WHOLODANCE

Whole-Body Interaction Learning for Dance Education

Call identifier: H2020-ICT-2015 - **Grant agreement no**: 688865

Topic: ICT-20-2015 - Technologies for better human learning and teaching

Deliverable 3.6

First report on software platform and libraries

Due date of delivery: March 31st, 2017

Actual submission date: April 3rd, 2017

Start of the project: 1st January 2016
Ending Date: 31st December 2018

Partner responsible for this deliverable: POLIMI

Version: 4.0

Dissemination Level: Public

Document Classification

Title	First Report on software platform and libraries
Deliverable	3.6
Reporting Period	1st
Authors	Massimiliano Zanoni, Michele Buccoli, Bruno Di Giorgi, Augusto Sarti, Fabio Antonacci, Paolo Alborno, Antonio Camurri, Nikolas De Giorgi, Ksenia Kolykhalova, Stefano Piana, Enrico Puppo
Work Package	WP3
Security	Public
Nature	Report
Keyword(s)	Software platform, software libraries, Eyesweb, Unity, Java script

Document History

Name	Remark	Version	Date
Massimiliano Zanoni	TOC and objectives	0.1	20/02/2017
Michele Buccoli	Add description of low-level feature analysis procedure and overview of the framework for audio analysis.	0.2	08/03/2017
Bruno di Giorgi	Add description of mid-level feature analysis procedure	0.3	10/03/2017
Massimiliano Zanoni	Compilation of Objects	0.4	08/03/2017
Paolo Alborno, Antonio Camurri, Nikolas De Giorgi, Ksenia Kolykhalova, Stefano Piana, Enrico Puppo	Add UNIGE software modules	1.0	28/03/2017

List of Contributors

Name	Affiliation
Massimiliano Zanoni	Polimi
Bruno Di Giorgi	Polimi

List of reviewers

Name	Affiliation
Michele Buccoli	Polimi
Augusto Sarti	Polimi
Fabio Antonacci	Polimi
Vladimir Viro	Peachnote
Antonella Trezzani	Lynkeus

1 Executive Summary

This Deliverable will be based on the outcomes of the task "T3.4. Development platform and software libraries" and aims at presenting a first collection of software libraries and applications developed by the partners within the WholoDance project. Since dance movement and behaviour analysis is preparatory for other tools, the Deliverable will be mainly focused on the description of libraries voted to the analysis of information related to dance performances.

Dance practice is multimodal by nature. For this reason, the development of software and tools for multimodal data analysis is the ground for other movement and behaviour analysis tools. The Deliverable includes *SW library for emotional and expressive analysis from musical signals (T3.4.1), SW Library for Emotion analysis from full-body movement and multimodal data (T3.4.2)* and *Multimodal analysis of qualities in individual dance (T3.4.4).*

Dancing can be an individual or social experience. This is the case when more than one dancer interact on the stage. The Deliverable includes *Software Library for Non-Verbal Social Signal Analysis (T3.4.3)*.

Table of Contents

2 Objectives

Main goals of WhoLoDancE project is the creation of interactive and immersive learning tools for different dance styles and appropriate for different teaching and learning modalities. The aforementioned tools include solutions for supporting the composition and distribution of educational content and services with assessment functionalities making use of real-time feedback mechanisms for dance movements and choreographies learning. All the tools are based on solid movement/behaviour analysis. The deliverable describes the first collection of software libraries and software applications for movement/behaviour analysis.

Dance practice presents a wide diversity across genres and contexts. WhoLoDancE project focuses on investigating different dance genres, learning principles and learning scenarios. In order to take into account different learning scenarios, various and appropriate technological solutions and analysis methodologies should be developed. Given the complexity of the global scenario, in this first stage of the project, the partners adopted a set of different programming languages, frameworks and tools, in order to find the best technological solution for each task. Programming languages used to develop the software and tools are: C++, Python, Matlab, as well as web-based languages such as Javascript. Frameworks and tools, instead, are: Eyesweb and Unity.

In a second stage of the project all the technological solutions will converge to build a unique integrated system. The Deliverable D3.7 will focus on the description of the integrated system.

