Dipartimento di Informatica, Bioingegneria,
Robotica ed Ingegneria dei Sistemi

Automated Analysis of Synchronization in Human Full-body Expressive Movement

by

Paolo Alborno

Theses Series **DIBRIS-TH-2018-XX**

DIBRIS, Università di Genova

Via Opera Pia, 13 16145 Genova, Italy http://www.dibris.unige.it/

Università degli Studi di Genova

Dipartimento di Informatica, Bioingegneria,
Robotica ed Ingegneria dei Sistemi

Ph.D. Thesis in Computer Science and Systems Engineering
Computer Science Curriculum

Automated Analysis of Synchronization in Human Full-body Expressive Movement

by

Paolo Alborno

February, 2018

Dottorato di Ricerca in Informatica ed Ingegneria dei Sistemi
Indirizzo Informatica
Dipartimento di Informatica, Bioingegneria, Robotica ed Ingegneria dei Sistemi
Università degli Studi di Genova

DIBRIS, Univ. di Genova
Via Opera Pia, 13
I-16145 Genova, Italy
`http://www.dibris.unige.it/`

Ph.D. Thesis in Computer Science and Systems Engineering
Computer Science Curriculum
(S.S.D. INF/01)

Paolo Alborno
DIBRIS, Università di Genova
Date of submission: 15 February 2018

Title: Automated Analysis of Synchronization in Human Full-body Expressive Movement

Advisor: Camurri Antonio, Volpe Gualtiero
Dipartimento di Informatica, Bioingegneria, Robotica ed Ingegneria dei Sistemi
Università di Genova
Ext. Reviewers:

Abstract

The research presented in this thesis is focused on the creation of computational models for the study of human full-body movement in order to investigate human behavior and non-verbal communication. In particular, the research concerns the analysis of synchronization of expressive movements and gestures. Synchronization can be computed both on a single user (intra-personal), e.g., to measure the degree of coordination between the joints' velocities of a dancer, and on multiple users (inter-personal), e.g., to detect the level of coordination between multiple users in a group. The thesis, through a set of experiments and results, contributes to the investigation of both intra-personal and inter-personal synchronization applied to support the study of movement expressivity, and improve the state-of-art of the available methods by presenting a new algorithm to perform the analysis of synchronization.

Table of Contents

Chapter 1

Introduction

Contents

1.1 Overview

A growing interest on the part of both the scientific community and industry towards systems that benefit of a very particular communication channel: *expressivity*, is undeniable. These typologies of human-machine interfaces can not be limited at processing verbal communication. Many studies proved that voice messages are only one of the channels that human beings use to express themselves effectively. Non-verbal modalities are needed to complement and reinforce the message to be communicated. In addition to the vocal component, non-verbal communication includes: prosody and expressiveness of speech, facial expressions, body language and other channels that depend on physiological/biometric signals. The specific interest of this research is directed to the automated analysis of how humans communicate through full-body movements.

1.1.1 Non-verbal communication

According to social psychologist Michael Argyle [Arg13], during a conversation, we mainly make use of: facial expressions, visual or gaze contact, gesticulation, posture, touch and spatial behaviour. Body language is partly inborn, and partly dependent on the processes of socialization. The mechanisms from which non-verbal communication flows are very similar in all cultures, even if distinct cultures tend to rework non-verbal messages differently.

According to linguists, body movements (especially facial expressions) convey more than 55% of the message [Meh72] in the sense that the effectiveness of the message depends only to a small extent on the literal meaning of what is said, and the way in which this message is perceived is heavily influenced by non-verbal communication factors. It is therefore necessary to create systems capable of decoding multi-modal signals.

Multi-modal signals are the product of coupling signals produced by different sensory modalities (sight, sound, touch, motion), that are integrated by the nervous system to produce higher-level information. *Multi-modal signal processing* is a research area related to how to extract, recognize, interpolate, and finally interpret information regarding human intentions, behaviour, and communication. The processing of multi-modal signals is in close contact with many research fields and is intrinsically multi-disciplinary, due to the nature of those kind of signals (i.e., a combination of physical dynamics and mental representations).

1.2 Research Focus

This thesis focuses on the design and development of methods and algorithms to efficiently measure the time relationships (as *synchronization*) in multi-modal signals with particular reference to *non-verbal expressive gestures and communication cues*.

The **analysis of synchronization** of expressive movements can be performed to different typologies of signals and provides quantitative measurements of, for example, how much two users are moving together or if they are communicating with the same expressive content. Two main approaches are possible:

- **Analysis of intra-personal synchronization**: the analysis is performed in the context of the personal sphere in order to detect expressive cues of full-body movements.
- **Analysis of inter-personal synchronization**: the analysis is performed in the context of the social sphere and the object of the study is a group of people considered as a single "organism". The analysis of synchronization can be used to measure the group coordination and understand the group dynamics and their temporal evolution.

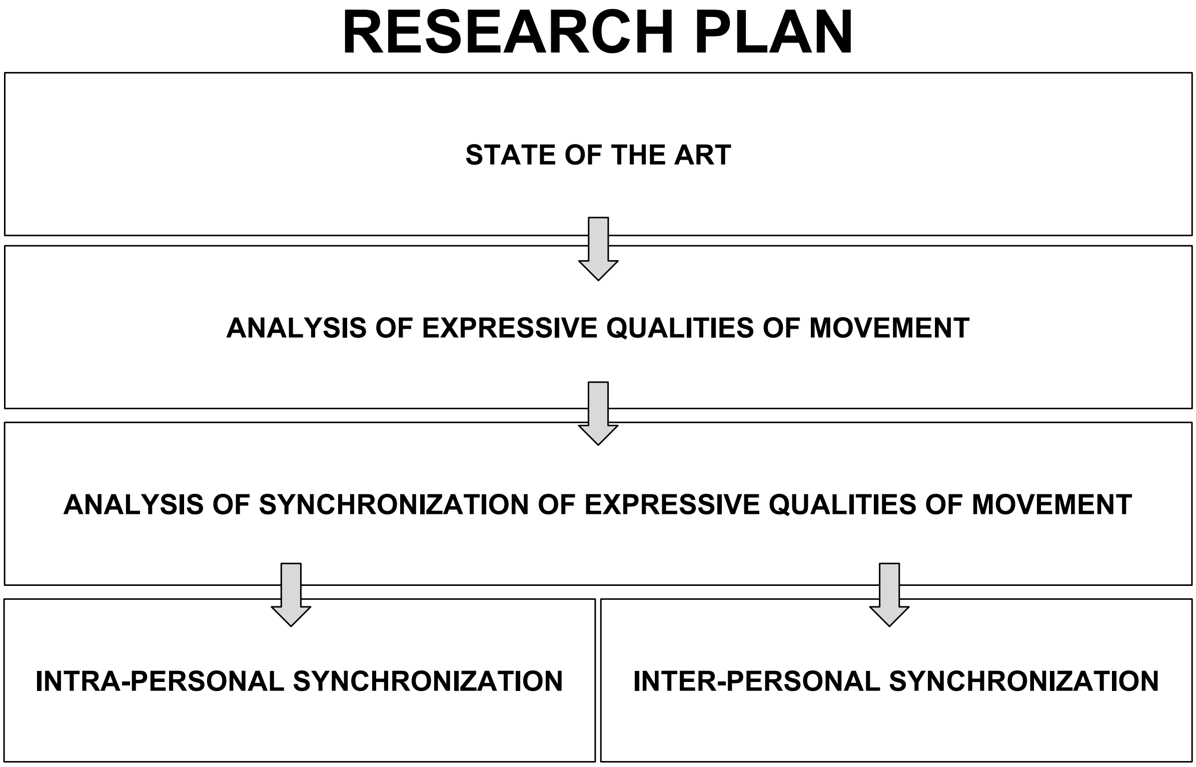

RESEARCH PLAN
STATE OF THE ART
ANALYSIS OF EXPRESSIVE QUALITIES OF MOVEMENT
ANALYSIS OF SYNCHRONIZATION OF EXPRESSIVE QUALITIES OF MOVEMENT
INTRA-PERSONAL SYNCHRONIZATION
INTER-PERSONAL SYNCHRONIZATION

Figure 1.1: The development of the research during the three years.

1.3 Methodology

The activity described in this thesis aims at the creation and refinement of computational models for non-verbal communication channels in order to improve the techniques to study human behavior and use their information content in innovative multi-modal interfaces. In particular, the presented research concerns the analysis of synchronization of expressive gestures and movements. Expressive gestures have been little considered because their expressive potential has only recently been recognized, unlike facial or vocal expressions, which have a more established background of studies, experiments, benchmarks, prototypes and scientific publications.

1.4 Document structure

The structure of the thesis follows the research plan shown in Figure 1.1.

After an introductory part describing the state of the art, the document is divided in conceptual blocks, which are mainly covering the following topics:

- Analysis of expressive qualities of movement
- Analysis of synchronization of expressive qualities of movement
- Case studies and experiments on inter-personal and inter-personal synchronization

The document is divided into several chapters, the content of each chapter is summarized below:

- **Chapter 2: State of the Art**

  Chapter 2 introduces the state of the art of the thesis, divided into three parts. In the first, the differences between gesture and expressive gesture are highlighted. The next section is dedicated to the Laban Movement Analysis theory, a conceptual model recently adapted to study the expressive content of generic movements. Then, the framework for the analysis of the expressivity of full-body movements ([CVP+16a]), which has been adopted and extended through the work presented in this thesis, is illustrated. The second part of the chapter concerns the analysis of synchronization in human movements and illustrates how it can be applied to investigate the movement's expressive qualities. A section regarding a specific topic is presented: musical entrainment. In this scenario, inter-personal synchronization analysis is used to support the study of a more complex phenomenon: entrainment between musicians. In the third and final part, an overview on the basic concepts and techniques to measure synchronization, from several typologies of signals and domains, is presented. Such techniques have been chosen according to different criteria (difficulty of application, linearity vs non-linearity and application domain etc.).

- **Chapter 3: A methodological approach to study the expressive movement qualities**

  The chapter opens with a theoretical section describing the methodological approach that have been followed to develop the definition and perform the analysis of expressive qualities of movement. Since the presented research was largely carried out in the context of the DANCE EU ICT H2020 Project, an introduction to the project context is given. In the final part of the chapter, two studies, performed within the context of DANCE, are presented. Both the studies can be considered as application instances of the approach explained at the beginning of the chapter, i.e., each study illustrates how, starting from meetings between interdisciplinary experts, the definition of an expressive quality and conceptual model with relative algorithms for automatic extraction, have been developed and validated.

- **Chapter 4: The Multi Event Class Synchronization algorithm**

  In this chapter, the Multi Event Class Synchronization (*MECS*), a novel algorithm to perform analysis of synchronization, is illustrated.

- **Chapter 5: Intra-personal synchronization**

  The main experimental results that were obtained by performing the analysis of synchronization of expressive qualities of movement, are illustrated in this chapter. Three studies are presented: while the first and second study were driven in the former period to MECS algorithm development (and therefore are making use of the techniques presented in the state-of-the-art), the third presents an application of the new algorithm on multi-modal data.

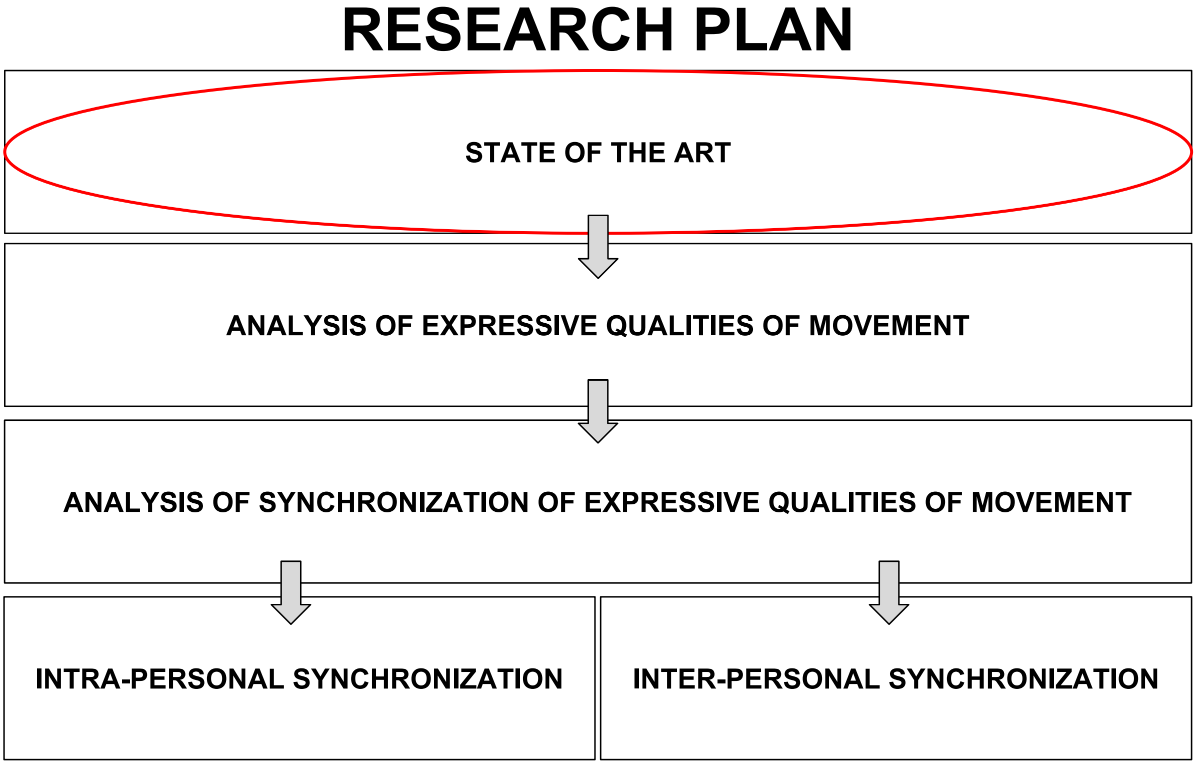
- **Chapter 6: Inter-personal synchronization**

  The last part of the thesis is focused on the study of musical entrainment. The content of Chapter 6 is the result of the studies carried out in the context of the international project IEMP (that is presented at the beginning of the chapter). The project is focused on the study of in inter-personal coordination, synchronization and entrainment in group music-making. Group music-making is a distinctive mode of human social interaction: it is a widespread activity that showcases the remarkable capacity for precision and creativity demonstrated in the coordination of rhythmic behavior between individuals. The chapter presents a pilot study aimed at investigating how the analysis of synchronization can support and contribute to the study of musical entrainment, when applied to a set of motion features that extracted from video recordings. In the last part of the chapter, a study on the robustness of a system to support the automated analysis of inter-personal entrainment, is presented.

- **Chapter 7: Conclusions**

Chapter 2

State of art


RESEARCH PLAN
STATE OF THE ART
ANALYSIS OF EXPRESSIVE QUALITIES OF MOVEMENT
ANALYSIS OF SYNCHRONIZATION OF EXPRESSIVE QUALITIES OF MOVEMENT
INTRA-PERSONAL SYNCHRONIZATION
INTER-PERSONAL SYNCHRONIZATION

Contents

2.1 Analysis of expressive qualities of movement

2.1.1 Gestures and movement analysis

As introduced in the previous chapter, communication between humans is rich of non-verbal aspects and cannot be reduced to mere speech support. Non verbal aspects heavily influence not only the meaning of the words, but also the perception, attitude and feelings of those who listen. Starting from this point of view, gestures are referred to unconscious actions conveying behavioral information (which can be decoded ([VPBP08]). Gestural input has just become popular recently with the introduction of new sensors (e.g., Kinect, HTC Vive, Myo, Leap Motion, etc.). As gesture recognition systems mature, movement and non verbal input become an integral part of human computer interfaces.

Many definitions and classifications of gestures can be found in literature. For example, gestures have been defined as movements produced mainly to support verbal communication and that is strongly synchronized with the flow of speech ([Ken04], [McN92],[McN00], [Cas98]. Among the many definitions, a common element is that *gestures are closely related to human movements with the purpose of communicating something*. A more formal distinction between gesture and movement was given by Kunterbach ([KH]). Kunterbach defined a gesture as a body movement that contains information and cannot be considered purely functional (i.e., the movement of a hand grabbing an object). A gesture can be of different kinds. The following chapters will be focused on a specific category of gestures: *expressive gestures*.

2.1.2 Expressive gestures

Ekman [EF69] affirms that gestures have expressive potential and can reflect complex emotions such as surprise anger, fear, interest. The definition of gestures given by Kurtenbach and Hulteen [KH] can be extended from *gestures* to *expressive gestures* as: an expressive gesture is a movement no longer characterized by informative content, but from *expressive* informative content. According to this extension, purely functional movements, which were not treated as a gestures, would be now considered whether they possess expressive content. Consider, as example, a very simple gesture: shaking your hand with another person. All of us have experienced different ways of shaking our hands (somebody shake his/her hand more angrily and with more energetic movements somebody else more calmly) and we are aware that each of these can be associated with very different meanings.

In the past, expressive gestures have been often studied in social interactions as a non-verbal medium to exchange affective and attitudinal information. A particular mention goes to the usage of expressive gestures during artistic performances. In music, gestures are emphasized, in dance or theatre, are used to transmit the information content to the audience according to specific codifications and expressive vocabularies. The discussed research deals with the investigation of hidden mechanics of expressive communication through a methodological and objective analysis of the performed movements.

2.1.3 Laban Movement Analysis

Human movement is a powerful tool available for communicating emotions, moods, and experiences. Rudolf Laban was a choreograph and dance theoretician who developed a representation of gestures expressivity and a movement analysis method called Laban Movement Analysis (LMA) [LL47a]. LMA has been originally created to support dance teaching and, only recently, it has been extended to study the expressive content of generic movements, using motion descriptors to classify gestures and communicative aspects of full-body movements.

LMA defines four major categories: Body, Space, Effort and Shape (see Figure 2.1):

- Body (WHAT): describes the structural and physical characteristics of the human body

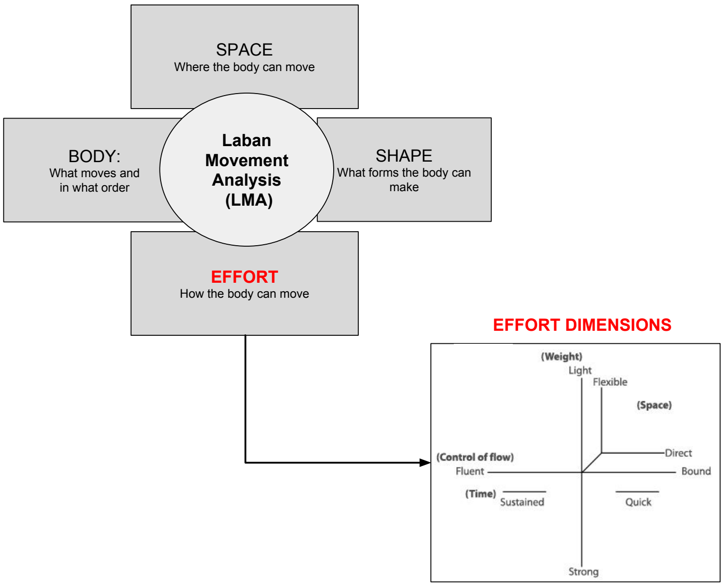


Figure 2.1: Laban Movement Analysis components. The Effort dimension is highlighted since it is of interest for this research.

while it moves. The Body category is responsible for describing which body parts are moving, are connected or influence and are influenced by other parts.

- Space (WHERE): how does movement relate to our *kinesphere* (i.e., the volume of space that our body is moving within). The space around our body that is referred to as our *personal sphere*.
- Shape (WHY): describes the morphology of the body during motion, i.e., the way and the reason why the body changes “shape” while moving.
- Effort (HOW): describes the more subtle characteristics of movement with respect to inner intention. Effort components describe what dance professionals call **movement qualities** (illustrated in detail in Section 3) and are related to the “effort” needed to perform a movement expressing such qualities.

  Effort dimensions: Effort is further decomposed into the following four elements (eight paired elements where each pair is represented as a continuum, as shown in Figure 2.1)

  - Space: ranges from direct (or straight) to indirect (or flexible) movements,
  - Time: ranges from sudden to continuous and sustained movements,
  - Flow: ranges from free to constrained movements,
  - Weight: ranges from heavy to light movements.

For this thesis, only Laban's **Effort** dimension is relevant[1]. The Effort dimension illustrates the qualitative features of movement i.e., how a movement is performed independently on its precise trajectory in space. The same sequence of movements can be performed smoothly or rigidly, lightly or heavily, etc. enabling us to study differences and invariants. In the next section, a conceptual framework for the analysis of expressivity in full-body movements, is introduced. The framework is inspired by and extends the Laban theories.

2.1.4 A conceptual framework for the analysis of expressivity in human movement

Several computational models to automatically detect and compute expressive movement qualities of a movement have been proposed. The content and the experiments presented in this thesis, are grounded on a conceptual framework for the analysis of the expressive qualities of movement[2] shown in Figure 2.2. The framework has been developed by Camurri and Volpe [CPLV01], in collaboration with other experts in movement analysis and computer science, in the early 2000's and refined in subsequent years (the last refinement has been recently published in [CVP$^+$16b][3]).

The framework grounds on the following basic assumptions:

Body Scales: different subsets of expressive movement features can be measured at different scales, ranging from a single part of the body (e.g., a hand), to the whole body, up to a group of users.

Temporal Scales: different time scales are applied to different feature-extracting techniques. The higher the feature is located in the layer stack, the larger is the time needed to extract that feature.

Each layer is related to different expressive movement features, and each user is represented by a single layer stack. The conceptual framework is presented in([CVP$^+$16b]), an excerpt from the description of the layers together with the definition of the features is given. It is needed to underline that each layer contains a vast set of features. The four layers are defined as:

- Layer 1: is the physical layer (e.g., pure positional data or simple kinematics (velocity, acceleration)). Layer 1 grounds on the concept of *virtual sensor*. A *virtual sensor* is an integration or fusion of data (coming from a broad range of physical sensors, including

[1] studies illustrated in Chapter 3 are based on Laban's theories.

[2] Note that movement *expressive features* and *expressive qualities* are used as synonyms throughout the thesis.

[3] The last version of the framework was the result of a close collaboration between the members of Casa Paganini - InfoMus Lab and a professional choreographer and dance teacher (Virgilio Sieni) occurred in the last three years. This activity led to the definition of additional movement qualities; some of them have been used in this thesis (See Section 3.2.3).

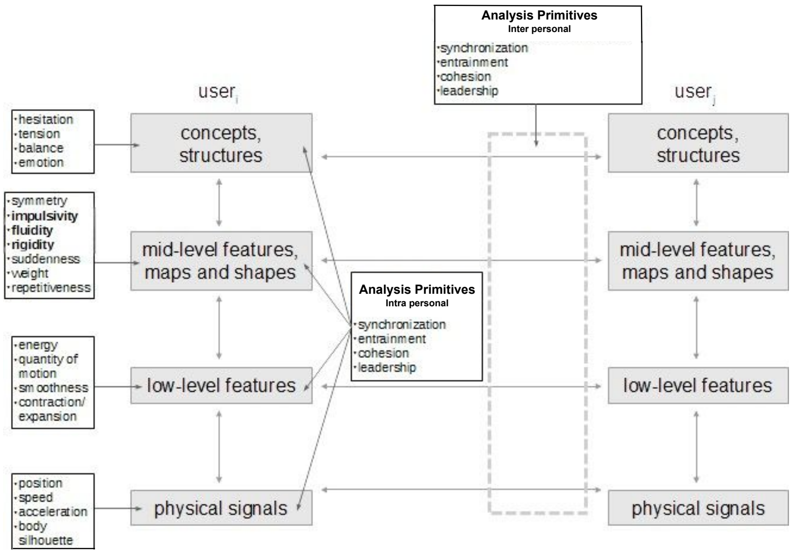


Figure 2.2: A graphic representation of the conceptual framework for the analysis of expressivity in human movement developed by Camurri et al. [CPLV01]

motion capture) and combined with signal conditioning (e.g., de-noising and filtering). For example, an RGB-D physical sensor (e.g., Kinect) may be associated with virtual sensors that provides the 3D trajectories of specific body parts, the silhouette of the tracked bodies, and the captured depth image. Each virtual sensor is characterized by:

- sampling rate
- typology of data it provides (e.g., 3D coordinates, a single numeric sample, a time series or buffer and so on).

At Layer 1, data is processed to get representations suitable for the next analysis layers.

- Layer 2: low-level features are motion features observable at a small time scale, frame-by-frame. Layer 2 receives the raw data coming from Layer 1 and extracts a collection low level qualities characterizing movement. Low-level features are usually computed instantaneously on the raw data or on small buffers of a few samples and are represented as time-series (usually sharing the same sampling rate as the raw data they are computed from). Time-series may be either univariate (e.g., smoothness) or multivariate (e.g., the three-dimensional components of a joint' acceleration).
- Layer 3: mid-level features (maps and shapes) are complex and expressive characteristics of the motion usually extracted on more than one joint and that are computed on movements' flow segments (gestures). Mid-level features require significantly longer temporal

intervals to be observed (between 0.5s and 5s), long enough to grab movement time evolution.

- Layer 4: high-level features (concepts and structures) represents even more abstract concepts such as emotional states, social attitudes, engagement, in full-body interactions. High level features mainly focuses on the non-verbal communication to an external observer. and require significantly long temporal observation, memory and context-awareness. The perception of an external observer of these features, is influenced by the past and by the context. For this reason, computational models, belonging to this layer, memorize and integrate data coming from different sensory channels.

A particular category of features is represented by the analysis primitives. **Analysis primitives** are unary, binary, or n-ary operators that describe (with a the result of their computation) the temporal development of the various features.

- The simplest unary analysis primitives are statistical moments (e.g., average, standard deviation). For example, unary primitives can be used to retrieve salient events (e.g., slope, valleys and peaks in the time-series), or to estimate the complexity or entropy of a movement or sequence of movements. They include various time-frequency transforms and models for predictions (e.g., HMM) that, for example, can be used to estimate the extent at which actual movement corresponds to or violates expectations.
- Binary and n-ary operators can be applied e.g., for measuring the relationships between time-series of low-level features computed on the movement of different body parts (limbs) of between more complex expressive qualities. Some of them are: Synchronization, Causality, etc. For example, the Synchronization primitive can be applied at low-level features, e.g., the ”Quantity of Movement” or the ”Smoothness”, computed from the movement of multiple user’ heads, as well as to mid-level qualities such as the Fluidity of their trunk movements.

Camurri’s framework has previously adopted for different research purposes: in order to describe the expression of human gestures [CTV02], for investigating the emotional mechanisms underlying expressiveness in music performances [VVC10] [CMC$^+$08] and as potential descriptors to infer the affective state of children with Autism Spectrum Condition [PSCO13].

2.2 Synchronization

Synchronization of interacting elements is a very widespread phenomenon in nature and, for this reason, is the subject of numerous researches in the physical, biological and social fields. Synchronization is *the order in the time of things*, and starts to exist when two or more events are repeated simultaneously.

Synchronization among humans (inter-personal) has been defined as: "*temporal coordination between several individuals during social interactions*" [DCM$^+$12]. One of the main contexts in which inter-personal synchronization has been studied is conversational communication. In fact, in the psychological literature, inter-personal synchronization is defined as the alignment of internal models, which, during an interaction, are governed by unconscious mechanisms [PG04]. It has been shown how, during a conversation, humans tend to align several multi-modal signals [CD01], in particular speech, body movements and postures [CO66, RKM11, SRD09]. In [vULD$^+$08], authors shown how users walking side by side tend to synchronize. Lorenz and colleagues [LMV$^+$11], measured the degree of synchronization between the movements performed by a human and a robotic partner, during a joint task. Moreover, human movement synchronization has been investigated in several rhythmical activities ([MNM09, VOD10]). The study of human rhythmic movements with respect to discrete, periodic stimuli is called *sensorimotor synchronization* (SMS) [RS13]. This line of research investigates the relationships between users' movements and an external rhythmical source (or an internalized representation of the source), often resulting in periodic dynamics (e.g., sequences of movements). Another successful approach to investigate movement synchronization involves the usage of particular physical models: *oscillators*. Models as [Kur75], [SS$^+$93] simplify movement complex dynamics back to a single variable (the *phase*) and provide mathematical conditions under which the synchronization is established (by studying phase relationships).

This research is focused on the study of synchronization between humans movements. A novelty of the presented approach is represented by the usage of the analysis of synchronization in order to investigate expressive characteristics of movements rather than to measure the temporal coordination of functional movements.

2.2.1 Analysis of synchronization of expressive qualities of movement

In the context of the conceptual framework, *Synchronization* has been introduced as one of the analysis primitives, i.e., operators that can be applied to different movement expressive qualities at different layers. Synchronization can be computed both on a single user (intra-personal), for example to measure the degree of coordination between the joints' velocities of a dancer, and on multiple users (inter-personal), for example to detect the level of coordination between multiple users and estimate the degree of collaboration and coalition between them.

Many studies made use of the analysis of synchronization to investigate movement expressive features. In [Kel95, NHP11] authors shown how synchronization is an important cue of several emotion displays. In [VVM11] Varni et. al investigate how motor synchronization can be used to analyze social group dynamics and detect dominant members (i.e., *leaders*), while in [LS11, HGP10] authors studied synchronization to measure the degree of cohesion of the whole group. In [Miy09], authors introduced a rehabilitation system based on limbs synchronization that demonstrated to be effective in stabilizing the walking of patients affected by Parkinson's disease and hemiplegia. In [LDL$^+$09], Leman et al. show how music might be an excellent domain to explore non-verbal communication and how synchronization can be used to measure collaboration and coalition between users. In [LMV$^+$13], authors focus on the effects of beat-synchronized walking on movement timing and vigor. Finally, in a recent work, Lussu and colleagues [LNVC16] compute synchronization between respiration and body movement energy to distinguish movements performed with different expressive qualities. In particular, they found that synchronization was higher in fragmented movements than in fluid movements.

2.2.1.1 Analysis of synchronization to study musical entrainment

Again related to mechanisms and temporal dynamics of interacting systems, entrainment is a more complex phenomenon in respect with synchronization. Entrainment, sometimes erroneously referred as synchronization, describes the mechanisms and temporal dynamics of interacting rhythmic systems [KNH14, Cla12].

Entrainment was scientifically observed for the first time the physicist Christiaan Huygens, who in 1665, observed that if two pendulums are coupled by a stand (of sufficiently elastic material, e.g., a wooden board), their oscillations influence each other. In particular, the system (composed by the two oscillators) tends to an stable (anti) phase synchronization. Furthermore Huygens also observes that even disturbing the fluctuations, after a transitional phase, the pendulums would be synchronized again.

In human interactions, entrainment affords forms of coordination which are particularly precise, complex, periodic, and it is not related to repetition or imitation, but it refers to *getting in tune* with another even by making very different movements. Entrainment describes the tendency of recurring behaviors, between two or more users, that seems to increase over time as a result of sharing or acting together. In physiologic literature, being "entrained" between two individuals has been linked to positive social affect and empathy ([WMS$^+$87]). Philips-Silver in [PSAB10] stated that entrainment can be defined as the ability of humans to *perceive and produce rhythmic action and real-time integration between sensory and motor systems*, then, the same author in collaboration with Peter Keller, presents in a subsequent publication [PSK12], a complementary definition of entrainment, i.e., *the spatio-temporal coordination between two or more individuals, often in response to a rhythmic signal*.

As briefly hinted, musical performances and, more in general, all the activities involving music (e.g., concerts, dances, sporting events and so on) are social moments that have been often considered an excellent field of analysis to study social interactions (such as entrainment) in an ecological way. Entrainment and rhythm are strictly linked; the interaction, coordination and synchronization of human beings can be described by rhythmic and oscillatory dynamics which evolve into a common state represented by period and phase lock or steady alignment [CSW05].

The work presented in the last part of this thesis has been partially realized in the context of the IEMP project (presented in Chapter 6), which is entirely focused on the study of musical entrainment. For musical entrainment it is intended the study of temporal and spatial coordination of movements (and its expressive qualities) between two or more players, during music performances. Music performances are usually much more complex to be analyzed than, for example, people tapping out in time with each other. Then, new methods for entrainment investigation are needed. A better understanding of this phenomenon will have important implications for the understanding of joint action and coordination from a psychological perspective, and is likely also to be applicable in entertainment, educational, and therapeutic contexts.

2.2.2 Key concepts

The following definitions should support a better understanding of the main properties of the phenomena of interest, that are featured by precise temporal characteristics (e.g., periodicity, recurrence etc.) and, among which, temporal relationships are established (e.g., phase locking). Subsequently, a set of measurement techniques and methods to perform synchronization analysis, on different typologies of signals, is presented.

Period P: for a (periodic) phenomenon or quantity with respect to time, the period is the minimum interval of time, starting from any time point, after which the characteristics of the phenomenon return to be the same. In such phenomena, P is the inverse of the frequency f. Given a periodic motion, e.g., uniform circular motion, and taking as reference a certain instant t, a certain position A and a certain speed and acceleration; P is equal to the minimum time interval, counted starting from *t*, after which the motion returns to A with the same speed and acceleration. In case of oscillatory motion, P is more simply identified by the time of a complete oscillation. Similarly to these examples, it is possible to compute the period P of a rotation, revolution, or wave propagation.

Frequency f: is the number of times a periodic phenomenon is repeated in the unit of time; f is the inverse of the period P and is measured in hertz (Hz - cycles per second).

Phase θ: in sinusoidal phenomena (in which a given quantity varies periodically according to a simple harmonic law e.g., sounds, monochromatic lights, electric oscillations, harmonic motions, etc.), the phase is the angle constituting the argument of the sine (or cosine). The phase is a function of time or space (depending on the case). We can define:

- *initial phase*: the value of the phase angle at the initial instant or in the initial spatial position;
- *phase difference*: the displacement between two sinusoidal quantities with the same period (or, equally, with the same frequency) computed as the difference between their phases;

Rhythm: it is a term to indicate a periodic phenomenon directly perceived by an human. A rhythm can be perceived by different senses (for example, we can listen to successions of sounds or look at succession of periodic movements). Rhythm is embodied i.e., modeled by an internal representation and conceived in our memory.

2.2.3 Methods

This section illustrates a set of methods, selected from the available literature, to measure interpersonal and intra-personal synchronization. In Section 2.2.1, the Synchronization primitive has been introduced within the context of the adopted conceptual framework. However, it is important to underline that, the choice of adopting such framework, has no impact on the conceptual meaning of synchronization but only affects its application modalities. This, to emphasize that, the presented methods can be considered as specific instances of the Synchronization primitive analysis and, subsequently, can be applied to all the features belonging to the framework's layers.

Referring to the literature[4], five main families of methods, to perform synchronization analysis of human movement, has been identified. For each family, only a subset of methods have been chosen, according to the following criteria: difficulty of application, linearity (or non-linearity) and application domain (time, frequency etc.).

The methods families are:

- Correlation analysis
- Spectral Analysis
- Phase Analysis
- Recurrence Quantification Analysis (RQA)
- Event Synchronization (ES) Analysis

[4] a summary table can be found at bottom of this section.

2.2.4 Correlation analysis

Correlation analysis techniques are used very commonly. One of the main reasons is certainly that these methods can be applied to a wide number of contexts as they are unrestricted, simple to implement and easy to apply.

2.2.4.1 Cross correlation

Cross correlation is a tool to establish the degree of similarity between two signals (continue) or time series (discrete). For example, it has been used to measure the degree of inter-personal coordination and the emergence of leaders in string quartets [VVM11]. Consider, as example scenario, a piano duo. Suppose we are interested in measuring how much the two pianists move in a coordinated or *correlated* way. First of all, we extract a set of movement features in a shape of time series of N values each.

To perform the correlation analysis and study the relationship between the first pianist (player x) and the second one (player y) of a particular feature f, we need to select the feature's time series, for both players, from the time series set: $x_f(n)$ and $y_f(n)$.

The cross correlation between two time series $x_f(n)$ and $y_f(n)$ is a function of l (or correlation *lag*) and it is expressed by:

$$C_{x_f,y_f}(l) = \sum_{n=1}^{n=N-l} x_f(n) y_f(n+l) \quad with \quad l = 0; +1; +2... \tag{2.1}$$

To quantify the strength of the correlation relationship, we can use the Pearson coefficient:

$$\rho(x_f, y_f) = \frac{\sum\limits_{n=1}^{n=N} (x_f(n) - \vec{x_f})(y_f(n) - \vec{y_f})}{\sqrt{\sum\limits_{n=1}^{n=N} (x_f(n) - \vec{x_f})^2 \sum\limits_{n=1}^{n=N} (y_f(n) - \vec{y_f})^2}} \tag{2.2}$$

The correlation coefficient ρ ranges from -1 to $+1$. A value of $+1$ implies that a linear equation perfectly describes the relationship between X and Y, with all data points lying on a line for which Y increases as X increases. A value of -1 implies that all data points lie on a line for which Y decreases as X increases. A value of 0 implies that there is no correlation between the time series.

Going back to our example, an high Pearson coefficient value, computed between the energy of the hand movements of the players, can be read as the indication that one player has a big influence on the other. A limitation of this method is the following: high correlation does not prove that x causes y or vice versa.

2.2.4.2 Auto correlation

The idea of auto correlation is to provide a measure of similarity between a signal and a delayed copy of itself, as a function of delay. The auto correlation function will show a global maximum at zero lag. As the lag increases or decreases, the value of the auto correlation will necessarily decrease.

$$C_{x,x}(l) = \sum_{n=1}^{n=N-l} x(n)y(n+l) \quad with \quad l = 0; +1; +2... \tag{2.3}$$

What it is interesting is that, when analyzing periodic or quasi periodic signals, $C_{x,x}(l)$ will be increasing again, reaching one or more local maximums.

2.2.5 Phase Analysis

2.2.5.1 Phase correction model

Consider another example: a performing orchestra. In group performances, musical times followed by the group members, may be different. Then, each member make an effort to synchronize with the others. According to several psycho-motor theories ([HM85],[VDSK13], synchronization can be achieved by tuning an internal timekeeper to reach the desired (shared) tempo. This situation can be modeled with *linear phase correction* models. A linear phase correction model describes the process of minimizing the asynchrony by predicting and adjusting the timing of each future movement. At each iteration, to reduce the error, a correction gain is applied.

Consider a common experiment where a subject is asked to tap her/his finger in-sync with a metronome. The process of synchronization between the subject and the metronome may be represented by the following equation:

$$t_n = t_{n-1} + T_n - \alpha A_{n-1} + \epsilon_n \tag{2.4}$$

In Equation 2.4, t_n and t_{n-1} identify the current and the previous observed tapping times (i.e., the instants when the finger comes into contact with the surface (e.g., a table)), T_n is the time interval generated by an assumed internal timekeeper, α is the correction gain (or correction strength), A_{n-1} is the measure of time asynchrony of the previous event (occurred at t_{n-1}) and ϵ_n is the random error term used to model the timekeeper error and the activation of motor components. In the model evolution, phase stabilization (i.e., $A_{n-1}{\sim}0$) is taken as strong evidence for motor synchronization.

2.2.5.2 Oscillators-based models

This family of (non linear) models, inspired by physics, describes interacting entities as oscillators. An *oscillator* is a system that follows and executes a periodic behavior. Each oscillator is modeled with its own period and natural oscillation frequency. Swinging pendulum are the typical example of oscillators. A pendulum returns to the same point in space x at regular intervals and at velocity v, which regularly decreases. Hyugens gave his definition of entrainment after having experienced that, interacting oscillators, whether they are physically coupled, tend to share the same oscillation period and phase.

A famous model of this family, has been developed by Kuramoto and colleagues [Kur75]. Kuramoto model has been used to study the behavior of interacting elements in large groups. The model allows to mathematically measure the synchronicity of the whole group and the contribute of each member. Given N coupled oscillators, each oscillator i is indicated by its phase θ_i and its natural frequency ω_i.

Two oscillators can be considered synchronised if they are locked to the same frequency ω_i, then, the system is totally synchronized if all N oscillators share ω_i. The dynamics of each oscillator is described by:

$$\theta_i(t+1) = \omega_i(t) + \frac{K\Delta T}{N}\sum_{j=1}^{n=N} sin(\theta_i(t) - \theta_j(t)) \quad with \quad i = 1; ...; N \tag{2.5}$$

ΔT is discrete fixed time step and K is called coupling constant. Kuramoto showed that, by increasing K the system experiences a transition towards complete synchronization, i.e., to a dynamical state in which $\theta_i = \theta_j$ for each i, j and for each t .

2.2.5.3 Circular scales and circular statistics

Normal statistics is not always the best way to go. Circular statistics are a set of methods and tools for data analysis. With circular statistics, data moves from a linear domain to a circular one. Data are represented differently, i.e., are shaped as vectors distributed around the circumference and are identified by angles. Datasets of circular data can be either created or derived from linear data.

Representing data on a circle is suitable for the analysis of phenomena of a periodic nature (common in biology, social sciences etc.). The circumference represents the range of data distribution and, at the same time, the period. Taking into account the natural periodicity of the circle, there is no zero point, no maximum or minimum, and any designation of *high* or *low*, and of *more* or *less*, is purely arbitrary. To convert our data in radians we use following equation:

$$\alpha = \frac{2\pi x}{k} \tag{2.6}$$

where x is the original data value, k is the number of steps that x was measured in (or the estimated period of x) and x/k gives the portion of the circle that x represents. Circular data can be represented on a two-coordinates system. On it, each sample is a vector centered in the origin and identified by a length and a direction.

From our dataset of vectors, several statistical quantities can be computed. For example, the average direction $\bar{R}$, computed as the sum of all our data vectors, can be a very useful information to understand if the studied phenomenon is periodic. A second interesting quantity is the circular variance $S = 1 - R$, where R is the length of the average direction vector. If all the data vectors share the same angle, R will be close to 1 and the variance will be almost 0.

Circular statistics is particularly relevant while investigating tapping synchronization. The timings of two users tapping a finger, are very likely collected as time series. Converting them from the linear domain to circular one and then apply circular statistics, will be really beneficial as shown, for example, in [DBS15]. Circular statistics can solve really complicated problems where relationships between data are hidden and challenging, as for example the analysis of entrainment.

2.2.6 Spectral Analysis

Full-body movement data contains a lot of powerful, but very difficult to interpret, information.

These data are affected by non-linearity and are *non-stationary* i.e., their probabilistic structure vary over time. Dealing with these typologies of data, in the time domain, can be really complex. Luckily, a family of mathematical operators, called *transforms*, can reduce their complexity. Mathematical transforms are defined as tools to trasnfer signals from the original domain to a different one where difficult problems become simpler to address.

