ each time. Then, in order to select the best combination of the hyperparameters H in a set of possible ones for a given algorithm A_H or, in other words, to perform the MS phase, we looking for the minimum error over the choice of the hyperparameters for the algorithm A_H. Since the data \in L are independent from the ones in V the idea is that H^*- the best choice of hyperparameters – should be the set of hyperparameters which allows to achieve a small error on a data set that is independent from the training set.
Then, in order to evaluate the performance of the optimal model which is $f_A^* = A_{H^*}(D_n)$ or, in other words, to perform the EE phase, the following procedure has to be applied:

$$M(f_A^*) = \frac{1}{n}\sum^{n_r} M(A_{H^*}(L \cup V), T)$$

Where $M(f_A^*)$ is the desired metric for the optimal model f_A^*.
Since the data in $L \cup V$ are independent form the ones in T, $M(f_A^*)$ is an unbiased estimator of the true performance, measured with the metric M, of the final model.
Furthermore these two methods are very simple to implement, they do not require particular or significant software libraries to be implemented. However, sample codes will be provided, where it will be possible to view the techniques mentioned above enriching the machine learning code library of the ETM project.

Oneto, L.: Model Selection and Error Estimation in a Nutshell. Springer (2019)

2.3.6 Feature Ranking

Once the models are built it is required to investigate how these models are affected by the different features used in the model identification phase in order to understand if the models have also a foundation which relies on the underline phenomena or if the model just captures spurious correlations (Guyon et al ., 2003, Calude et al., 2017). This procedure is called Feature Ranking (FR) and allows to detect if the importance of those features, that are known to be relevant from a physical perspective, are appropriately taken into account by the learned models. The failure of the computational model to properly account for the relevant features might indicate poor quality in the measurements or spurious correlations. FR therefore represents an important step of model verification, since it should generate consistent results with the available knowledge of the phenomena under exam.
Several measures are available for feature importance in machine learning. One approach is the one based on the Permutation Importance or Mean Decrease in Accuracy (MDA), where the importance is assessed for each feature by removing the association between that feature and the target. This is achieved by randomly permuting (Good et al. 2013) the values of the feature and measuring the resulting increase in error. The influence of the correlated features is also removed.
Another FR technique are the Attention Networks which allow to visualize through an heatmap the weight – and then the importance – of each feature for (deep) neural-network models.
The feature ranking techniques can be multiple and much greater than the two mentioned above as examples. It will depend on the machine learning model used to find the best technique to highlight the importance of the features. A library for the machine learning code is under construction and will be enriched with the continuation of the project where the various techniques will be analyzed and structured.

Calude, C.S., Longo, G.: The deluge of spurious correlations in big data. Foundations of science 22(3), 595–612 (2017)

Good, P.: Permutation tests: a practical guide to resampling methods for testing hypotheses. Springer Science & Business Media (2013)

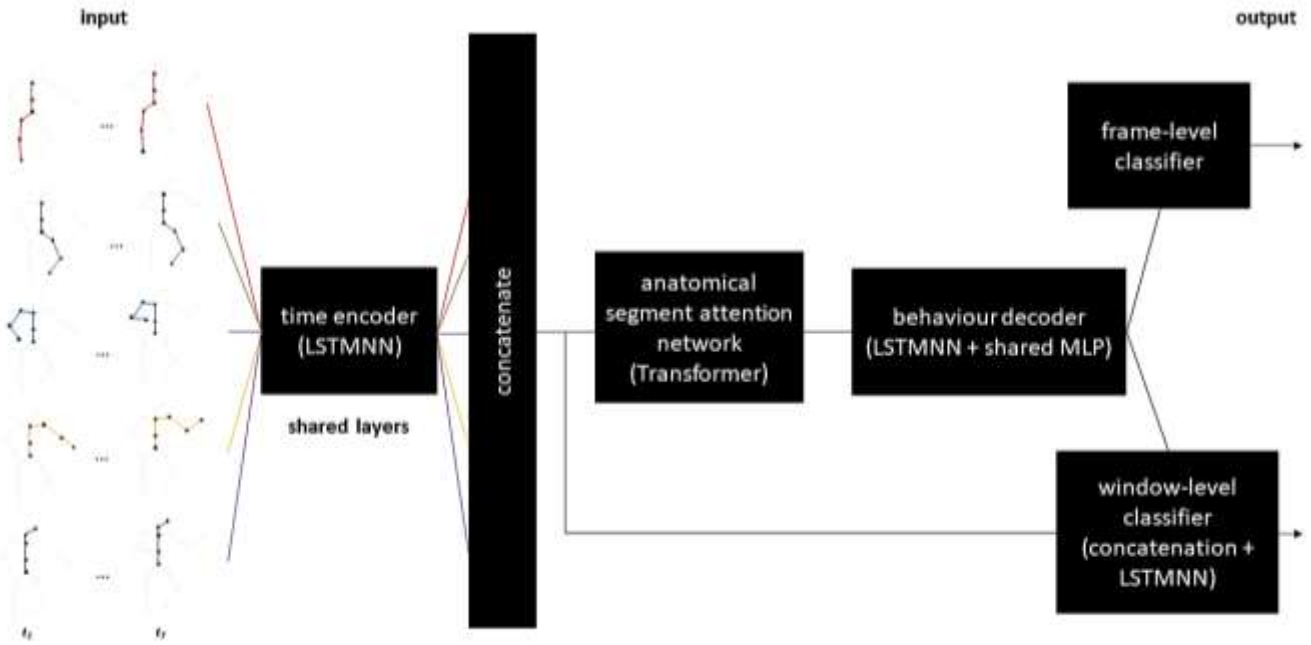
Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning Research 3(Mar), 1157–1182 (2003)

2.3.7 Multi-Time Neural Network (MTNN) Architecture

We propose a neural network architecture (named Multi-Time Neural Network) for learning movement behaviour at multiple time scales of interpretation, e.g. frame-level versus window-level, and further based on segment-distributed encoding of low-level temporal body movement information.

The architecture draws from the multitask learning paradigm and consists of four main modules: 1) a shared time encoder module (shared by multiple anatomical segments) implemented using Long Short-Term Memory Neural Networks (Hochreiter and Schmidhuber 1997 and Gers et al. 2000), LSTMNNs, which are standard for timeseries data processing in machine learning; 2) an attention module which we implement based on the transformer architecture of (Vaswani et al. 2017); 3) a behaviour decoder module based on LSTM as well as fully connected network layers such as in a multilayer perceptron (MLP); and 4) a set of classifiers predict the same behaviour (or affective state) but at different time scales.

The figure below provides an overview of the MTNN architecture for classification at two timescales, i.e. at the frame level and at the window level. In this case the window-level classifier is implemented based on concatenation operation and LSTM layers.



F. A. Gers, J. Schmidhuber, and F. Cummins. (2000). "Learning to forget: Continual prediction with LSTM." Neural Computation, 12(10), 2451–2471.

S. Hochreiter and J. Schmidhuber (1997). "Long Short-Term Memory." Neural Computation, 9(8), 1735–1780.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. (2017). "Attention is all you need." Advances in neural information processing systems, 5998-6008.