Through interviews to dance students and choreographers the consortium defined a set of learning scenarios to be considered within the project. An exhaustive description of the scenarios is included in *D7.1 Usability and Learning Experience Evaluation report* and spans from scenarios where the teacher shows to students a movement to mimic, to scenarios where students mimics a pre-recorded set of movements. In the latter, a repository and an off-line analysis system of pre-recorder movements are needed. In both cases, real-time systems that provides feedback on the quality of the performance are required. Adopted learning scenarios require the use of different technologies for data acquisition. Some examples are high-end quality engine for 3d motion capture (mocap), low-end quality engines (Kinects, accelerometer sensors, etc.), video and music.

In sections 3.2.2 and 3.2.3 software libraries for off-line expressive and human behaviour measurement through multimodal analysis are described. In order to access and visualize recordings from different media of the same performance and to build an effective vocabulary of movements, automatic data synchronization and segmentation algorithms are needed. In section 3.2.1 a method for synchronize video and motion capture data is presented. Whereas, in section 5.2 an algorithm for automatic movement segmentation is described.

Assessment of a performance requires the real-time measuring of a set of qualities by means of a set of wearable sensors. In section 4 we describe the developed tools for the real-time analysis of movement on the Laban's cube.

Dancing can be an individual or a social experience. In the social experience, more than one dancer is involved in the performance. The learning of the interaction between dancers on the stage is part of the teaching process. The automatic real-time or off-line analysis techniques of this interaction is described in section 4 where Synchronization and Dynamic Symmetry are considered.

In several dance styles, music plays an important role in the performance. This is the case of Greek folk dances, Ballet, Flamenco, which are styles considered in the project among the others. For this reason, joint dance-music analysis algorithms are mandatory in WhoLoDancE project. Software framework developed within the project for dance-music analysis is presented in Section 3.1.

3 Software and libraries for multimodal data analysis

In this section we describe the software and libraries used for processing and feature extraction from multimodal data. This is related to the Task 3.1.3 “Joint music-dance representation models”, dealing with the analysis of the relations and dependencies between descriptors of music and dance. In Sect. 4.1, we describe the features and software used to analyse musical audio signals. The corresponding description for movement and multimodal signal will be found in Sect. 4.2.

3.1 SW library for emotional and expressive analysis from musical signals

The emotional and expressive analysis of music involves social information on the listener and the modelling of human intellectual perception, i.e., from acoustic stimuli to higher-level information. It is possible to extract semantic information on the emotional content or expressivity of the musical performance by means of machine learning techniques trained over a set of annotated examples. This approach, however, requires an intermediate representation of the musical signal, that captures the most relevant properties of the musical content. These properties involve three main levels of information: the physical level, which is related to the origin and propagation of the audio wave and can be analysed with signal processing techniques; the musical level that describes aspects from musical theory such as the rhythm and the harmony; and the perceptual level, which concerns how the human body and brain perceive and process the audio information.

The Music Information Retrieval (MIR) community developed and designed several techniques to describe the musical signal by means of automatic extraction of *features*, which capture different aspects of the sound at various levels of abstraction, from those related to the energy, the spectrum or the timbre, to those regarding musical aspects of the music performance. The features are traditionally classified into two levels of abstraction: Low and Middle Level. Low-Level Features (LLFs) capture the properties concerning the physical and perceptual level of information by using a clear mathematical formulation. They are extremely objective, but they can only be interpreted by researchers and provide little insight of the musical content. Mid-Level Features (MLFs), instead, capture the musical properties by taking into consideration prior knowledge from musical theory, such as the notion of notes, chords and harmony, or information about beats, tempo and rhythmic patterns. The mid-level representation is closer to the concepts musicians and composers use to understand music. LLF and MLF are directly computed on the musical signal. In Figure 1 we show a representation of the features from the two levels of abstraction, i.e., those that are directly computed from the signal. A further level, composed of the descriptors for the expressive and emotional analysis, can be then computed on LLF and MLF by means of machine learning techniques.