One of the most famous examples is the Fourier Transform that transforms the signal from the time domain to the frequency one. Fourier Transform is very useful to study, for example, rhythm, periodical and recurrent behaviors. A well-known problem with the Fourier Transform is that it provides frequency information of the signal (i.e., how much of each frequency exists in the signal spectrum) but, at the same time, loses completely any information about what spectral component occurs at what time interval. In order to gain information in both time and frequency domain we can use other transforms as, for example, the Wavelet transform.

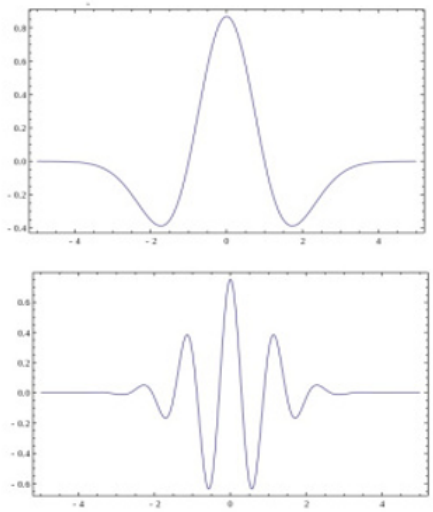


Figure 2.3: Examples of wavelets

2.2.6.1 Wavelet transform

Wavelet are relevant tools for the study of interaction phenomenon between different components of a living system or between different living systems [IBGM15]. Analogously to Fourier, the Wavelet transform exist in both continuous (CWT) and discrete (DWT) forms[5]. Wavelet Transforms scan the time and frequency domains simultaneously in order to detect particular aspects of the signal such as drift, trends, abrupt changes.

The Wavelet transform uses a linear combination of shifted and scaled non-infinite base functions called *wavelets* (i.e., the input signal is *decomposed* into wavelets). The resulting new signal representation is the *wavelet spectrum* of the signal.

Summarizing, a Wavelet transform:

- provides the time-frequency representation of a time series named *spectrum* (alternatively to the Fourier transform that does not preserve temporal information on the frequencies).
- detects singularity, i.e., where a time series signal behaves irregularly.

Every signal has its own spectrum. The *Wavelet analysis* consists detecting the major components of the signal within the resulting spectrum (similarly to a analysis of a Fourier transforms). Given a wavelet spectrum, the original signal can be reconstructed from it.

[5] CWT and DWT are quite complex transforms, for more details on the mathematical fundamentals refer to [LY94].

Initially applied in signal processing for financial forecasting, the Wavelet transform has been increasingly employed in other fields, among these, to the study of human movement. For example, authors in [WW04], applied wavelets to study human movements trajectories. In particular, they proposed a system to classify 3D trajectories by decomposing them using wavelets and selecting the ones that are most representative for each considered trajectory.

More interesting for our studies is the Cross Wavelet Transform (XWT).

XWT is used to perform the analysis of the interaction between the wavelet transforms of two individual time-series. The time-frequency representation of the XWT provides information about the intensity of the interaction between two input time-series, for each frequency, as a function of time. Roughly speaking, XWT detects the spectrum regions characterized by high power that are shared by the two input time series. Moreover, it reveals useful information about phase relationship i.e., it measures how much the time series are physically related (if a relationship between them exists, their phase difference will be varying slowly).

2.2.7 Recurrence Quantification Analysis (RQA)

Recurrences can be defined as trajectories outlined by a system (in the space of the system states) that have been already visited by the system itself.

In synchronization analysis, systems coincide with human beings. By so doing, the systems states can be represented as “snapshots” of movement features values. For example, a dancer moving an arm, displays a recurrent behaviour, if and only if, her arm revisits certain regions of space by following the same trajectories, presents the same velocity (or the same acceleration profiles) or displays the same movement qualities, *in separate moments of the dance performance*.

Recurrence Quantification Analysis (RQA) is a method to evaluate the presence of such recurrent patterns in time series data. RQA allows to identify the degree to which a considered time series repeats itself and provide a measure of reliability i.e., quantifies how much the recurrent patterns reflect the presence of predictable system’s dynamics.

2.2.7.1 Cross Recurrence Quantification Analysis (CRQA)

Cross-recurrence quantification analysis is the multivariate extension of RQA used to determine the presence and duration of shared patterns and behaviours between the dynamics of two different time series by quantifying their regularity, predictability, and stability.

CRQA, for example, can be applied to measure if two persons come to exhibit similar patterns of behavior. In addition, CRQA quantifies the *lag* for one individual to maximally match the other, or whether there is a leader–follower type of relationship.

2.2.7.2 Recurrence Plots

Recurrence plots (RP) are a two-dimensional analytical tool to visualize recurrences of data. RP are binary matrices, which depicts the pairs of times at which the trajectory of the system recurs [6]. Given a time window with N instants, each point of the system trajectory is represented by a state vector $v(n)$, with $n \in [0; N]$.

An element (or pixel) RP_{ij} of the matrix is *recurrent* if $D_{ij} = ||v(n_i) - v(n_j)||$ (i.e., the distance between two state vectors at different time instants) is smaller than a threshold τ. If $D_{ij} < \tau$ a recurrence point is identified and a 1 (or a black pixel) is inserted in the NxN matrix, otherwise will be inserted a 0 (or a white pixel).

Useful information regarding the dynamics of the system are represented by regular structures within the RP, as shown in Figure 2.4. RQA defines a set of indexes based on the distribution of white and black pixels within the RP to study determinism, entropy, trends of the evolution of the system.

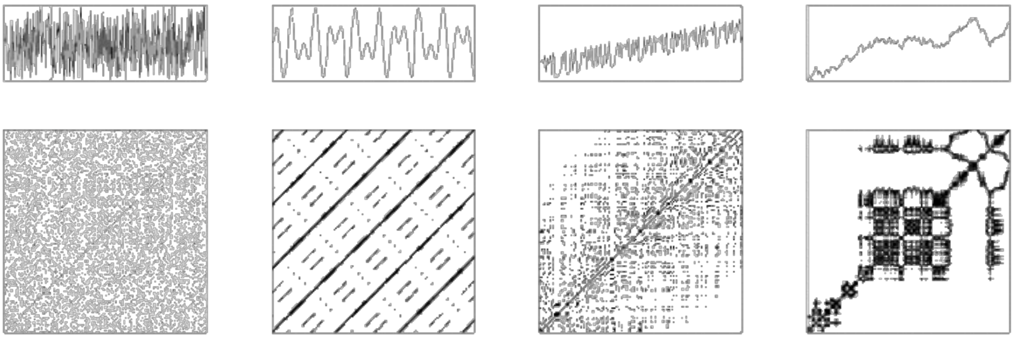


Figure 2.4: Recurrence plots examples.

2.2.8 Event Synchronization (ES) Analysis

Event Synchronization (ES) measures synchronization and time-delay patterns between two time series [QKG02]. The method relies on time difference between events occurrence-timings. To be applied, ES requires the following pre-processing steps:

1. to conceptually define what an event is within the application context;

2. to model the conditions required for the creation of a new event occurrence;

[6] For detailed information about RP refer to the following website http://www.recurrence-plot.tk/glance.php.

3. to generate two binary time series containing information about events occurrence-timings. Such time series are used as input for the algorithm and contain a “1” element to indicate the presence of an event or a “0” to indicate the contrary. Each time-series has the same number of n items.

Given two time binary series of n samples $x = x_1, ..., x_n$ and $y = y_1, ..., y_n$, with $x_i, y_i \in R$, events are detected and the time instants t_i^x and t_i^y at which events occur in time series x and y respectively are computed. A measure of the synchronization Q_τ between the two time series is computed as:

$$Q_\tau = \frac{c^\tau(y|x) + c^\tau(x|y)}{\sqrt{m_x m_y}} \tag{2.7}$$

where $c^\tau(y|x)$ and $c^\tau(x|y)$ are, respectively, the number of times an event in time series y appears within a time interval defined by parameter τ after an event appears in time series x, and vice-versa. m_x is the number of events detected in time series x and m_y is the number of events detected in time series y. Then, $c(x|y)$ is computed as:

$$c^\tau(x|y) = \sum_{j=1}^{m_x} \sum_{i=1}^{m_y} J_{ij}^\tau \tag{2.8}$$

where J_{ij}^τ is defined as follows:

$$J_{ij}^\tau = \begin{cases} 1 & \text{if } 0 < t_i{}^x - t_j{}^y \leq \tau \\ 1/2 & \text{if } t_i{}^x = t_j{}^y \\ 0 & \text{otherwise} \end{cases} \tag{2.9}$$

ES can be used to quantify intra-personal and inter-personal synchronization in various human multi-modal behaviors. In particular, in Chapter 4, we have extended the original algorithm with new complementary characteristics, in order to model and solve an higher number of problems.

Reference	Experiment Details	# Users	Movement Data	Method
[YWS12]	Measure the synchronization degree between fingertip movements and neural activity	20	Finger Positional data, EEG	Correlation analysis
[ODGJ$^+$08]	Measure the synchronization of finger movements	12	Positional data	Relative Phase and frequency analysis
[MSBC13]	Quantify the strength of interpersonal inter-limb coordination	12	Angle of lower lib and knee positions	CRQA
[RDR$^+$11]	Experiment 1: Measure coordination at the intrapersonal (i.e., hand–torso coordination for individual participants) and interpersonal (hand–hand or torso–torso coordination between individuals) performing a joint task	24	Hand and torso motion data	CRQA
//	Experiment 2: Effect on coordination increasing the task difficulty by varying individual task	//	//	//
[FD16]	Measure interpersonal synchrony during an unstructured conversation	62	Full-body movement data, extracted using a video-image analysis software	Cross Wavelet analysis
[MLV$^+$12]	Measure synchronization during a goal-directed experimental task	20	Hand movements	Phase Analysis, extended Kuramoto model
[VCCV08]	Measure entrainment between four violin players	20	Head motion extracted from video recordings	RQA and Phase analysis
[VVC10]	Quantify the level of synchronization of the affective behavior within a small group and the emergence of functional roles, such as leadership	20	Head MoCap position	RQA and Event Synchronization Analysis
[RGF$^+$12]	Assessing group (sat and rocked in six identical wooden rocking chairs) synchrony for objectively determining degree of group cohesiveness	6	Positional data of each rocking chair	Phase analysis
[DBS15]	Uncovering rhythm disorders, such as beat deafness, by measuring synchronization of finger tapping to a beat	122	Tapping events	Phase analysis, circular statistics
[OMI$^+$12]	Measure correlation of synchronization of head movements and degree of understanding on interpersonal communication	2	Vertical and front-back directions of head acceleration	Correlation analysis

2.3 Contributions

The main contributions of this research can be summarized as follows:

- With regard to the analysis of expressive gestures: a possible approach to objectively quantifying expressive characteristics of movement is provided. A methodology and possible guidelines on how to proceed are presented together with two applications on real case studies.

- With regard to the analysis of synchronization: there is a lack of studies in literature on intra-personal synchronization in favor of the much more common inter-personal synchronization ones. The thesis, through a set of experiments and results, contributes to the investigation of intra-personal synchronization, specifically applied to the study of movement expressivity.

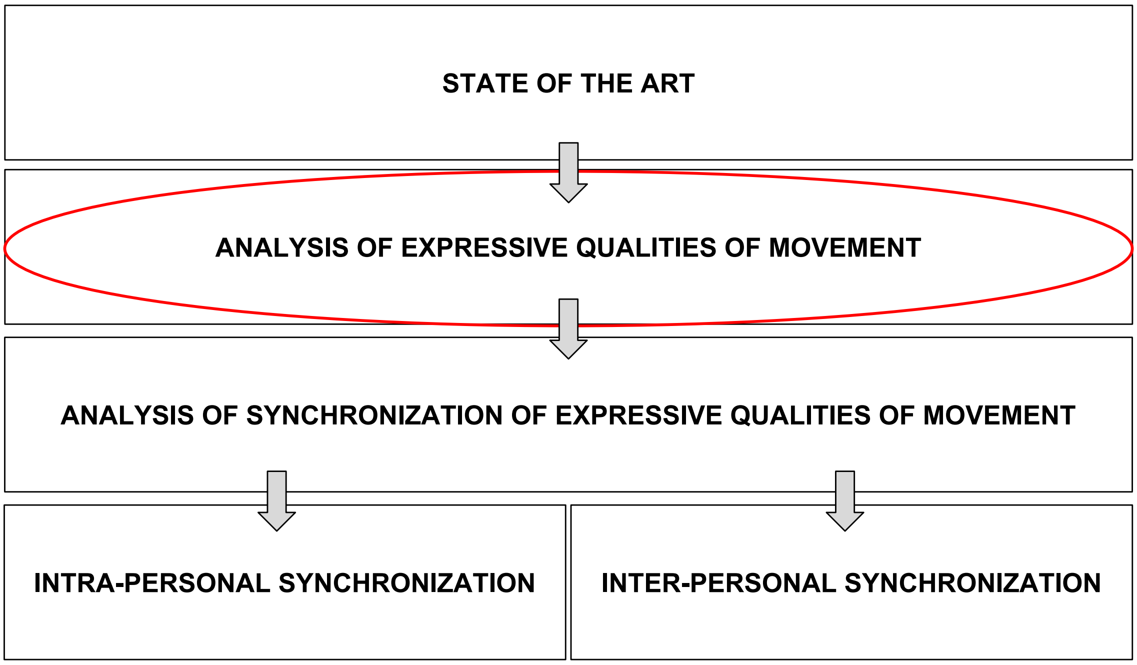
- With regard to the methods to calculate a measure of synchronization: a contribution to the state-of-the-art of the available techniques is represented by the development of the MECS algorithm (presented in Chapter 4).

- With regard to the case of inter-personal synchronization analysis: a system for the automated extraction of features for the analysis of synchronization in order to find out evidence of entrainment, from video recordings of music performances, has been developed and tested in preliminary experiments.

Chapter 3

A methodological approach to study the expressive qualities of movement

RESEARCH PLAN


STATE OF THE ART
ANALYSIS OF EXPRESSIVE QUALITIES OF MOVEMENT
ANALYSIS OF SYNCHRONIZATION OF EXPRESSIVE QUALITIES OF MOVEMENT
INTRA-PERSONAL SYNCHRONIZATION
INTER-PERSONAL SYNCHRONIZATION

Contents

The first part of this chapter presents a methodological approach to study the expressive qualities of movement.

Then, the DANCE EU ICT H2020 Project (DANCE) is introduced. In the context of DANCE two experiments, carried out by following the methodological approach, are illustrated.

The first experiment regards the process that led to the definition and development of the computational model of a specific movement expressive quality: *Fluidity*. The work has been published in [PAN+16]. In relation to it, I mainly contributed to the creation of the dataset, to the definition and development of the computational model for Fluidity and to the application of the model on recorded data.

The second work (published in [NMP+17]) presents a low-intrusive system for detection and classification of expressive qualities of movement that make use of machine learning. My contribution was to define the computation model of one of the two movement qualities considered in the study (Fragility) and to train and test the three machine learning models that have been compared on their classification precision.

3.1 Approach

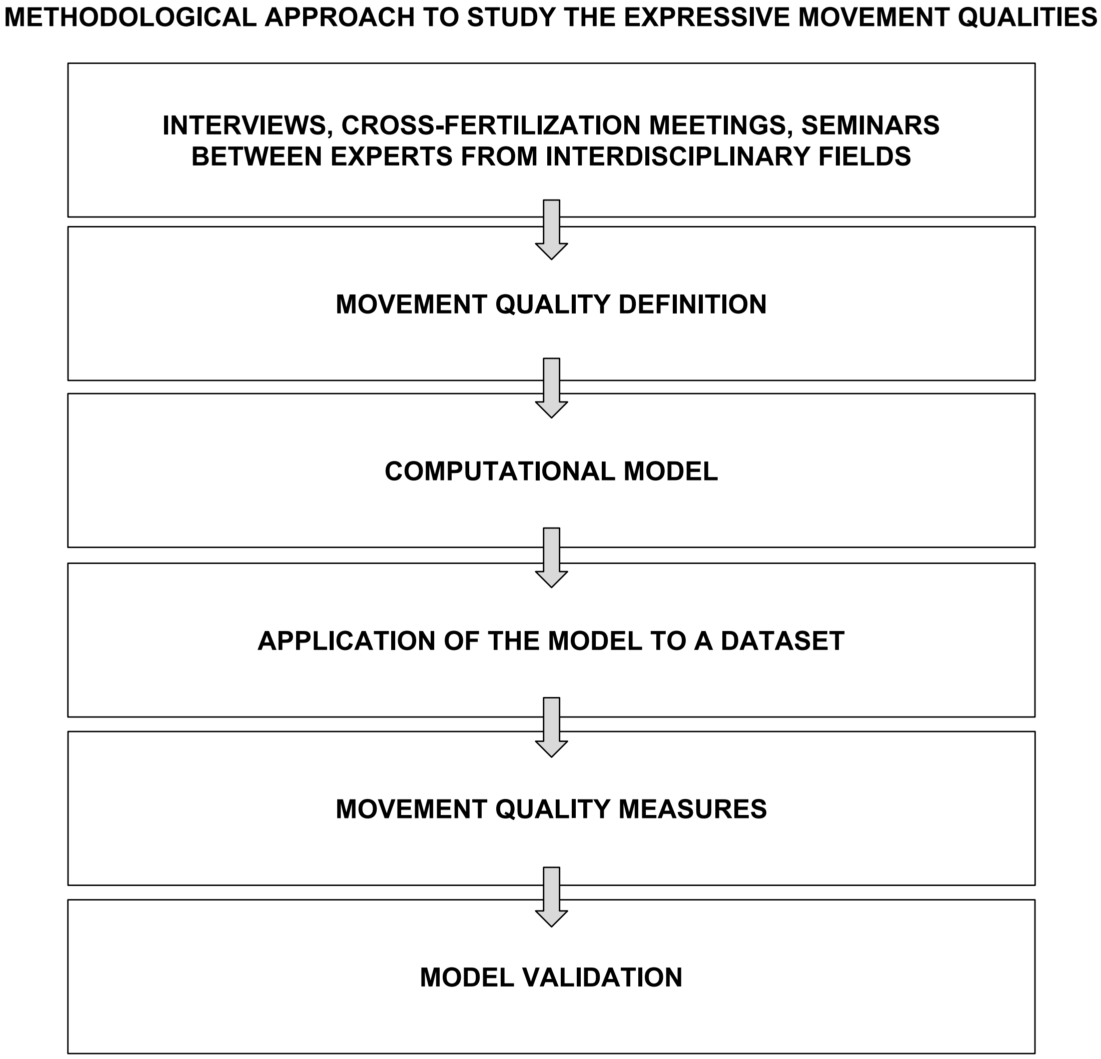

METHODOLOGICAL APPROACH TO STUDY THE EXPRESSIVE MOVEMENT QUALITIES
INTERVIEWS, CROSS-FERTILIZATION MEETINGS, SEMINARS BETWEEN EXPERTS FROM INTERDISCIPLINARY FIELDS
MOVEMENT QUALITY DEFINITION
COMPUTATIONAL MODEL
APPLICATION OF THE MODEL TO A DATASET
MOVEMENT QUALITY MEASURES
MODEL VALIDATION

Figure 3.1: Methodological approach schematic view.

The proposed approach (described by Figure 3.1) starts with an active and continuous collaboration and cross-fertilization between science and art. That is, besides being grounded on scientific evidence, e.g., from psychology and motor sciences, movement qualities models and definitions result from discussion with choreographers and dancers, i.e., the most skilled people in conveying expressivity through movement. After successive iterations, a confirmed and agreed definition of quality is proposed.

From the definition, which is halfway between the artistic and computational worlds, a computational model is designed, i.e., the properties of the quality of movement, specified in the definition, are translated into computerized terms, so that they can be quantified.

Once the computational model is finalized, the experimental phase begins, i.e., given an hypothesis, the model is applied to a dataset in order to extract movement qualities measures. Finally, the last step is represented by the model validation, i.e., to verify that values computed by the model are consistent with annotations given by large piers of users or movement experts.

In conclusion, the proposed approach sets out as a guideline for the development of further models of distinctive expressive qualities.

3.2 Analysis of expressive qualities of movement in dance performances

3.2.1 The DANCE EU ICT H2020 Project

The initial phase of the research presented in this thesis, strictly related to the study of movement and its expressive qualities, was largely carried out in the context of the DANCE EU ICT H2020 Project (DANCE).

The main goal of the DANCE project is to enable visually impaired people to perceive dance movements through sensory substitution. Such a process is achieved by implementing innovative interactive sonification techniques that allow the communication of the qualitative aspects of movement through active music experience. In other words, in DANCE we aim at studying the relationship between movement quality and interactive sound spaces as a form of sensory substitution for both blind and non-blind people.

Dancers, choreographers, and musicians can then contribute to find the best way how to convey expression and emotion by means of non-verbal full-body movement and gesture. Artists are an important source of inspiration to build computational models capable to analyze expressive qualities. All the studies presented in this chapter have been carried out in DANCE and rely on the conceptual framework for the analysis of expressive content conveyed by full-body movement [CVP+16b] (Chapter 2, Section 2.1.4).

3.2.2 A case study: a computational model for Fluidity

3.2.2.1 Fluidity of movement: Definition

Fluidity belongs to the third layer of the conceptual framework and it is associated to "good" and "soft" movements (e.g., in certain dance styles). Fluidity's formal definition required a deep analysis in bio-mechanics and psychology literature, and was finally achieved by conducting a series of interviews and motion capture recordings with experts in human movement (such as choreographers and dancers).Fluidity is one of the properties that seem to contribute significantly to perception of emotions [CMR$^+$04].

Fluidity is inspired from Laban's Flow ([LL47b]) (already introduced in Chapter 2, Section 2.1.3), in particular its two variants: Bound and Free. Fluidity extends Laban's original definition, introducing the concept of *wave propagation*. A fluid movement can be of two types: (i) not controlled, entirely unimpeded and difficult to stop suddenly (Free) or (ii) confident and fully controlled (Bound). Moreover, the fluid movement is characterized by a smooth movement of each joint involved in the movement, and presents a wave-like movement propagation which, for example, is originated by the shoulder and propagates to the arm, the forearm, the hand, the fingertips to the outer space.

The proposed definition of *Fluidity* of movement is the following: a fluid movement can be performed by a part of the body or by the whole body and is characterized by the following properties:

- **Property 1 (P1)**: the movement of each involved joint is smooth, following the standard definitions in the literature of bio-mechanics [VF95, Mor81, PSCO15].
- **Property 2 (P2)**: the energy of movement (energy of muscles) is free to propagate along the kinematic chains of (parts of) the body (e.g., from head to trunk, from shoulders to arms) according to a coordinated wave-like propagation. That is, there is an efficient propagation of movement along the kinematic chains, with a minimization of dissipation of energy.

This approach, based on the conceptual framework, can be applied to a number of other movement qualities, e.g., Lightness, Fragility, etc.

3.2.2.2 Fluidity of Movement: a Mass-Damper-Spring Model

Starting from the provided definition, we developed an implementation of Fluidity in terms of a simple physical model, based on Mass-Damper-Springs.

Mass-Damper-Spring models have been used to analyze human movement: in [WT09, DBBL98, GMM11, NT98] authors created simple mass-spring models to simulate human gait, run, and jump. Authors of [HL05] generated a set of mass-spring models to simulate different dance verbs.

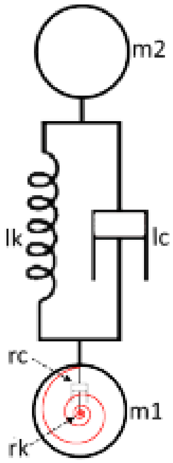
The model we propose represents the human body as a set of interconnected masses, each mass (estimated using anthropometric tables [MCC^{+}80]) represents a body's joint.

The model contains two kinds of spring:

- *longitudinal springs* (lk) or links: springs that connect two joints together. We define *body segments* two masses connected by a link;
- *rotational springs* (rk): springs that impress rotational forces on body segments;

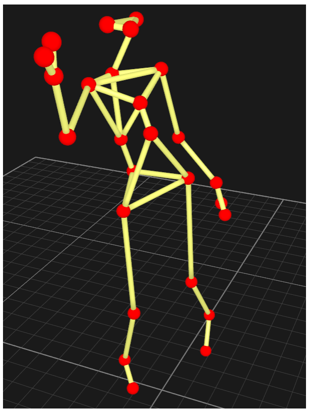
Figure 4.1 represents an example of the model with a single body segment. The proposed model can be used to analyze, filter, synthesize and/or alter movements. The response of the model to the same stimuli can vary tuning its parameters (i.e., spring stiffness, masses of the joints, damping coefficients) allowing to simulate a very large number of different conditions (i.e., a stiff / rigid body vs a fluid one).

Figure 3.2: Two masses (m1 and m2) are linked by a spring (lk), and the resulting body segment is influenced by a rotational spring (rk) that controls its rotation and movement.



3.2.2.3 Dataset

Figure 3.3: a frame of stick-figure animation.



We recorded short performances of professional dancers who were asked to exhibit full body movements with a requested expressive quality. Two professional female dancers participated to the recording sessions. Short instructions (scenarios) were given to the dancers. At the beginning of each session, the dancers were given our definition of *Fluidity*, then, they were instructed to repeat predefined movements (e.g., to pick up the object from the floor, or to throw an object) using the requested expressive qualities (e.g., fluid vs non-fluid).

We then created set of mocap recordings. For the recordings, a Qualisys motion capture system was used at 100Hz, synchronized with video (1280x720, 50fps). The resulting data consists of 3D positions of twenty six markers (see Figure 3.3).

3.2.2.4 Dataset Evaluation

We need to understand if the above defined computational model is detecting the presence of Fluidity. The objective is to validate our recordings from the observer's perception point of view and in terms of fluid vs non-fluid qualities. For this reason, we set-up an online perceptive study. Participants were asked to watch stick-figure animations of a skeleton (i.e., with *no* audio and *no* facial expressions, see Figure 3.3).

After seeing an animation they had to answer whether the following properties were present in the animation, by using 5 point Likert scale from "I totally disagree" to "I totally agree":

> *The person's energy of movement (e.g., energy of muscles) is free to flow between body's regions (e.g., from trunk, to arms, from head to trunk to feet, and so on), the same way a wave propagates in a fluid (e.g., when a stone is thrown into a pond, and circular waves radiate from the point where the stone fell down)*

It is worth to notice that the proposed text in the evaluation study did not contain the name of the expressive quality the study was focused on (i.e., Fluidity). This choice was made intentionally to avoid participants (in particular those who do not have any experience in dance) to provide their own interpretation of Fluidity. The evaluation set consisted of 42 stick-figures animations: 21 segments where the dancer was asked to explicitly perform movements characterized by high

Fluidity and 21 segments, where she was expressing non-fluid qualities (e.g., Rigidity, Impulsivity). Segments duration is between 3 and 22 seconds (total duration 5m and 34s).

A web page displayed single full-body skeleton animations of motion capture data corresponding to one segment. Participants could watch each animation as many times as they wanted. Each participant had to evaluate maximum 20 animations. Animations were displayed in a random order: each new animation was chosen among the animations that received the smaller number of evaluations. In this way, we obtained a balanced number of evaluations for all segments.

3.2.2.5 Fluidity computational algorithm

In this work, as a proof of concept, the Spring Mass model was used to simulate a dancer's body and to compare the various recorded performances with the movements generated by the model. Since the model was designed (by experimentally tuning its parameters) to generate the smoothest trajectories, it has been used as reference to estimate Fluidity.

Algorithm 1 Fluidity estimation from 3D coordinates: for each frame k of a MoCap segment i, an estimation of the jerkness JI_k is computed, finally an estimation of the mean jerkiness JI_i of the segment is calculated

```
while A new frame of the segment $s_i$ is available do
    let $X_k$ be a set of coordinates measured at frame $k$;
    set $X_k$ as target position for the model
    let the model evolve and get the simulated set $Y_k$;
    evaluate $JI_k$ as in Equation 3.1;
    update mean value $JI_i$;
    wait next data frame;
```

In particular, we computed the mean jerk values of the shoulders (s), elbows (e) and hands (h) for both the original measurements and the ones simulated by the model. By measuring the distance of the overall jerk of the captured data and the synthesized one, we defined the quantity JI as a rough estimation of Fluidity in the movement of a given segment. The JI index, at frame k, is computed as:

$$JI_k = JI_k^l + JI_k^r \tag{3.1}$$

where JI_k^l and JI_k^r are respectively:

$$JI_k^l = |(\dddot{X}_k^{ls} + \dddot{X}_k^{le} + \dddot{X}_k^{lh}) - (\dddot{Y}_k^{ls} + \dddot{Y}_k^{le} + \dddot{Y}_k^{lh})| \tag{3.2}$$

and:

$$JI_k^r = |(\dddot{X}_k^{rs} + \dddot{X}_k^{re} + \dddot{X}_k^{rh}) - (\dddot{Y}_k^{rs} + \dddot{Y}_k^{re} + \dddot{Y}_k^{rh})| \tag{3.3}$$

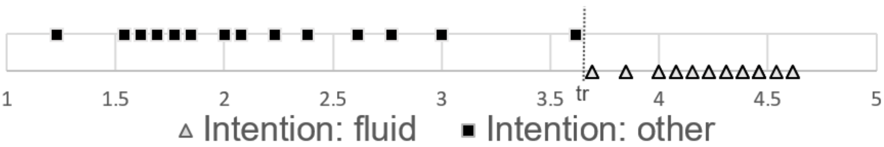


Figure 3.4: the average scores for 42 segments divided according to the dancers' intention.

The procedure for evaluating the Fluidity estimation of a segment is explained in Algorithm 1.

3.2.2.6 Results

We collected 546 answers from 41 participants (15 females, age = 23.5 (Mean = 30.7, SD = 5.8), 11 Nationalities (63% Italy, 10% France)). Each animation was evaluated 13 times. Figure 3.4 presents average of User Evaluations, UE_i, of segments s_i.

Figure 3.4 shows how segments intended to be fluid can be easily separated from the other segments. We defined two sets of segments by applying a threshold on the user evaluation scores ($tr = 3.65$): ***High Fluidity Segments***, i.e., $HFS = \{s_i : UE_i \geq tr\}$, and ***Low Fluidity Segments***, i.e., $LFS = \{s_i : UE_i < tr\}$. In this way, HFS contains all segments in which dancers were asked to express high Fluidity.

The means of UE_i are bigger than 3.65 for segments where the dancers were asked to move fluid, and they are lower than 3.65 for all other segments. Next, we also applied ANOVA with *intention* (i.e., Fluid Vs. Other) as independent variable and the *average participants' answers* UE_i as dependent variable. Participants' answers were significantly higher for segments intended to express a high Fluidity ($F(1, 41) = 215.102, p < .001$).

	Fluid	Other
UE	4.2 ± 0.29	2.05 ± 0.61

Table 3.1: results of the perceptive users' evaluation (UE).

Table 3.2 shows results on the computation of JI on the recorded data. Movement segments identified as fluid are characterized by a statistically significant lower JI index ($F(1, 41) = 11.45, p < .001$) than non fluid movements.

	Fluid	Other
JI	0.0198 ± 0.0004	0.043 ± 0.0005

Table 3.2: results of the index computed by the proposed algorithm (JI).

Results indicate that the JI index may be useful in identifying fluid movements.

3.2.3 Automated detection of expressive qualities of movement: a machine learning approach

The study presented in this section was carried out within context of the DANCE project and represents an attempt of application of an experimental methodology (in this particular case, concerning the automated classification and real-time detection of expressive movement qualities) to a subset of expressive features.

As illustrated at the beginning of the chapter, in the past 2 years, several meetings with experts in dance and choreography has been organized to investigate how physical and low-level movement features (e.g., speed, acceleration, direction, energy, and so on) can be exploited and integrated to model expressive ones (e.g., Fluidity, Rigidity). In particular, through meetings, interviews, and movement recording sessions with the famous contemporary dance choreographer Virgilio Sieni[1], we defined an *expressive vocabulary of movement basics* allowing a person to communicate, for example, emotional states by non verbal communication.

More precisely, we investigated two qualities, belonging to the defined expressive vocabulary and added to the third layer of the conceptual framework (introduced in Chapter 2). The names of the qualities are *Lightness* and *Fragility*, which are defined as follows:

- **Lightness**: A movement expressing Lightness should include at least one of the following characteristics: (i) it should exhibit a low amount of downward vertical acceleration following gravity (in particular on forearms and knees), (ii) each possible downward acceleration should be counterbalanced by an opposite upward movement (simultaneous or consequent); (iii) vertical downward acceleration movements should be finalized on the horizontal plane.
  An example of a dancer performing Light movements can be seen at: `https://youtu.be/5Yk35QgyQ1A`
- **Fragility**: A sequence of non-rhythmical upper body cracks and leg releases. It emerges, for example, when moving at the boundary between balance and fall. Can be described as a sequence of short movements with continuous interruption and re-planning of motor plans. The resulting movement is non-predictable, interrupted, and uncertain.
  An example of a dancer performing Fragile movements can be seen at: `https://youtu.be/XcEhc0_uuvA`

3.2.3.1 Classification of Movement Qualities

Recent computational models and analysis techniques were developed to automatically compute and analyze different movement expressive qualities, e.g., [NF03, CRB$^+$07, KBB13] (see

[1] `http://www.virgiliosieni.it`

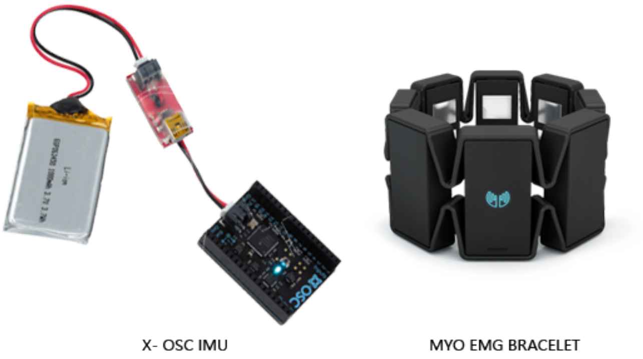


Figure 3.5: Typologies of wearable sensors: accelerometers (gyroscopes and magnetometers) and electromyograph.

[NMP13] for a more complete review). It can be observed that the majority of the existing works make use of high precision MoCap devices to perform accurate movements recordings.

One of the objective of DANCE is find a way to analyze the expressiveness of movement during live performances (and then translating it into sounds, an experiment on this topic is reported in the Appendix of the thesis). With this regard, we developed a *low-intrusive* approach that integrates data recorded by various type of wearable sensors such as Inertial Movement Units (IMUs) and electromyograph bracelets (EMG).

Being less intrusive as possible is the only way to collect data with an ecological validity. Our setup and approach can be used outside the scientific laboratory and during artistic performances, without interfering with them. Using a MoCap system or a Kinect during a live performance is problematic.

MoCap is affected by several issues:

- is not practical (long time is needed to dress dancers; the system needs to be calibrated before use),
- it may limit the dancers' movements (inertial MoCap),
- is difficult in public spaces (optical MoCap).

Kinect, instead is too sensitive to:

- changing light conditions,
- 360 degrees movements,
- dimensions of the stage and number of participants.

3.2.3.2 Dataset

We recorded a dataset of short performances. At the beginning of each session, dancers were given the definitions of the expressive qualities (Lightness and Fragility). Next, the dancers were asked to perform an improvised choreography containing movements that, in their opinion, express the qualities. 13 female dancers, with different dance backgrounds (classic dance, pop, contemporary dance), and different levels of professional experience, participated to the experiment. They performed five repetitions for each expressive quality, each trial had a duration of 1 minute. All dancers were wearing black clothes.

Data Streams:

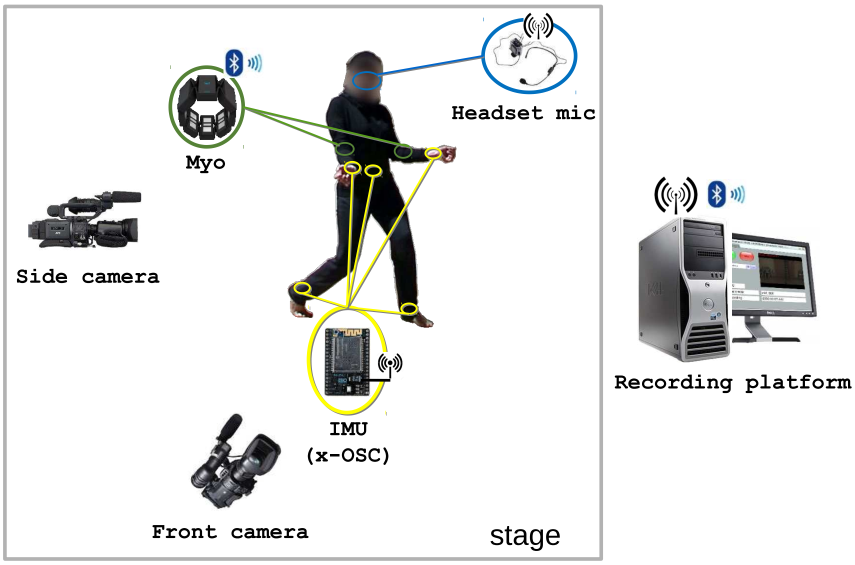


Figure 3.6: Setup for multi modal recordings

We recorded data streams from the following devices (see Figure 3.5):

- 5 IMU sensors (x-OSC[2]) placed on the dancer's body limbs; the data is captured at 50 frames per second; each frame consists of 9 values: (x, y, z) of accelerometer, gyroscope, and magnetometer;
- 2 video cameras (1280x720, at 50fps);
- 2 EMG armband placed on the dancer's forearms (MYO[3]); data is captured at 50 frames per second. Each armband streams 8 different EMG signals at each frame;

[2] http://x-io.co.uk/x-osc

[3] https://www.myo.com

- one wireless microphone (Mono, 48 kHz) placed below the dancer's nose, to record breathing;

Figure 3.6 shows the recording setup. Data was recorded and synchronized the EyesWeb XMI platform (see Chapter 8.2 for more details about the platform). Data streams synchronization is obtained by using SMPTE timecodes [4].

Segmentation:

The recorded video streams were evaluated by dance experts and expressive movement analysis experts. For every trial, they identified segments of about 10s each corresponding to a uniform, coherent sequence of movements. For each dancer and each expressive quality, between 5 and 6 segments were selected, for a total of 150 segments. The details of the segmentation are presented in Table 3.3.

Table 3.3: List of segments

Quality	No. Segments	Mean duration	Total duration
Lightness	77	10.2s	13m 6s
Fragility	73	10.4s	12m 41s
Total	150	10.3s	25m 46s

Ranking:

Five raters watched the 150 segments resulting from the segmentation. They observed each video segment and they were asked to rate the global level of Fragility and Lightness they perceived by using two independent 5-point Likert scales (from 0 to 4).

The raters did not hear any audio. We blurred the face of the dancer to prevent the rater to identify her and to avoid that facial expressions could affect the ratings. The raters were given the definitions of the expressive qualities and how they were computed (to avoid confusion and, for example, letting them know that there is no computation involving feet movements, as explained in the next section).

We checked the inter-rater reliability between the raters using weighted Cohen κ and Pearson correlation r. The mean pairwise linear weighted Cohen agreement for 5-point scale values are: 0.30 for Lightness and 0.40 for Fragility. The mean correlation values are: 0.46 for Lightness and 0.58 for Fragility.

Next, for each segment we computed the average scores $RankLI$ and $RankFR$ of the perceived Lightness and Fragility between the 5 raters.

[4] SMPTE is a standard in multimedia content production.

Table 3.4: Number of segments per rank interval

	Lightness	Fragility
Average rank 1 or less	39	73
Average rank between 1 and 2	60	39
Average rank between 3 and 2	35	32
Average rank between 4 and 3	16	6
Total	150	150

As Table 3.4 shows, many segments received only medium average levels of Lightness and Fragility ranks. Additionally, Lightness scores are better distributed than Fragility scores.

Indeed, many segments were perceived as not expressing high Fragility even when the dancers were asked to do it. This might be due to different dance backgrounds of the dancers. In particular, Fragility is a cue that does not appear in classical ballet and thus it might be difficult for some of the dancers to express it in a way that raters can perceive it. Consequently, we decided to define four subsets containing segments of high and low rank for each quality:

- FR_{low} contains 48 segments (out of 150) for which the ranked average Fragility score was below 0.4,
- FR_{high} contains 44 segments (out of 150) for which the ranked average Fragility score was above 2,
- LI_{low} contains 48 segments (out of 150) for which the ranked average Lightness score was below 1.2,
- LI_{high} contains 40 segments (out of 150) for which the ranked average Lightness score was above 2.3.

The choice of the thresholds was made to balance the number of segments in each subset.

3.2.3.3 Features and Descriptors

In this section, the features and descriptors used to classify Lightness and Fragility are presented. It is important specify that only data from the two IMUs placed on the participant's hands and from the two EMG bracelets placed on the forearms were used.

We defined a set of four features:

- Features I_1 and I_2 are inspired by the movement qualities and are computed from two IMU sensors placed on the participant's wrists.

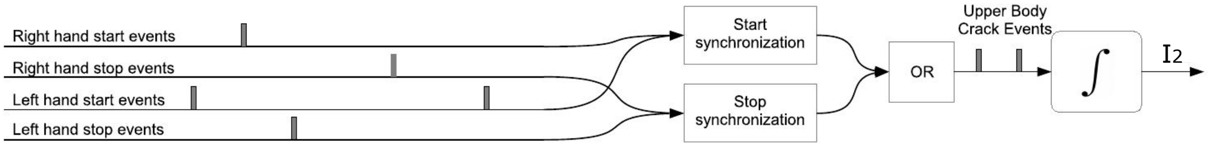


Figure 3.7: Feature I_2 extraction algorithm. Acceleration and deceleration phases were extracted on both hands. If synchronization between the two hands peaks is present, then an upper body crack is detected.

- Features E_1 and E_2 are computed from two EMG sensors placed on the participant's forearm.
- Additionally, for each feature, we extracted a set of descriptors (e.g., mean, standard deviation, and so on).

$\boldsymbol{I_1}$ is inspired by the definition of Lightness and it is computed as:

$$I_1 = 1 - \frac{W_r + W_l}{2} \tag{3.4}$$

where W_r and W_l are the ratios of the energy' vertical component on the total energy. *r* and *l* indicates left and right hand, respectively.

$\boldsymbol{I_2}$ is inspired by the definition of Fragility. We detect the *start* and *stop* instants of hands movements using the 3-axis linear acceleration of the two inertial sensors attached to the participant's wrists, by looking for acceleration and deceleration peaks. The output of this process are four binary time series, where a 1 or 0 encode whether a peak is detected or nor respectively.

Then, the synchronization between *start* and *stop* instants of both hands is measured (i.e., whether both hands *start* and *stop* to move simultaneously). If synchronization emerges we identify an *Upper body crack*. Peak synchronization is computed by using the Event Synchronization algorithm with Tau = 10 samples. Finally, I_2 is computed sum of the Upper body cracks, normalized on a fixed time window.

For both I_1 and I_2 the following descriptors are computed: $MEAN_{I1}$, STD_{I1}, MIN_{I1}, MAX_{I1}. and $MEAN_{I2}$, STD_{I2}, MIN_{I2}, MAX_{I2}.

$\boldsymbol{E_1}$ and $\boldsymbol{E_2}$ are both related to EMG signals. EMG are supposed to characterize Fragility movements by modeling the presence of quick alternation between high and low muscle tension. As previously mentioned, each MYO armband is made by eight sensors. Then, at each frame d_k (with $k = 1..N$), we receive a vector of eight EMG measures x_j^k where $j = 1..8$.

We define E_1 and E_2 for the left hand as:

$$E_{1,l}(d_k) = \frac{\sum_j^8 \|x_j^k\|}{8} \tag{3.5}$$

and

$$E_{2,l}(d_k) = max_j(x_j^k) \tag{3.6}$$

Analogously to I_1 and I_2 the following descriptors are computed: $MEAN_{E1}$, STD_{E1}, MIN_{E1}, MAX_{E1}. and $MEAN_{E2}$, STD_{E2}, MIN_{E2}, MAX_{E2}. Additionally, Willison Amplitude (WAMP) and Waveform Length (WL) descriptors are computed as:

$$\begin{aligned} WAMP_{Ei} &= \sum_{k=1}^{N} f(\|E_i(d_k) - E_i(d_{k+1})\|), \\ where \quad f(x) &= \begin{cases} 1 & \text{if } x < threshold \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{3.7}$$

and

$$WL_{Ei} = \sum_{k=1}^{N} \|E_i(d_{k+1}) - E_i(d_k)\| \tag{3.8}$$

Note: for features E_1 and E_2 descriptors are computed for each hand separately. To obtain a single value. the mean, between left and right hand descriptors, is computed.

3.2.3.4 Validation

The features proposed in the previous section were extracted on the segments in sets FR_{low}, FR_{high}, LI_{low} and LI_{high}. Then, we applied a statistical analysis to check whether there are significant differences between the values of each feature for the two pairs of sets: FR_{low}, FR_{high} and LI_{low}, LI_{high}.

First, the values of all the descriptors were normalized in interval $[0, 1]$. Next, we applied ANOVA with Lightness (FR_{low} vs. FR_{high}) as independent variable and the descriptors $MEAN_{I1}$, STD_{I1}, MAX_{I1}, MIN_{I1}, $MEAN_{E1}$, STD_{E1}, $WAMP_{E1}$, WL_{E1}, MAX_{E1}, MIN_{E1}, $MEAN_{E2}$, STD_{E2}, $WAMP_{E2}$, WL_{E2}, MAX_{E2}, MIN_{E2} as dependent variables (see Table 3.5). As a result, descriptors $MEAN_{I1}$ ($F(1, 86) = 57.743$, $p < 0.001$), MAX_{I1} ($F(1, 86) = 4.315$,

$p < 0.05)$, MIN_{I1} $(F(1, 86) = 65.102, p < 0.001)$ had a significantly higher value for segments perceived to express a high Fragility, whilst descriptors STD_{I1} $(F(1, 86) = 72.865, p < 0.001)$, $MEAN_{E1}$ $(F(1, 86) = 5.918, p < 0.05)$, STD_{E1} $(F(1, 86) = 21.383, p < 0.001)$, WL_{E1} $(F(1, 86) = 4.974, p < 0.05)$, MAX_{E1} $(F(1, 86) = 29.814, p < 0.001)$ $MEAN_{E2}$ $(F(1, 86) = 4.748, p < 0.05)$, STD_{E2} $(F(1, 86) = 12.624, p < 0.01)$, MAX_{E2} $(F(1, 86) = 11.585, p < 0.01)$ had significantly lower values. Similarly, we applied ANOVA with Fragility (FR_{low} vs FR_{high}) as independent variable and the descriptors $MEAN_{I2}$, STD_{I2}, MAX_{I2}, MIN_{I2}, $MEAN_{E1}$, STD_{E1}, $WAMP_{E1}$, WL_{E1}, MAX_{E1}, MIN_{E1}, $MEAN_{E2}$, STD_{E2}, $WAMP_{E2}$, WL_{E2}, MAX_{E2}, MIN_{E2} as dependent variables (see Table 3.5).

Table 3.5: Means and standard deviations obtained for each descriptor and subset of the dataset. Significant differences are in bold.

Desc.	$LIlow$	$LIhigh$	$FRlow$	$FRhigh$
$MEAN_{I1}$	**0.59 (0.24)**	**0.90 (0.09)**	-	-
STD_{I1}	**0.49 (0.21)**	**0.17 (0.13)**	-	-
MAX_{I1}	**0.94 (0.19)**	**0.99 (0.0)**	-	-
MIN_{I1}	**0.39 (0.23)**	**0.77 (0.20)**	-	-
$MEAN_{I2}$	-	-	**0.05 (0.10)**	**0.43 (0.30)**
STD_{I2}	-	-	**0.08 (0.14)**	**0.39 (0.21)**
MAX_{I2}	-	-	**0.09 (0.16)**	**0.50 (0.24)**
MIN_{I2}	-	-	**0.01 (0.02)**	**0.13 (0.20)**
$MEAN_{E1}$	**0.40 (0.23)**	**0.28 (0.18)**	**0.32 (0.19)**	**0.24 (0.17)**
STD_{E1}	**0.47 (0.24)**	**0.25 (0.19)**	0.31 (0.20)	0.29 (0.18)
MAX_{E1}	**0.51 (0.20)**	**0.28 (0.20)**	0.34 (0.20)	0.36 (0.19)
MIN_{E1}	0.35 (0.20)	0.34 (0.20)	**0.38 (0.20)**	**0.26 (0.15)**
$WAMP_{E1}$	0.74 (0.12)	0.74 (0.16)	**0.76 (0.14)**	**0.68 (0.15)**
WL_{E1}	**0.36 (0.21)**	**0.27 (0.19)**	0.29 (0.18)	0.24 (0.17)
$MEAN_{E2}$	**0.43 (0.26)**	**0.32 (0.22)**	0.35 (0.20)	0.27 (0.23)
STD_{E2}	**0.50 (0.23)**	**0.33 (0.20)**	0.36 (0.18)	0.32 (0.20)
MAX_{E2}	**0.85 (0.20)**	**0.68 (0.26)**	0.74 (0.23)	0.70 (0.28)
MIN_{E2}	0.27 (0.18)	0.27 (0.19)	**0.31 (0.20)**	**0.19 (0.13)**
$WAMP_{E2}$	0.73 (0.19)	0.73 (0.19)	**0.77 (0.15)**	**0.60 (0.23)**
WL_{E2}	0.45 (0.26)	0.35 (0.25)	0.37 (0.23)	0.29 (0.24)

As a result, descriptors $MEAN_{I2}$ $(F(1, 90) = 67.368, p < 0.001)$, STD_{I2} $(F(1, 90) = 70.368, p < 0.001)$, MAX_{I2} $(F(1, 90) = 92.379, p < 0.001)$, MIN_{I2} $(F(1, 90) = 20.435, p < 0.001)$ had significantly higher values for segments perceived to express a high Fragility while descriptors $MEAN_{E1}$ $(F(1, 90) = 3.986, p < 0.05)$, $WAMP_{E1}$ $(F(1, 90) = 6.778, p < 0.05)$, MIN_{E1} $(F(1, 90) = 9.733, p < 0.01)$, $WAMP_{E2}$ $(F(1, 90) = 18.326, p < 0.001)$, MAX_{E2} $(F(1, 90) = 11.576, p < 0.01)$ had significantly lower values.

3.2.3.5 Classification

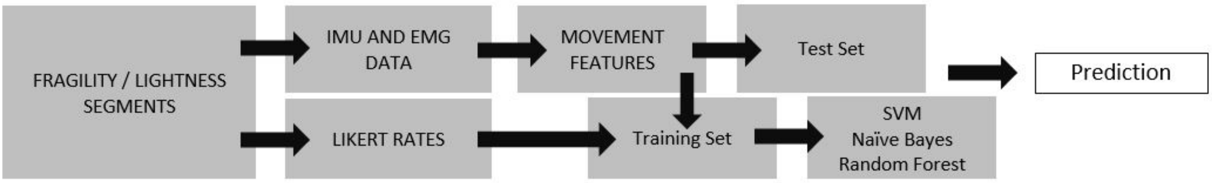


Figure 3.8: Graphical visualization of the classification study

We build two different models: one for Fragility and one for Lightness using different supervised machine learning algorithms and a subset of 11 descriptors out of the 20 described in the previous section: i.e., $MEAN_{I1}$, STD_{I1}, MIN_{I1}, $MEAN_{I2}$, STD_{I2}, MAX_{I2}, MIN_{I2}, STD_{E1}, MAX_{E1}, MIN_{E2} and $WAMP_{E2}$.

To build the Lightness model we split the dataset into two classes by applying the threshold of 1.6 on $RankLI$ (i.e., the median of $RankLI$; $RankLI$ and $RankFR$ are the averarge rating scores introduced in Section 3.2.3.2). As a consequence, we obtain 72 segments expressing Lightness and 78 expressing no or low Lightness. Similarly, we applied the threshold of 1.2 on $RankFR$ (i.e., the median value of $RankFR$) to split the segments in those where Fragility was perceived (68 segments) and those where no or low Fragility was observed (82 segments).

For both models, we tested the performance of 3 supervised machine learning algorithms: a Support Vector Machine (SVM) with polynomial kernel, a Random Forest (RF) and a Naive Bayes (NB) classifier.

The averaged performance of each classifier was assessed via a multiple run and Leave-One-Out Method. In our study, we adopted 100 runs. Table 3.6 shows the performance of each classifier in terms of average Accuracy, Precision, Recall, and F-score.

Additionally, we also ran the same machine learning algorithms on the pairs of sets FR_{low}, FR_{high} and LI_{low}, LI_{high}. As expected, performances are higher with Random Forest compared to the results computed on the whole dataset, obtaining an average F-score of 0.86 for Lightness and of 0.8 for Fragility. Table 3.7 shows the results.

Table 3.6: Average Accuracy, Precision, Recall, F-score for all segments

	Lightness				Fragility			
	Avg. Accuracy	Avg. Recall	Avg. Precision	Avg. F-score	Avg. Accuracy	Avg. Recall	Avg. Precision	Avg. F-score
SVM	0.68	0.65	0.67	0.66	0.79	0.81	0.74	0.77
NB	0.77	0.79	0.74	0.77	0.75	0.59	0.8	0.68
RF	0.75	0.71	0.75	0.73	0.77	0.72	0.75	0.74

Table 3.7: Average Accuracy, Precision, Recall, F-score for “best” segments

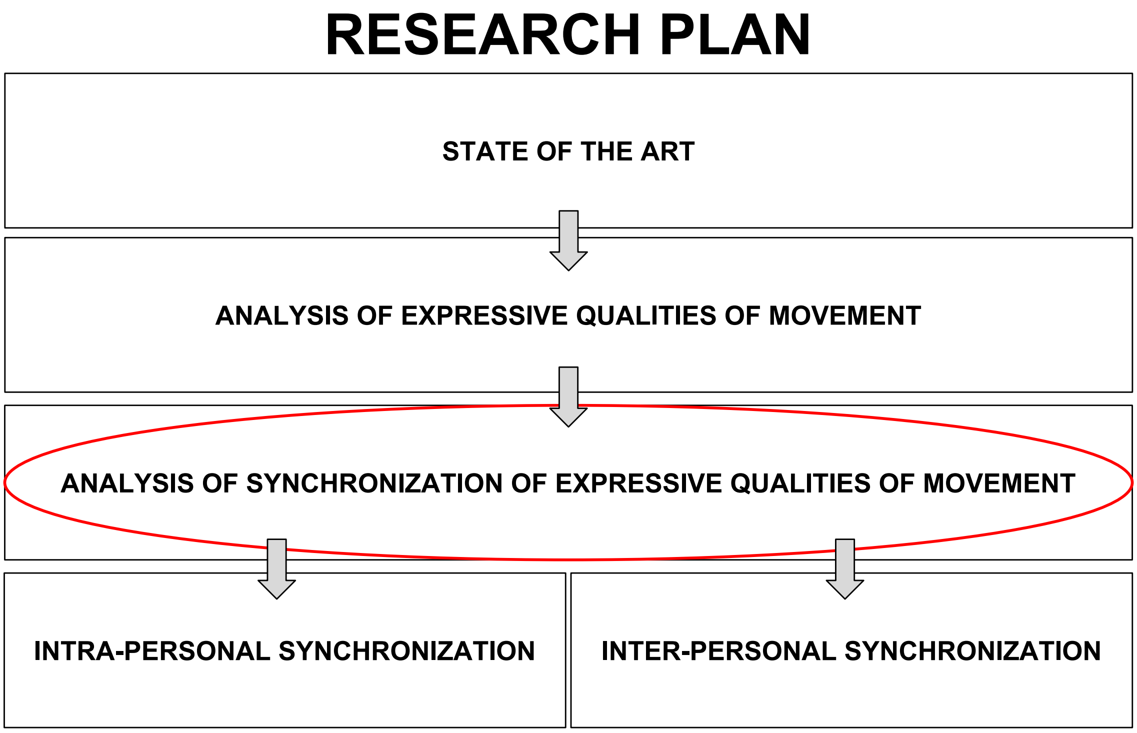
	Lightness				Fragility			
	Avg. Accuracy	Avg. Recall	Avg. Precision	Avg. F-score	Avg. Accuracy	Avg. Recall	Avg. Precision	Avg. F-score
SVM	0.76	0.73	0.74	0.73	0.76	0.84	0.79	0.81
NB	0.80	0.73	0.81	0.76	0.76	0.84	0.77	0.80
RF	0.86	0.9	0.82	0.86	0.77	0.81	0.78	0.80

3.2.3.6 Conclusions

We developed and trained two supervised machine learning models to detect Fragility and Lightness. Results of our statistical analysis show that the proposed features permit to distinguish segments according to the degree of Lightness and Fragility perceived by human observers. In more detail, the best trained model (a Random Forest) is able to detect the two expressive qualities with an F-score of 0.77.

Chapter 4

The Multi Event Class Synchronization (MECS) algorithm


RESEARCH PLAN
STATE OF THE ART
ANALYSIS OF EXPRESSIVE QUALITIES OF MOVEMENT
ANALYSIS OF SYNCHRONIZATION OF EXPRESSIVE QUALITIES OF MOVEMENT
INTRA-PERSONAL SYNCHRONIZATION
INTER-PERSONAL SYNCHRONIZATION

Contents

In Section 2.2.2, we introduced the original algorithm to perform the Event Synchronization Analysis. Such algorithm, illustrated in [QKG02], has been subject to a series of extensions, thus originating family of new methods and algorithms, all of them based on the relative timings of events, applied mainly in the measurement of synchronization in biological signals.

A novel extension is presented in this chapter: the *Multi-Event-Class Synchronization* (*MECS*) is an algorithm to measure the amount of synchronization between relevant events, detected in two or more time series. MECS was entirely developed during the two years preceding the formulation of this dissertation. The MECS algorithm can be applied to a large variety of problems. Among others, it can be used by human centred systems and multi-modal interfaces with the long-term goal of endowing machines with the capability to "decode" human behaviors.

4.1 Introduction

As previously mentioned, recently, Quiroga and colleagues' algorithm has been extended in order to measure the degree of synchronization between events occurring in a set of time series (instead of a single pair). These family of techniques and event-sync based algorithms (e.g., [KCH^{+}12], [IR16]), are also known under the name of *Measures of spike train synchrony* [Kre11].

With respect to other existing techniques, MECS can deal with multiple classes of events. The term “events” denote a significant behavior for a system, an application and so on. We can think to events as reduced and collapsed representations of continuous information into the discrete domain (often events are equivalent to time stamps). Usually events are detected by recognizing particular characteristics of the time series profiles (e.g., peaks or valleys of the profile). After grouping events in 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*). Moreover, events can be combined in *macro-events* on which synchronization is measured. A relevant example of macro-event is a *sequence* of events. Finally, events and macro-events can be grouped in *macro-classes* and synchronization can be computed within and between them.

4.2 Related work

Iqpal and Riek [IR16] proposed an extension of the original ES algorithm to deal with multiple types of events. In their approach, given the two time series x and y and the event type E_i, $i = 1...K$, they first compute the synchronization index $Q_\tau(E_i)$ for all events of type E_i by adopting Quiroga and colleagues' equations 2.7 - 2.9. Next, they compute synchronization of multiple types of events between x and y as the average of $Q_\tau(E_i)$, weighted by the number of events of type E_i. In the last step, they consider M different time series $z_1, ...z_M$ and they compute their pairwise and overall synchronization.

In particular, the individual synchronization of a given time series z_i is the average of all the pairwise synchronizations between z_i, and each time series $z_j : i \neq j$, that are beyond a certain fixed threshold Q_{tresh}. The overall synchronization is the average of the products of the individual synchronization indexes multiplied by their connectivity values, where the connectivity value is the number of time series pairs having a pairwise synchronization above the threshold Q_{tresh} divided by the total number of possible pairs. Kreuz and colleagues presented an improved and simplified extension of event synchronization holding for both the bivariate and the multivariate case to measure the degree of synchrony from the relative number of appearances of spikes.

The Multi-Event Class Synchronization (MECS) algorithm is an extension of the background works by Quian Quiroga [QKG02], Iqpal and Riek [IR16], and Kreuz [KCA$^+$09]. MECS introduces new complementary characteristics that are missing in these works.

In Table 4.1 we compare these four algorithms in terms of: the number of input time series; the number of event classes; the possibility of handling macro-events, such as for example those induced by temporal constraints (sequences of events), and hierarchies of classes (macro-classes). If the MECS algorithm is applied on two time series considering only one class of events, it provides the same result as [QKG02]. If it is applied on pairs taken from a set of M time series considering N event classes, it provides the same synchronization as in [IR16]. Finally, if applied

Table 4.1: Comparison between existing synchronization algorithms [QKG02, IR16, KCA$^+$09] and the proposed one. MECS introduces the computation of synchronization between N event classes over M time series and it handles manipulations of the events as macro-events (e.g., sequences) and of the event class set as macro classes.

Reference	**Number of time series**	**Number of classes**	**Macro events**	**Macro classes**
Quian Quiroga et al. [QKG02]	2	1	not handled	not handled
Iqpal and Riek [IR16]	M	N	not handled	not handled
Kreuz et al. [KCA$^+$09]	M	1	not handled	not handled
MECS	M	N	handled	handled

on M time series and a single event class it provides the same synchronization as in [KCA$^+$09].

Differently from Quian Quiroga and colleagues, but similarly to Iqpal and Riek, we consider events belonging to multiple classes. We, however, differentiate from Iqpal and Riek, as they compute synchronization class by class (i.e., intra-class synchronization, involving all the events belonging to a class in particular), whilst we also compute synchronization between events belonging to different classes (inter-class synchronization). Similarly to Kreuz and colleagues, MECS considers multivariate time series, but differently from them MECS manages multiple classes of events. Moreover, MECS introduces the computation of synchronization between N event classes over M time series and it handles manipulations of the events as macro-events (e.g., sequences) and of the event class set as macro classes.

4.3 Multi-Event Class Synchronization

Multi-Event Class Synchronization (MECS) computes the amount of synchronization between events occurring in a set of time series. Events may belong to several different classes.

MECS can compute: (i) a separate synchronization index for each class (*intra-class synchronization*), (ii) a synchronization index for specific aggregations of classes (*inter-class synchronization*), and a global synchronization index for all classes.

In presenting MECS, we make reference to a sample scenario, which will help explaining the major features of the algorithm in a concrete way. Suppose we are interested into measuring the level of motor coordination between the members of a group of N users performing a motor task (e.g., a fitness exercise). A measure of coordination can be obtained by evaluating the amount of synchronization between the movements the users.

Let us consider:

- A set T of N time series: $T \equiv \{TS_1, ...TS_N\}$

- A set E of K event classes: $E \equiv \{E_1, ..., E_K\}$

A time series is sequence of discrete-time data that are modeling behaviors of phenomena of the real world. Time-series can either contain events or events can be identified within them automatically (for example, events can be described by points in time where the data behavior changes from increasing to decreasing or vice versa, or in correspondence of peaks or valleys) or artificially (i.e., through manual annotations that denote significant instants and are encoded within the time series data).

Additionally, events can belong to several classes. For example, they can describe different behavior changes in the data, such as peaks and valleys. Each class $E_1, ..., E_K$ characterizes a different typology of event i.e., it represents something relevant, that can either be detected or annotated within the time series.

Coming back to our example, we can think of the time series $TS_1, ...TS_N$ as if they completely describe the motor activity of each user (i.e., time series TS_j models the motor activity of user j) and can potentially contain events of any class $E_1, ..., E_K$. Each class may identify a specific movement of interest (for example movements like “step performed”, “object grabbed”, “object released”, and so on).

4.3.1 Intra-class synchronization

MECS relies its computations on the temporal distances between events. Similarly to [QKG02] [Kre11], it consists of two steps:

- detects coincidences in the time instants at which events occur in different time series (*coincidence detection*) and counts them.
- normalizes the number of detected coincidences is with respect to the total number of possible coincidences that may happen (*normalization*).

We associate to each event h of class $E_k \in E$, occurring in time series $TS_n \in T$, its occurrence time as:

$$t_h^{n,k} \qquad h = 1, ..., m_{n,k} \tag{4.1}$$

where $m_{n,k}$ represents the total number of detected events of class E_k occurring in the n-th time series TS_n. For example, $t_4^{2,3}$ represents the time at which the fourth event belonging to the third class, i.e., E_3, occurred in TS_2.

In the coincidence detection phase, for each pair of time series $< TS_i \ ; \ TS_j >$, the MECS algorithm computes the temporal coincidence between an event x detected on time series TS_i and another event y (of the same class of x) detected on time series TS_j (with $i \neq j$) by measuring the extent to which they are close in time within a certain interval τ_k (*coincidence window*) that may depend on the class of the events.

The temporal distance d between events x and y is computed as:

$$d(x,y) = |t_x^{i,k} - t_y^{j,k}| \tag{4.2}$$

The amount of coincidence c_k between x and y is defined as follows:

$$c_k(x,y) = \begin{cases} 1 - \frac{d(x,y)}{\tau_k} & \text{if} \quad 0 \leq d(x,y) \leq \tau_k \\ 0 & \text{otherwise} \end{cases} \tag{4.3}$$

Differently from [QKG02] and [Kre11], where coincidence is only detected, here it is also quantified, being $c_k(x,y) \in [0,1]$. The coincidence window τ_k can either be determined or estimated from the dynamics of the phenomenon the specific class of events refer to, or it can be automatically calculated for each pair of events x and y as proposed in [Kre11], i.e.:

$$\begin{aligned} \tau_k^{x,y} = \frac{1}{2}\dot{\min}\{t_{x+1}^{i,k} - t_x^{i,k}, t_x^{i,k} - t_{x-1}^{i,k}, \\ t_{y+1}^{j,k} - t_y^{j,k}, t_y^{j,k} - t_{y-1}^{j,k}\} \end{aligned} \tag{4.4}$$

For each class E_k, the overall coincidence $C_k(i|j)$ of all the events of class E_k in time series TS_i with respect to the events of the same class E_k in time series TS_j is computed as follows:

- First, the average coincidence of each event x in time series TS_i with respect to all the events in time series TS_j is calculated;
- Then the sum of the averages over all the events in time series TS_i is taken.

That is:

$$C_k(i|j) = \sum_{x=1}^{m_{i,k}} \frac{1}{m_{j,k}} \left[\sum_{y=1}^{m_{j,k}} c_k(x,y) \right] \tag{4.5}$$

Equation 4.5 shows that event x in TS_i can contribute to the overall coincidence by being coincident (at different extents) with more than one event in TS_j. Multiple coincidences are usually avoided and the computation of the coincidence window as in equation 4.4 is often performed with the exact purpose of minimizing the likelihood of counting multiple coincidences. Since MECS enables to weight coincidences so that a perfect coincidence has a weight of 1.0, and the amount of coincidence decreases along the coincidence window, it supports managing multiple coincidences that may indeed happen in some application contexts. Analogously, the overall coincidence $C_k(j|i)$ of all the events of class E_k in time series TS_j with respect to the events of class E_k in time series TS_i is computed by taking the average coincidence of each event y in time series TS_j with respect to all the events in time series TS_i and then summing such averages:

$$C_k(j|i) = \sum_{y=1}^{m_{j,k}} \frac{1}{m_{i,k}} \left[\sum_{x=1}^{m_{i,k}} c_k(y,x) \right] \tag{4.6}$$

Pairwise synchronization of the events of class E_k for the pair of time series $< TS_i\ ;\ TS_j >$ is computed as:

$$S_k(i,j) = \frac{C_k(i|j) + C_k(j|i)}{m_{i,k} + m_{j,k}} \quad S_k(i,j) \in [0,1] \tag{4.7}$$

Having defined the set $P \equiv \binom{T}{2}$ of all the 2-combinations of the set T (i.e., each element $p \in P$ is a distinct pair $< TS_i\ ;\ TS_j >$, with TS_i, $TS_j \in T$, $i \neq j$), the overall synchronization for the events of class E_k is finally obtained as:

$$Q_k = \frac{1}{|P|} \sum_{p \in P} S_k(i,j) \quad Q_k \in [0,1] \tag{4.8}$$

where the cardinality $|P|$ of set P is given by the number of 2-combinations of T, that is:

$$|P| = \binom{N}{2} = \frac{N!}{2!(N-2)!} \tag{4.9}$$

4.3.2 Global intra-class synchronization

To compute a global synchronization index SI for the events of all classes, we define the multi-class synchronization vector $\vec{Q}$ as:

$$\vec{Q} = [Q_1, ..., Q_K] \tag{4.10}$$

SI is obtained as a function of $\vec{Q}$, i.e., $SI = f(\vec{Q})$. A straightforward choice for f is the average over the K components of $\vec{Q}$. If event classes have e.g., different priorities, a set of weights

$\vec{W} = [W_1, ..., W_K]$ can be associated to each class of events and a weighted synchronization index is computed as:

$$SI_W = f(\vec{W}, \vec{Q}) \tag{4.11}$$

4.3.3 Inter-class synchronization

Given the set of event classes E, we may want to compute inter-class synchronization, i.e., between events that do not belong to the same class E_k, but rather to a pair of different classes E_{k1} and E_{k2}.

For each couple of time series $< TS_i \ ; \ TS_j >$, $TS_i, TS_j \in T$, $i \neq j$, the temporal coincidence between an event x found in the first time series TS_i and another event y found in the second time series TS_j measured by releasing the constraint that they belong to the same event class E_k, i.e., x belongs to class E_{k1} and y to class E_{k2} (E_{k1} and $E_{k2} \in E$). The measure of how much events x and y are close in time is computed within a certain interval $\tau_{k1,k2}$ (*coincidence window*) that may depend on the considered pair of classes of events. The temporal distance d between events x and y (refer to equation 4.2) is reformulated as:

$$d(x,y) = |t_x^{i,k1} - t_y^{j,k2}| \tag{4.12}$$

Accordingly, the relative amount of coincidence $c_{k1,k2}$ (refer to equation 4.3) becomes:

$$c_{k1,k2}(x,y) = \begin{cases} 1 - \frac{d(x,y)}{\tau_{k1,k2}} & \text{if} \quad 0 \leq d(x,y) \leq \tau_{k1,k2} \\ 0 & \text{otherwise} \end{cases} \tag{4.13}$$

For a pair of classes E_{k1} and E_{k2}, the overall coincidence $C_{k1,k2}(i|j)$ of all the events of class E_{k1} in time series TS_i with respect to the events of class E_{k2} in time series TS_j and analogously, the overall coincidence $C_{k1,k2}(j|i)$ of all the events of class E_{k1} in time series TS_j with respect to the events of class E_{k2} in time series TS_i are computed by:

$$C_{k1,k2}(i|j) = \sum_{x=1}^{m_{i,k1}} \frac{1}{m_{j,k2}} \left[\sum_{y=1}^{m_{j,k2}} c_{k1,k2}(x,y) \right] \tag{4.14}$$

$$C_{k1,k2}(j|i) = \sum_{y=1}^{m_{j,k2}} \frac{1}{m_{i,k1}} \left[\sum_{x=1}^{m_{i,k1}} c_{k1,k2}(y,x) \right] \tag{4.15}$$

Finally, inter-class pairwise synchronization of the events of class E_{k1} and events of class E_{k2}, for the pair of time series $< TS_i \ ; \ TS_j >$ is computed as:

$$S_{k1,k2}(i,j) = \frac{C_{k1,k2}(i|j) + C_{k1,k2}(j|i)}{m_{i,k1} + m_{j,k2}} \tag{4.16}$$

and the overall synchronization for the pair of class E_{k1} and E_{k2} (refer to equation 4.7):

$$Q_{k1,k2} = \frac{1}{|P|}\sum_{p \in P} S_{k1,k2}(i,j) \quad Q_{k1,k2} \in [0,1] \tag{4.17}$$

where P is the set of all the 2-combinations of the set T (refer to equation 4.9), and $S_{k1,k2}(i,j)$ and $Q_{k1,k2}$ both belong to $[0,1]$.

4.3.4 Event Class set manipulation

The MECS algorithm accepts as input a set of event classes E and a set of time series T and it is independent on how event classes are defined and how events are identified. In this section we will show how we can increase the number of cases of study that can be modeled and investigated just by performing simple logical operations on the event class set. In particular, we introduce two possible extensions:

- *Macro classes*
- *Macro events (in particular sequences)*

The first extension introduces the possibility to regroup the classes and compute the synchronization on different levels of abstraction corresponding to a hierarchical organization of the classes. The second extension permits to compute the synchronization between aggregations of events belonging to different classes.

4.3.5 Macro Classes

Let's define as $Pow(E)$ the power set of the event classes set E minus the empty set, i.e., $Pow(E) = \mathcal{P}(E)/\emptyset$.
$Pow(E)$ has cardinality $2^K - 1$. For example, if $E = \{E_1, E_2, E_3\}$, $Pow(E)$ will contain the following elements:

$$\begin{aligned} Pow(E) =& \{\{E_1\}, \{E_2\}, \{E_3\}, \{E_1, E_2\}, \\ & \{E_1, E_3\}, \{E_2, E_3\}, \{E_1, E_2, E_3\}\} \end{aligned} \tag{4.18}$$

We then define $\dot{E} \subseteq Pow(E)$ e.g., $\dot{E} = \{\{E_1, E_2\}, \{E_1, E_2, E_3\}\})$ and we consider $\dot{E}$ as the new set of event classes, i.e., in the presented example, $\dot{E_1} = \dot{E_1}, \dot{E_2}$ with $\dot{E_1} = \{E_1, E_2\}$ and $\dot{E_2} = \{E_1, E_2, E_3\})$.
In practice it is possible to take $2^K - 1$ subsets of the original elements of E, combining classes and merging them in *macro classes*.

Each generated macro class is actually a single class, or the combination of two or more classes of E. To compute synchronization, MECS will consider each item of each set in $\dot{E}$ as belonging to the same class. Synchronization is computed using the same methodology explained in section 4.3.1 and section 4.3.3 by using $\dot{E}$ as input event class set.

To be noticed that events that belong to one of the original K classes can belong to more than one macro class e.g., events of class E_1 belong to both $\dot{E_1}$ and $\dot{E_2}$ in $\dot{E}$.

4.3.6 Macro events

Events can be grouped in *macro-events* i.e., aggregations that require constraints to be satisfied. As an example, we illustrate sequences of events, where the constraint to be satisfied is the order of occurrence of each event in the sequence.

By considering again $Pow(E)$, a sequence S is defined as an ordered n-uple of elements (with repetitions) where each element is referred by a sequence index: $(1, 2, 3, ..., s - 1, s)$ Starting from the elements of E, examples of sequences are $S_1 = \{E_2, E_1, E_3\}$, $S_2 = \{E_1, E_1, E_3, E_1\}$, $S_3 = \{E_2, E_1\}$ and so on..

Let's define the following quantities:

1. $S[i]$: i-th element of sequence S that is an event class i.e., $S[i] \in E, \forall i \in 1, ..., s$
2. $t_e^{n,S[i]}$: the occurrence time of a generic event e found in the time series TS_n and that belongs to the $i - th$ class of sequence S i.e., any $t_e^{n,S[i]}$ with $e \in 1, ..., m_{S[i]}$.
3. IEI: the *Inter-Event Interval* i.e. the maximum time allowed between two events that belongs to two consecutive classes $S[i]$ and $S[i + 1]$ of sequence S to not interrupt the sequence.

Then, within a time series TS_n, a particular sequence S is detected if three conditions are true:

$\forall i \in 1, ..., s$

(a) $t_e^{n,S[i+1]} > t_e^{n,S[i]}$

(b) $t_e^{n,S[i+1]} - t_e^{n,S[i]} \leq IEI$

(c) no other $t_e^{n,l}$ occurs in $[t_e^{n,S[i+1]}$, $t_e^{n,S[i]}]$ where $l \in S$

When a sequence S is detected within a time series TS_n, it will be treated as an event belonging to a new class named E_S. Since sequences allows repetitions of the same elements, it is possible to define an *infinite* number of sequences. Let us call $\dot{S}$ the set of all the sequences defined starting by E. The synchronization degree between sequences is computed using the same methodology explained in 4.3.1 with $\dot{S}$ used as input event class set.

4.3.7 Extending the MECS Algorithm

So far, the MECS algorithm calculates the contribution each pair of events brings to synchronization using equation 4.3 that we recall here:

$$c_k(x,y) = \begin{cases} 1 - \frac{d(x,y)}{\tau_k} & \text{if} \quad 0 \leq d(x,y) \leq \tau_k \\ 0 & \text{otherwise} \end{cases} \tag{4.19}$$

The algorithm is based on a function of distance $d(x, y)$. This formulation assumes that the distance in time between two events and their contribution to the synchronization index are linearly inversely proportional. This assumption may limit the potential of the algorithm and its use in specific cases. An extension of MECS consists of modeling the relationship between the previously mentioned quantities as a different (non linear) function, that we define as the *kernel function* ($Kern(d(x, y))$) of the MECS algorithm. That is:

$$c_k(x,y) = Kern(d(x,y)) \quad Kern(d(x,y)) \in [0,1] \tag{4.20}$$

Designing different shapes for this function allows the algorithm to better adapt to different contexts. The following section introduces a palette of candidate kernel functions.

The graph presented with each candidate kernel function shows how the contribution of an events pair to synchronization $c_k(x, y)$ (on the ordinate) varies with the distance in time between the events in the pair ($d(x, y)$ on the abscissa).

Some kernel functions (exponential and sigmoid kernels for example) are indeed a family of functions depending on some parameters. Such parameters can be tuned so that the kernel function covers the whole $[0, 1]$ interval for $c_k(x, y)$ and it reaches its maximum (i.e., $c_k(x, y) = 1$) at a time distance $d(x, y)$ between events, which is the most appropriate for the specific application where MECS is employed.

4.3.8 Kernel functions for MECS

Linear kernel

This is the original design of the kernel function $Kern(d(x, y))$, where the relationship between the distance in time between two events and their contribution to the synchronization index is linear and inverse proportional, as shown in Figure 4.1.

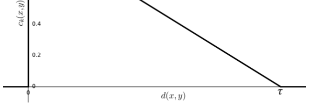


Figure 4.1: Example of a linear kernel function.

$$Kern(d(x,y)) = \begin{cases} 1 - \frac{d(x,y)}{\tau_k} & \text{if} \quad 0 \leq d(x,y) \leq \tau_k \\ 0 & \text{otherwise} \end{cases}$$

Step kernel

$Kern(d(x,y))$ consists of a step function as described below:

$$Kern(d(x,y)) = \begin{cases} 1 & \text{if} \quad 0 \leq d(x,y) \leq \tau_k \\ 0 & \text{otherwise} \end{cases} \tag{4.21}$$

The MECS algorithm with this kernel function is very similar to a multi class extension of the original even synchronization algorithm [QKG02]. The two algorithms are totally equivalent if there are only two time series ($N = 2$), a single event class (i.e., $K = 1$) and the following step kernel is chosen:

$$Kern(d(x,y)) = \begin{cases} 1 & \text{if} \quad 0 < d(x,y) \leq \tau_k \\ 1/2 & \text{if} \quad d(x,y) = 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.22}$$

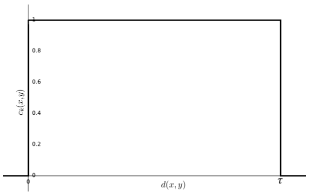


Figure 4.2: Example of a step kernel function.

Exponential kernel

$Kern(d(x,y))$ can be chosen as an exponential function (see Figure 4.3):

$$Kern(d(x,y)) == \begin{cases} e^{-sd(x,y)} & \text{if} \quad 0 \leq d(x,y) \leq \tau_k \\ 0 & \text{otherwise} \end{cases} \tag{4.23}$$

where parameter s represents the slope of the function.

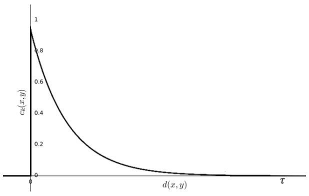


Figure 4.3: Example of an exponential kernel function.

Sigmoid kernel

$Kern(d(x,y))$ can have a sigmoid shape (see Figure 4.4)

$$Kern(d(x,y)) = \begin{cases} 1 - \frac{1}{1+e^{-\sigma d(x,y)+\mu}} & \text{if} \quad 0 \leq d(x,y) \leq \tau_k \\ 0 & \text{otherwise} \end{cases}$$

where σ and μ are parameters to set the steepness of the sigmoid and its mean value, respectively.

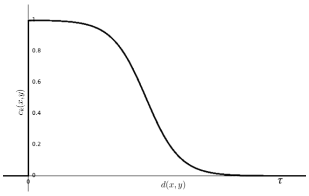


Figure 4.4: Example of a sigmoid kernel function.

Gaussian kernel

$Kern(d(x,y))$ can be conceived as a Gaussian function (see Figure 4.5):

$$Kern(d(x,y)) = \begin{cases} e^{-\frac{(d(x,y)-\mu)^2}{2\sigma^2}} & \text{if} \quad 0 \leq d(x,y) \leq \tau_k \\ 0 & \text{otherwise} \end{cases} \tag{4.25}$$

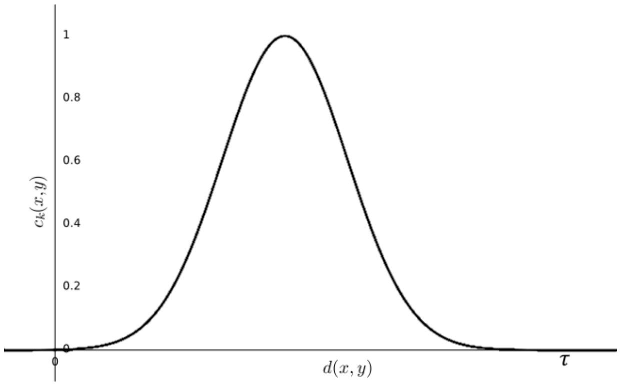


Figure 4.5: Example of a Gaussian kernel function.

where σ and μ ($\mu \in (0, +\infty)$) are parameters to set the width of the Gaussian and its mean value, respectively. This particular function shape was designed to extend the MECS synchronization algorithm towards the computation of causality, that connects one process (the *cause*) with another one (the *effect*), where the former is at least partly responsible for the latter, and consequently the latter is dependent on the former.

An initial, simple, characterization of causality is that of chronemic causality, based on the evidence that the *effect* always comes after the *cause*, specifically at an instant in time which is not too close to the cause (that would be perceived as synchronous) and not too far (in that case the two events would be perceived as independent). Grounding on this assumption, the MECS algorithm with a Gaussian kernel function can provide a rough but simple measure of chronemic causality. Please note that MECS can give an estimation of chronemic causality but it will not indicate which is the cause and which the effect, as we will show with examples in Section 4.4.

4.3.9 MECS Implementation

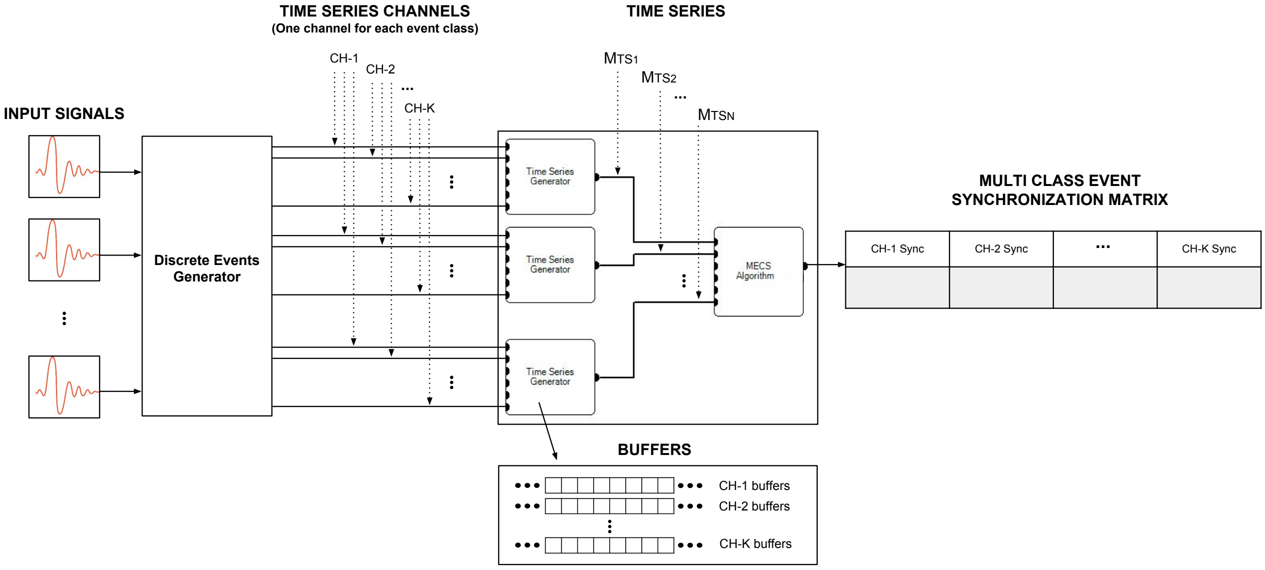


Figure 4.6: The MECS algorithm computing the degree of synchronization of K classes of events that have been extracted from N input signals.

In this section we present a possible of implementation of the MECS algorithm. The schema represented in Figure 4.6 provides a logical, graphical representation of a sample MECS application. Individual elements are shown with their interrelations. In the figure is shown, from left to right, how from a set of input signals, MECS provides an output synchronization matrix Q.