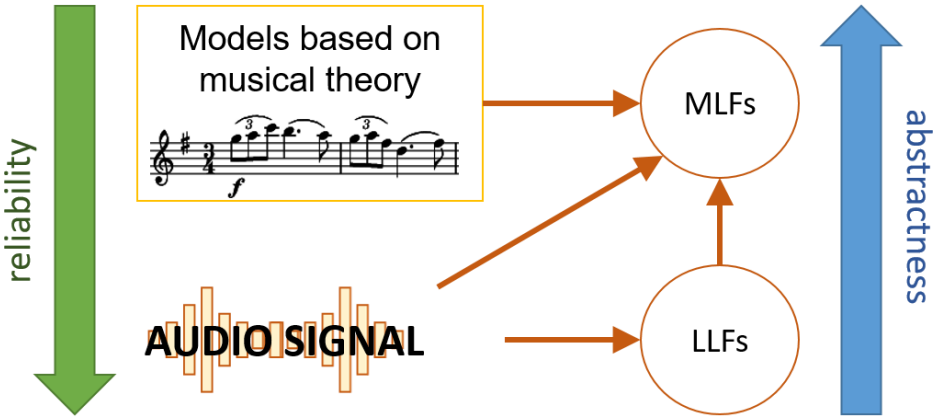


Figure 1 Schema of the features extracted from the musical signal

Several software tools have been developed for the task of automatic extraction of features. The *MIRToolbox* is a library, developed in the Finnish Centre of Excellence in Interdisciplinary Music Research, that allows to extract and manipulate features in the MATLAB environment. Also, Python language has received great attention by the MIR community, which leads to the tool *LibROSA* which are used for generic feature extraction and manipulation. Finally, the batch tool *Sonic Annotator*, developed in the Queen Mary University of London, is widely employed within the MIR community by applying the VAMP plugins, which are also supported by other applications such as Audacity or Sonic Visualiser.

For the sake of the WhoLoDancE project, the partners from Polimi developed a Python-based framework able to interface with the aforementioned tools. A simplified visualization of the proposed framework is shown in Figure 2.

The framework interfaces with the LibROSA Python library and it implements a parser to import the features extracted by means of the VAMP plugins and MirToolbox. Some feature extraction techniques are also implemented directly in the framework, which is the reason why the framework includes the Input/Output module for the analysis and possible playing of audio signals. Users can interact with the framework to provide annotations or visualize features and results by means of a web-based interface. The framework also communicates with other systems developed within the project by means of TCP or Open Sound Control (OSC) connections. OSC is a protocol for networking sound that is used to deliver data, including series of features, and it is widely supported by many tools, such as the EyesWeb platform developed by UniGe and the Unity application under development by Motek. The TCP connections are used over the internet to load data from or store preliminary results in the CKAN repository set up by Athena.

The analysis and processing of the extracted features leads to the possible generation of more abstract descriptors, which are then all fed to machine learning algorithms. These techniques require annotations that are under collection by means of a web-based interface, which will interact with the CKAN repository. The generated expressive or emotional descriptors will be transmitted to EyesWeb or Unity applications for multimodal analysis.

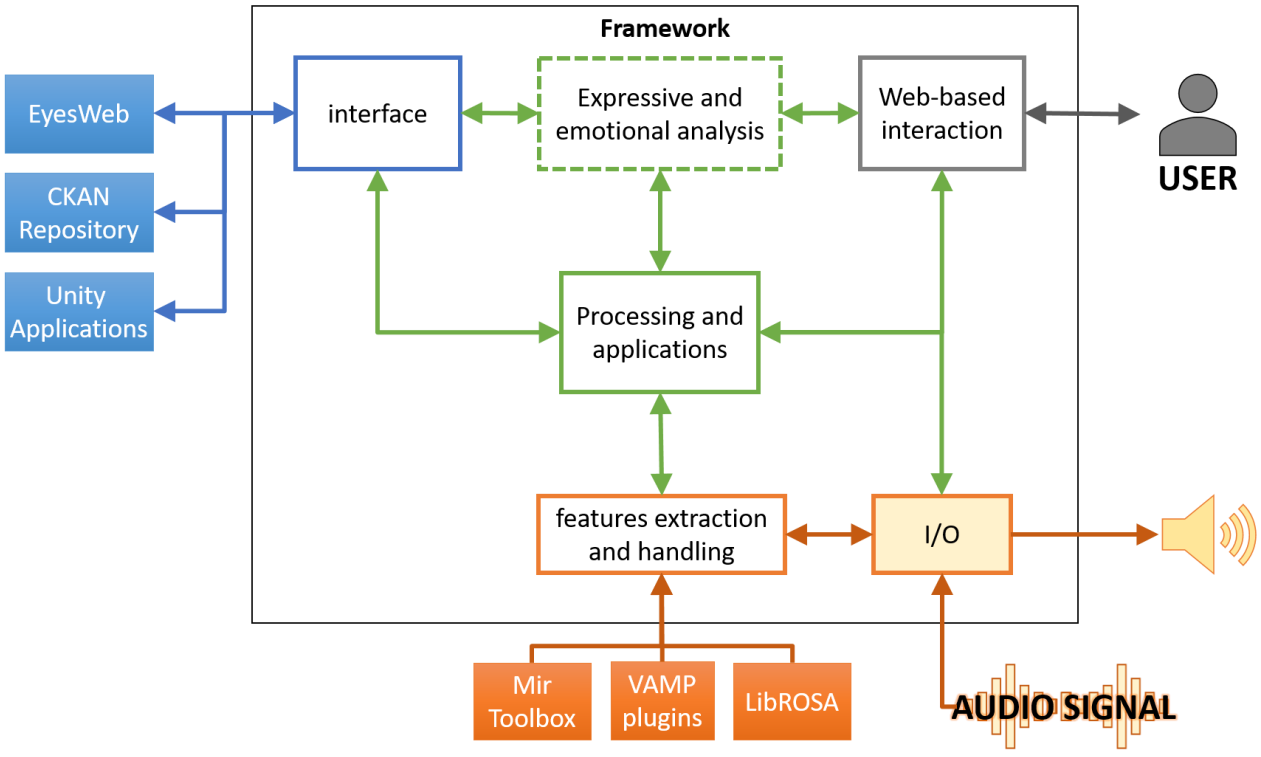


Figure 2 Schema of the Python-based framework for the sake of the project

In the following sections, we briefly present the most widely used features within the project. A more comprehensive list of the features available for the different tools is provided in Table 1.

3.1.1 Low-Level Features

LLFs include descriptors directly extracted from the signal with a clear mathematical formulation. The basic approach is to analyse the time-domain representation of the signal, e.g., for the *Root Mean Square Energy* (RMSE) or the *Zero Crossing Rate* (ZCR). The RMSE is a measure of the energy of the signal, it is linked with its amplitude and with the *Loudness*, which estimates the perception of the energy. ZCR estimates the frequency at which the signal crosses the zero axes and it is used as an estimator of the noisiness of the signal. Other time-domain features are related to the amplitude envelope of the signal, from which some information on the sound quality (the timbre) are inferred. The envelope of a note is often modelled by means of four phases: Attack, Decay, Sustain, Release. Without going into the details, these phases are analysed by means of features such as the *Attack Time*, *Attack Duration*, *Attack Slope*, etc.

Since the human auditory system mainly focus on frequency content of sounds, the frequency-domain representations (e.g. spectrum and spectrogram) of the signal is mostly used. From these representations, so-called spectral features are extracted. Some example are the statistical moments of the signal that are computed from the distribution of the magnitude of the spectrum over the frequency bins (*Spectral Centroid*, *Spectral Spread*, *Spectral Skewness* and *Spectral Kurtosis*). The *Spectral Centroid* represents the "centre of gravity" (first moment) of the magnitude spectrum, i.e., the frequency at which the energy of the spectrum is more concentrated. The *Spectral Centroid* is used as a descriptor of the brightness of the sound: the higher the *Centroid*, the brighter the sound and vice-versa. The *Spectral Spread* measures the standard deviation of the spectrum from its frequency mean, i.e., from the *Spectral Centroid*. The *Spread* estimates the noisiness of the sound, since the lower it is, the more the spectrum is distributed around its centroid and hence, the more it resembles a pure tone. The *Spectral Skewness* is the third statistical moment and it computes the coefficients of the skewness of the frequency distribution, i.e., the degree of its symmetry around the *Spectral Centroid*; the *Spectral Kurtosis* is the fourth statistical moment and it indicates the resemblance of the spectral shape with a Gaussian bell curve. Other spectral features considered in the project are *Spectral Entropy*, *Flatness*, *Roll-Off*, *Inharmonicity*, *Brightness*, *Contrast*, *Roughness*.