For simplicity, we consider all the input signals being generated at the same time and at a fixed frequency. Input signals are sampled and streamed to the *Discrete Events Generator* module, that:

- identifies the presence of events in the input streams and dispatches them into classes (through event detection techniques).
- generates K discrete output streams $CH_1, ..., CH_K$ called *Channels* and forward them to the *Time Series Generators* modules. Each channel CH_i correspond to a single event class of the set E (see Section 4.3).

Event detection and differentiation techniques are deliberately undefined because they strictly depends on each specific application context. From the The Discrete Events Generator events are sent to the Time Series Generators. Each *Time Series Generators* module performs the following actions:

- for each channel CH_i fills a buffer B_i with $buffDim$ samples taken from the channel streams.
- fills an internal matrix M_{TS_i} (K rows and $buffDim$ columns) with the produced K buffers, where $i \in 0, .., N$.
- forwards the TS_i matrix to the MECS algorithm module.

Finally, before explaining the MECS algorithm module, let us first introduce a set of auxiliary data structures:

- CH_i: data streams used by the *Discrete Events Generator* that identifies events, and dispatches them correctly, i.e., CH_i contains events of class E_i.
- $Buffer\ B$: represents a single portion of data. During the execution of the algorithm, each channel stream is divided into buffers of size $buffDim$.
- M_{TS_i}: matrix of channels and samples. Namely each M_{TS_i} has K rows and $buffDim$ columns. The value of each element of the matrix determines the presence (value $\neq 0$) or absence (value $= 0$) of an event.
- $ECM(channel, ts)$ or Event Class Matrix: stores all the absolute positions of all the detected events.
- $Sync(pair, channel)$ and $Tot_{Sync}(pair, channel)$ are internal data structures used to store the values of $C_k(i|j)$ and $S_k(i, j)$.

The ECM, $Sync$ and Tot_{Sync} data structure are re-initialized every time a new buffer arrives.

Algorithm 2 MECS

```
for each new $Buffer\ B$ do
    Init()
    Compute()
    Finalize()
    $n_{buffers}$++
```

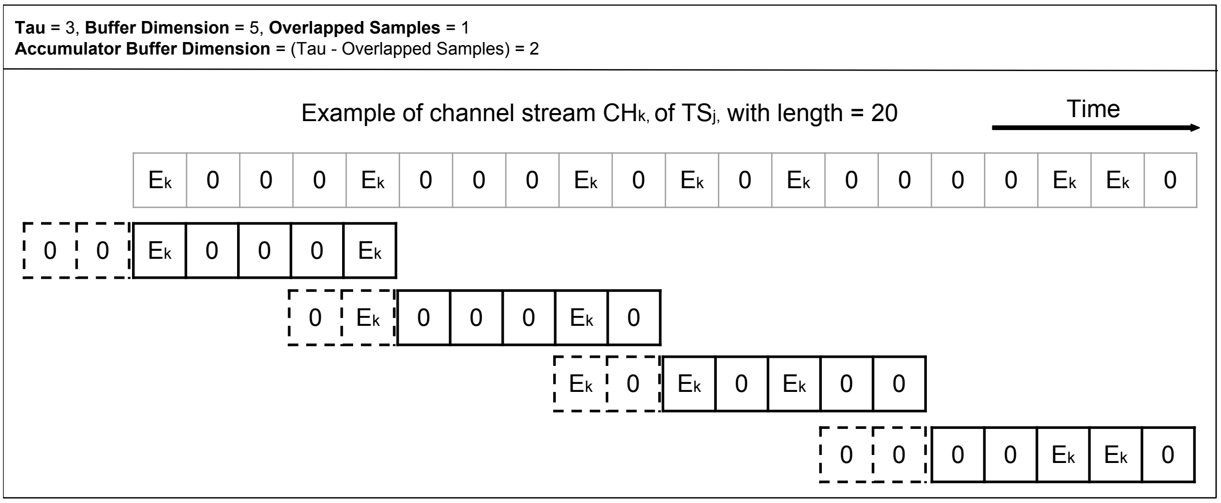

Tau = 3, Buffer Dimension = 5, Overlapped Samples = 1
Accumulator Buffer Dimension = (Tau - Overlapped Samples) = 2
Example of channel stream CHk, of TSj, with length = 20
Time
Ek 0 0 0 Ek 0 0 0 Ek 0 Ek 0 Ek 0 0 0 0 Ek Ek 0
0 0 Ek 0 0 0 Ek
0 Ek 0 0 0 Ek 0
Ek 0 Ek 0 Ek 0 0
0 0 0 0 Ek Ek 0

Figure 4.7: Use of the *Accumulator buffer* structure.

4.3.9.1 Initialization

In Algorithm 2, the body of the main routine of MECS is reported. The main routine runs at every received buffer. To correctly compute the synchronization vector, the algorithm stores in memory a certain number of samples from the last processed buffer at each execution cycle. Such samples are stored in a support data structure called *Accumulator buffer* which dimension is variable and determined before starting the algorithm execution.

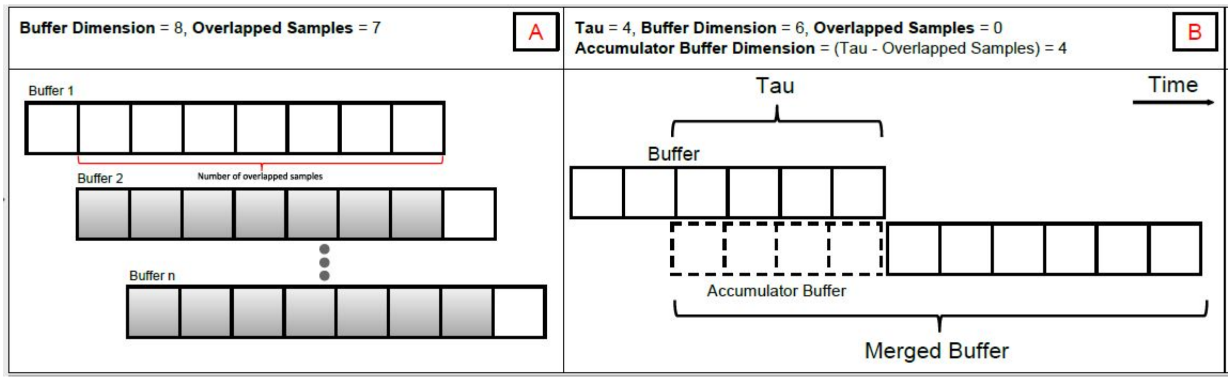
The correct dimension of *Accumulator buffer* is given by three different cases which depend on the value of Tau and on the number of overlapped samples $n_{overlapped}$. Representative cases are represented in Figures 4.8 and 4.9.

Algorithm 3 Init():

```
$MergBuff = createMergedBuffer(B)$
for $k \in \{CH_1, ..., CH_K\}$ do
    for $ts \in \{M_{TS_i}, ..., M_{TS_N}\}$ do
        for all $sample \neq 0 \in MergBuff$ do
            $abs_{Pos} = n_{buffers} * offset_{set} + rel_{Pos}$
            $ECM(k, ts).insert(abs_{Pos})$
```

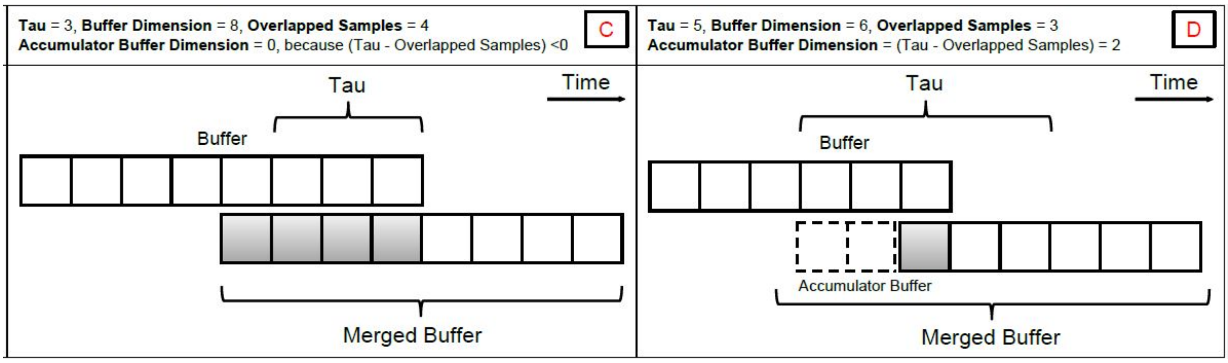
Successively, the content of the *Accumulator buffer* is attached to the next input buffer as explained in Figure 4.7 resulting a (*Merged Buffer*) of size $mergDim$.

The initialization function *Init()* fills the ECM (*Event Class Matrix*) with all the positions of all the events found in all the available channels. Absolute positions abs_{Pos} represents the occurrence timings of the events in the whole period of execution.



Buffer Dimension = 8, Overlapped Samples = 7
A
Buffer 1
Buffer 2
Number of overlapped samples
Buffer n
Tau = 4, Buffer Dimension = 6, Overlapped Samples = 0
Accumulator Buffer Dimension = (Tau - Overlapped Samples) = 4
B
Tau
Time
Buffer
Accumulator Buffer
Merged Buffer

Figure 4.8:
(A) Example of consecutive buffers with $buffDim$ = 8 and $n_{overlapped}$ = 7.
(B) $Tau = 4$, $n_{overlapped} = 0 \implies$ *Merged Buffer* is bigger than the original buffer.



Tau = 3, Buffer Dimension = 8, Overlapped Samples = 4
Accumulator Buffer Dimension = 0, because (Tau - Overlapped Samples) <0
C
Tau
Time
Buffer
Merged Buffer
Tau = 5, Buffer Dimension = 6, Overlapped Samples = 3
Accumulator Buffer Dimension = (Tau - Overlapped Samples) = 2
D
Tau
Time
Buffer
Accumulator Buffer
Merged Buffer

Figure 4.9:
(C) $Tau < n_{overlapped} \implies$ *Merged Buffer* has the same dimension of the original buffer.
(D) $Tau > n_{overlapped} \implies$ *Merged Buffer* is bigger than the original buffer.

To compute the absolute position abs_{Pos} of the detected events, the MECS algorithm uses the following quantities:

- the number of received buffers $n_{buffers}$
- the original dimension of the buffers $buffDim$
- the dimension of the accumulator buffer $accDim$
- the number of overlapping samples $n_{overlapped}$
- the relative position of an event $relPos$ (the position of the sample in the buffer B).

The absolute position of each event abs_{Pos} is computed by using the number of received buffers ($n_{buffers}$) and an $offset$ equals to the difference between $mergDim$ and $n_{overlapped}$, as explained in Algorithm 3.

4.3.9.2 Execution

Algorithm 4 Compute():

```
for $k \in \{CH_1, ..., CH_K\}$ do
    for all $couples$ of time series $< tsi,tsj >$
     with $tsi \neq tsj$ do
        for all $x \in ECM[k][tsi]$ do
            for all $y \in ECM[k][tsj]$ do
                $d$ = ComputeDist$(x, y)$
                $sync$ = $C_k(i|j)$
                $couple$ = $< tsi,tsj >$
                $Sync[couple, k]$ += $sync$
```

The *Compute()* routine calculates the distances between all the events found in all the possible pairs of time series, using the absolute positions stored in ECM, and saves the results in the $Sync(pair, channel)$ matrix.

When the Compute routine completes its execution, the $Sync$ matrix stores the total contribution to synchronization for each pair of time series $< TS_i, TS_j >$ and for each event class k.

The *Finalize()* routine performs the following steps:

- computes pairwise synchronization of the events of class k for each pair of time series $< TS_i, TS_j >$.
- computes the overall synchronization for each class k dividing pairwise synchronizations S_k by the set of all the 2-combinations of the considered time series.

Algorithm 5 Finalize():

```
for $k \in \{CH_1, ..., CH_K\}$ do
    for all $couples$ of time series $< tsi, tsj >$
     with $tsi \neq tsj$ do
        $couple$ = $< min(tsi, tsj), max(tsi, tsj) >$
        $Tot_{Sync}[couple, k]$ = $S_k(i, j)$
for $k \in \{CH_1, ..., CH_K\}$ do
    for all $couples$ of time series $< tsi, tsj >$
     with $tsi \neq tsj$ do
        $couple$ = $< min(tsi, tsj), max(tsi, tsj) >$
        $Q[k]$ += $Tot_{Sync}[couple, k]$ / Comb($N$, 2)
```

4.4 Application on synthetic data

In this section, we present two application examples of the MECS algorithm. We manually construct some signals and events sequences, and we provide them as input to the algorithm, reporting and commenting the corresponding output.

4.4.1 Inter-class synchronization between two time series

We suppose that the input signal under investigation (shown in the top plot in Figure 4.10) is the result of the composition of three simpler signals:

- a large sinusoidal signal with constant frequency (the second plot from top),
- a smaller sinusoidal signal with decreasing frequency (the third plot from top),
- and a noise component (the fourth plot from top).

The aim of this example is to demonstrate how the MECS algorithm can be exploited to find out the frequency of the main harmonic of a signal. The main harmonic is the second signal shown from the top of Figure 4.10. To do that, we propose to compute the synchronization of the input signal with a reference signal corresponding to the input signal main harmonic (see Figure 4.11). We first extract the peaks values of the three input signal components and the reference signal: the peaks are highlighted as dots in Figure 4.10 and 4.11. Afterwards, we create 2 time-series TS_1 and TS_2:

- TS_1 contains the events of classes E_1, E_2, E_3, E_4 corresponding to the peaks of, respectively, the input signal and three components;

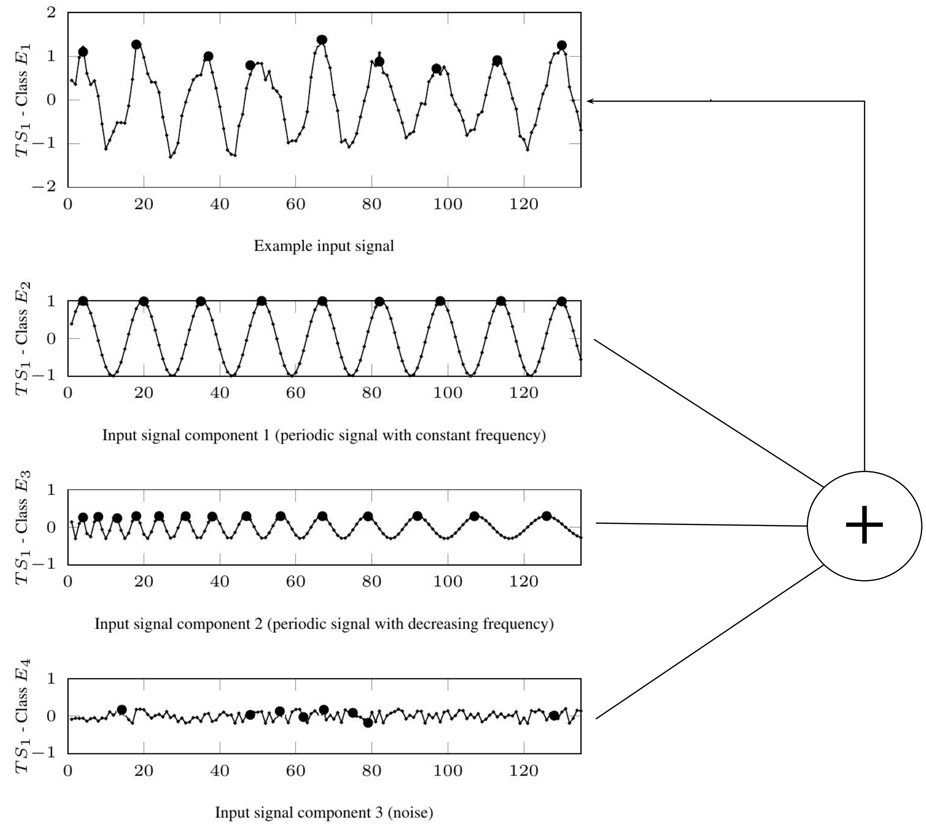


Figure 4.10: The input signal (highest plot) is the result of the composition of its three components: a sine wave with constant frequency and amplitude (second plot from top); a sine wave with constant amplitude and decreasing frequency (third plot from top); a noise signal (lowest plot). Dots highlight the peaks (automatically detected) of the four signals.

- TS_2 contains the events of class E_5 corresponding to the reference signal peaks.

TS_1 and TS_2 are provided as input to the MECS algorithm, which computes inter-class synchronization between all the pairs of classes in TS_1 and TS_2 (that is, for each $i \in 1, 2, 3, 4$ it computes Q between events of E_i and E_5) and provides as output the values reported in Figure 4.12. In this example, the MECS algorithm is set up with a value of $\tau = 5$ and a linear kernel (for details about kernels function see Section 4.3.7).

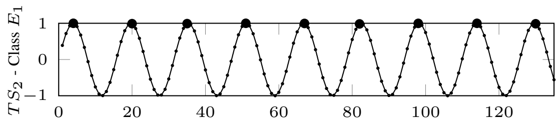
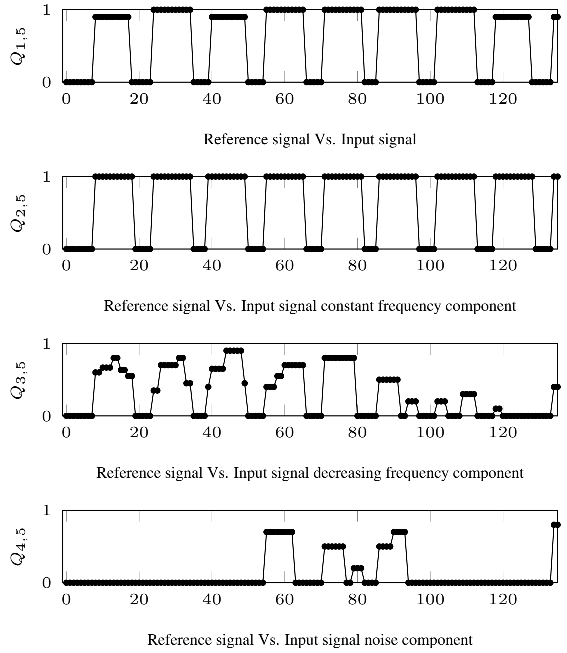


Figure 4.11: An example reference signal. The frequency and amplitude are identical to the first component of the input signal in Figure 4.10. Dots highlight the peaks of the signal.



Reference signal Vs. Input signal

Reference signal Vs. Input signal constant frequency component

Reference signal Vs. Input signal decreasing frequency component

Reference signal Vs. Input signal noise component

Figure 4.12: Synchronization results between the 3 signal components and the input signal reported in Figure 4.10 and the reference signal reported in Figure 4.11. The four synchronization values are computed at the same time by applying the MECS algorithm.

As illustrated in Figure 4.12, the synchronization between the input signal and the reference signal (the first plot from top) exhibits a rhythmical (quasi-periodic) pattern with a high constant amplitude. The same happens for the first component, which is identical to the reference signal.

Conversely, the second and third components of the input signal, that is, the signal with a variable frequency and the noise component, do not exhibit the same kind of pattern.

4.4.2 Macro events synchronization between two time series

In this second example we apply MECS to two input time series TS_1 and TS_2 consisting of events belonging to three classes (i.e., E_1, E_2, E_3) (see Figure 4.13). Such events are represented within the time series by the corresponding positive integer numbers $1, 2, 3$, while, when no event is triggered, a value of zero is stored. For example, TS_2 in Figure 4.13 starts with an event of class E_1, then no events are triggered for the second sample of the time series, then an event of class E_2 is triggered, and so on.

We set up the MECS algorithm to detect synchronization of macro events (sequences) consisting of an event of class E_1, followed by an event of class E_2, followed by an event of class E_3 (the concept of *sequence* has been introduced in Section 4.3.6). We chose a value of τ is 20 and run the algorithm 3 times to compare different kernels: uniform, linear and Gaussian. The output of the algorithm, illustrated in the 3 lower plots, is the amount of synchronization between $S = \{E_1, E_2, E_3\}$ events in the two input time series. The figure shows that an uniform kernel (the third plot from top) provides only two possible outputs, 0 or 1, that depend on the presence or absence of event sequences within a distance of τ samples. Conversely, the linear kernel (the fourth plot from top) provide non-discrete values, depending on the distance, and the Gaussian depending on the Gaussian mean parameter.

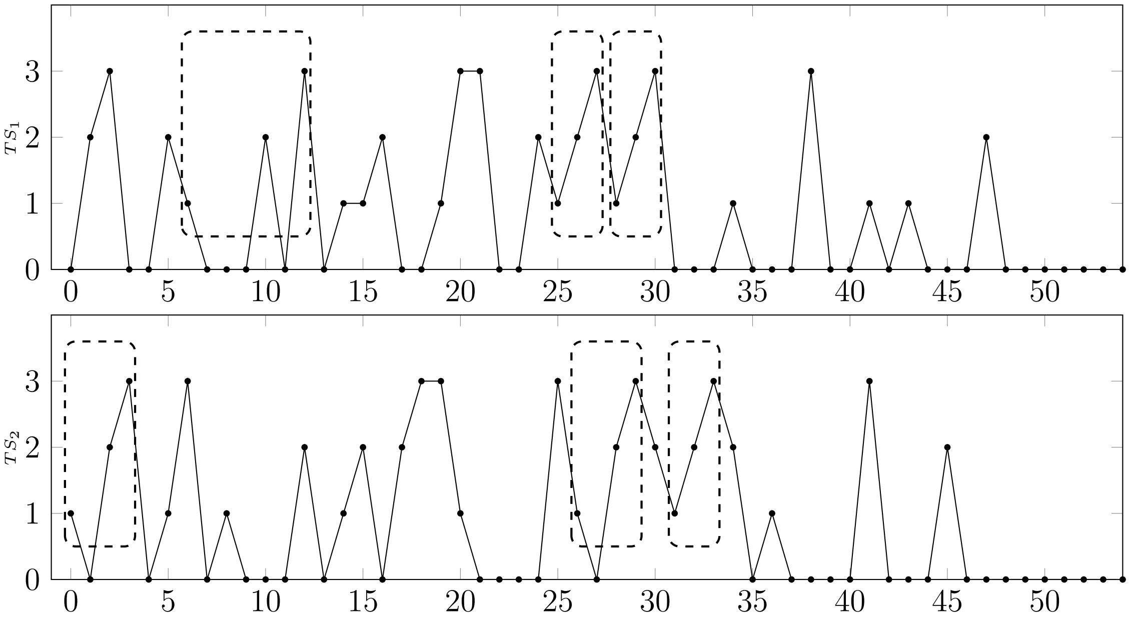


Figure 4.13: MECS applied to two time series. Macro events are triggered when the following sequence of events $S = \{E_1, E_2, E_3\}$ is detected (see Section 4.3.6). The three events must occur in the specified order (in this case any amount of samples with no events can appear between them, i.e., $IEI = \infty$). Detected macro events are highlighted in the figure with dashed boxes enclosing each single event that contributes to the final sequence. For example, the first macro event found in TS_1 (in the top left corner of the figure) consists of the event of class E_1 followed by three samples with no events, then the event of class E_2, another empty sample and finally an event of class E_3.

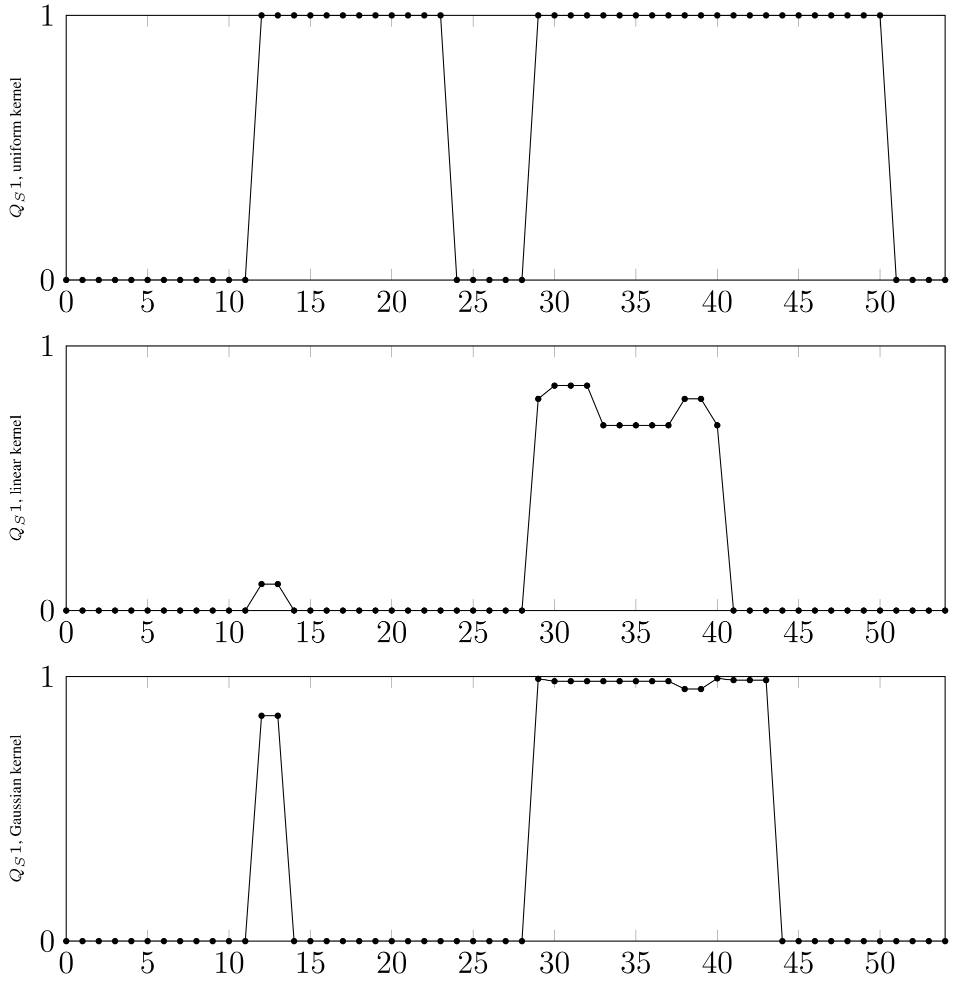
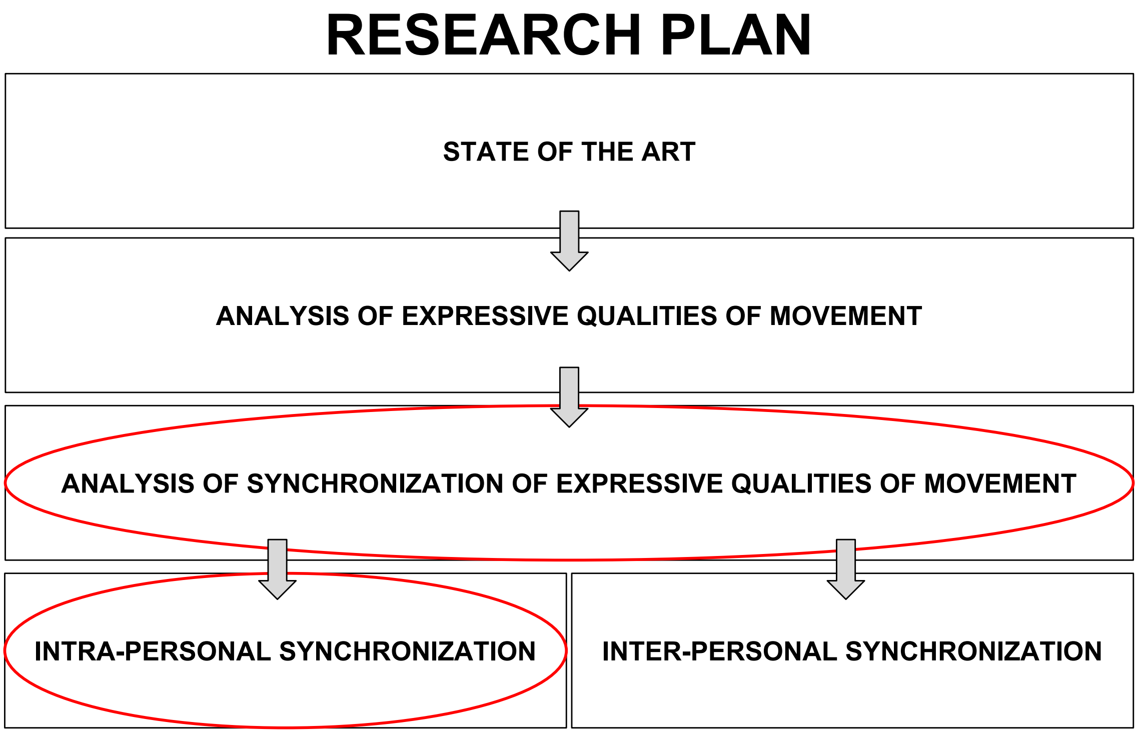


Figure 4.14: MECS algorithm output, between TS_1 and TS_2, with $\tau = 20$ samples using respectively, an uniform, a linear and a Gaussian kernel.

Chapter 5

Intra-personal synchronization


RESEARCH PLAN
STATE OF THE ART
ANALYSIS OF EXPRESSIVE QUALITIES OF MOVEMENT
ANALYSIS OF SYNCHRONIZATION OF EXPRESSIVE QUALITIES OF MOVEMENT
INTRA-PERSONAL SYNCHRONIZATION
INTER-PERSONAL SYNCHRONIZATION

Contents

In this chapter, we present three case studies where the analysis of synchronization is used to answer different research questions. Presented studies face problems related to the **intra-personal** synchronization analysis. Complementary studies on **inter-personal** synchronization are presented in Chapter 6, which addresses musical entrainment thematics.

In the first study, published in [APM^{+}16], we show how, by the analysis of synchronization of physical, low level signals (e.g., limbs' accelerations), extracted from dance recordings, it is possible to automatically detect which qualities of movement are expressed. For this first study, we used the algorithm presented in Section 2.2.8.

The second study focuses on the investigation of expressive qualities of movement in sport, specifically in karate. In the proposed approach, the analysis of synchronization is complementary to another technique: the *Multi Scale Analysis*, used for noise-filtering and for detecting of most significant time instants in the flow of movements. The goal of this work is to quantify the quality of kata executions. We built a system for the automated measure of the degree of *cleanness* of karate movements focusing on the starts and stop of the movements. This work has been published in [ADGCP17]. The followed approach derives from interviews with experts in karate.

The third and last study is an application of the MECS algorithm on a real case. In particular, we automatically distinguish two respiration phases (inspiration and expiration), that are related to different expressive qualities of the movement), through the analysis of synchronization of multimodal data (movement and breath). MECS allows us to evaluate not only these different data at the same time, but also to measure the synchronization of temporal sequence of events (see Chapter 4.3, Section 4.3.6), necessary to model correctly this problem and verify the hypothesis.

5.1 Study 1: Analysis of synchronization to distinguish between movements qualities

We present an approach to dance movement analysis in which a set of dance performances are analyzed and classified according the analysis of synchronization of body joint's velocity in a dance performance. Our goal, in particular, is to demonstrate that the level of synchronization of the joints composing the kinematic chains of a person can help to automatically distinguish movements displaying following expressive qualities: *Fluidity*, *Impulsivity* and *Rigidity*.

- **Fluidity** the definition of Fluidity used in this experiment is described in Chapter 3, Section 3.2.2.
- **Impulsivity**: an *impulsive movement* is characterized by a sudden and non-predictable change of velocity, which is usually produced without exhibiting a preparation phase [NMVC15]. According to Bishko [Bis91], in dance, an impulsive movement can be characterized as *a movement of increasing intensity ending with an accent is considered impactive*. In Physics, the impulse is defined as the variation of an object's momentum in time. The momentum depends on the object's mass and velocity. So basically an impulse corresponds to a high variation of the object's speed or, in other words, to high object's acceleration/deceleration. A similar concept can be found in psychological studies, for example in [DB93]:"actions that are poorly conceived, prematurely expressed". An impulse can be considered as a movement with high acceleration performed with no premeditation. Examples of impulsive movements are: avoidance movements (e.g., when hearing a sudden and unexpected noise) and movements made to recover from a loss of balance.
- **Rigidity**: a movement quality strictly linked with the internal emotional state of a user. Being rigid by performing a movement can be a consequence of stress, fear or tension. For example, a stressed person tends to increase the tension in her muscles, producing rigid movements [CDT15]. A better understanding and automatic detection of rigidity

could greatly improve the adaptability of Human-Computer interfaces. In [SMBC$^+$14], rigidity is considered as one of the motor cues to recognize emotions and mental states of children characterized by Autism Spectrum Conditions and it is measured as of the relative movement of different parts of the body. Moreover, Rigidity is one of the few movement qualities that are addressed in credibility assessment in the information systems area. In [TEBN14] authors developed automated interviewing systems based on kinetic rigidity detection, in order to detect the amount of non-credible information during an interview.

5.1.1 Experimental set-up

We recorded short performances of professional dancers who were asked to exhibit full-body movements with one among the considered expressive qualities. Two professional female dancers participated in the recording sessions. At the beginning of each session, the dancers were given definitions of the expressive qualities. For each expressive quality, the instruction we provided to the dancers were the following:

1. to perform several repetitions of predefined movements (e.g., avoiding an imaginary and sudden danger, throwing an object with a wave-like arm movement) by focusing on the expressive quality;

2. to perform an improvised choreography containing movements that, in the opinion of the dancer, better expressed the expressive quality;

For the recordings, we used a Qualisys motion capture system sampling dancers' movement at 100 Hz and synchronized with a video recording system (1280x720, 50fps). We placed 6 single markers and 11 rigid bodies plates on the dancer body, as illustrated in Figure 5.3. The resulting data consisted of the 3D positions of 19 markers: 6 corresponding to the single markers plus 11 corresponding to the rigid bodies' barycentre and 3 corresponding to all the markers attached rigid body placed on the dancer's head (see Figure 5.1).

Two experts in the domain of expressive movement analysis segmented the recorded data. They were instructed to select segments that exhibited each expressive quality in a regular way. Feature segments were not validated in a formal manner: the identification of the most representative segments is the result of a discussion afterwards with the domain experts. Thus, segmentation was based on the observer's perception of the dancer's expressive quality, and not on the dancers' expressive intention. We obtained a dataset of 60 segments: it contains 10 highly impulsive, 10 highly fluid, and 10 highly rigid segments for each dancer. The mean segments duration is 5.85 seconds ($SD = 3.76$) and the total duration is 5 minutes 51 seconds.

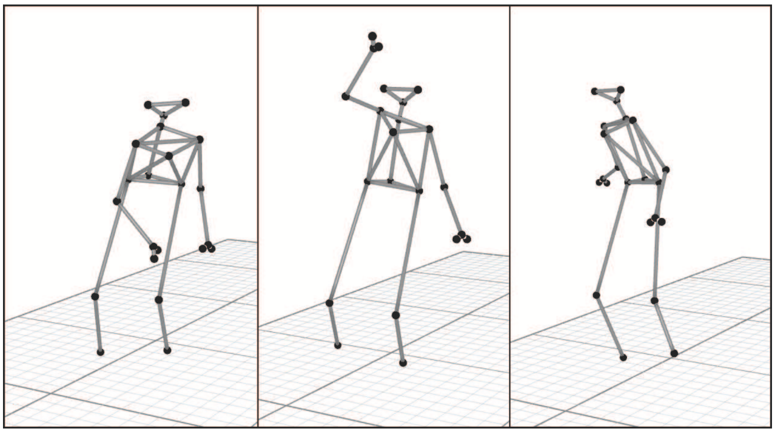


Figure 5.1: The motion captured 3D skeleton of a dancer.

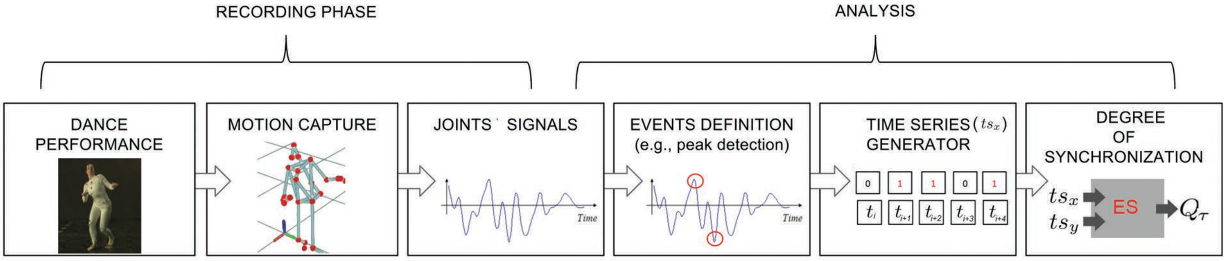


Figure 5.2: A schematic view of the experiment dataflow: from the dance performance (on the left) to the computed degree of synchronization (on the right).

5.1.2 Data analysis and discussion

In this first study, event synchronization analysis has been performed in its original version (see Section 2.2.8) on the segments dataset as shown in 5.2.

Events were defined as abrupt changes of limbs velocity during the performances. We extracted events by detecting peaks of the velocity module (velocity is computed as the derivative of position, given joint's 3D coordinates frame-by-frame).

For each segment S of N frames in the dataset, we selected three joints of the right arm J_w, J_e and J_s (wrist, elbow, and shoulder, respectively) and we extracted the corresponding velocity modules v_w, v_e and v_s.

Then, we applied a supervised event detection algorithm (i.e., parameterized peak- detector) on velocity signals v_w, v_e and v_s in S to extract significant events. We obtained three binary time-series ts_{vw} , ts_{ve} and ts_{vs} containing all the events occurrences coupled with the exact time of the occurrence (e.g., ti is the time of the i-th event that occurred in the time series).

Table 5.1: Descriptive Statistics.

	Fluidity			Rigidity		
	Dancer 1	Dancer2	Total	Dancer1	Dancer2	Total
	0.253 (0.145)	0.325 (0.153)	0.289 (0.150)	0.355 (0.255)	0.413 (0.163)	0.384 (0.211)
	0.277 (0.146)	0.319 (0.083)	0.298 (0.117)	0.595 (0.188)	0.347 (0.117)	0.471 (0.199)
	0.169 (0.075)	0.265 (0.098)	0.217 (0.098)	0.354 (0.276)	0.380 (0.276)	0.367 (0.269)
Total	0.233 (0.131)	0.303 (0.115)	0.268 (0.127)	0.435 (0.261)	0.380 (0.192)	0.407 (0.229)

	Impulsivity			Total		
	Dancer1	Dancer2	Total	Dancer1	Dancer2	Total
	0.532 (0.171)	0.557 (0.275)	0.545 (0.223)	0.380 (0.223)	0.432 (0.220)	0.410 (0.221)
	0.730 (0.208)	0.572 (0.256)	0.651 (0.241)	0.534 (0.261)	0.413 (0.200)	0.473 (0.237)
	0.530 (0.272)	0.434 (0.229)	0.482 (0.250)	0.351 (0.266)	0.359 (0.220)	0.355 (0.242)
Total	0.598 (0.233)	0.521 (0.253)	0.559 (0.244)	0.422 (0.261)	0.401 (0.213)	0.411 (0.238)

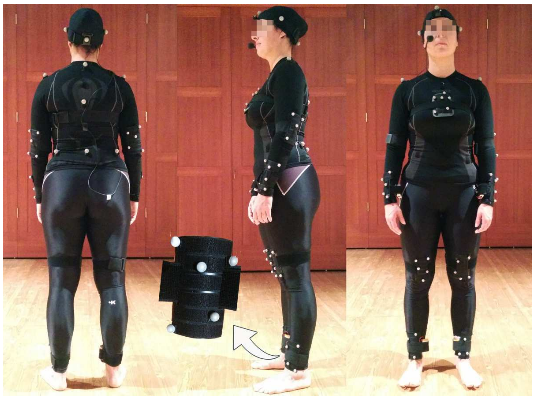


Figure 5.3: Markers and rigid bodies placed on the dancer's body. A rigid body is a rigid plate on which several markers are attached. This configuration allows us to extract not only the position but also the rotation of body joints. In the figure, a rigid body is highlighted.

We computed the average Q_τ various for three pairs of joints: elbow-shoulder, elbow-wrist and wrist-shoulder (corresponding to the following pairs of time-series: (ts_{vw}, ts_{ve}),(ts_{ve}, ts_{vs}), (ts_{vw}, ts_{vs})), setting-up a value of $Tau = 20$ frames (corresponding to 20 ms at 100 Hz). Analysis has been performed for all the 60 segments. Detailed results are presented in Table 5.1 and Figure 5.4.

A two-way MANOVA revealed a significant multivariate main effect for Quality, Wilks' $\lambda = .540$, $F(6, 104) = 6.248$, $p < .001$, partial $\eta^2 = .265$ (power to detect the effect .998), and for Dancer, Wilks' $\lambda = .812$, $F(3, 52) = 4.011$, $p < .05$, partial $\eta^2 = .188$, (power to detect the effect .810).

The interaction effect between Quality and Dancer on the combined dependent variables was not observed, $F(6, 104) = 1.560$, $p = .166$; Wilks' $\lambda = .842$, partial $\eta^2 = .083$. Given the significance of the overall test for Quality, the univariate main effects were examined. A significant univariate main effect of Quality for elbow-shoulder pair was observed, $F(2, 54) = 8.325$, $p < .01$, for elbow-wrist pair, $F(2, 54) = 20.125$, $p < .001$, and shoulder-wrist pair $F(2, 54) = 7.229$,

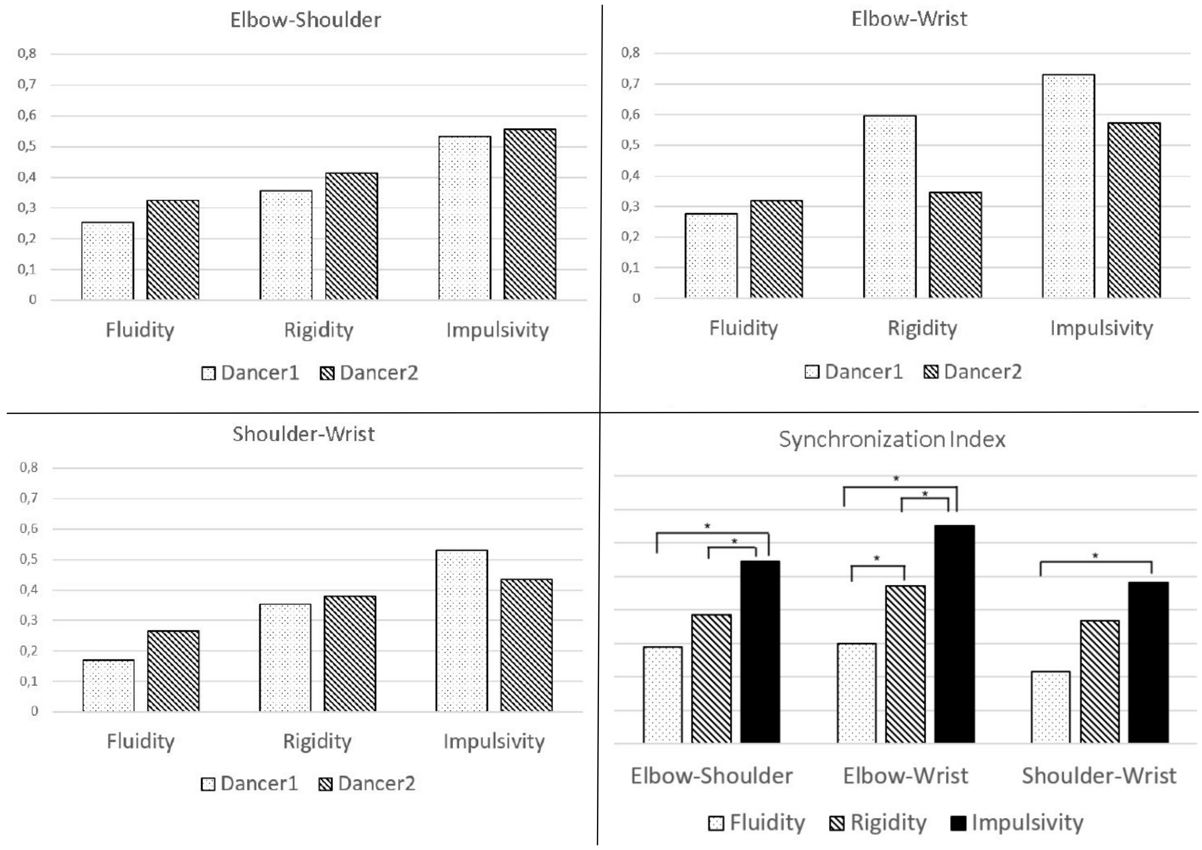


Figure 5.4: The mean values of Q_τ for three pairs of joints. In the bottom right graph are shown the mean values of Q_τ for each quality. Significant differences are marked with a *.