Some spectral features are modelled by taking into account the perceptual qualities of sounds. Such a perceptual representation is built by exploiting information on how the human auditory system works. From the studies on psychoacoustics, it is known that the human perception of frequency is logarithmic, as well as the perception of loudness. The decomposition of a sound in the correspondent frequency components is performed within the spiral-shaped cavity in the inner ear named *cochlea*, which acts as a filter bank for different ranges of frequencies. The *Mel-Frequency Cepstral Coefficients* (MFCCs) are a set of low-level features that exploit a model from psychoacoustics on the human auditory system to provide perceptual cepstral features. Due to their ability to provide a compact representation of the distribution of the energy values of the spectrum, they have also been widely employed to capture properties of the timbre and therefore they are also referred to as *timbral descriptors*. The MFCCs are computed by applying a mel-based filter bank whose bands model the auditory response of the cochlea. The mel scale follows the logarithmic perception of the frequencies, so there are more filter banks in the lower frequencies than in the higher ones. Then, for each band of the mel-filtered spectrum, the log-energy is computed, where the logarithmic scale takes the human perception of loudness into consideration. Finally, the Discrete Cosine Transform (DCT) is applied for the extraction of a complete yet compact representation of the log-energy, commonly with only 13 coefficients per frame.

LLFs are designed and employed to extract a high number of properties of the signal domain. The musical content can be further analysed by considering not only the time-domain or frequency-domain representation, but also their interpretation considering the musical theory, as done by MLFs.

3.1.2 Mid-Level Features

Mid-level features (MLFs) describe music using music theory concepts, such as harmony, melody and rhythm. Several algorithms have been recently developed at Polimi that deal with the extraction of harmonic and rhythmic descriptors, as shown in Figure 3. A brief description of these methods is provided in the next sections.

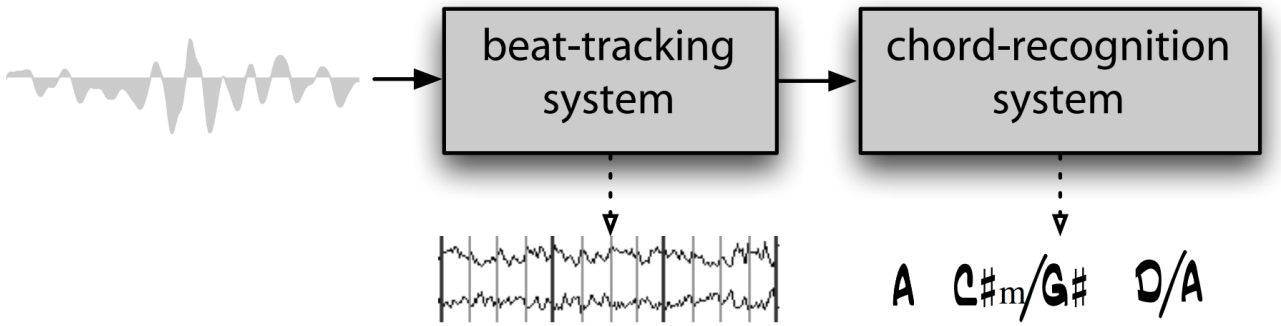


Figure 3 A visual representation of the extraction of the Mid-Level Features from musical signals

Beat tracking

Most musical pieces evolve according to an underlying unit of time, sometimes implied and sometimes clearly audible, called the beat. Beat trackers are essential components of rhythmic analysis systems in a wide range of applications of musical information extraction. Beat instants, in fact, can be used for music segmentation and processing (e.g., structural segmentation, interactive accompaniment, cover song detection, music similarity, score following, etc.); for beat-synchronous analysis (e.g., drum transcription; chord extraction); and more. Beat is also one of the most relevant features for joint music-movement analysis since for some dance genres the sequence of beats is used as a grid on which to lay out the choreographies and many dance movements visually highlight the beat to show synchronicity with the background musical piece. In this project we focus on automatic extraction of the beats from audio signal, starting from a LLF called Onset Detection Function (ODF) that highlights the transients of the signal. The musical transients are the moments when for example a new note is played, which usually happens with the beat.