$p < .01$.

Additionally a significant univariate effect of Dancer for elbow-wrist pair was observed ($F(2, 54) = 7.140$, $p < .05$) but not for the remaining two pairs (elbow-shoulder: $F(2, 54) = .995$, $p = .323$ and shoulder-wrist: $F(2, 54) = .022$, $p = .882$) The Levene's statistics for the three dependent variables are all non-significant.

For the elbow-shoulder pair (see Figure 5.4) post hoc comparisons using the LSD test with Bonferroni correction indicated that the Fluidity synchronization indexes were significantly lower than Impulsivity ones ($p < .01$). Additionally synchronization indexes of Rigidity were significantly lower than Impulsivity ones ($p < .05$). There was no significant difference between the synchronization indexes of Fluidity and Rigidity ($p = .328$).

For the elbow-wrist pair (see Figure 5.4) post hoc comparisons using the LSD test with Bonferroni correction indicated that the synchronization indexes of Fluidity were significantly lower compared to Rigidity ones ($p < .01$), and Impulsivity ones ($p < .001$). Additionally the synchronization indexes of Rigidity scores were significantly lower than Impulsivity ones ($p < .01$).

For the shoulder-wrist pair (see Figure 5.4) post hoc comparisons using the LSD test with Bonferroni correction indicated that only the synchronization indexes of Fluidity were significantly lower than Impulsivity ones ($p < .01$). There was no significant difference neither between the synchronization indexes of Fluidity and Rigidity ($p = .108$) nor between Rigidity and Impulsivity ($p = .320$).

Results indicates that it is possible to distinguish between movements performed with different qualities by exploiting the arm's joints velocity synchronization analysis.

Figure 5.4 shows that impulsive movements are the most synchronized, the fluid ones show the lowest values of synchronization, and the rigid ones lay in between. Although results show differences between the three qualities the differences between dancers were also significant for one pair of joints (elbow-wrist).

The significant differences in synchronization between all thee considered expressive qualities occur in the most external joints of the body (elbow-wrist) (see Figure 5.4). Less strong differences in synchronization occur between pairs: elbow-shoulder and wrist-shoulder. It might be due to the fact that not all the movements have to include the shoulder motion. Indeed the average synchronization indexes Q^{τ} for two joint pairs that include shoulder are lower compared to elbow-wrist correspondences.

5.1.3 Conclusion

The presented study showed how intra-personal synchronization might contribute to distinguish movements performed with different expressive qualities: Fluidity, Impulsivity and Rigidity. In detail, we performed an event synchronization analysis to the right arm joint's velocity and we found out that different synchronization patterns characterize each considered quality. This work can be extended by:

- investigating how synchronization analysis can be used to distinguish a larger set of movement qualities.
- enhancing the event detection step by taking into account the characteristics of the input signal (e.g., for a slowly varying signal a more "sensible" analysis will be needed in order to detect peaks).

5.2 Study 2: Synchronization as a quality measure for karate performances

The second experiment concerns the study of movement synchronization in sport activities. In sport, systems capable to perform automated analysis of movement qualities can be useful for more efficient training, and for evaluating the effectiveness of a specific physical gesture. In this sense, we address the problem of evaluating the overall quality of martial arts performances starting from motion capture recordings.

More specifically, we aim to quantify how karate performances at different levels of expertise are perceived by an external (not necessarily expert) observer. To this aim, we hypothesize synchronization of movements between specific limbs as a main feature capable to contribute to explain the differences between the same kata[1] expressed by different athletes with different levels of expertise.

The motivations to ground our measure of quality on synchronization is rooted on well-known common assumptions in sport practice: an experienced athlete makes a more "neat" execution of a kata, in which the starting and ending moments of each movement (e.g. punches, kicks) are performed by an expert with less or no fluctuations or ripples between joints movements, with respect to a less skilled athlete.

That is, in the case of an expert athlete, each movement exhibits a high synchronization between the involved limbs (i.e., the arms or the legs). This also corresponds to the concept of soft entrainment in music performance, a similar case of motor task characterized by high skills and expressiveness. Stability is obtained by a high level control of the movement, emerging from a strong synchrony between limbs, in particular at the starting and ending moments of single movements, such as punches, strikes and kicks.

We analyze a set of performances (**katas**) to distinguish and classify each on a measure of a definition of "overall quality". Several potential problems arise while analyzing this kind of data: although preprocessing to clean the raw data has been done, performing sessions are of different length, due both to the different speed at which the athletes perform and to the fact that recordings do not start and stop exactly at the beginning and end of the kata. Further, motion capture data can be noisy, not just because of the acquisition process, but also because of noise intrinsic in the biological movement itself. These issues are tackled by using a multi-scale analysis to get rid of the noise and extract relevant information present in the input signal, which is then fed to an event synchronization algorithm. Our experiments confirm that the analysis of synchronization can be used to characterize the quality of performance of athletes from different levels of skill.

[1] in karate a *kata* is a ordered sequence of movements

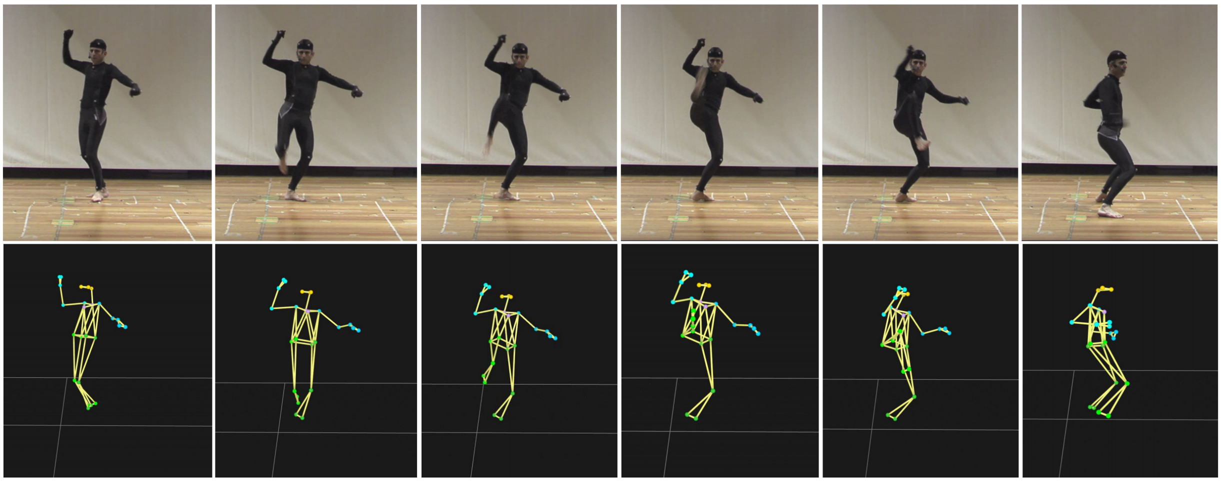


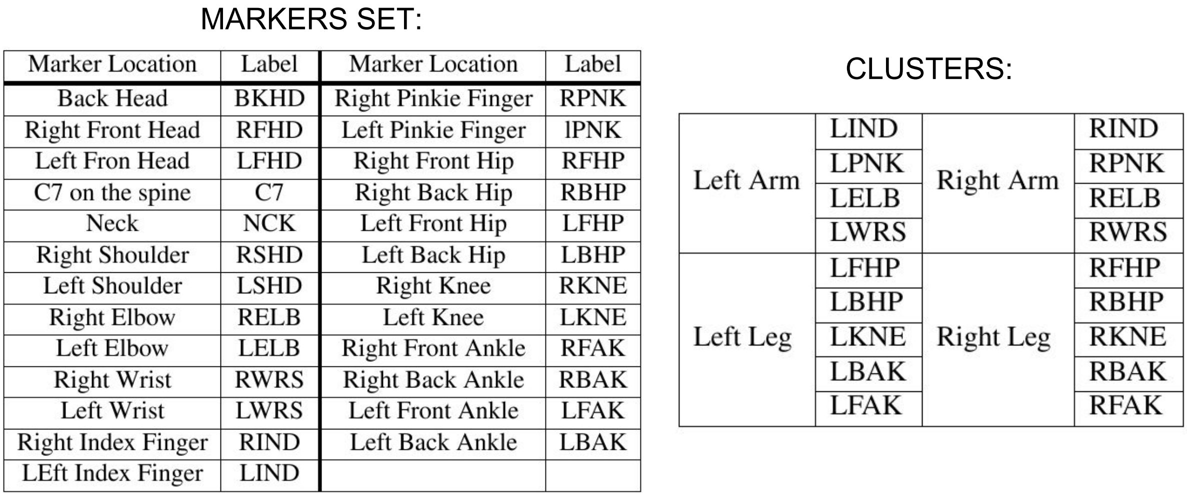
Figure 5.5: Example of MoCap data of an athlete performing a kata.

5.2.1 Dataset and participants

Analogously for the previous study, recordings were acquired using a Qualisys motion capture system, synchronized with a video recording system (1280x720, 50fps) but with an higher frame rate: 250Hz (such a high sampling rate was necessary to capture the very fast movements of karate experts). This dataset recorded is presented in [KCV$^+$15].

Post-processing has been applied to get a labeled dataset, with a skeleton model, in which noise from "ghost" or jitter markers has been cleaned out. The recordings were realized by 5 participants with different levels of experience and skill in Karate (when assessing their experience on a 1 to 5 scale, three of them were placed at 3, the others two at 5). The recorded trials have been watched by both the participants and their teacher to make sure that the performances where adequate with respect to the assessed level. The participants were asked to perform two different katas, namely: Heian Yondan and Bassai Dai. However, our analysis is independent from the particular kata and makes no assumption based on the kind of movement the kata requires.

The systems tracked the position of 25 markers placed on the performer. Our analysis is based on 3D motion capture data. The various participants were let free to perform at their own speed and rhythm, and the output recordings were not cut to have the same starting and ending moment, nor warped to have the same length; therefore there are significant differences in lengths for the different recordings: recordings of kata Bassai Dai range from 71s to 115s; recordings of Heian Yondan range from 50s to 97s. Our method, though, does not make any assumptions on the length of input and it is independent from it, since the scale-space and persistence analysis extract relevant events independently to when they occur.



MARKERS SET:

Marker Location	Label	Marker Location	Label
Back Head	BKHD	Right Pinkie Finger	RPNK
Right Front Head	RFHD	Left Pinkie Finger	lPNK
Left Fron Head	LFHD	Right Front Hip	RFHP
C7 on the spine	C7	Right Back Hip	RBHP
Neck	NCK	Left Front Hip	LFHP
Right Shoulder	RSHD	Left Back Hip	LBHP
Left Shoulder	LSHD	Right Knee	RKNE
Right Elbow	RELB	Left Knee	LKNE
Left Elbow	LELB	Right Front Ankle	RFAK
Right Wrist	RWRS	Right Back Ankle	RBAK
Left Wrist	LWRS	Left Front Ankle	LFAK
Right Index Finger	RIND	Left Back Ankle	LBAK
LEft Index Finger	LIND		

CLUSTERS:

Left Arm	LIND	Right Arm	RIND
	LPNK		RPNK
	LELB		RELB
	LWRS		RWRS
Left Leg	LFHP	Right Leg	RFHP
	LBHP		RBHP
	LKNE		RKNE
	LBAK		RBAK
	LFAK		RFAK

Figure 5.7: On the left the list of markers traced by the motion capture system. On the right, the four clusters defined by computing the barycentre of the grouped markers.

5.2.2 Marker set and 3D skeleton

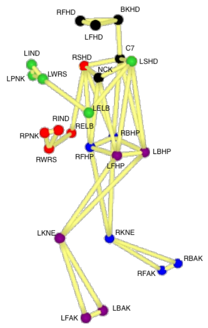


Figure 5.6: The Motion Capture 3D Model. In red, green, purple and blue the groups of markers used to define the clusters.

Motion tracking has been realised with 25 makers placed on the body (as shown in Figure 5.6), each providing a 3D trajectory of samples at 250hz, where every sample consists of a triple (x,y, z) of coordinates representing the position of that marker at each time instant, in a calibrated reference system. Since our analysis is focused on the movements of the limbs, we do not use every marker of the model but we rather extract four clusters in order to exploit redundant information and obtain a reduced yet more stable representation

Table 5.7 shows all the markers in the original dataset and the defined clusters. The simplified model thus consists of four trajectories, each subsuming the movement of one limb through time.

5.2.3 Techniques for the analysis of karate performances quality

As outlined in the previous sections, we aim at analyzing the level of synchronization between limbs while executing a session of *kata*. Given that, the feature that was found to be more representative for our analysis is the intensity of limbs' acceleration (and deceleration) that allows us to distinguish the initial and final phases of the basic movements, such as punches, strikes, and kicks. Based on the simplified model described in the previous section (see Table 5.7), we have extracted the clusters' barycentre acceleration for each trial, frame by frame, obtaining four different time series, associated with (left and right) arm and leg, respectively.

5.2.3.1 Multi-Scale Analysis

Peaks of acceleration and deceleration characterize instants of time that are relevant for our analysis, but they also appear along trajectories, because of noise, uncertainty and ripple in the movement. Since data are rather noisy, isolating relevant peaks from unimportant ones is a challenging task. We tackle this problem by performing a multi-scale analysis of our time series, which allows us to rank peaks of acceleration by relevance. In order to build our multi-scale model we combine two independent techniques, one working on the domain of frequency (linear scale-space) and the other working on the domain of amplitude (topological persistence).

5.2.3.2 Linear scale-space

Linear scale-space is a formal theory for multi-scale signal representation [Lin94], which in based on the regressive smoothing of an input signal f with Gaussian filters of increasing variance. Progressive filtering makes the signal become smoother and smoother, thus progressively reducing the number of critical points of the signal. This process induces a ranking of importance for such points, based on their resistance to filtering: each critical point can be labeled with the variance of the filter that makes it disappear; the larger the variance, the more relevant the critical point.

Despite being continuous in its formulation, scale-space is most often implemented in a discrete way; in our work, we make use of the virtually continuous extension of scale-space described in [RP13], which supports a robust tracking of critical points through scales, inducing a more reliable and precise ranking. More precisely, each critical point c_k of the input signal is associated to a real number l_k, representing an approximation of how much it will survive a continuous smoothing process, since the method computes a linear interpolation between two steps of the discrete scale-space. We convert the result of our analysis in a time series S consisting of impulse signals, defined as follows:

$$Si = \begin{cases} l_k & \text{if } f(i) \text{ is a critical point } c_k \\ 0 & \text{otherwise} \end{cases} \tag{5.1}$$

5.2.3.3 Topological persistence

Topological persistence is a concept related to Morse theory [ELZ00], which provides a characterization of a function in terms of topology of its level sets. For the simple case of a one-dimensional signal, persistence can be easily explained through a flooding process that starts at the absolute minimum of the function and progressively fills the basins that have their lowest point at the local minima; this process forms pairs between each minimum m and a maximum M at which the basin related with m is merged with a "more relevant" basin, i.e., a basin related to a minimum that is lower than m.

The flooding process produces pairs progressively until all maxima and minima have been paired; the difference in amplitude between the maximum and the minimum within each pair defines their persistence. Qualitatively speaking, persistence is a measure of how much a critical point resists to the flood, which can be regarded as a filtering process in the amplitude domain. Each critical point c_k of the input signal in this case is associated to a real number p_k, which is the computed persistence for that critical point. As in the previous case, we encode the result of our analysis into a time series P, defined as follows:

$$Pi = \begin{cases} p_k & \text{if } f(i) \text{ is a critical point } c_k \\ 0 & \text{otherwise} \end{cases} \tag{5.2}$$

Note that S and P are congruent impulse signals that have non-null values only at critical points of the input signal f.

5.2.3.4 Combined measure of relevance

We rank critical points of the input signal f by giving high importance to points that have both long life in the scale-space and high persistence. After computing time series S and P for each limbs' acceleration, we normalize each of them in range $[0, 1]$ range and we sum them to obtain a measure of importance of the critical points in the input signal. We call it the *multi-scale index, or MSI for short.*

An empirical analysis of the computed signal over the various trials shows that this generates a distribution of values in which the vast majority of critical points are associated to low values (< 0.1) while just a small fraction get higher values (> 0.25). This reflects the fact that most

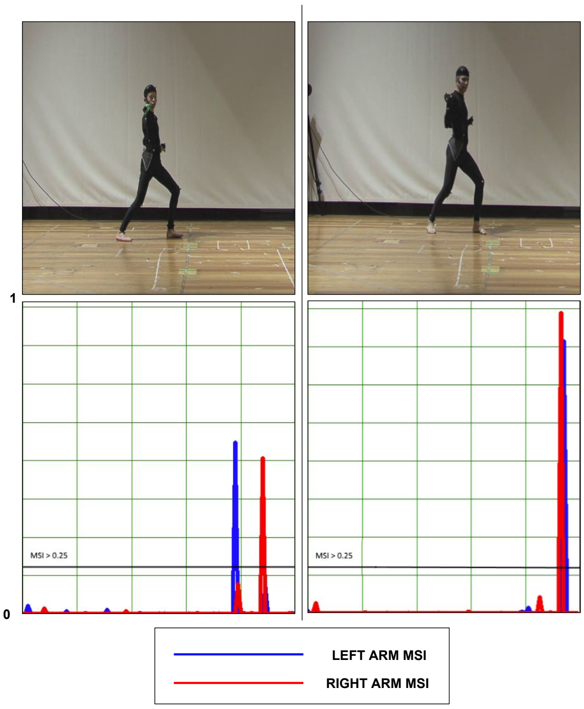


Figure 5.8: Difference in synchronization between athletes: the same movement performed by a less skilled performer (left) and a more skilled one (right). We see that events related to the arms of the more skilled athlete are much more synchronized.

critical points of the acceleration are related to noise in the movements or in the acquisition process, while just a few critical points identify relevant events in the sequence of movements.

5.2.3.5 Event Synchronization Analysis

We consider the MSI series related to the different limbs to measure their level of synchronization. We first filter the multi-scale index by thresholding, in order to consider just important points with high values ($MSI > 0.25$). Again, as for Experiment I, to evaluate the synchronization degree, we apply the Event Synchronization analysis on the filtered MSI time series (see Section 2.2.8). The output of multi-scale analysis consists of the following time-series:

$$ts_i \quad \text{with} \quad i \in leftArm,\ rightArm,\ leftLeg,\ rightLeg$$

containing for each frame the value of the MSI index. Such time series are used to estimate the synchronization on limbs. The time-series have been paired as follows:

$$(ts_{leftArm}, ts_{rightArm}) \quad (ts_{leftLeg}, ts_{rightLeg})$$

For each limb time series, events are generated every time the acceleration value showed a $MSI > 0.25$. We set-up a value of time interval τ of 12 frames that in our context is a reasonable time for abrupt changes in limbs' acceleration, to be perceived by an observer. For each trial, we computed the average degree of synchronization Q_τ between the two arms and between the two legs. Q_τ is a value in the range $[0, 1]$, where 0 means a complete lack of synchronization, while 1 means that all the detected events of the two limbs are synchronous within threshold τ.

5.2.4 Statistical Summary

As Tables 5.2 and 5.3 show, the synchronization indexes computed on high skilled performer's recordings are generally much higher than the ones computed on the less skilled athlete's performances.We thus proceed to test the significance of our sample data, by hypothesizing that synchronization indexes for high and low skilled performers are statistically significantly different. Statistical analyses of the data was performed between two groups by Excel Data Analysis ToolPak.

5.2.4.1 Arms

Since data did not deviate from a normal distribution (Shapiro-Wilk test, (low skilled, $p > 0.1$); (high skilled, $p > 0.3$)), an independent two tails t-test was conducted to examine whether there was a significant difference between the synchronization index of the arms of high skilled performers (M = 0.734, SD = 0.01) in relation with low skilled performers (M = 0.251, SD = 0.05) The t-test revealed a statistically significant difference in the arm synchronization index between the two groups ($t = -6.14, p < .001$).

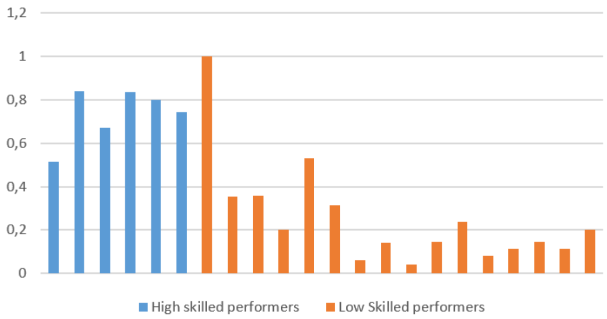


Table 5.2: Synchronization of arms for the different trials.

5.2.4.2 Legs

Since the data deviates from a normal distribution (Shapiro-Wilk test), a Mann-Whitney test for two independent samples was conducted to examine whether there was a significant difference between the synchronization index of the legs of high skilled performers (M = 0.570, SD = 0.05) in relation with low skilled performers (M = 0.201, SD = 0.028) The Mann-Whitney test revealed a statistically significant difference in the legs synchronization index between the two groups ($p < .001$).

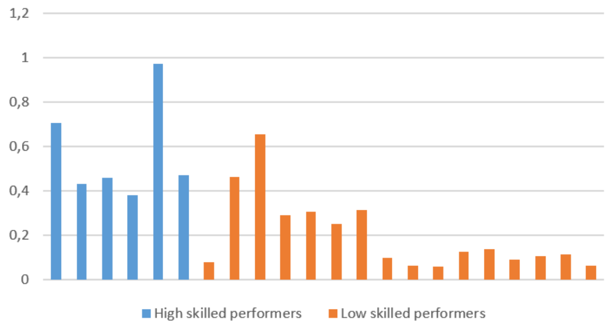


Table 5.3: Synchronization of legs for the different trials.

5.2.5 Conclusion

The results presented show statistical significance in discriminating between differently skilled performers; this is more evident on the arms than on the legs, but it can be explained by looking at the two kata performed: the harder one, in which we expect to see a greater difference in movement quality, is *Bassai Dai*, which focuses mainly on arm movements, while *Heian Yondan*, which involves more leg movement, is much easier and we can expect the difference between different performers to be smaller.

Although data comes from a small population, statistical tests have been used which are designed to work with small samples size; anyway, more data can be gathered by either recording more trials or by extracting more information from the synchronization process (e.g. by looking at how the synchronization factor changes when we change τ).

In second study, we presented a method to estimate a karate performance quality by studying how much the limbs are synchronized during relevant motion phases. Event synchronization analysis has been applied to a set of events (starts and stops of relevant motions) to provide a qualitative measure of the karate performance, in terms of "cleanness" of motion. Results are showing that analysis of synchronization can be used to discriminate between athletes at different levels of skill. For instance, katas usually contain a many sudden and fast movements, and intuition suggests that experienced athletes are better at having a cleaner transition when starting the movement and, most important, when ending it. That is, good performers tend to have just one strong deceleration from high velocity to a still position, while less skilled performers could have trouble performing a sudden stop, which would be surrounded by a “trembling” of the part of the body that moved.

5.3 Study 3: Analysis of synchronization of multi-modal data

The research challenge that led to the development of the MECS algorithm is to quantify intra-personal and inter-personal synchronization in various human multi-modal behaviors. We describe here an example of application of MECS to measure multi-modal, intra-personal synchronization between respiration phases and kinetic energy captured while performing body movements characterized by different expressive qualities.

Taking a breath is a physical action that can influence the body movements performed at the same time. Similarly, body movements expressing abrupt changes of velocity and acceleration can influence the respiration pattern. Rhythm of respiration synchronizes with repetitive motoric activities such as running [BK93, HTB12], or rowing [BMB$^+$06]. Moreover, respiration plays an important role in learning physical activities, such as for example yoga or tai-chi. In this last case study, we focus on dance to investigate the expressive qualities through the analysis of synchronization on multi-modal data.

5.3.1 Hypothesis

In this study, we focus on movements displaying two different expressive qualities: fluid and impulsive movements.

- **Fluidity** the definition of Fluidity used in this experiment is described in Chapter 3, Section 3.2.2.
- **Impulsivity**: the definition of Impulsivity used in this experiment is described in Section 5.1 of this chapter.

Imagine a dancer mostly dancing fluidly and suddenly displaying one or more very impulsive movements. Figure 5.9 shows two examples of multi-modal data that may be captured in such a situation: Figure 5.9-A displays an example of impulsive movement, and Figure 5.9-B is a fluid movement performed by the same dancer. The two signals refer to *movement energy*, computed as the kinetic energy of the whole body, and to *respiration*, computed as the energy of the audio signal capture by a microphone (that was located near to the dancer's mouth). Energy peaks of the audio respiration signal and those of the kinetic energy are closer in Figure 5.9-A than in Figure 5.9-B. This is reasonable, as we may expect that different modalities may be more synchronized at the time of an impulse.

The respiration rhythm is interrupted by the impulse: a new respiration phase starts and it is synchronized with the kinetic energy peak caused by a sudden increase of velocity. Formally, we hypothesize that during an impulsive movement there is a single strong peak in the respiration

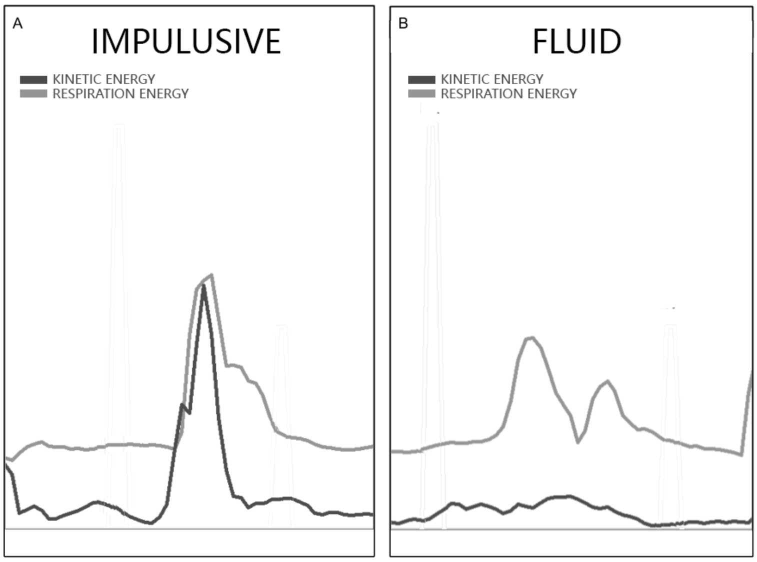


Figure 5.9: Two parts of the same dance performances, alternating impulsive and fluid movements.

signal that appears immediately after the beginning of the respiration phase and which is synchronized with a similar peak in kinetic body energy. Such co-occurrence may depend on the respiration phase and it is not observed for fluid movements.

5.3.2 Input

We applied the MECS algorithm on 90 seconds of a recordings of a female professional dancer (see [PCG$^+$16] for more details on the recording setup). The dancer was asked to move fluidly most of the time, and to perform several impulsive movements in between. The data consisted of: the audio of the respiration captured using a wireless microphone (mono, 48kHz), two accelerometers (xOSC [2]) placed on the arms of the dancer, as well as 2 video cameras (1280x720, at 50fps).

[2] http://x-io.co.uk/x-osc

5.3.3 Features extraction

5.3.3.1 Respiration Energy (RE)

The audio stream was segmented in frames of 1920 samples. The instantaneous energy of the audio signal was computed on each single frame using Root Mean Square (RMS). Next, we extracted the envelope of the instantaneous audio energy using an 8-frames buffer.

5.3.3.2 Kinetic Energy (KE)

Kinetic energy was computed from the data captured by the two accelerometers placed on the dancer's arms. Velocity was obtained by integrating the values obtained from the accelerometers. Next, we computed the average kinetic energy by taking the mean value of the kinetic energies obtained from the data from the two accelerometers. The accelerometers signals and the kinetic energy obtained thereof were sampled at 50fps. To keep synchronization with the audio energy signal, every second value was taken.

5.3.4 Manual annotation

5.3.4.1 Expressive Quality Annotation (EQA)

An expert in expressive movement qualities annotated the beginning of each segment of the video where impulsive or fluid movements can be perceived. EQA was used to distinguish peaks and respiration phases belonging to segments of impulsive and fluid movements.

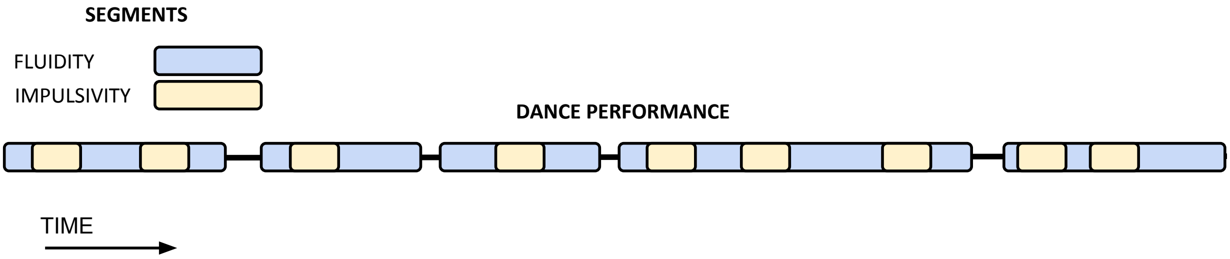


Figure 5.10: A simplified view of the segmentation of the dance performance. The interval of each segment has been annotated manually. Some parts of the performance do not present any of the two considered qualities. Movement and respiration features (KE and RE) has been extracted on the entire performance without differentiating between the segments. Annotations are used to define the time series and the set of events used to perform the analysis of synchronization.

5.3.4.2 Respiration Phase Annotation (EX/IN)

Although the inspiration and expiration phases can be automatically extracted from the audio signal (see for example [YF14]), to ensure high precision of the segmentation we opted for manual annotation. The audio signal was annotated by an expert who used the Audacity software[3] to assign the start and end time of each occurrence of each inspiration and expiration phase detected in the audio signal.

5.3.5 Applying MECS

In our initial and simple study, the hypothesis is to understand if, given the two respiration phases, the synchronization between a single respiration phase's energy and the kinetic energy of the body, grows in correspondence of impulsive behaviors.

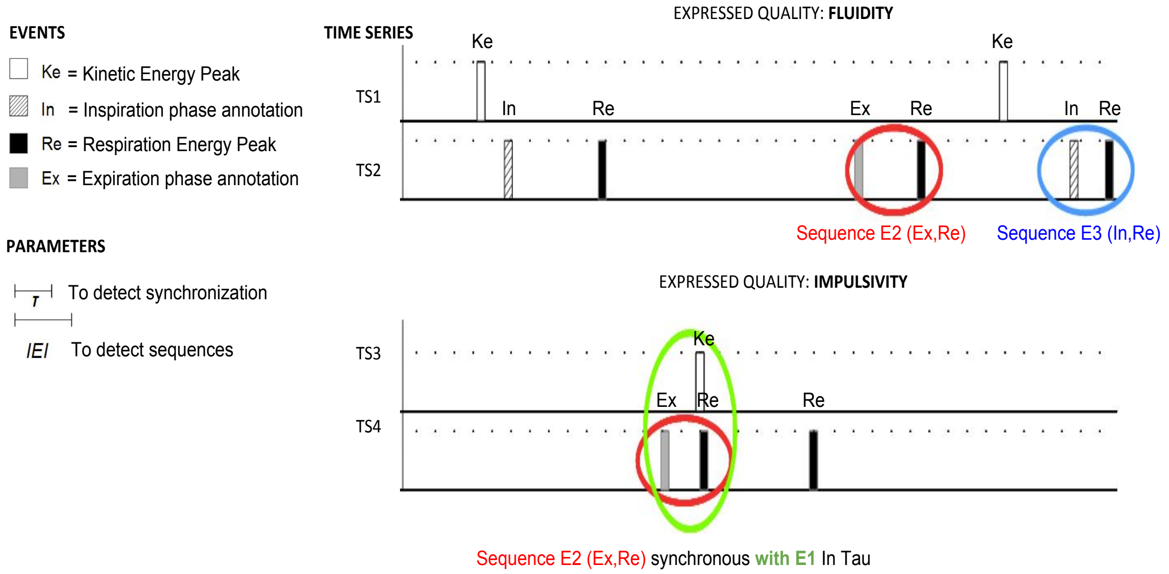

EVENTS
□ Ke = Kinetic Energy Peak
▨ In = Inspiration phase annotation
■ Re = Respiration Energy Peak
▩ Ex = Expiration phase annotation
PARAMETERS
T To detect synchronization
IEI To detect sequences
TIME SERIES
EXPRESSED QUALITY: FLUIDITY
TS1
TS2
Ke
In
Re
Ex
Re
Ke
In
Re
Sequence E2 (Ex,Re)
Sequence E3 (In,Re)
EXPRESSED QUALITY: IMPULSIVITY
TS3
TS4
Ke
Ex
Re
Re
Sequence E2 (Ex,Re) synchronous with E1 In Tau

Figure 5.11: Synchronization between respiration and kinetic energy: the four time series of the extracted features before sequence detection.

In order to apply the MECS algorithm peaks, were extracted from the kinetic energy signal KE and from the respiration energy signal RE. Following the EQA annotations, a set of $N = 4$ time series $T = \{TS_1, TS_2, TS_3, TS_4\}$ was obtained, where:

- TS_1 contains a value different from zero in correspondence of peaks in the kinetic energy signal KE during an **fluid movement**.

[3] http://www.audacityteam.org

- TS_2 contains a value different from zero in correspondence of:
  - peaks in the respiration energy signal RE
  - the start time of an inspiration phase (i.e., an *IN* annotation)
  - the start time of an expiration phase (i.e., an *EX* annotation)

  during an **fluid movement**.
- TS_3 is the equivalent of TS_1 during an **impulsive movement**.
- TS_4 is the equivalent of TS_2 during an **impulsive movement**.

The MECS algorithm was then applied to measure inter-class synchronization between $K = 3$ classes of events, i.e., $E = \{E_1, E_2, E_3\}$, in the time series belonging to T. The three classes were defined as follows:

- E_1: is the class of the events consisting of peaks in the kinetic energy signal KE. These events are contained in time series TS_1 and TS_3.
- E_2: is the class of the macro-events, in the respiration energy signal RE, consisting of the following sequence $Seq = (IN, RE)$, i.e., the beginning of an inspiration is nearly immediately followed by a peak in the respiration energy signal RE. These sequences can be detected in time series TS_2 and TS_4.
- E_3: is the class of the macro-events, in the respiration energy signal RE, consisting of the following sequence $Seq = (EX, RE)$, i.e., the beginning of an inspiration is nearly immediately followed by a peak in the respiration energy signal RE. These sequences can be detected in time series TS_2 and TS_4.

To model the occurrence of a peak in the respiration energy signal RE nearly immediately after the beginning of a respiration phase IN or EX, we tuned the IEI parameter in the conditions that events should satisfy for a sequence to be detected (see Section 4.3.6).

We ran MECS to compute: $C_1 = S_{1,2}(1,2)$, $C_2 = S_{1,3}(1,2)$, $C_3 = S_{1,2}(3,4)$, $C_4 = S_{1,3}(3,4)$.

MECS parameters were given the following values: $\tau_{k_1,k_2} = 15$ samples (18ms) for every pair (k_1, k_2) of event classes, $IEI = 25$ samples (50ms). A linear kernel was used. For each condition C_h, time series TS_1, TS_2, TS_3 and TS_4 were divided into N_b buffers and, for each buffer, the synchronization value S_{k_1,k_2} was computed, and finally used to compute an overall synchronization score.

For each condition C_1 - C_4, we count the number of times inter-class pairwise synchronization $S_{k_1,k_2}(i,j) > 0$ (i.e., the number of times at least a partial synchronization is detected) and we normalize such a number by dividing it by the number of occurrences of the respiration phase events relate to (i.e., inspiration for C_1 and C_3, and expiration for for C_2 and C_4). That is, the final synchronization score for condition C_h, $h = 1...4$ is obtained as:

$$Q_{C_h} = \frac{1}{Y} \sum_{t=1}^{N_b} \Theta(S_{k_1,k_2}(i,j)) \tag{5.3}$$

where: i and j, and k_1 and k_2 are the indexes of the time series TS_i and TS_j and of the event classes E_{k_1} and E_{k_2} involved in condition C_h, respectively; $\Theta(z)$ is the Heaviside function; Y is the number of occurrences of the respiration phase (inspiration or expiration) involved in condition C_h.

It is important to notice that the original ES algorithm is not powerful enough to model these complex relationship between modalities: we want to analyze sequences of peaks in two modalities (respiration audio energy and kinetic energy), which should be treated as a single event, if and only if such peaks appear in a sufficiently "short" time window. The MECS algorithm allows to detect sequences and to analyze their development along time. Moreover, MECS also allows us to distinguish between two different classes of events related to respiration: inspiration and expiration. In this way, we are able to check whether the sequence of peaks in impulsive movements happens for any respiration phase, or whether it is rather related to one of the two respiration phases.

5.3.6 Results

The number of inhalations and exhalations in the analyzed trial was similar (45 vs. 48). Much less inspiration intervals were observed, however, for impulsive movements than for fluid ones (12 vs. 33). When considering the results the MECS algorithm provided for impulsive movements (conditions C_3 and C_4), the total number of the synchronized events was much higher for the expiration phases than for the inspiration phases (13 vs. 7), but the normalized score for both respiration phases is similar with $Q_{C_3} = 0.58$ and $Q_{C_4} = 0.65$. When comparing these results with those obtained for fluid segments the difference is strong: only few synchronized fluid segments were observed for the inspiration phase with $Q_{C_1} = 0.12$, and even less for the expiration phase $Q_{C_2} = 0.04$. To further investigate this result, we repeated the same procedure with different values of τ_{k_1,k_2} and we obtained similar results, see Figure 5.12.

5.3.7 Discussion and conclusions

We can say that applying MECS algorithm allowed us to investigate the difference in synchronization between different respiration phases and expressive qualities of the movement in the analyzed trial. It seems that the beginning of an impulsive movement co-occurs more often with

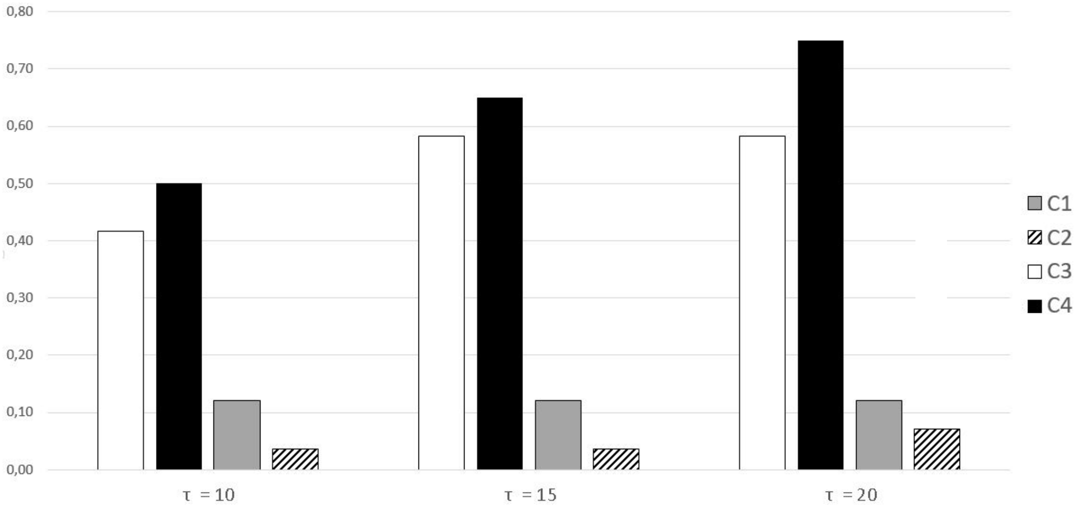
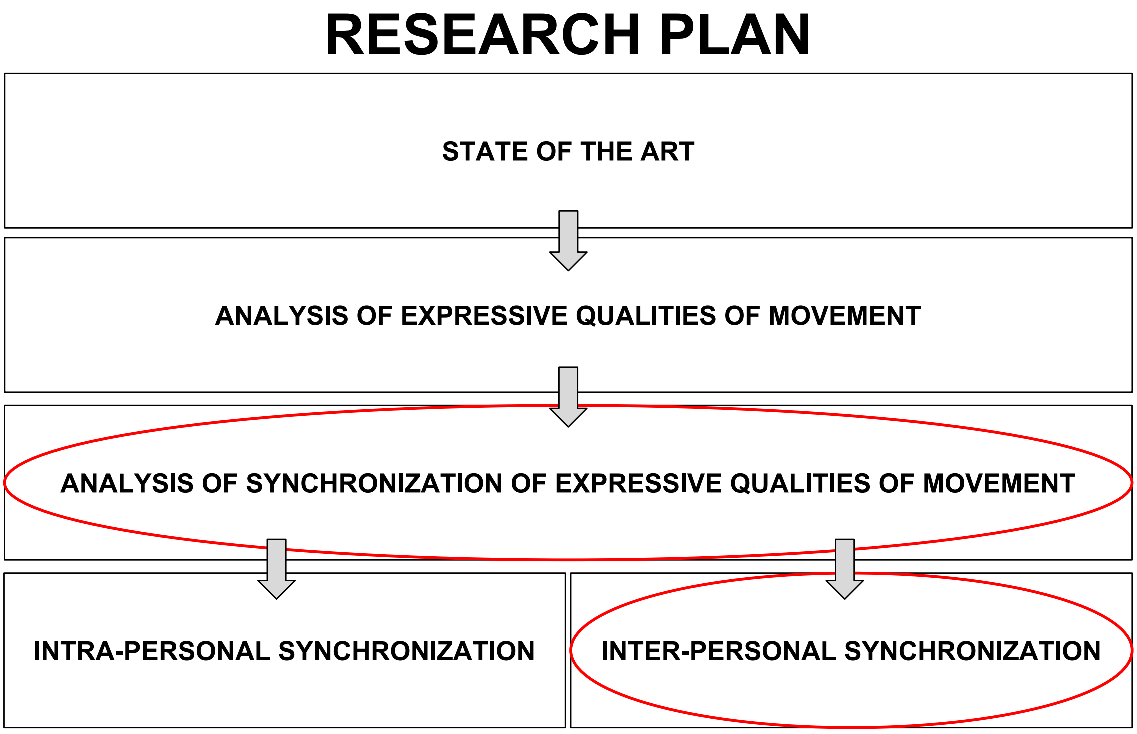


Figure 5.12: Synchronization results for τ_{k_1,k_2} equal to 10, 15, and 20 samples (at 50fps, the correspond to 12, 18 and 24 milliseconds). The same value is used for every pair of event classes (k_1, k_2).

an exhalation phase, and for both respiration phases in impulsive movements a strong peak in the energy of the respiration signal often appears nearly immediately after the beginning of the respiration phase and is synchronized with the peak of body kinetic energy. The same relationship was very rarely observed for fluid movements. It must be highlighted that the analysis above only serves as an illustration of the application of our algorithm to a real research question related to synchronization of multi-modal behavior. We illustrated how MECS has been applied to artificially generated data, and to data from a real case study, to evaluate its correctness and provide a practical application example. More applications of the algorithm would be needed to effectively highlight the real potentialities of our approach. For example, MECS can be used to analyze the degree of synchronization of movements performed by multiple users (inter-personal synchronization), or to measure coordination between an user and a robotic agent.

Chapter 6

Inter-personal synchronization


RESEARCH PLAN
STATE OF THE ART
ANALYSIS OF EXPRESSIVE QUALITIES OF MOVEMENT
ANALYSIS OF SYNCHRONIZATION OF EXPRESSIVE QUALITIES OF MOVEMENT
INTRA-PERSONAL SYNCHRONIZATION
INTER-PERSONAL SYNCHRONIZATION

Contents

The following section, which concerns the study of musical entrainment, was carried out in the context of an international project: the IEMP project, in collaboration with Durham University (coordinator), University of Genoa and Western Sydney University. Interpersonal Entrainment in Music Performance (IEMP) is an interdisciplinary research project focused on the study of in inter-personal coordination, synchrony and entrainment in group music-making.

Group music-making is a distinctive mode of human social interaction: it is a widespread activity that showcases the remarkable capacity for precision and creativity demonstrated in the coordination of rhythmic behavior between individuals. Understanding musical entrainment requires contributions from several disciplines, in particular ethnomusicology, music cognition and computing. The way in which entrainment in music is manifested appears to vary as a function of differences in social, ritual and musical conventions. A better understanding of the process of inter-personal entrainment and its cultural variation is therefore imperative. The IEMP research focus is to investigate the key-aspects of inter-personal musical entrainment in a comparative study of a variety of cultural settings; it does so through the establishment of an international and interdisciplinary team, and by creating a shared corpus of prepared and annotated performance data. The main objective of the IEMP project is to comprehend how can inter-personal entrainment be reliably and efficiently measured using video collections recorded in ecologically-valid context.

My role, within the project, was to discover which video extraction techniques are robust across an evaluation subset of the whole IEMP collection. This is a challenging process due to the difficulty in standardizing the video analysis methods because of many differences in the recorded material. My contribution involved the following activities:

- Identify a set of vision computer methods to perform a video tracking on a set of a large variety of music performance recordings (comprising a variety of setups, light conditions, camera positions etc.).

- Develop a system prototype that implements the set of identified computer vision techniques and extract movement tracking measures.
- Validate the system robustness.
- Integrate in the system an *analysis* module that implements the main techniques to perform the analysis of synchronization (including the MECS algorithm) on the extracted data.
- Test the system on the project dataset.

6.1 Entrainment objective measures

In Section (2.2.1.1) the concept of inter-personal entrainment is defined as the process whereby two or more users interact with each other perceiving and producing rhythmic movements. To objectively quantify entrainment, we focused on the definition given by [PSK12] Phillips-Silver and Keller i.e., entrainment can be conceived as "*the spatio-temporal coordination between two or more individuals, often in response to a rhythmic signal.*"

Activities involving the creation or listening to music, are an excellent test-bed to validate algorithms and methods that provide measurements of entrainment. In IEMP, analysis of synchronization techniques can be used to measure the coordination of expressive movements (or their features) of performers, extracted from video recordings. Coordination of movements performed in musical and rhythmic activities are a form of estimation of entrainment. In order to obtain more accurate results, other data would be needed to supplement what can be extracted from video recordings, such as physiologic data (e.g., breathing, hearth beating and so on). To objectively measure the degree of entrainment it is necessary to identify the main entities and their interactions. Entrainment between involved entities (i.e., musicians during a performance) will emerge if the dynamics of their interactions (e.g. movements and expressive qualities) evolves to shared synchronization patterns.

During a musical performance the main entities are the interacting users, represented as systems receiving stimuli and reacting to them. Suppose to be able to record a music performance with non-invasive sensors (i.e., cameras and microphones) obtaining a set of time series containing both audio and video features (e.g., event onsets from each performer and the quantity of motion and head/hand movements of each performer). Then, the time series are analyzed using the techniques to perform the analysis of synchronization (e.g., cross-recurrence quantification analysis, event synchronization) in order to provide quantitative coordination measures. An example of application of such approach is the pilot study, presented in the next section.

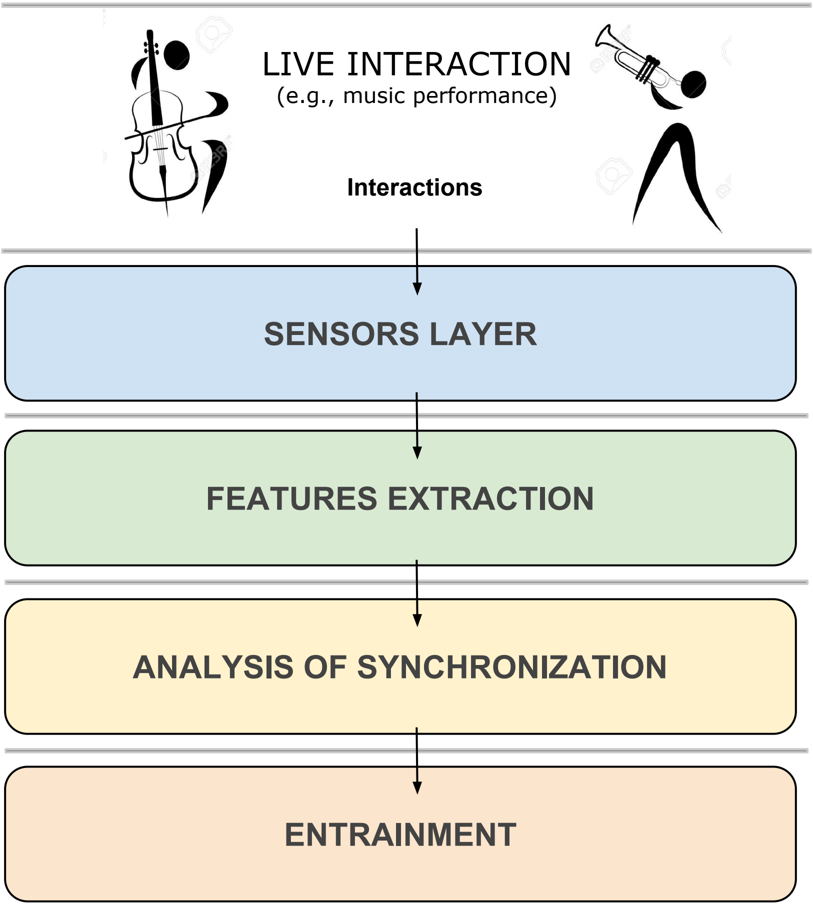


Figure 6.1: Musical entrainment computational model schematic representation. In the bottom layers, the analysis of synchronization of movement features supports entrainment analysis.

6.2 Entrainment analysis on videos datasets

6.2.1 Pilot study

We developed a system prototype for the automated extraction of features for the analysis of synchronization in order to find evidence of entrainment from video recordings of music performances. First of all we selected, from the IEMP dataset, an Indian music performance, on which the prototype's functionalities can be tested and, at the same time, the following hypothesis could be verified: *is the synchronization between the Indian singer and the tabla accompanist following a fixed pattern?* The performance in divided in several musical cycles. In particular, within a single musical cycle, synchronization between the players is relatively loose over the course of the middle part of the cycle, and is becoming more precise in the approach to the

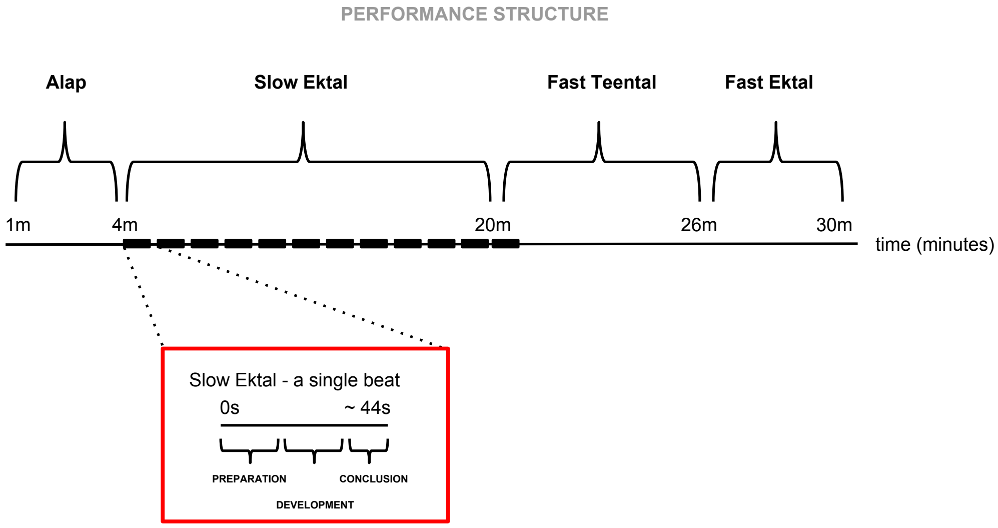


Figure 6.2: The structure of the musical performance

final part (*sam*), and can this be demonstrated by an automated analyzing the body movement of musicians? This hypothesis refer to the soft entrainment model ([YTY02]) that describes how, in a musical phrase, interactions are less synchronized at the beginning of the phrase while they tend to be increasingly synchronized and reach their maximum in a point near the phrase end.

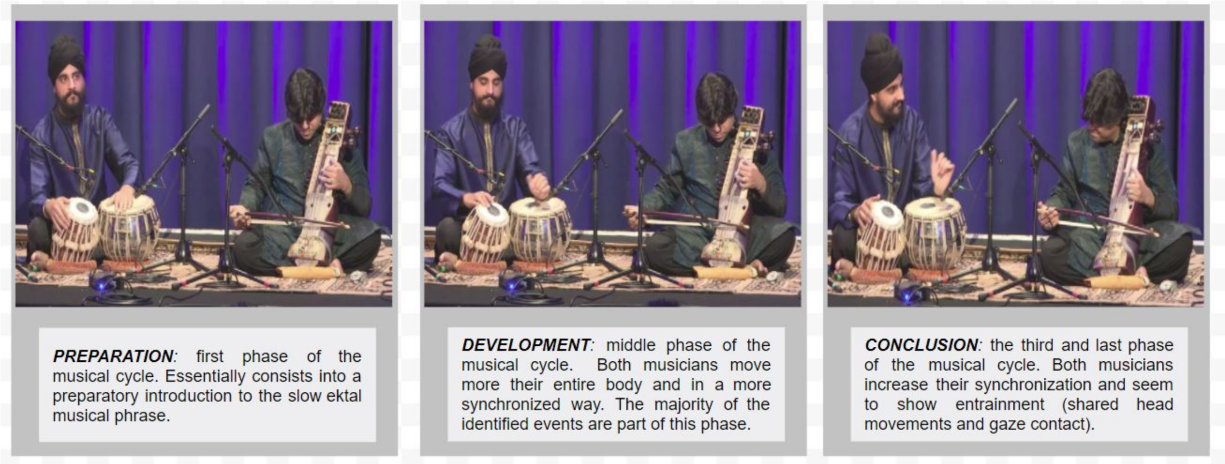
6.2.1.1 Performance description

The performance recordings were made at a public concert promoted by Gem Arts at the Sage Gateshead on 24th September 2012, by a team from Durham University using two HD video cameras (Canon XF305), with separate audio tracks fed via the mixing desk to a Tascam HS-P82 recorder. The soloist, Murad Ali (MA), is highly-regarded sarangi (bowed lute) player from India; his tabla accompanist, Gurdain Rayatt (GR), is a young UK-based musician. The sarangi is the favored melodic accompaniment instrument for Hindustani vocal performance. It became established as a solo concert instrument in the latter half of the twentieth century: when used as a solo instrument, as here, its repertory and style remain very closely modeled on vocal style.

The main reasons for the choice of this recording were the good quality of both performance and recording, and the use of the sarangi, which means that vocal-style music can be studied in a duo format (without additional melodic accompaniment).

The thirty minute performance has 4 sections:

- S1: 0:00-4:30: Alap. MA on unaccompanied sarangi.



PREPARATION: first phase of the musical cycle. Essentially consists into a preparatory introduction to the slow ektal musical phrase.

DEVELOPMENT: middle phase of the musical cycle. Both musicians move more their entire body and in a more synchronized way. The majority of the identified events are part of this phase.

CONCLUSION: the third and last phase of the musical cycle. Both musicians increase their synchronization and seem to show entrainment (shared head movements and gaze contact).

Figure 6.3: A single *slow ektal cycle* division

- S2: 04:30-20:10: **Slow Ektal** (the one considered in this pilot study) formed by 12 cycles (matras), each cycle lasts about 44s at the beginning, accelerating to c.35s by the end.
- S3: 20:10-26:15: Fast Teental. 16 cycles, each cycle lasts c. 2.5s.
- S4: 26:15-30:50: Fast Ektal. 12 cycles, each cycle c. 2.5s.

Every slow ektal cycle has been considered for the analysis and divided into three different parts: preparation, development and conclusion. This is a novel division: it reflects the fact that the soloist typically fills the cycle with one coherent episode of improvisation, in the course of which a new idea is presented, then further developed, before he starts to orient himself towards the conclusion of the cycle. Note that the division into equal thirds is arbitrary.

6.2.1.2 Manual annotations

In order to refine questions for investigation here, the frontal video view was watched several times, beats of the tala cycles were labeled as a reference (tapping the beats in Sonic Visualiser) and initial observations noted. Since it is impossible to visually identify many periodic movements – other than the recurrence of head nods at the start of each cycle in the slow ektal section – the following observations are largely qualitative in nature:

- In the alap (S1) there is no visible coordination between MA (who plays) and GR (who does not), although the latter does maintain a respectful and attentive demeanor.
- In the slow ektal portion (S2), most occurrences of sam (beat one) are clearly marked by shared head movements by the two musicians. Other than this there is little obvious gestural communication between the two.

- Looking at the joint head nods in the slow ektal (S2) in more detail, sometimes the musicians look towards each other, making eye contact; at others both musicians look ahead at the audience. Sometimes there seem to be clear preparatory movements: either a nodding of MA's head in the last beat or so, or a joint head-nod on beat 9; sometimes MA looks to GR and makes eye contact just before the nod. It was noticed that after periods in which MA has been giving a lot of visual attention to a section of the audience, he deliberately reconnects with GR in this way (see e.g. cycle 15, s 769-799).

- In cycle 20 (s 968-1001) the musicians make a mistake, adding an extra beat to the cycle. Their movements suggest a lack of mutual attention here, although it is not clear what caused the error. (As is usually the case after such mistakes, the musicians make no obvious reference to the fact either to each other or to the audience).

- In contrast to the slow ektal, in the two fast sections (S3, S4) the joint head nods on sam are only occasional. However, much more miscellaneous gestural communication occurs: in particular, instructions from MA to GR, mostly to either adjust the tempo (he asks GR to speed up at least 5 times) or to take a solo (4 times).

- Transitions between sections are obviously significant in terms of coordination. Observation of these points (c. 20-21 mins and 26-26:30) does not reveal much movement coordination, however: what seems to happen is that MA stops one section, then after a few sections begins the next, and GR starts up when he has picked up the new tala pattern (during the transition to S3 MA also briefly re-tunes his instrument).

- There is nothing in the video to suggest that MA's leadership is being contested by GR. The soloist role and the fact that MA is a touring artist and GR a local accompanist mean that the musical 'seniority' clearly lies with the former. When GR takes solos, for instance, this is clearly at the invitation of MA. h. It was also noticed in a few occasions that, although there is no eye contact or shared gesture, MA is clearly working on a rhythmic figure while concentrating hard on the tabla

6.2.1.3 System architecture

This pilot study is focused on analyzing the relationship between the two musicians in S2 (the slow ektal portion of the performance) between the three phases of each single matra: preparation (beats 1-4), development (5-8) and conclusion (9-12); based on the observations described above (see 6.2.1.2), we hypnotize that inter-personal synchronization increases in the last third of the cycle.

In particular for this first analysis, we focused on the head movements of both musicians. Figure 6.4 shows an overview of the prototype architecture and its major layers. The main building blocks, or modules, are described below.

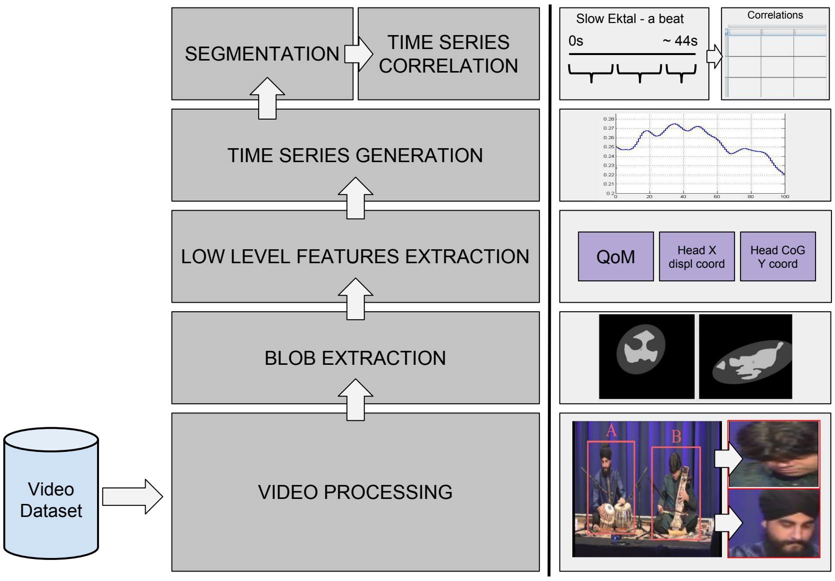


Figure 6.4: The system architecture for automatic feature extraction from music performance video recordings

Layer description:

- **Video processing**: the system accepts as input a video of a music performance (fixed camera view, high quality colour video). This low-level video analysis layer consists of the automated tracking of the regions of interests (ROIs), i.e. the minimum rectangles containing the blobs corresponding to the locations of musician's heads. A blob is defined as a cluster of pixels extracted from the ROI characterized by invariant properties (i.e. pixels that are part of the same cluster vary within a statistically coherent range of values). In this case, two ROIs are identified, corresponding to the two musicians' heads. Once identified, a video binarization is applied.

- **Blob extracting**: the output of this layer is a binarized image containing blobs rendered as white textures. The extracted blobs correspond to the heads of the two performers.

- **Low-level features extraction**: low-level features are extracted from blobs. Computer vision techniques are applied to compute features such as velocities, accelerations, energy, contraction, directness and so on. In this case, three features of interest were identified:

  - Quantity of Motion: the amount of movement detected by the video camera and computed as the pixel-based difference between two consecutive frames containing head blobs.
  - Y coordinate of the head Centre of Gravity (CoG): the blob's barycenter coordinate

on the vertical axis. This feature was used to analyze joint head movements (nods) that typically characterize the end of musical (tala) cycles.

  - Head X displacement: the overall translation and rotation components of the head movements, computed from optical flow techniques applied to the head of each musician. This measure can be useful for approximating the gaze direction, to identify whether a musician is looking for eye contact during the performance.

- **Time Series Generation**: the values of the features are logged in a set of files as time series. Time series are synchronised at a fine-grained level.

- **Segmentation**: performances are manually segmented into different sections, corresponding to sections in the musical structure. The slow ektal portion is segmented into 22 cycles (average duration is between 35 and 44 s).

- **Time Series Correlation**: the Pearson's correlation coefficient between the features' time series is computed. Statistical analysis is performed on the correlation values.

6.2.1.4 Results

To assess synchronization between the movements of the two musicians, the Pearson's linear correlation coefficient was computed on the time series of each extracted movement feature for each of the three parts of each slow ektal cycle (preparation, development, and conclusion).

The final part of each cycle (conclusion) proved to be the one displaying the highest synchronization between the two musicians. Table I reports mean and standard deviation of the correlation coefficient computed for each movement feature.

A one-way repeated-measures ANOVA was conducted to compare the effect of every cycle part (preparation, development, and conclusion) on the synchronization value for each feature. ANOVA hypotheses were checked with the commonly used Shapiro-Wilk and Mauchly's tests. Quantity of Motion: data did not deviate from a normal distribution (Shapiro-Wilk test). Mauchly's Test of Sphericity indicated that the assumption of sphericity was not violated ($p = .302$). The effect of cycle part on synchronization was significant ($F(2, 42) = 27.73, p < 10^{-7}$).

Three paired samples t-tests were used to make post hoc comparisons between conditions. A Bonferroni correction was applied to account for multiple comparisons.

The first paired samples t-test indicated that there was no significant difference in synchronization for the preparation part ($M = -.006$, $SD = .125$) and the development part ($M = .036$, $SD = .204$), $p = .999$. The second t-test indicated that there was a significant difference in synchronization for the preparation part ($M = -.006$, $SD = .125$) and the conclusion ($M = .391$, $SD = .289$), $p < 10^{-4}$. The third t-test indicated that there was a significant difference

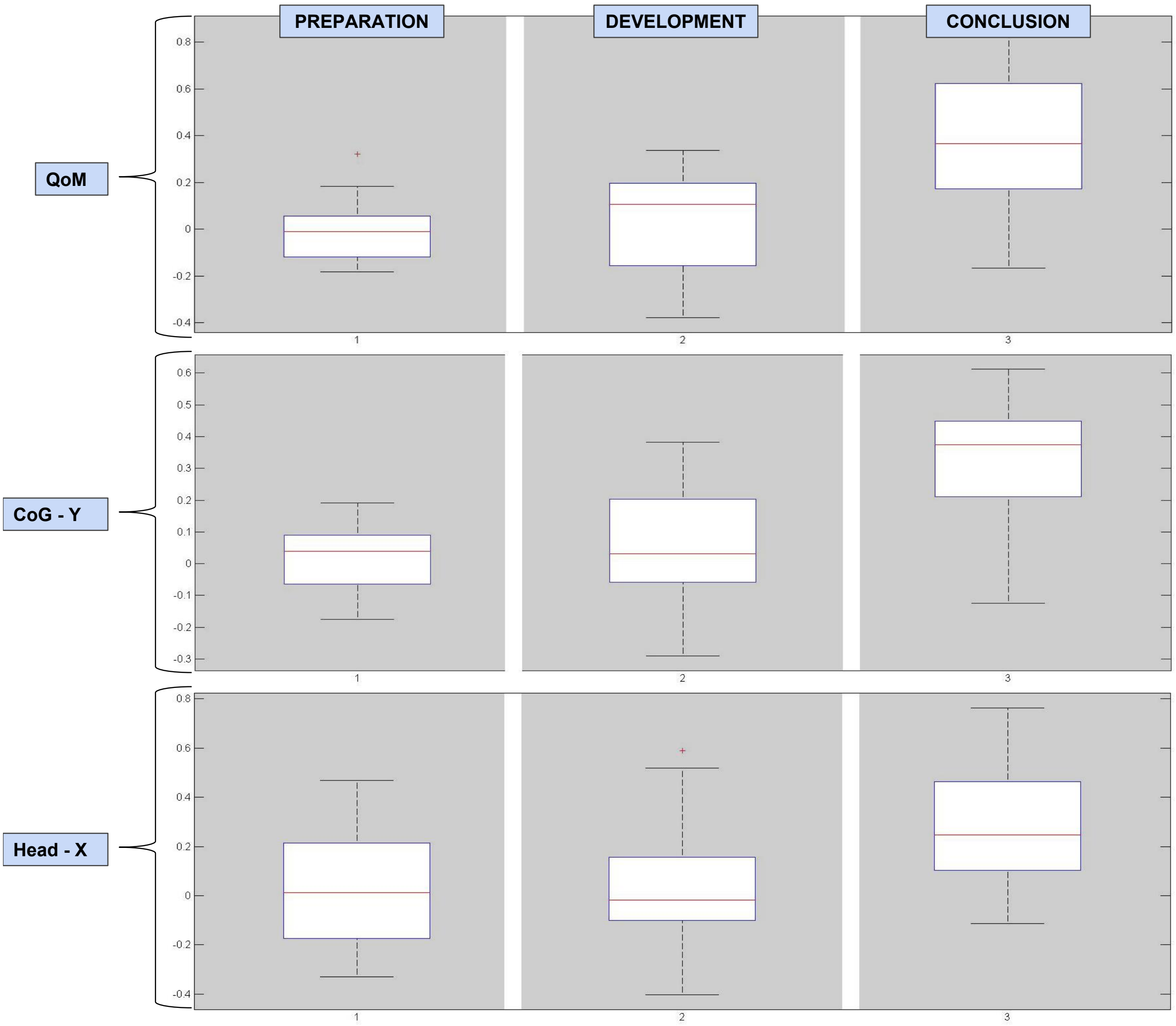


Figure 6.5: Boxplots relative to the three parts of a cycle. (A) Quantity of Motion. (B) Y coordinate of CoG. (C) Head X displacement.

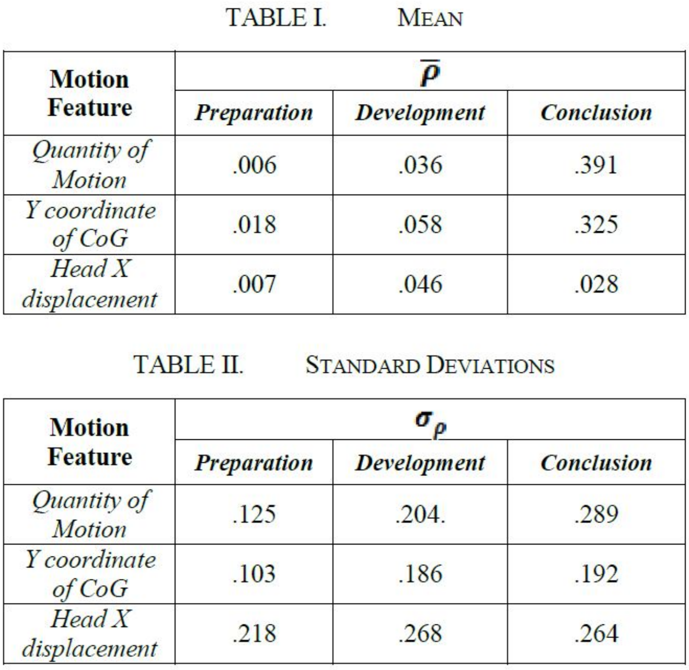


TABLE I. MEAN

Motion Feature	$\overline{\rho}$		
	Preparation	*Development*	*Conclusion*
Quantity of Motion	.006	.036	.391
Y coordinate of CoG	.018	.058	.325
Head X displacement	.007	.046	.028

TABLE II. STANDARD DEVIATIONS

Motion Feature	σ_{ρ}		
	Preparation	*Development*	*Conclusion*
Quantity of Motion	.125	.204.	.289
Y coordinate of CoG	.103	.186	.192
Head X displacement	.218	.268	.264

Figure 6.6: Pilot study results: Pearson coefficient and standard deviation of the three motion features measured on the two musicians

in synchronization for the development part ($M = .036$, $SD = .204$) and the conclusion ($M = .391$, $SD = .289$), $p < 10^{-4}$.

Table I and II (see Figure 6.6) and 6.5 summarise some preliminary results on the analysis of entrainment.

Data analysis reveals how the synchronization between Indian sarangsoloist MA and tabla accompanist GR is relatively loose over the course of a long cycle, becoming more precise in the approach to the *sam*. The research output of this pilot study is a system prototype that was demonstrated to be suitable for the analysis of synchronization of musical performances video recordings. In the next section, an extension of the system is presented and validated.

6.2.2 A robust system to extract features for entrainment analysis

On the basis of what has been verified in the pilot study and according to the IEMP project's finalities, the next step is to enhance the functionalities of the above-presented system and to verify its robustness and usability. The enhanced version of the system should offer a way to extract useful information from a diverse range of settings, including field research and other ecological contexts in which the implementation of complex motion capture (MoCap) systems is not feasible or affordable, such ecological performance contexts (e.g., rehearsals in music practice rooms, concerts, ritual ceremonies and religious events, etc.).

The content of this section is based on the work published in [JEA$^+$17]. Our hypothesis is that, computer vision methods applied to video recordings (as in the pilot study), can perform similar tracking of body movements to more expensive techniques, such as MoCap systems, under certain conditions. Use video recordings instead of specialized MoCap technologies can be really advantageous: motion capture system are not only costly but also invasive (i.e., markers need to be fixed to a person's body, or for some systems a specialized suit needs to be worn), time consuming in terms of setup and calibration procedures, and difficult to implement in ecological settings and outside of specialized MoCap laboratories [1]. My contribution to the research presented in [JEA$^+$17], was to develop and implemented the software architecture of the system to perform the automated video tracking and feature extraction (presented in more detail in Section 6.3).

To verify the hypothesis, we applied three automated computer vision techniques (Frame Differencing, Optical Flow, and Kernelized Correlation Filter) to a set of video recordings comprising a variety of performers, performance settings, instrumentations, and musical styles. We selected three video datasets of ensemble performances from different genres, again from the IEMP dataset: a pop piano duo, three jazz duos, and a string quartet. The next section will discuss about:

- the robustness of three computer vision techniques for capturing body movements across the different performance conditions.
- how these techniques were able to capture the actual motion of performers, compared by MoCap data from the same performances.

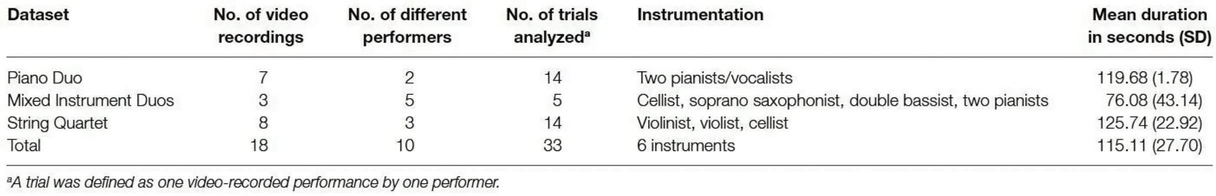
Finally, as previous studies comparing MoCap data to computer vision techniques have primarily examined full-body movements ([RAF$^+$17]), we included analysis of video data within predefined regions of interest (ROIs, e.g., regions of the head or upper body) to verify whether the video analysis techniques could also be effective in quantifying movements of specific parts of the body.

[1] This topic has already been covered in Section 3.2.3

6.2.2.1 Materials

We made use of three datasets (see Table 6.7). Datasets are made up of video recordings and MoCap data of the same musical performances that had been collected for other research purposes.

- The first dataset (hereafter referred to as the "Piano Duo") comprised seven songs performed by singer-songwriters Konstantin Wecker and Jo Barnikel. Wecker has been described as one of Germany's most successful singer-songwriters, with a career spanning 40 years at the time of the recording, and Barnikel is a leading film and TV composer who had been accompanying Becker on recordings and concert tours for over 15 years.
- The second dataset (hereafter referred to as "Mixed Instrument Duos") consisted of three performances by jazz duos, a subset of the Improvising Duos corpus ([MHBK15]). Mixed Instrument Duos is a collection of two duos performed free jazz improvisations. Performers were recruited on the basis of public performance experience of around 10 years in their respective styles.
- The third dataset (hereafter referred to as "String Quartet") comprised eight recordings by the Quartetto di Cremona performing the first movement of Schubert's String Quartet No. 14 ("Death and the Maiden"; [GMC$^+$13]). Two of these recordings featured only the first violinist performing his part alone. For the other six recordings, two of the four performers were selected for whom the least occlusions were observed (i.e., another player was not moving in front of him/her regularly).



Dataset	No. of video recordings	No. of different performers	No. of trials analyzed[a]	Instrumentation	Mean duration in seconds (SD)
Piano Duo	7	2	14	Two pianists/vocalists	119.68 (1.78)
Mixed Instrument Duos	3	5	5	Cellist, soprano saxophonist, double bassist, two pianists	76.08 (43.14)
String Quartet	8	3	14	Violinist, violist, cellist	125.74 (22.92)
Total	18	10	33	6 instruments	115.11 (27.70)

[a] *A trial was defined as one video-recorded performance by one performer.*

Figure 6.7: Summary of performance details for each dataset.

In total, the three datasets allowed for the analysis of 33 cases of 10 different performers playing six different instruments. Recordings were made under stable conditions (e.g., all string quartet recordings were made with performers situated in a similar position on the same stage using the same video camera and MoCap system).

Dataset were analyzed in respect of performers' permissions on data reuse. The Piano Duo and Mixed Instrument Duos were both recorded at the Max Planck Institute in Leipzig, Germany, using a Vicon Nexus 1.6.1 optical MoCap system with 10 cameras and a sampling rate of 200

Hz. A SONY HDR-HC9 camera was used to make the video recordings. The video files were recorded in AVI format at a frame rate of 25 fps and frame size of 720 x 576 pixels. The String Quartet was recorded at Casa Paganini Research Centre (University of Genova, Italy), using a Qualisys Oqus300 MoCap system with 11 cameras and a sampling rate of 100 Hz. A JVC GY-HD-251 camera was used to capture video of the performances. The video files were recorded in AVI format at a frame rate of 25 fps and frame size of 720 x 576 pixels.

6.2.2.2 MoCap data

All MoCap data were processed using the MoCap Toolbox ([BT13]) in Matlab. Each dataset was first rotated in order to orient the MoCap data to the same perspective as the camera angle of the video recording. This adjustment was done manually by inspecting animations generated from the MoCap data in comparison to the video recording (see Figure 6.9). Once the optimal rotation was achieved, a subset of markers was selected from each performer, comprising one marker from the head and one from the torso or each shoulder (if a torso marker was not present, as was the case for the String Quartet dataset). Markers were also selected in consideration of the camera angle of the video. For instance, if only the back of the head of a performer was visible in the video, a marker from the back of the head was selected. The horizontal and vertical coordinates of the MoCap data are subsequently referred to as the x- and y-dimensions, respectively, which were compared to the two-dimensional data that were derived from the video recordings by the computer vision techniques.

6.2.2.3 Video data

Computer vision techniques :

The computer vision techniques, implemented for the study (see Section 6.3), are: Frame Differencing, Optical Flow and Kernelized Correlation Filter. The motivations for selecting such techniques are twofold: they apply to different contexts and they provide different outputs.

- **Frame Differencing (FD)**: In frame differencing, the foreground, i.e., the moving element(s) of interest (in this case, the performers), is separated from the background and further processing is performed on the foreground. Frame Differencing was implemented in the EyesWeb library using the *Pfinder* algorithm of Wren et al.([WADP97]). The *Pfinder* algorithm uses adaptive background subtraction, in which the background model that is subtracted from the foreground is constantly updated throughout the analysis process. The speed at which the background model is updated is determined by a the *alpha* parameter (set in the present study to 0.4). Following an optimization process, alpha was manually adjusted to a range of values and tested on a subset of the present videos. The analysis that was performed on the foreground elements measures the overall Quantity of Motion

(QoM) in each ROI for each frame, which has been computed based on the number of pixels that change in the foreground from one frame to the next. Frame Differencing block produces a single column of output values for each performer, containing the value of the QoM instant by instant)

- **Optical Flow (OF)**: Optical flow is the distribution of apparent velocities of movement of brightness patterns in an image. Initially, graphical features of the image (such as edges or angles) are identified within each section of the video frame. In the next frame, such characteristics are sought again. A speed is then associated to each pixel in the frame; the movement is determined by the ratio between the distance in pixels of the displacement of the characteristic in question and the time between one frame and another. The version of Optical Flow implemented in this study is known as dense Optical Flow and is based on the algorithm of Farnebäck ([Far03]). Traditional Optical Flow methods [e.g., as implemented by Lucas and Kanade ([LK$^+$81])] compute optical flow for a sparse feature set, i.e., using only specific parts of the image, such as detected corners. Dense optical flow, as implemented by Farnebäck ([Far03]), performs optical flow computation on all pixels in the image for each frame. The use of dense optical flow can increase the accuracy of the results, with a trade-off of slower computation speed. A similar optimization procedure was followed to that used for frame differencing in which the “pyramid layers” parameter was adjusted to a range of values and tested on a subset of the present videos. This parameter allows for the tracking of points at multiple levels of resolution; increasing the number of layers allows for the measurement of larger displacements of points between frames and also increases the number of necessary computations. The optimal value that was selected for this parameter was 12. The resulting output that was provided by the optical flow analysis was two columns of data per performer (or tracked body part), which represent movement of the barycenter of the ROI along the x- (horizontal) and y- (vertical) axes. The barycenter of the ROI is computed based on pixel intensities. The video image is converted to grayscale and the barycenter coordinates are calculated as a weighted mean of the pixel intensities within the ROI; this is done separately for the x- and y-dimensions.

- **Kernelized Correlation Filter (KCF)**: The kernelized correlation filter (KCF) tracker is a relatively recently developed tracking technique (Bolme et al., 2009 in [BDB09], based on older correlation filter methods (Hester and Casasent, 1980?), that works using pattern similarity calculations on a frame-by-frame basis. As for Frame Differencing and Optical Flow, KCF was implemented and added to the EyesWeb synchronization library (8.2). Its implementation is based on the OpenCV C++ version released by Henriques et al. ([HCMB15]). When the KCF algorithm is initialized, a visual tracker is placed at the center pixel of the predefined ROI for the first frame of the video. In the second frame, similarity and classification computations are performed by searching for the set of pixels with the maximum correlation to the initial tracker position in terms of its multichannel RGB color attributes, and so on for each subsequent frame. In effect, this allows the technique to track the movement of the performers across the ROI. Similar to Optical Flow, the output of the

KCF analysis is two columns of data per performer (or tracked body part), which represent movement of the barycenter of the ROI along the x- and y-axes. In this case, since the ROI moves dynamically with the performer, the barycenter that is used is the geometric barycenter at the intersection of the two diagonals of the rectangular ROI.

ROI definitions:

The first step when applying each technique was to manually define relevant ROIs on which to apply the technique to each video.

A rectangular ROI was selected around each performer while ensuring that only that individual performer was serving as the main source of motion in the ROI. This was generally achieved to a high standard, although there were a few cases in which the hands or bows of another performer occasionally moved into the ROI in the Piano Duo and String Quartet. Two sets of ROIs were defined for each performer in each video a larger ROI that comprised the upper body (from the mid-chest or the waist up to the top of the head, depending on how much of the performer could be seen in the video) and a smaller ROI around the head only (see Figure 6.8).

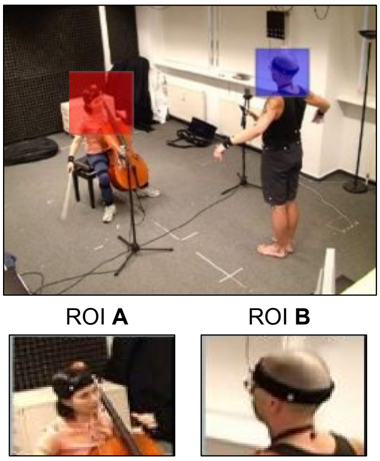


Figure 6.8: Example of ROIs.

ROI differences per computer vision method:

Frame differencing and Optical Flow were both applied using the same sets of upper body and head ROIs for each video. A slightly different set of upper body and head ROIs were defined for KCF, due to the way this technique is implemented. In typical implementations of KCF, the entire ROI moves dynamically throughout the process of tracking the performer. Conversely, Frame Differencing and Optical Flow were applied on static ROIs that do not move during the analysis process. As such, larger ROIs were needed that could encompass the whole range of movement of a performer for Frame Differencing and Optical Flow, whereas KCF is more suited to smaller ROIs since the ROI shifts from frame to frame.

6.2.2.4 Analysis: MoCap and Video data comparison

As video data collection was not the primary focus of the original studies, the video and MoCap data were not synchronized with an external timecode. As such, these two data sources were aligned in this study using automated cross-correlational methods. Each video analysis output from EyesWeb was cross-correlated with its corresponding MoCap target (e.g., the x-coordinate of the head from the Optical Flow analysis within the head ROI was cross-correlated with the x-coordinate of the MoCap head marker). This allowed us to determine the optimal lag time for each trial, which was defined as the lag at which the maximum correlation value between the video and MoCap data was reached.

The median optimal lag time from all cross-correlational analyses from the same video (taking account of analysis of all position data from both performers in each video) was taken as the optimal lag time for that particular video.

Before computing any statistical comparisons between the video and MoCap data, the MoCap data were down-sampled to match the lower sampling rate of the videos at 25 fps, and all video and MoCap data outputs were de-trended and normalized. Figure 6.9 depicts the data preparation and extraction process for video and MoCap for one example performance from the Mixed Instrument Duos.

Although the primary research question is focused on evaluating and comparing the three computer vision techniques within the two ROIs (upper body and head), *dataset* is also included as an independent variable in subsequent analyses to take account of the fact that the three datasets (Piano Duo, Mixed Instrument Duos, and String Quartet) vary on a number of parameters, including setting, recording session, lighting, camera angle, and instrumentation.

For the upper body ROI, we compared the outputs of the computer vision analyses (tracking data) to the coordinates of the torso marker from the MoCap data or the shoulders markers (left or right, depending on the dataset). For the head ROI, we compared the tracking data to the coordinates of the MoCap head marker.

Since Frame Differencing provides a single, overall estimate of movement of each performer (rather than two-dimensional tracking); Optical Flow and KCF tracking data together with the corresponding MoCap data were converted from Cartesian (x and y) to polar (radial and angular) coordinates. We then computed the absolute change of the radial coordinate on a frame-by-frame basis for each trial; this absolute change measure allowed comparisons to the one-dimensional frame differencing results.

6.2.2.5 Results

The video and MoCap data for each trial were then compared using correlations (Pearson's r); a summary of these comparisons is reported, by dataset, in Table 6.10. These descriptive statistics suggest that the two-dimensional tracking methods (Optical Flow and KCF) tend to perform more accurately than the more coarse-grained method (Frame Differencing) and that performance of all three computer vision techniques is improved when concentrated on a smaller ROI (head, as compared to upper body).

For the data using the upper body ROI, a 3 x 3 mixed ANOVA was conducted to test the effects of computer vision technique (Frame Differencing, Optical Flow, KCF) and dataset (Piano

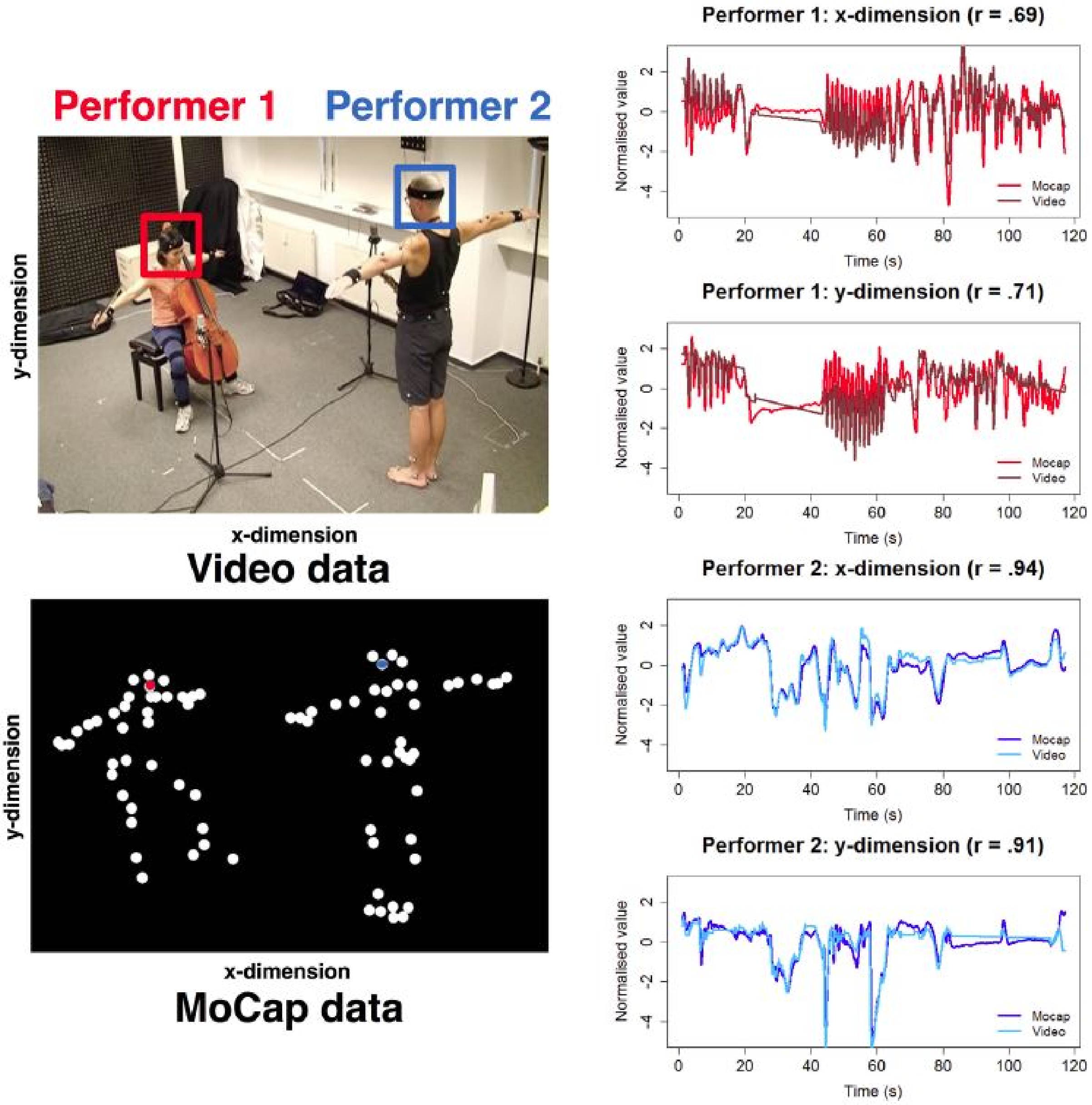
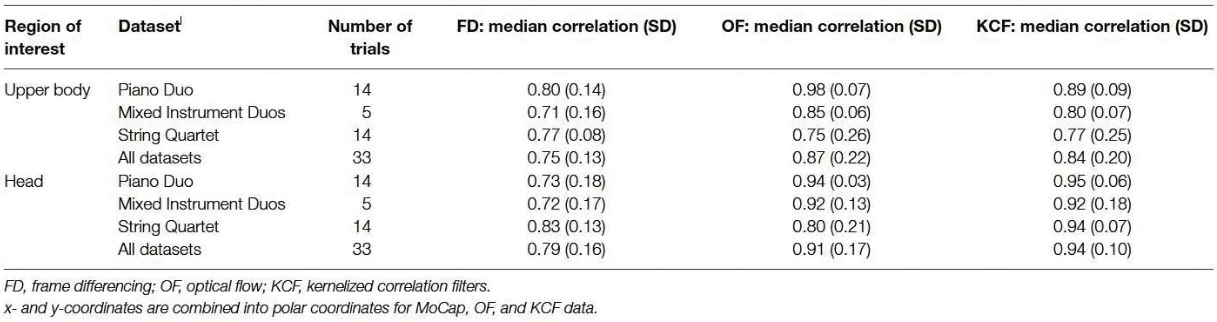


Figure 6.9: MoCap and video data comparison.

Duo, Mixed Instrument Duos, String Quartet) on accuracy of overall movement measurement (as indexed by the correlation of each video analysis output with the MoCap data; see Table 6.10).



Region of interest	Dataset	Number of trials	FD: median correlation (SD)	OF: median correlation (SD)	KCF: median correlation (SD)
Upper body	Piano Duo	14	0.80 (0.14)	0.98 (0.07)	0.89 (0.09)
	Mixed Instrument Duos	5	0.71 (0.16)	0.85 (0.06)	0.80 (0.07)
	String Quartet	14	0.77 (0.08)	0.75 (0.26)	0.77 (0.25)
	All datasets	33	0.75 (0.13)	0.87 (0.22)	0.84 (0.20)
Head	Piano Duo	14	0.73 (0.18)	0.94 (0.03)	0.95 (0.06)
	Mixed Instrument Duos	5	0.72 (0.17)	0.92 (0.13)	0.92 (0.18)
	String Quartet	14	0.83 (0.13)	0.80 (0.21)	0.94 (0.07)
	All datasets	33	0.79 (0.16)	0.91 (0.17)	0.94 (0.10)

FD, frame differencing; OF, optical flow; KCF, kernelized correlation filters.
x- and y-coordinates are combined into polar coordinates for MoCap, OF, and KCF data.

Figure 6.10: Median correlations between MoCap data and data extracted from the computer vision techniques.

6.3 Research Output

In this section is presented a tool bundle that has been initially developed to support the work presented in the previous section, and subsequently has been adapted to be used by non-IT researchers and permit them to take advantage of its functionalities.

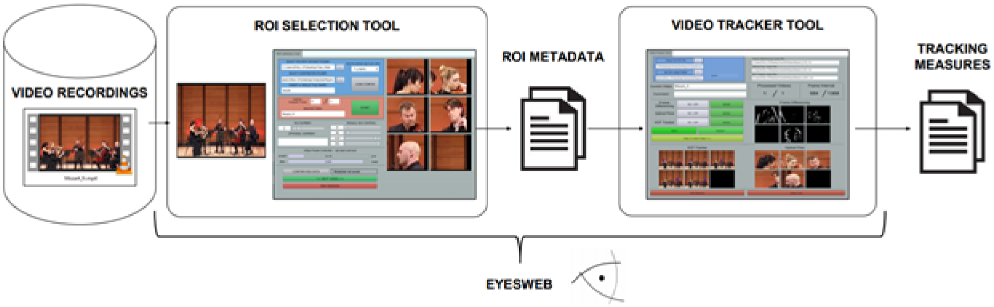


Figure 6.11: Overall bundle architecture. The bundle consists of two major components: the *ROI Selection Tool* enabling to define specific Region of Interest (ROIs) to be tracked and the *Video Tracking Tool*, which actually performs motion tracking.

Figure 6.11 displays the overall system architecture. It consists of two major tools.

The first one (*ROI selection*) takes as input a dataset of video recordings and allows the user to define one or more Regions Of Interest (ROIs) to be tracked in the input movies. A ROI is a specific rectangular region within a video frame containing individual people, parts of the body, or objects to be tracked. The ROI Selection Tool generates as output a file of metadata containing the information concerning the selected ROIs.

The second tool (*Video Tracking*) takes as input the ROIs metadata and the dataset of video recordings and tracks the selected ROIs, by automatically tuning and applying a selected motion tracking technique. The output of this tool is a file containing the obtained measures, e.g., the (x,y) coordinates of the baricenter of the tracked ROIs.

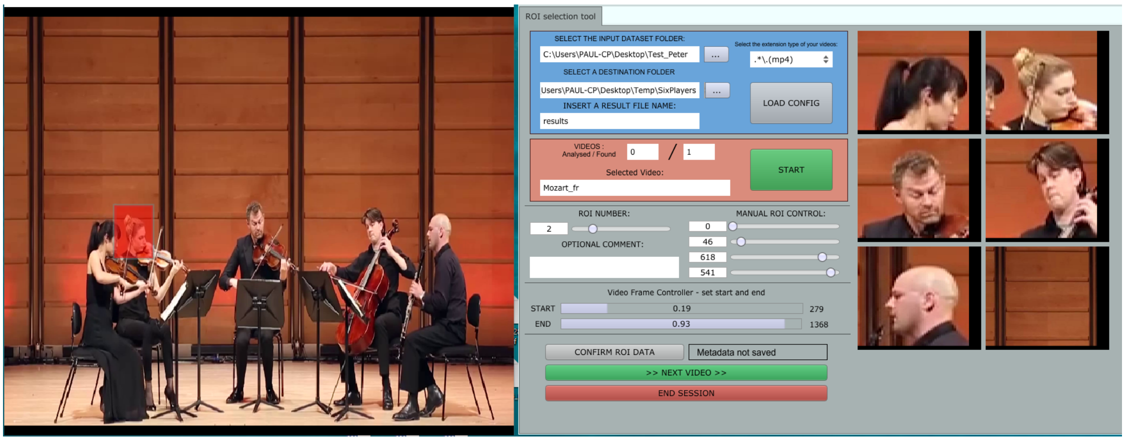
6.3.1 ROI Selection Tool

The user interface of the ROI Selection tool is shown in Figure 6.12. It enables several actions:

- Select a dataset of video recordings and navigate through it.
- Define one or more ROIs (maximum 6) for each video. The definition of each ROI can be performed in two different ways: (i) by moving a cursor over a series of sliders allowing the user to position and resize the rectangular area to be tracked, or (ii) more simply, by

using the mouse to draw a rectangular area directly on a frame taken from the video of interest.

- Check whether the defined ROIs, applied to the first frame of the video, contain the subject/object that should be tracked along the whole video. The ROI Selection Tool allows the user to check how the content of the ROIs evolves during the course of the whole video by moving a couple of sliders to set the boundaries (start and stop) of a specific video segment. While playing the selected fragment back, the interface shows each video frame together with the content of each ROI.


SELECT THE INPUT DATASET FOLDER: C:\Users\PAUL-CP\Desktop\Test_Peter

Select the extension type of your videos: .*\.(mp4)

SELECT A DESTINATION FOLDER: Users\PAUL-CP\Desktop\Temp\SixPlayers

INSERT A RESULT FILE NAME: results

VIDEOS : Analysed / Found 0 / 1

Selected Video: Mozart_fr

ROI NUMBER: 2

OPTIONAL COMMENT:

MANUAL ROI CONTROL: 0, 46, 618, 541

Video Frame Controller - set start and end

START 0.19 279

END 0.93 1368

Metadata not saved

Figure 6.12: The interface of the ROI Selection Tool.

- Extract, from each video, one or more video segments to be analyzed, avoiding possible non-significant segments. Indeed, it might not be necessary to analyze an entire video, e.g., an initial or ending part that is not interesting may be cut away. The ROI Selection Tool allows the user to define the boundaries (start and stop) of the video segments which she deems relevant for the analysis. By default, the starting frame is set to the first frame and the ending frame is set to the last frame of the video. It is possible to segment the same video in several different ways. Every time a new video is loaded, its length is computed and the start and ending frames are set to default.

- Export the metadata that contains all the information about the selected ROIs and video segments. The information stored in each metadata file is structured as depicted in Figure 6.13. For each video segment (possibly consisting of a whole video), a new row is inserted in the output file. Each row includes the number of selected ROIs, the position and size of each ROI, the absolute path of the video file containing the segment, the start and stop frames of the video segment, and an optional text comment (e.g., to remember the reasons for selecting a certain ROI or for performing a certain analysis).

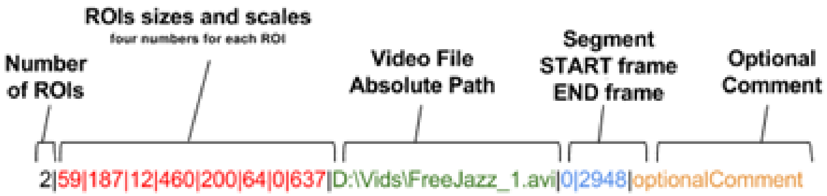
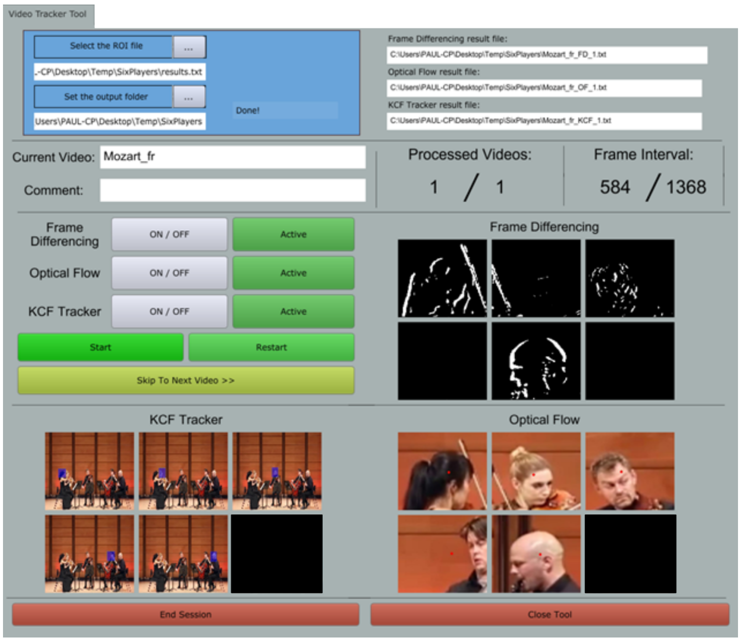

ROIs sizes and scales
four numbers for each ROI
Number of ROIs
Video File Absolute Path
Segment START frame END frame
Optional Comment
2|59|187|12|460|200|64|0|637|D:\Vids\FreeJazz_1.avi|0|2948|optionalComment

Figure 6.13: Representation of the metadata the ROI Selection Tool generates. This includes start and stop frames of the segment of video file to be analyzed and the position and size of each ROI to be tracked.

6.3.2 Video Tracking Tool

Figure 6.14 shows the user interface of the Video Tracking Tool. This tool takes as input a metadata file created by the ROI Selection Tool and applies different automated tracking techniques to each extracted segment and each defined ROI. The Video Tracking Tool allows the user to:

- Import and read ROI metadata files.



..-CP\Desktop\Temp\SixPlayers\results.txt

Users\PAUL-CP\Desktop\Temp\SixPlayers

Frame Differencing result file: C:\Users\PAUL-CP\Desktop\Temp\SixPlayers\Mozart_fr_FD_1.txt

Optical Flow result file: C:\Users\PAUL-CP\Desktop\Temp\SixPlayers\Mozart_fr_OF_1.txt

KCF Tracker result file: C:\Users\PAUL-CP\Desktop\Temp\SixPlayers\Mozart_fr_KCF_1.txt

Current Video: Mozart_fr

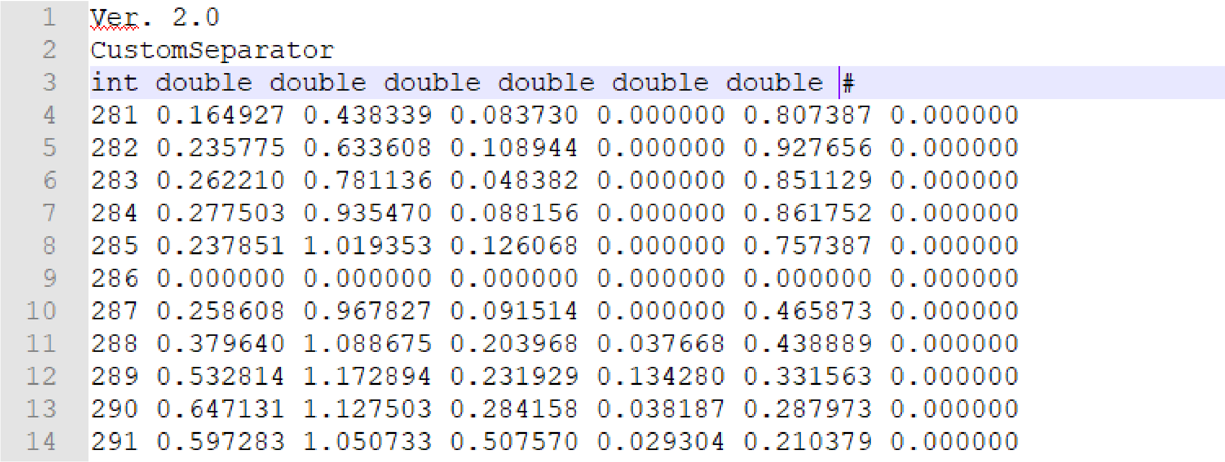
Comment:

Processed Videos: 1 / 1

Frame Interval: 584 / 1368

Figure 6.14: The interface of the Video Tracking Tool.

- Apply various automated tracking techniques to track the ROIs. Frame Differencing, Optical Flow and Kernelized Correlation Filter, already explained in the previous section, are the techniques currently available.



```
Ver. 2.0
CustomSeparator
int double double double double double double #
281 0.164927 0.438339 0.083730 0.000000 0.807387 0.000000
282 0.235775 0.633608 0.108944 0.000000 0.927656 0.000000
283 0.262210 0.781136 0.048382 0.000000 0.851129 0.000000
284 0.277503 0.935470 0.088156 0.000000 0.861752 0.000000
285 0.237851 1.019353 0.126068 0.000000 0.757387 0.000000
286 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
287 0.258608 0.967827 0.091514 0.000000 0.465873 0.000000
288 0.379640 1.088675 0.203968 0.037668 0.438889 0.000000
289 0.532814 1.172894 0.231929 0.134280 0.331563 0.000000
290 0.647131 1.127503 0.284158 0.038187 0.287973 0.000000
291 0.597283 1.050733 0.507570 0.029304 0.210379 0.000000
```

Figure 6.15: A fragment of an output file containing the data extracted by the Video Tracking Tool. This is made of a header and a data section. The first column in the data section reports the index of the video frame data refers to.

- Extract and make the tracking results available to the user for further analyses. This is done by storing the data in text files according to a specific format. Figure 6.15 shows an example of such a file. It consists of a header and of a data section. The first row of the header reports the version of the system, the second one a comment (e.g., useful to keep track of what the data refers to), the third one the datatype for each column in the data section (e.g., integer for the (x,y) position of a tracked target in image coordinates, and double for normalized coordinates or for normalized amount of motion). The data section is a matrix whose first column represents the index of the video frame data refers to and the other columns contain the actual data.

Chapter 7

Conclusions

Contents

7.0.1 Research results

The research presented in this thesis addressed the following problem: *can the automated analysis of synchronization support the study of expressive qualities of movement?* First of all, the state-of-the-art is presented. In particular, two baseline concepts are presented: the Laban Movement Analysis and the conceptual framework for the analysis of the expressive qualities of movement. An overview of the main techniques for the analysis of synchronization of multi-modal signals is then provided. Thereafter, the object of the analysis of synchronization, i.e., the expressive qualities of movement, are discussed in more detail. In particular it is explained how starting from a description of an expressive characteristic of the movement already founded in performing arts such as martial arts or dance (e.g., Fluidity), an experimental and computational model can be developed. This part of the thesis aims at informing whoever reads on how to deal with defining and evaluating expressive movement characteristics. The latter task is far from being simple since expressive qualities of movements are not easily quantifiable by virtue of their non-functional nature. To this end, a methodological approach to define such movement characteristics is presented, together with two case studies. Within the context of the adopted conceptual framework, synchronization is used as an operator (or analysis primitive) that can be applied to different typologies of signals, i.e., ranging from the direct measures of physical motions to semantic-based representations of expressivity features. In particular, there are two ways to investigate movements expressivity through synchronization analysis:

1. **The study of the synchronization of movements (and expressive movements) performed by a single user or between the different parts of his body.** In literature, the study of expressivity in non-verbal channels has often been focused on the individual. In the context of the individual, the analysis of synchronization can be applied and used to investigate different expressive qualities and this was one of the main results reported in the body of the thesis in the three case studies. The expressive qualities considered, evaluated in the movements carried out by various parts of the body, have different characteristics. Obtained results show how the analysis of synchronization can be applied to classify and automatically recognize such qualities, either offline (processing dataset of motion capture recordings) or real-time (for example during a dance choreography).

2. **The study of synchronization of movements (and expressive movements) between different individuals.** When referring to coordination and synchronization, it is more natural to imagine situations involving and studying several entities at the same time. Indeed, several works can be found in literature on the study of ecological data recorded by group of individuals. As a contribution to this typology of studies, a system for the automated extraction of movement features (from video recordings of music performances) and automated analysis of synchronization has been developed and tested in preliminary experiments.

In order to be able to deal with these typologies of problems and as contribution to the state-of-the-art of the available techniques, a new algorithm has been theorized and developed (the MECS algorithm, presented in Chapter 4), with the objective of being complementary and bridging the gaps of the other available algorithms belonging to the event-synchronization family.

7.0.2 Future prospectives

Human-machine interactions based on input commands and output feedbacks (as is the case with interfaces such as keyboard and mouse) are long established. One of the main objectives of HCI researchers is to make the interaction process more natural, as if the interaction were with another human. Then, computer interfaces should be no longer restricted to the use of the keyboard or mouse, but controlled also through words, facial expression and gestures. Those who have been using a computer for some time, can no longer realize that to interact with them, it is necessary to use a precise standard of communication between user and machine. At present, interfaces have evolved and continue to improve, but they still do not reach the levels of natural language. Techniques and systems presented in this thesis can contribute to the development of innovative interfaces as they can improve the ability of recognize expressive movements and behaviors, and consequently improve the overall quality of the interaction between human and machine.

Chapter 8

Appendix

Contents

8.1 List of publications

Conferences and Workshops:

1. **Alborno Paolo**, Gualtiero Volpe, Antonio Camurri, Martin Clayton, and Peter Keller. Automated video analysis of interpersonal entrainment in Indian music performance (2015, June). In Intelligent Technologies for Interactive Entertainment (INTETAIN), 2015 7th International Conference on (pp. 57-63). IEEE.

2. Ghisio Simone, Coletta Paolo, Piana Stefano, **Alborno Paolo**, Gualtiero Volpe, Camurri Antonio, Primavera Ludovica et al. (2015, June). An open platform for full body interactive sonification exergames. In Intelligent Technologies for Interactive Entertainment (INTETAIN), 2015 7th International Conference on (pp. 168-175). IEEE.

3. **Alborno Paolo**, Kolykhalova Ksenia, Frid Emma, Malafronte Damiano, Huis in 't Veld. L., Analysis of the qualities of human movement in individual action, in Proceedings of eNTERFACE 2015 Workshop.

4. Piana Stefano, **Alborno Paolo**, Niewiadomski Radoslaw, Mancini Maurizio, Volpe Gualtiero, and Camurri Antonio. "Movement Fluidity Analysis Based on Performance and Perception." In Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems, pp. 1629-1636. ACM, 2016.

5. **Alborno Paolo**, Piana Stefano, Mancini Maurizio, Niewiadomski Radoslaw, Volpe Gualtiero, and Antonio Camurri. ”Analysis of Intrapersonal Synchronization in Full-Body Movements Displaying Different Expressive Qualities” in Proceedings of the International Working Conference on Advanced Visual Interfaces. ACM, 2016.

6. Kolykhalova Ksenia, **Alborno Paolo**, Camurri Antonio, and Volpe Gualtiero. ”A serious games platform for validating sonification of human full-body movement qualities.” in Proceedings of the 3rd International Symposium on Movement and Computing, p. 39. ACM, 2016.

7. Volpe Gualtiero, **Alborno Paolo**, Camurri Antonio, Coletta Paolo, Ghisio Simone, Mancini Maurizio, Massari Alberto, Niewiadomski Radoslaw, Piana Stefano, Sagoleo Roberto “Designing Multimodal Interactive Systems Using EyesWeb XMI” in Smart Ecosystems creation by Visual design Workshop (SERVE 2016), 2016

8. **Alborno Paolo**, Cera Andrea, Piana Stefano, Canepa Corrado, Volpe Gualtiero, Camurri Antonio. “Interactive Sonification of Movement Qualities - a Case Study”, in Proceedings of ISon 2016, 5th Interactive Sonification Workshop, CITEC, Bielefeld University, Germany, December 16, 2016

9. Volpe Gualtiero, Kolykhalova Ksenia, Volta Erica, Ghisio Simone, Waddell George, **Alborno Paolo**, Piana Stefano, Canepa Corrado, Ramirez-Melendez Rafael “A multimodal corpus for technology-enhanced learning of violin playing” in Proceedings of the 12th Biannual Conference on Italian SIGCHI, Article No. 25.

10. **Alborno Paolo**, De Giorgis Nikolas, Camurri Antonio, and Puppo Enrico, “Limbs synchronisation as a measure of movement quality in karate”. In Proceedings of the 4th International Conference on Movement Computing (MOCO ’17), Kiona Niehaus (Ed.). ACM, NY USA, Article 29, 6 pages.

11. Niewiadomski Radoslaw, Mancini Maurizio, Piana Stefano, **Alborno Paolo**, Volpe Gualtiero, and Antonio Camurri, “Low-Intrusive Recognition of Expressive Movement Qualities.” In Proceedings of ICMI 2017, Glasgow,Scotland, 2017, 8 pages.DOI: 10.475/123 4.

12. Ghisio Simone, Volta Erica, **Alborno Paolo**, Gori Monica and Volpe Gualtiero “An open platform for full-body multisensory serious-games to teach geometry in primary school” in ICMI 2017- Proceedings of the 19th ACM International Conference on Multimodal Interaction.

13. Ghisio Simone, **Alborno Paolo**, Volta Erica, Gori Monica and Volpe Gualtiero “A multimodal serious-game to teach fractions in primary school”, in ICMI 2017- Proceedings of the 19th ACM International Conference on Multimodal Interaction.

14. Olugbade Temitayo, Cuturi Luigi, Cappagli Giulia, Volta Erica, **Alborno Paolo**, Newbold Joseph, Bianchi-Berthouze Nadia, Baud-Bovy Gabriel, Volpe Gualtiero and Gori Monica, “What Cognitive and Affective States Should Technology Monitor to Support Learning?”, in ICMI 2017- Proceedings of the 19th ACM International Conference on Multimodal Interaction.

Journals:

1. Frid Emma, Bresin Roberto, **Alborno Paolo** and Elblaus Ludvig. “Interactive Sonification of Spontaneous Movement of Children - Cross-modal Mapping and the Perception of Body Movement Qualities through Sound” in Frontiers in Neuroscience 10:521 · November 2016.

2. Niewiadomski Radoslaw, Kolykhalova Ksenia, Piana Stefano, **Alborno Paolo**, Volpe Gualtiero, Camurri Antonio 2017. “Analysis of Movement Quality in Full-Body Physical Activities”. ACM Transactions on Interactive Intelligent Systems. 1, 1, Article 1 (January 2017), 20 pages. DOI: 10.1145/3132369.

3. Kelly Jakubowski, Eerola Tuomas, **Alborno Paolo**, Volpe Gualtiero, Camurri Antonio and Clayton Martin, “Extracting Coarse Body Movements from Video in Music Performance: A Comparison of Automated Computer Vision Techniques with Motion Capture Data” in Frontiers in Digital Humanities, 10:3389, April 2017.
4. **Alborno Paolo**, Volpe Gualtiero, Mancini Maurizio, Niewiadomski Radoslaw, Piana Stefano, and Camurri Antonio, “The Multi-Event-Class Synchronization (MECS) Algorithm” submitted to IEEE Transactions on Human-Machine Systems, under review.

Posters:

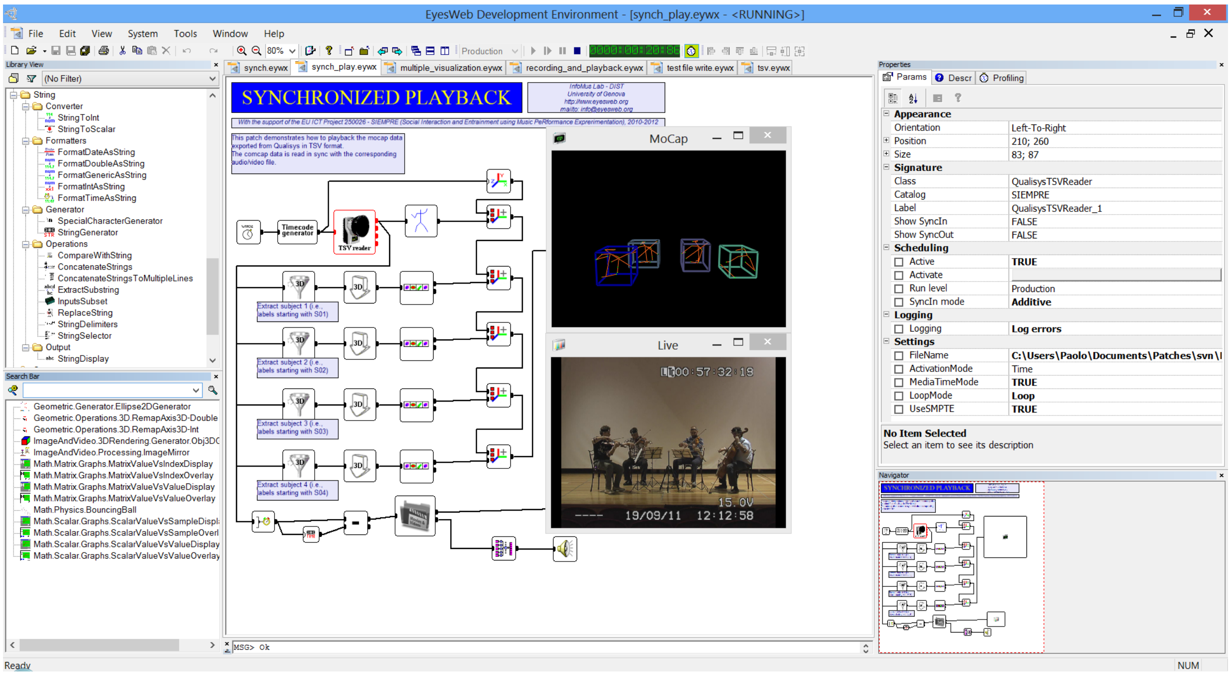
1. Volta Erica, **Alborno Paolo**, Piana Stefano, Volpe Gualtiero “Exploiting multisensory modalities for mathematics learning based on multimodal technology and serious games“, Germany 40th European Conference on Visual Perception ECVP 2017, 27–31 August 2017, Berlin.
2. Volta Erica, **Alborno Paolo** and Volpe Gualtiero “Informing bowing and violin learning using movement analysis and machine learning” at 10th International Workshop on Machine Learning and Music (MML) , October 6, 2017, Barcelona.

8.2 The EyesWeb XMI Platform

The EyesWeb XMI platform[1] (for eXtended Multimodal Interaction) is a tool for fast prototyping of multimodal systems, including interconnection of multiple smart devices, e.g., smartphones. EyesWeb is endowed with a visual programming language enabling users to compose modules into applications. Modules are collected in several libraries and include support of many input devices (e.g., video, audio, motion capture, accelerometers, and physiological sensors), output devices (e.g., video, audio, 2D and 3D graphics), and synchronized multimodal data processing. Specific libraries are devoted to real-time analysis of nonverbal expressive motor and social behavior. The EyesWeb platform encompasses further tools such EyesWeb Mobile supporting the development of customized Graphical User Interfaces for specific classes of users.

EyesWeb has a special focus on higher-level nonverbal communication, i.e., EyesWeb provides modules to automatically compute features describing the expressive, emotional, and affective content multi-modal signals convey, with particular reference to full-body movement and gesture. With respect to its previous versions, the current version of EyesWeb XMI encompasses enhanced synchronization mechanisms, improved management and analysis of time-series (e.g., with novel modules for analysis of synchronization and coupling), extended scripting capabilities (e.g., a module whose behavior can be controlled through Python scripts), and a reorganization of the EyesWeb libraries including novel supported I/O devices (e.g., Kinect V2) and modules for expressive gesture processing.

[1] Eyesweb is a non-profit open software platform developed by CasaPaganini InfoMus - DIBRIS - University of Genoa, Italy. For more information see: www.infomus.org/eyesweb_eng.php



EyesWeb Development Environment - [synch_play.eywx - <RUNNING>]

SYNCHRONIZED PLAYBACK

InfoMus Lab - DIST
University of Genova
http://www.eyesweb.org
mailto: info@eyesweb.org

With the support of the EU ICT Project 250026 - SIEMPRE (Social Interaction and Entrainment using Music PeRformance Exprerimentation), 2010-2012

This patch demonstrates how to playback the mocap data exported from Qualisys in TSV format.
The comcap data is read in sync with the corresponding audio/video file.

Extract subject 1 (i.e., labels starting with S01)

Extract subject 2 (i.e., labels starting with S02)

Extract subject 3 (i.e., labels starting with S03)

Extract subject 4 (i.e., labels starting with S04)

Appearance	
Orientation	Left-To-Right
Position	210; 260
Size	83; 87
Signature	
Class	QualisysTSVReader
Catalog	SIEMPRE
Label	QualisysTSVReader_1
Show SyncIn	FALSE
Show SyncOut	FALSE
Scheduling	
□ Active	**TRUE**
□ Activate	
□ Run level	Production
□ SyncIn mode	**Additive**
Logging	
□ Logging	**Log errors**
Settings	
□ FileName	**C:\Users\Paolo\Documents\Patches\svn\I**
□ ActivationMode	Time
□ MediaTimeMode	**TRUE**
□ LoopMode	**Loop**
□ UseSMPTE	**TRUE**

No Item Selected
Select an item to see its description

MSG> Ok

Ready

File
Edit
View
System
Tools
Window
Help
80%
Production
Library View
(No Filter)
String
Converter
StringToInt
StringToScalar
Formatters
FormatDateAsString
FormatDoubleAsString
FormatGenericAsString
FormatIntAsString
FormatTimeAsString
Generator
SpecialCharacterGenerator
StringGenerator
Operations
CompareWithString
ConcatenateStrings
ConcatenateStringsToMultipleLines
ExtractSubstring
InputsSubset
ReplaceString
StringDelimiters
StringSelector
Output
StringDisplay
Search Bar
Geometric.Generator.Ellipse2DGenerator
Geometric.Operations.3D.RemapAxis3D-Double
Geometric.Operations.3D.RemapAxis3D-Int
ImageAndVideo.3DRendering.Generator.Obj3DG
ImageAndVideo.Processing.ImageMirror
Math.Matrix.Graphs.MatrixValueVsIndexDisplay
Math.Matrix.Graphs.MatrixValueVsIndexOverlay
Math.Matrix.Graphs.MatrixValueVsValueDisplay
Math.Matrix.Graphs.MatrixValueVsValueOverlay
Math.Physics.BouncingBall
Math.Scalar.Graphs.ScalarValueVsSampleDispl
Math.Scalar.Graphs.ScalarValueVsSampleOverl
Math.Scalar.Graphs.ScalarValueVsValueDisplay
Math.Scalar.Graphs.ScalarValueVsValueOverlay
synch.eywx
synch_play.eywx
multiple_visualization.eywx
recording_and_playback.eywx
test file write.eywx
tsv.eywx
Timecode generator
TSV reader
3D
MoCap
Live
00:57:32:18
15.0V
19/09/11 12:12:58
Properties
Params
Descr
Profiling
Navigator
NUM

Figure 8.1: A view of the EyesWeb Graphic Development Environment (GDE), with a sample patch for synchronizing motion capture, audio, and video recordings of the music performance of a string quartet.

EyesWeb end users usually do not directly deal with the platform, but rather experience the multimodal interactive systems that are implemented using the platform. Their interface is therefore a natural interface they can operate by means, e.g., of their expressive movement and gesture to act upon multimedia content. In such a way, end users can also interact with smart devices they may wear or hold (e.g., smartphones). Smart devices can either be used to directly operate on content, or the data they collect can be presented back to end users, e.g., using data sonification and visualization technologies.

EyesWeb developers make applications (*patches*) using the EyesWeb GDE and its visual programming language. This implements all the basic constructs of programming languages such as sequences (a chain of interconnected blocks), conditional instructions (e.g., by means of switch blocks that direct the flow of data to a specific part of a patch when a given condition is matched), iterative instructions (by means of a specific mechanism that allows to execute a given sub-patch repetitively), and subprograms (implemented as sub-patches). Because of the visual programming paradigm, EyesWeb developers do not need to be computer scientists or expert programmers. In our experience, EyesWeb patches were developed by artists, technological staff of artists (e.g., sound technicians), designers, content creators, students in performing arts and digital humanities, and so on. Still, some skills in usage of computer tools and, especially, in algorithmic thinking are required. EyesWeb developers can exploit EyesWeb as a tool for fast-prototyping of applications for smart objects: EyesWeb can receive data from such objects by means of its input devices and the task of the developer is to design and implement in a patch the control flow of the application.

EyesWeb is nowadays employed by thousands of users spanning over several application domains. It was adopted in both research and industrial projects by research centers, universities, and companies. A one-week tutorial, the EyesWeb Week is organized every two years at our research center.

8.3 Movement and sound

In recent years, it became evident the need for research to address a thorough interaction and fruition of information across the auditory channel. Such interaction can be represented by the link between sound and movement. Both passive and active musical listening can induce movement: [JG03] has shown that music stimulates the activation of human brain regions interested in the production of movements. Authors in [GJ09] stated that music-related movements often match expressive and emotional features of a musical sound, for example, musical conductors guidelines state that legato should be evoked by smooth and connected gestures, while staccato by shorter and more jerky movements [Bla07]. For a better understanding of the topics discussed in this section, we need to illustrate the following notions:

- *Interactive Sonification*: the discipline of interactive representation of data (and data relationships) by means of sound.
- *Sound Model*: consists of set of equations and laws (possibly simplified) that govern the production of sound. Resulting sound is controlled by several parameters, some are constants and describe fixed characteristics of the desired output, others are functions depending on (one of more) temporal variables that describe the interaction between the user and the synthesized sound.
- *Mapping*: how input measures are mapped to sound model parameters in order to convey information through perceptually relevant acoustic features.

In this thesis, input is represented by human movement data. Then, the number of mappings that could be used within the context of sonification of human movement is potentially infinite (see e.g. [DB13] for an overview). However, only a small subset mappings will produce perceptually relevant results [RF14].

8.3.1 Sonification of movement qualities

Sonifications and auditory displays are increasingly becoming an established technology for exploring data, monitoring complex processes, or communicating information. Sonification addresses the auditory sense by transforming data into sound, allowing the human user to get valuable information from data by using their natural listening skills. This relationship between movement and sound becomes an integral part of interaction process and this has implications on modern humans-computers interfaces.

In this section, we propose a mapping between sound properties and movement qualities and serves as first investigation in a series of attempts aimed at finding perceptually relevant attributes of sound synthesis for sonification of human movement qualities. As mentioned in the previous section, the idea is to investigate which is the most suitable mapping to evoke (induce spontaneously) through the auditory channel the qualities of movement that are perceived by the visual channel. Possible mappings (or sonifications models) are potentially infinite, so we need to find a way to evaluate the effectiveness of the designed mappings. This is precisely the objective of the experiments presented in the next section.

The following topics will be discussed:

- how to design the best sonification model for movement Fluidity (which has already been introduced in Section 3.2.2.1).
- how to validate the designed sonification model.

The research question is the following: *is it possible to sonify a movement to enable a listener to perceive Fluidity even without seeing it?*

8.3.1.1 A scaled model for Fluidity

The adopted computational model for this study is a scaled version of the model presented in Section 3.2.2.1. A scaled model provides a rough and less precise Fluidity measure, but it has been developed mainly due to the following two reasons:

- the model should estimate Fluidity in real time (from IMUs (Inertial Measurement Units) data (the same sensors used in Section 3.2.3).
- the model should estimate Fluidity from a small number of less intrusive and less accurate sensors.

First of all, we compute the following low-level movement features:

Jerkiness: at frame f, hands linear accelerations (computed by subtracting to the data measured by the device the component corresponding to the gravity) $H^N_{lin_{x,y,x}}$ with $N \in [R, L]$ are used to calculate the squared jerkiness as:

$$J^N = (\dot{H}^N_{lin_x})^2 + (\dot{H}^N_{lin_y})^2 + (\dot{H}^N_{lin_z})^2 \tag{8.1}$$

Then, J^N is normalized over a buffer of 20 values:

$$J^N_{tot} = \frac{\sum_{i=1}^{20} J^N_i}{Max(J^N_i)} \tag{8.2}$$

Kinetic Energy: it is computed as the global kinetic energy of the wearer's hands, whose mass is approximated to 1 for sake of simplicity. Each velocity component is obtained by integrating the corresponding linear acceleration component:

$$E^N = 1/2\left[\int H^N_{lin_x})^2 + \int H^N_{lin_y})^2 + \int H^N_{lin_z})^2\right] \tag{8.3}$$

By integrating the linear acceleration components, we obtain the velocity components, necessary to compute kinetic energy. As before, we normalize the resulting value:

$$E^N_{tot} = \frac{E^N}{Max(E^N_i)} \tag{8.4}$$

Finally, we compute the user's movement Fluidity as:

$$F^N_{tot} = \frac{1}{J^N_{tot}/E^N_{tot}} \tag{8.5}$$

Fluidity Index FI is the mean of the Fluidity computed on the two hands:

$$FI = \frac{F^R_{tot} + F^L_{tot}}{2} \tag{8.6}$$

8.3.1.2 Fluidity sonification strategy

To identify the best approach to design the most effective sonification to convey Fluidity, we focused on two different sources of inspiration. On the one hand, we analyzed the state of the art in the expression of extra-musical qualities in sound design and electroacoustic music. Works of Wishart (Wishart, 1986), Tagg (Steedman, 1981), Middleton (Middleton, 1993), Kahn (Kahn, 1999) on cross-modality, studies by Carron (Carron, Rotureau, Dubois, Misdariis, Susini) on the sound designers' production techniques to convey specific extra-musical meaning provide a very useful and rich background of methodological guidelines for sonically rendering the Fluidity of a movement.

On the other hand, we took inspiration from cinematographic works. Sound design in cinematography can indeed provide a popular vocabulary, representing a largely shared way to associate sound and physical qualities (Hermann, Hunt, Neuhoff, 2011). We selected few sequences from very well know blockbusters such as The Matrix (Warner-Bros), The Fantastic Four (Marvel-Studios), Alien (Twentieth-Century-Fox-Film) in which movement Fluidity is clearly shown and sonically underlined: an example is given by the sequence in The Matrix movie (Warner-Bros) where the main character connects to the Matrix for the first time and his body becomes liquid. We empirically analyzed the sequences to find which sound cues and features are mainly used to induce the sensation of fluidity in the spectator.

On the basis of the described background, we then identified a set of common elements to *fluid* sounds: smooth attack and release curves, smooth dynamic profiles (i.e., no audible jumps in dynamics and no cuts), and smooth timbral evolution. In particular, the timbral content is close to the sound produced by flowing water, sounds audible underwater or sounds of bubbles. Often these sounds are more pitched than noisy, even if with non-harmonic relation between partials.

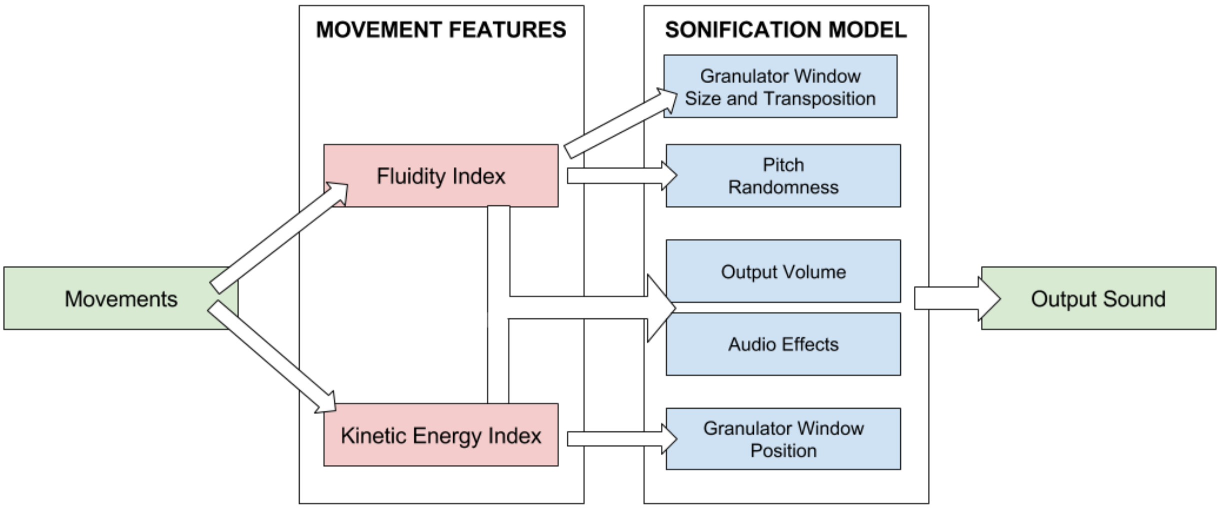


Figure 8.2: Schematic representation of the sonification system. The two features indexes (Kinetic Energy and Fluidity) control the sound parameters.

Our hypothesis is that Fluidity can be sonified using continuous sounds with a high value of spectral smoothness, evolving timbrically and dynamically with continuity, without audible steps. Sounds with low/medium spectral centroid are the first choice. On the contrary, sounds with a very high spectral centroid, even if continuous and smooth, may remind of non-fluid phenomena (i.e., friction, noise), and convey the impression of something moving against a resistance.

The model we propose is based on granular synthesis: it easily allows to change the basic sound materials (buffers), to obtain a wide timbral variety. The model is parametrized as follows:

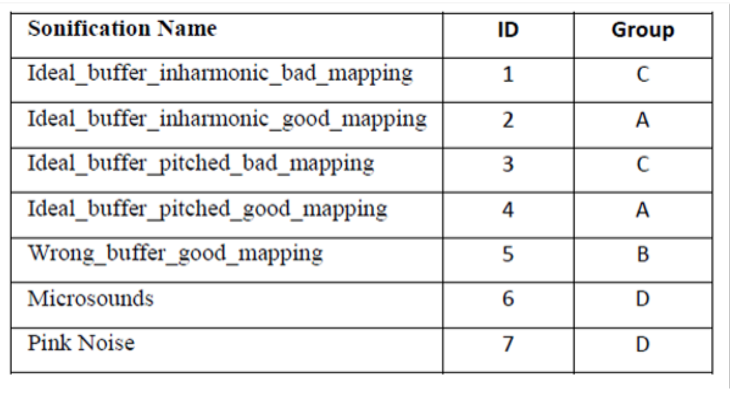
- Fluidity Index: to detect fluid movements, on a large temporal scale (1-3 seconds).
- Kinetic Energy Index: (the mean between E^R and E^L) to detect little, short and fast changes in the movements which otherwise would not have been considered by FI, potentially causing a lack of synchronization between the visual flow and the sonification.

Starting from the hypotheses described so far, we designed seven different sonifications and we tested them by mapping them to a short (30 seconds) recorded dance sequence in which the dancer was instructed to characterize her movements by high and continuous fluidity. Sonifications have been synthesized in order to assess several degrees of correspondence between the qualities of the dance and the sound. The granular synthesis model is implemented as MAX/MSP patches and uses a 20 seconds long buffer. The granulator window position is modulated by the value of the Kinetic Energy index. Fluidity Index is then used to make small displacements and to tune the window size. Finally, the combination of the two indexes controls the output volume. The audio synthesis process as shown in 8.2.

The mapping between the movement indexes and the granulator's parameters is designed to provide a smooth and fluid control of the model patch (low-pass filtering and temporal interpolations with ramps of at least 100 milliseconds duration). The "ideal" buffers for conveying Fluidity are based on either pitched and harmonic or inharmonic sounds. The frequency content of the buffers varies over time: in their initial portion (explored by the granulator's window when the energy of the movement is low) the buffers are fitted with a low centroid, and in the final portion (explored by the granulator's window when the energy of the movement is high) the centroid is higher.

Fluid characterized by low energy (slow, calm) are sonified with low centroid and dark timbre instead fluid movements characterized by high energy (fast, circular) are sonified with higher centroid and brighter timbre. High energy is translated to a richer sound in high frequencies, brighter, "energetic", and vice-versa. The final portion of the buffer is exploited for movements that are very energetic and presumably moving towards less-fluid qualities.

8.3.1.3 Sonifications



Sonification Name	ID	Group
Ideal_buffer_inharmonic_bad_mapping	1	C
Ideal_buffer_inharmonic_good_mapping	2	A
Ideal_buffer_pitched_bad_mapping	3	C
Ideal_buffer_pitched_good_mapping	4	A
Wrong_buffer_good_mapping	5	B
Microsounds	6	D
Pink Noise	7	D

Figure 8.3: Sonification table

We developed seven different sonifications: they differ in the type of buffer ("ideal" or "wrong /

contrasting") and type of mapping ("good" or "bad"); "ideal" buffers have been designed to comply with the description of fluid sounds given in the previous section. On the contrary, "wrong" buffers have been designed following the opposite criteria. "Good" mappings are characterized by smooth transitions and continuity while "bad" ones use steps and discontinuities. The seven sonifications belong to four different groups:

- Group A: Two sonifications generated using an "ideal" buffer and "good" mapping, identified by numbers 2 and 4. Sonifications belonging to Group A contain the patches designed to sonify Fluidity in the best possible way. In this group the buffers are based on a sound material characterized by absence of audible steps, in timbral and dynamic evolution and by high values of spectral smoothness and low centroid, according to the observation that fluid movements show no jerks, sudden stops or sudden changes of direction.

- Group B: A single sonification generated using a "ideal" buffer and "wrong" mapping, identified by number 5. Group B contains a single sonification. It uses the same mapping of Group A, but the buffer is designed in the opposite way: the average spectral smoothness value is low, showing a chaotic behavior and the centroid of the buffer is higher in the initial portion then it decreases.

- Group C: Two sonifications generated using an "ideal" buffer, but with "bad" mapping, numbers 1 and 3. Group C sonifications are based on the same buffer of Group A but make use of a non-fluid mapping characterized by a ten-step discretization of the indexes values controlling the granulator's parameters, furthermore interpolation ramps length is decreased from 10 to 5 milliseconds.

- Group D: Two control sonifications, identified by numbers 6 and 7. Group D contains two control sonifications, generated with a different synthesis technique (not based on granular synthesis):

  - Microsounds is designed to provide a sonic behavior deliberately contrasting with the idea of Fluidity. It is based on the superposition of four short loops made of percussive sounds (each loop is between 400 and 700 milliseconds). Kinetic Energy controls the speed of each loop, the playback rate and amplitude of the samples and the cutoff frequency of a low-pass filter. The result is a contrasted and irregular sonic material, which segments the sound continuity into a myriad of micro-events.
  - Pink Noise conveys a fluid, smooth timbral profile, realized with a simple technique, less evocative than granular synthesis. The sound is rendered by a pink noise generator and a low-pass filter. The cutoff frequency and the output volume are controlled by the energy while and the filter slope is piloted by the Fluidity index.

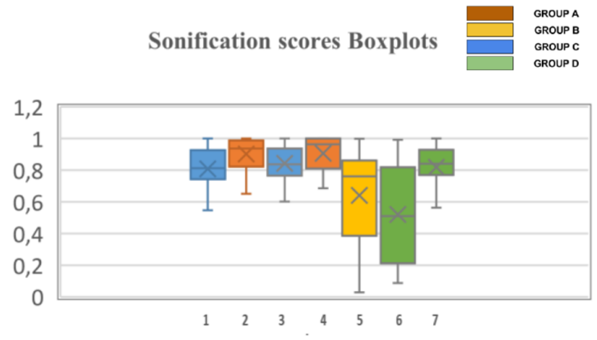


Figure 8.4: Sonification total scores on 22 players. Each column represents the sum of the Performance Indexes achieved by all the participants among the seven sonifications

8.3.1.4 Experimental setup

We developed a software platform to create and flexibly configure several serious games to evaluate how users “hear” a movement or a dance. The game platform and this specific instance were implemented in the EyesWeb XMI platform (illustrated in Section 8.2). A detailed description of the platform architecture can be found in [KACV16]). For this experiment, we generated an instance of this platform to validate the seven sonifications described in Section 3.

We carried out the experiment on a group of 22 adult participants. We placed two IMUs H^R and H^L, respectively, on the users’ right and left wrists. To extract the movement data we used the X-OSC sensors (X-IO Technology) that provide 9-axis inertial measurements of, respectively, the participant’s right and left hand.

To engage the users in the experiment and facilitate them in focusing on the task, we designed the experiment as a competitive game between two players/participants. An entire game session consisted of listening to seven sonifcations produced from a pre-recorded short dance performance (not visible to the participants). Sonifications are presented in a random order at each new session. While listening, each player was asked to move freely following what they listened to. Players were not aware of the origin of the sonic material they listened to. To avoid mutual influence between the players and to increase their sensitivity on the auditory perception they were blindfolded before starting the experiment.

Original IMU motion data (recorded during the dance performance) were used to generate the sonification, and, at the same time, each participant/player motion data were received from two IMU sensors on the wrists. Recorded dancer and the player motion data were both used to compute the value of the movement qualities described in Section 2.1, but only the dancer data were sonified. At every time instant, the Performance index is computed as:

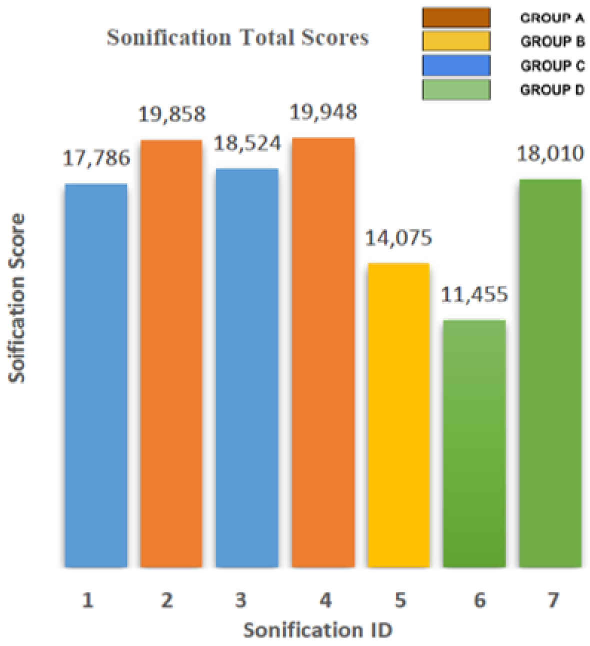


Figure 8.5: Sonifications total scores

$$PerformanceIndex_{pi} = \frac{F_{pi}}{F_d} \quad i \in [1,2] \tag{8.7}$$

where F_pi and $F_(d)$ are the player's and the dancer's Fluidity index respectively. This index is used to compute each player's game score.

The score reflects how much each player is able to "understand" the dancer movement's qualities (in this case Fluidity) through the sound. Our hypothesis is that sonifications following our model will communicate more efficiently to the players the original quality of movement, resulting in higher scores. Sonification scores are computed as the sum of all the Performance indexes of all players, for each one of the seven sonifications. To take into account the player subjective expressive style, sonification scores were normalized to the maximum score obtained by each player. The evaluation of the sonifications is based on the differences among the sonification scores given by the game to all players. The game's total score is calculated as the sum of each player's scores during a whole gaming session, but it remains relevant only to entertainment purposes i.e., to find out which of the two players has won the competition and it does not represent an interesting factor with regard to the validation of the sonifications.

8.3.1.5 Data Analysis

Sonification scores are computed as the sum of all the Performance indexes of all players, for each one of the seven sonifications. To take into account the player subjective expressive style, sonification scores were normalized to the maximum score obtained by each player. The evaluation of the sonifications is based on the differences among the sonification scores given by the game to all players. The game's total score is calculated as the sum of each player's scores during a whole gaming session, but it remains relevant only to entertainment purposes i.e., to find out which of the two players has won the competition and it does not represent an interesting factor with regard to the validation of the sonifications.

A statistical analysis on the sonification scores was performed, to confirm the hypothesis that sonifications following our model are the best candidate to convey Fluidity through the auditory channel.

To test our hypothesis a one-way repeated-measures ANOVA was conducted with one within-subject measure: Condition (1-7) as sum of the Performance Index for the different sonfiications as dependent value. Since Mauchly's Test of Sphericity indicated that the assumption of sphericity had been violated $(p < 0.005)$, Greenhouse Geisser correction was used ($\epsilon = .507$). Within-subject analysis showed a significant effect of Condition $F(3.041; 63.852) = 14.081$; $p < 0.001$ after application of Greenhouse-Geisser correction of Sphericity).

Next, the effect of Condition was analyzed using a post hoc tests with Bonferroni correction. Post hoc comparisons indicated that the Performance Index sum in Condition 6 was significantly lower than in Condition 2 $(p < 0.001)$, Condition 3 $(p < 0.005)$ and Condition 4 $(p < 0, 001)$. Moreover, comparisons indicated that the Performance Index sum in Condition 5 was significantly lower than in Condition 2 $(p < 0.005)$ and Condition 4 $(p < 0.005)$. In addition, sonification 6 showed a significant difference from sonification 7 $(p < 0.001)$.

Results suggest that sonifications in groups A and C better convey movement fluidity than sonifications in group B. In group D Microsounds, as expected, did not perform as good as Group A and C sonification. We did not observe the same behavior for the Pink Noise sonification.

8.3.1.6 Conclusions

The results from the experiment confirmed the validity of our sonification model as a promising starting point to convey Fluidity: sonifications in Group A obtained significantly better scores and seem to be more effective. As expected, the control sonification microsounds resulted less effective, followed by Group B (that is designed deliberately contrasting with the model). No statistically significant difference between Groups A, C and the control sonification Pink Noise was found.

Bibliography

[ADGCP17] Paolo Alborno, Nikolas De Giorgis, Antonio Camurri, and Enrico Puppo. Limbs synchronisation as a measure of movement quality in karate. In *Proceedings of the 4th International Conference on Movement Computing*, page 29. ACM, 2017.

[APM$^+$16] Paolo Alborno, Stefano Piana, Maurizio Mancini, Radoslaw Niewiadomski, Gualtiero Volpe, and Antonio Camurri. Analysis of intrapersonal synchronization in full-body movements displaying different expressive qualities. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, AVI '16, pages 136–143, New York, NY, USA, 2016. ACM.

[Arg13] Michael Argyle. *Bodily communication*. Routledge, 2013.

[BDB09] David S Bolme, Bruce A Draper, and J Ross Beveridge. Average of synthetic exact filters. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2105–2112. IEEE, 2009.

[Bis91] L Bishko. The use of laban-based analysis for the discussion of computer animation. In *The 3rd Annual Conference of the Society for Animation Studies*, 1991.

[BK93] Paolo Bernasconi and Jana Kohl. Analysis of co-ordination between breathing and exercise rhythms in man. *J. Physiol*, 471:693–706, 1993.

[Bla07] Alfred Blatter. *Revisiting music theory: a guide to the practice*. Taylor & Francis, 2007.

[BMB$^+$06] AH Bateman, AH McGregor, AMJ Bull, PMM Cashman, and RC Schroter. Assessment of the timing of respiration during rowing and its relationship to spinal kinematics. *Biology of Sport*, 23:353–365, 2006.

[BT13] Birgitta Burger and Petri Toiviainen. Mocap toolbox-a matlab toolbox for computational analysis of movement data. In *10th Sound and Music Computing Conference, SMC 2013, Stockholm, Sweden*. Logos Verlag Berlin, 2013.

[Cas98] Justine Cassell. A framework for gesture generation and interpretation. *Computer vision in human-machine interaction*, pages 191–215, 1998.

[CD01] Thierry Chaminade and Jean Decety. A common framework for perception and action: neuroimaging evidence. *Behavioral and Brain Sciences*, 24(5):879–882, 2001.

[CDT15] Baptiste Caramiaux, Marco Donnarumma, and Atau Tanaka. Understanding gesture expressivity through muscle sensing. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 21(6):31, 2015.

[Cla12] Martin Clayton. What is entrainment? definition and applications in musical research. *Empirical musicology review.*, 7(1-2):49–56, 2012.

[CMC$^+$08] Ginevra Castellano, Marcello Mortillaro, Antonio Camurri, Gualtiero Volpe, and Klaus Scherer. Automated analysis of body movement in emotionally expressive piano performances. *Music Perception: An Interdisciplinary Journal*, 26(2):103–119, 2008.

[CMR$^+$04] Antonio Camurri, Barbara Mazzarino, Matteo Ricchetti, Renee Timmers, and Gualtiero Volpe. Multimodal analysis of expressive gesture in music and dance performances. In *Gesture-based communication in human-computer interaction*, pages 20–39. Springer, 2004.

[CO66] William S Condon and William D Ogston. Sound film analysis of normal and pathological behavior patterns. *The Journal of nervous and mental disease*, 143(4):338–347, 1966.

[CPLV01] Antonio Camurri, Giovanni De Poli, Marc Leman, and Gualtiero Volpe. A multilayered conceptual framework for expressive gesture applications. In *In Proceedings of MOSART: Workshop on Current Directions in Computer Music*, pages 29–34, 2001.

[CRB$^+$07] George Caridakis, Amaryllis Raouzaiou, Elisabetta Bevacqua, Maurizio Mancini, Kostas Karpouzis, Lori Malatesta, and Catherine Pelachaud. Virtual agent multimodal mimicry of humans. *Language Resources and Evaluation*, 41(3):367–388, 2007.

[CSW05] Martin Clayton, Rebecca Sager, and Udo Will. In time with the music: the concept of entrainment and its significance for ethnomusicology. In *European meetings in ethnomusicology.*, volume 11, pages 1–82. Romanian Society for Ethnomusicology, 2005.

[CTV02] Antonio Camurri, Riccardo Trocca, and Gualtiero Volpe. Interactive systems design: A kansei-based approach. In *Proceedings of the 2002 Conference on New Interfaces for Musical Expression*, Proceedings of the International Conference on New Interfaces for Musical Expression (NIME02, pages 1–8. National University of Singapore, 2002.

[CVP$^+$16a] Antonio Camurri, Gualtiero Volpe, Stefano Piana, Maurizio Mancini, Radoslaw Niewiadomski, Nicola Ferrari, and Corrado Canepa. The dancer in the eye: towards a multi-layered computational framework of qualities in movement. In *Proceedings of the 3rd International Symposium on Movement and Computing*, page 6. ACM, 2016.

[CVP$^+$16b] Antonio Camurri, Gualtiero Volpe, Stefano Piana, Maurizio Mancini, Radoslaw Niewiadomski, Nicola Ferrari, and Corrado Canepa. The dancer in the eye: towards a multi-layered computational framework of qualities in movement. In *Proceedings of the 3rd International Symposium on Movement and Computing*, page 6. ACM, 2016.

[DB93] Jorge H. Daruna and Patricia A. Barnes. A neurodevelopmental view of impulsivity. In William G. McCown, Judith L. Johnson, and Myrna B. Shure, editors, *The impulsive client: Theory, research, and treatment*, pages 23–37. American Psychological Association, 1993.

[DB13] Gaël Dubus and Roberto Bresin. A systematic review of mapping strategies for the sonification of physical quantities. *PloS ONE*, 8(12):e82491, 2013.

[DBBL98] Georges Dalleau, Alain Belli, Muriel Bourdin, and Jean-René Lacour. The spring-mass model and the energy cost of treadmill running. *European journal of applied physiology and occupational physiology*, 77(3):257–263, 1998.

[DBS15] Simone Dalla Bella and Jakub Sowiński. Uncovering beat deafness: detecting rhythm disorders with synchronized finger tapping and perceptual timing tasks. *Journal of visualized experiments: JoVE*, (97), 2015.

[DCM$^+$12] Emilie Delaherche, Mohamed Chetouani, Ammar Mahdhaoui, Catherine Saint-Georges, Sylvie Viaux, and David Cohen. Interpersonal synchrony: A survey of evaluation methods across disciplines. *IEEE Transactions on Affective Computing*, 3(3):349–365, 2012.

[EF69] Paul Ekman and Wallace V Friesen. The repertoire of nonverbal behavior: Categories, origins, usage, and coding. *semiotica*, 1(1):49–98, 1969.

[ELZ00] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. In *Proceedings of the 41st Annual Symposium on Foundations of*

Computer Science, pages 454–, Washington, DC, USA, 2000. IEEE Computer Society.

[Far03] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. *Image analysis*, pages 363–370, 2003.

[FD16] Ken Fujiwara and Ikuo Daibo. Evaluating interpersonal synchrony: Wavelet transform toward an unstructured conversation. *Frontiers in psychology*, 7, 2016.

[GJ09] Rolf Inge Godøy and Alexander Refsum Jensenius. Body movement in music information retrieval. In *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*, pages 45–50, Kobe, Japan, 2009.

[GMC$^+$13] Donald Glowinski, Maurizio Mancini, Roddy Cowie, Antonio Camurri, Carlo Chiorri, and Cian Doherty. The movements made by performers in a skilled quartet: a distinctive pattern, and the function that it serves. *Frontiers in Psychology*, 4(841), 2013.

[GMM11] Olivier Girard, Jean-Paul Micallef, and Grégoire P. Millet. Changes in spring-mass model characteristics during repeated running sprints. *European journal of applied physiology*, 111(1):125–134, 2011.

[HCMB15] João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):583–596, 2015.

[HGP10] Hayley Hung and Daniel Gatica-Perez. Estimating cohesion in small groups using audio-visual nonverbal behavior. *IEEE Transactions on Multimedia*, 12(6):563–575, Oct 2010.

[HL05] Chi-Min Hsieh and Annie Luciani. Generating dance verbs and assisting computer choreography. In *Proceedings of the 13th Annual ACM international Conference on Multimedia*, pages 774–782. ACM, 2005.

[HM85] David Hary and George P Moore. Temporal tracking and synchronization strategies. *Human Neurobiology*, 4(2):73–79, 1985.

[HTB12] Charles P. Hoffmann, G?rald Torregrosa, and Beno?t G. Bardy. Sound stabilizes locomotor-respiratory coupling and reduces energy cost. *PLoS ONE*, 7(9), 09 2012.

[IBGM15] Johann Issartel, Thomas Bardainne, Philippe Gaillot, and Ludovic Marin. The relevance of the cross-wavelet transform in the analysis of human interaction–a tutorial. *Frontiers in psychology*, 5:1566, 2015.

[IR16] Tariq Iqbal and Laurel D. Riek. A method for automatic detection of psychomotor entrainment. *IEEE Transactions on Affective Computing*, 7(1):3–16, Jan 2016.

[JEA$^+$17] Kelly Jakubowski, Tuomas Eerola, Paolo Alborno, Gualtiero Volpe, Antonio Camurri, and Martin Clayton. extracting coarse body movements from video in music performance: a comparison of automated computer vision techniques with motion capture data. *Frontiers in Digital Humanities*, 4:9, 2017.

[JG03] Petr Janata and Scott T Grafton. Swinging in the brain: shared neural substrates for behaviors related to sequencing and music. *Nature Neuroscience*, 6(7):682–687, 2003.

[KACV16] Ksenia Kolykhalova, Paolo Alborno, Antonio Camurri, and Gualtiero Volpe. A serious games platform for validating sonification of human full-body movement qualities. In *Proceedings of the 3rd International Symposium on Movement and Computing*, MOCO '16, pages 39:1–39:5, New York, NY, USA, 2016. ACM.

[KBB13] Andrea Kleinsmith and Nadia Bianchi-Berthouze. Affective body expression perception and recognition: A survey. *IEEE Transactions on Affective Computing*, 4(1):15–33, 2013.

[KCA$^+$09] Thomas Kreuz, Daniel Chicharro, Ralph G Andrzejak, Julie S Haas, and Henry DI Abarbanel. Measuring multiple spike train synchrony. *Journal of neuroscience methods*, 183(2):287–299, 2009.

[KCH$^+$12] Thomas Kreuz, Daniel Chicharro, Conor Houghton, Ralph G Andrzejak, and Florian Mormann. Monitoring spike train synchrony. *Journal of neurophysiology*, 109(5):1457–1472, 2012.

[KCV$^+$15] Ksenia Kolykhalova, Antonio Camurri, Gualtiero Völpe, Marcello Sanguineti, Enrico Puppo, and Radosław Niewiadomski. A multimodal dataset for the analysis of movement qualities in karate martial art. In *Intelligent Technologies for Interactive Entertainment (INTETAIN), 2015 7th International Conference on*, pages 74–78. IEEE, 2015.

[Kel95] Dacher Keltner. Signs of appeasement: Evidence for the distinct displays of embarrassment, amusement, and shame. *Journal of Personality and Social Psychology*, 68:441–454, 1995.

[Ken04] Adam Kendon. *Gesture: Visible action as utterance*. Cambridge University Press, 2004.

[KH] G Kurtenbach and E Hulteen. The art of human computer interface design, chapter gestures in human. *Computer Communications*, page 309.

[KNH14] Peter E Keller, Giacomo Novembre, and Michael J Hove. Rhythm in joint action: psychological and neurophysiological mechanisms for real-time interpersonal coordination. *Phil. Trans. R. Soc. B*, 369(1658):20130394, 2014.

[Kre11] Thomas Kreuz. Measures of spike train synchrony. *Scholarpedia*, 6(10):11934, 2011.

[Kur75] Yoshiki Kuramoto. Self-entrainment of a population of coupled non-linear oscillators. In *International symposium on mathematical problems in theoretical physics*, pages 420–422. Springer, 1975.

[LDL$^+$09] Marc Leman, Michiel Demey, Micheline Lesaffre, Leon Van Noorden, and Dirk Moelants. Concepts, technology, and assessment of the social music game" sync-in-team'. In *Computational Science and Engineering, 2009. CSE'09. International Conference on*, volume 4, pages 837–842. IEEE, 2009.

[Lin94] Tony Lindeberg. *Scale-space theory in computer vision.* Kluwer Academic, Boston, 1994.

[LK$^+$81] Bruce D Lucas, Takeo Kanade, et al. An iterative image registration technique with an application to stereo vision. 1981.

[LL47a] Rudolf Laban and Frederick Charles Lawrence. *Effort*. Macdonald & Evans, 1947.

[LL47b] Rudolf Laban and Frederick Charles Lawrence. *Effort*. Macdonald & Evans, 1947.

[LMV$^+$11] Tamara Lorenz, Alexander Mörtl, Björn Vlaskamp, Anna Schubö, and Sandra Hirche. Synchronization in a goal-directed task: human movement coordination with each other and robotic partners. In *RO-MAN, 2011 IEEE*, pages 198–203. IEEE, 2011.

[LMV$^+$13] Marc Leman, Dirk Moelants, Matthias Varewyck, Frederik Styns, Leon van Noorden, and Jean-Pierre Martens. Activating and relaxing music entrains the speed of beat synchronized walking. *PloS one*, 8(7):e67932, 2013.

[LNVC16] Vincenzo Lussu, Radoslaw Niewiadomski, Gualtiero Volpe, and Antonio Camurri. Using the audio respiration signal for multimodal discrimination of expressive movement qualities. In Mohamed Chetouani, Jeffrey Cohn, and Albert Ali Salah, editors, *Human Behavior Understanding: 7th International Workshop, HBU 2016, Amsterdam, The Netherlands, October 16, 2016, Proceedings*, pages 102–115. Springer International Publishing, 2016.

[LS11] Daniel Lakens and Marielle Stel. If they move in sync, they must feel in sync: Movement synchrony leads to attributions of rapport and entitativity. *Social Cognition*, 29:1–14, 2011.

[LY94] Daniel TL Lee and Akio Yamamoto. Wavelet analysis: theory and applications. *Hewlett Packard journal*, 45:44–44, 1994.

[MCC$^+$80] John T. McConville, Charles E. Clauser, Thomas D. Churchill, Jaime Cuzzi, and Ints Kaleps. Anthropometric relationships of body and body segment moments of inertia. Technical report, DTIC Document, 1980.

[McN92] David McNeill. *Hand and mind: What gestures reveal about thought*. University of Chicago press, 1992.

[McN00] David McNeill. *Language and gesture*, volume 2. Cambridge University Press, 2000.

[Meh72] Albert Mehrabian. *Nonverbal communication*. Transaction Publishers, 1972.

[MHBK15] Nikki Moran, Lauren V Hadley, Maria Bader, and Peter E Keller. Perception of 'back-channeling'nonverbal feedback in musical duo improvisation. *PloS one*, 10(6):e0130070, 2015.

[Miy09] Yoshihiro Miyake. Interpersonal synchronization of body motion and the walk-mate walking support robot. *Robotics, IEEE Transactions on*, 25(3):638–644, 2009.

[MLV$^+$12] Alexander Mörtl, Tamara Lorenz, Björn NS Vlaskamp, Azwirman Gusrialdi, Anna Schubö, and Sandra Hirche. Modeling inter-human movement coordination: synchronization governs joint task dynamics. *Biological cybernetics*, pages 1–19, 2012.

[MNM09] Lynden K Miles, Louise K Nind, and C Neil Macrae. The rhythm of rapport: Interpersonal synchrony and social perception. *Journal of experimental social psychology*, 45(3):585–589, 2009.

[Mor81] Pietro Morasso. Spatial control of arm movements. *Experimental brain research*, 42(2):223–227, 1981.

[MSBC13] Cristiana Mercê, Cátia Santos, Marco Branco, and David Catela. Recurrence aanalysis of interpersonal synchronization in children during tap side of aerobics. 2013.

[NF03] Michael Neff and Eugene Fiume. Aesthetic edits for character animation. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '03, pages 239–244. Eurographics Association, 2003.

[NHP11] Radoslaw Niewiadomski, Sylwia Julia Hyniewska, and Catherine Pelachaud. Constraint-based model for synthesis of multimodal sequential expressions of emotions. *Affective Computing, IEEE Transactions on*, 2(3):134–146, 2011.

[NMP13] Radoslaw Niewiadomski, Maurizio Mancini, and Stefano Piana. Human and virtual agent expressive gesture quality analysis and synthesis. *Coverbal Synchrony in Human-Machine Interaction*, pages 269–292, 2013.

[NMP$^+$17] Radoslaw Niewiadomski, Maurizio Mancini, Stefano Piana, Paolo Alborno, Gualtiero Volpe, and Antonio Camurri. Low-intrusive recognition of expressive movement qualities. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pages 230–237. ACM, 2017.

[NMVC15] Radoslaw Niewiadomski, Maurizio Mancini, Gualtiero Volpe, and Antonio Camurri. Automated detection of impulsive movements in hci. In *Proceedings of the 11th Biannual Conference on Italian SIGCHI Chapter*, CHItaly 2015, pages 166–169, New York, NY, USA, 2015. ACM.

[NT98] Luciana P. Nedel and Daniel Thalmann. Real time muscle deformations using mass-spring systems. In *Computer Graphics International, 1998. Proceedings*, pages 156–165. IEEE, 1998.

[ODGJ$^+$08] Olivier Oullier, Gonzalo C De Guzman, Kelly J Jantzen, Julien Lagarde, and JA Scott Kelso. Social coordination dynamics: Measuring human bonding. *Social neuroscience*, 3(2):178–192, 2008.

[OMI$^+$12] Eisuke Ono, Masanari Motohasi, Yuki Inoue, Daisuke Ikari, and Yoshihiro Miyake. Relation between synchronization of head movements and degree of understanding on interpersonal communication. In *System Integration (SII), 2012 IEEE/SICE International Symposium on*, pages 912–915. IEEE, 2012.

[PAN$^+$16] Stefano Piana, Paolo Alborno, Radoslaw Niewiadomski, Maurizio Mancini, Gualtiero Volpe, and Antonio Camurri. Movement fluidity analysis based on performance and perception. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 1629–1636. ACM, 2016.

[PCG$^+$16] Stefano Piana, Paolo Coletta, Simone Ghisio, Radoslaw Niewiadomski, Maurizio Mancini, Roberto Sagoleo, Gualtiero Volpe, and Antonio Camurri. Towards a multimodal repository of expressive movement qualities in dance. In *3rd International Symposium on Movement and Computing, MOCO 2016*, 2016.

[PG04] Martin J Pickering and Simon Garrod. Toward a mechanistic psychology of dialogue. *Behavioral and brain sciences*, 27(2):169–190, 2004.

[PSAB10] Jessica Phillips-Silver, C Athena Aktipis, and Gregory A Bryant. The ecology of entrainment: Foundations of coordinated rhythmic movement. *Music Perception: An Interdisciplinary Journal*, 28(1):3–14, 2010.

[PSCO13] Stefano Piana, Alessandra Staglianò, Antonio Camurri, and Francesca Odone. A set of full-body movement features for emotion recognition to help children affected by autism spectrum condition. In *IDGEI International Workshop*, 2013.

[PSCO15] Stefano Piana, Alessandra Stagliano', Antonio Camurri, and Francesca Odone. Adaptive body gesture representation for automatic emotion recognition. In *Transactions on Interactive Intelligent System*, page in printing. ACM press, 2015.

[PSK12] Jessica Phillips-Silver and Peter E Keller. Searching for roots of entrainment and joint action in early musical interactions. *Frontiers in human neuroscience*, 6, 2012.

[QKG02] Rodrigo Quian Quiroga, Thomas Kreuz, and Peter Grassberger. Event synchronization: a simple and fast method to measure synchronicity and time delay patterns. *Physical review E*, 66(4):041904, 2002.

[RAF$^+$17] Veronica Romero, Joseph Amaral, Paula Fitzpatrick, RC Schmidt, Amie W Duncan, and Michael J Richardson. Can low-cost motion-tracking systems substitute a polhemus system when researching social motor coordination in children? *Behavior research methods*, 49(2):588–601, 2017.

[RDR$^+$11] Verónica C Ramenzoni, Tehran J Davis, Michael A Riley, Kevin Shockley, and Aimee A Baker. Joint action in a cooperative precision task: nested processes of intrapersonal and interpersonal coordination. *Experimental brain research*, 211(3-4):447–457, 2011.

[RF14] Stephen Roddy and Dermot Furlong. Embodied aesthetics in auditory display. *Organised Sound*, 19(01):70–77, 2014.

[RGF$^+$12] Michael J Richardson, Randi L Garcia, Till D Frank, Madison Gergor, and Kerry L Marsh. Measuring group synchrony: a cluster-phase method for analyzing multivariate movement time-series. *Frontiers in physiology*, 3, 2012.

[RKM11] David Reitter, Frank Keller, and Johanna D Moore. A computational cognitive model of syntactic priming. *Cognitive science*, 35(4):587–637, 2011.

[RP13] Luigi Rocca and Enrico Puppo. *A Virtually Continuous Representation of the Deep Structure of Scale-Space*, pages 522–531. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[RS13] Bruno H Repp and Yi-Huang Su. Sensorimotor synchronization: a review of recent research (2006–2012). *Psychonomic bulletin & review*, 20(3):403–452, 2013.

[SMBC$^+$14] Björn Schuller, Erik Marchi, Simon Baron-Cohen, Helen O'Rielly, Peter Robinson, Ian Davies, Ofer Golan, Shimrit Friedenson, Shahar Tal, Shai Newman, et al. ASC-inclusion: Integrated internet-based environment for social inclusion of children with autism spectrum conditions. *arXiv preprint arXiv:1403.5912*, 2014.

[SRD09] Kevin Shockley, Daniel C Richardson, and Rick Dale. Conversation and coordinative structures. *Topics in Cognitive Science*, 1(2):305–319, 2009.

[SS$^+$93] Steven H Strogatz, Ian Stewart, et al. Coupled oscillators and biological synchronization. *Scientific American*, 269(6):102–109, 1993.

[TEBN14] Nathan W Twyman, Aaron C Elkins, Judee K Burgoon, and Jay F Nunamaker. A rigidity detection system for automated credibility assessment. *Journal of Management Information Systems*, 31(1):173–202, 2014.

[VCCV08] Giovanna Varni, Antonio Camurri, Paolo Coletta, and Gualtiero Volpe. Emotional entrainment in music performance. In *Automatic Face & Gesture Recognition, 2008. FG'08. 8th IEEE International Conference on*, pages 1–5. IEEE, 2008.

[VDSK13] Maria Christine Van Der Steen and Peter E Keller. The adaptation and anticipation model (adam) of sensorimotor synchronization. *Frontiers in Human Neuroscience*, 7:253, 2013.

[VF95] Paolo Viviani and Tamar Flash. Minimum-jerk, two-thirds power law, and isochrony: converging approaches to movement planning. *Journal of Experimental Psychology: Human Perception and Performance*, 21(1):32, 1995.

[VOD10] Piercarlo Valdesolo, Jennifer Ouyang, and David DeSteno. The rhythm of joint action: Synchrony promotes cooperative ability. *Journal of Experimental Social Psychology*, 46(4):693–695, 2010.

[VPBP08] Alessandro Vinciarelli, Maja Pantic, Hervé Bourlard, and Alex Pentland. Social signals, their function, and automatic analysis: a survey. In *Proceedings of the 10th international conference on Multimodal interfaces*, pages 61–68. ACM, 2008.

[vULD$^+$08] Niek R van Ulzen, Claudine JC Lamoth, Andreas Daffertshofer, Gün R Semin, and Peter J Beek. Characteristics of instructed and uninstructed interpersonal coordination while walking side-by-side. *Neuroscience letters*, 432(2):88–93, 2008.

[VVC10] Giovanna Varni, Gualtiero Volpe, and Antonio Camurri. A system for real-time multimodal analysis of nonverbal affective social interaction in user-centric media. *IEEE Transactions on Multimedia*, 12(6):576–590, 2010.

[VVM11] Giovanna Varni, Gualtiero Volpe, and Barbara Mazzarino. Towards a social retrieval of music content. In *SocialCom/PASSAT*, pages 1466–1473, 2011.

[WADP97] Christopher Richard Wren, Ali Azarbayejani, Trevor Darrell, and Alex Paul Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7):780–785, 1997.

[WMS$^+$87] Rebecca M Warner, Daniel Malloy, Kathy Schneider, Russell Knoth, and Bruce Wilder. Rhythmic organization of social interaction and observer ratings of positive affect and involvement. *Journal of Nonverbal Behavior*, 11(2):57–74, 1987.

[WT09] Ben R. Whittington and Darryl G. Thelen. A simple mass-spring model with roller feet can induce the ground reactions observed in human walking. *Journal of biomechanical engineering*, 131(1):011013, 2009.

[WW04] Shu-fai Wong and Kwan-yee Kenneth Wong. Fast and reliable recognition of human motion from motion trajectories using wavelet analysis. In *Proc. the Symposium on Professional Practice in AI within the First IFIP Conference on Artificial Intelligence Applications and Innovations in IFIP World Computer Congress*, 2004.

[YF14] Omar Yahya and Miad Faezipour. Automatic detection and classification of acoustic breathing cycles. In *Proceedings of the 2014 Zone 1 Conference of the American Society for Engineering Education*, pages 1–5, April 2014.

[YTY02] Tomoyoshi Yoshida, Shoichi Takeda, and Sayoko Yamamoto. The application of entrainment to musical ensembles. 2002.

[YWS12] Kyongsik Yun, Katsumi Watanabe, and Shinsuke Shimojo. Interpersonal body and neural synchronization as a marker of implicit social interaction. *Scientific reports*, 2:959, 2012.