The problem of beat tracking can be solved by first analysing the periodicities of the ODF with the purpose of finding an initial estimate of the Inter-Beat Interval (IBI), which is generally time-varying. The IBI is usually also called *tempo* and can be expressed in beats per minute. Successively, a dynamic programming algorithm can efficiently find the sequence of beats through a joint optimization process that jointly maximizes the ODF values of the beat instants and match the estimated IBI.

We focused on this last joint optimization process and proposed a novel strategy based on an efficient generation and joint steering of multiple trackers (paths) [Di Giorgi, 2016]. This solution leads to improved computational efficiency with respect to dynamic programming methods; a computational improvement factor of 2 has been obtained with a negligible accuracy loss.

For the implementation of the beat tracking algorithm we used the Python libraries Numpy, Scipy and Cython. The core of our algorithm is implemented in C++ and available at the following link: http://ispg.deib.polimi.it/private/multipath.zip .

Chord and key recognition

Chords and keys are among the most exhaustive descriptors of songs. Chord sequences, in particular, provide a simple and effective representation of the harmony. Among the many applications of chord recognition there are cover identification, interactive music training and automated transcription. In the first part of the project we focused on automatic chord and key recognition. The notions of chord and key are well known aspects of the music theory. Specifically, a chord is a set of harmonically related musical pitches (notes) that sound almost simultaneously. The key can be divided in the *key root*, also called *tonic*, that is defined as the most important pitch class and the *key mode* that is the subset of pitch classes used in a song, relatively to the key root (e.g., major or *Ionian*, minor or *Aeolian*), . The basic feature used for the analysis is the chromagram, also called pitch class profiles (PCPs), a time-varying 12-dimensional signal obtained from the spectrogram of the audio signal.

The probabilistic formulation for the automatic chord recognition system is modelled using the Dynamic Bayesian Network (DBN), a graphical model that generalizes the Hidden Markov Model (HMM), managing graphs with any number of hidden and observed nodes. In this model, nodes can represent observed random variables, in our case the chromagram, or hidden random variables that can be inferred given the values of the observed variables, in our case the chord, key and bass note. Similarly to HMM, also DBN explicitly model time dependencies; in particular, this is achieved through a set of transition probability distributions, which model the probability of a particular variable, conditioned on the values of a set of related variables at the previous time frame. A natural discretization of the time dimension is obtained using beat instants, which are estimated using our beat extraction algorithm; this process is called beat-synchronization.

Differently from other methods in the literature, which only considers the major and minor modes, we can extract a richer description of key by including two diatonic modes not previously considered, i.e., the *Mixolidian* and the *Dorian*. This allows to provide a more complete and expressive model of the harmonic induced mood, which is more suitable for tonal music languages typically used as the background music in Flamenco and ballet dance genres.

In order to achieve this goal, we proposed a set of new parametric distributions. Comparing with similar chord recognition algorithms we found a significant advantage in using such a larger state space for the key variable. Finally, exploiting this rich description of key and chords, we proposed three simple harmonic features that relate chord sequences to perceived emotion.

For the implementation of the dynamic Bayesian network model we used the library Bayes Net Toolbox written in Matlab.

Table 1 List of the main features extracted by the considered tools.

Type	Name	Description	Tools
LLFs	Root Mean Square Energy (RMSE)	An indicator of the energy of the audio signal in a frame from time-domain or frequency-domain representation.	Librosa, MirToolbox, VAMP
LLFs	Zero Crossing Rate	Rate with which a frame of an audio signal crosses the zero; it is an estimation of the noisiness of the signal.	Librosa, MirToolbox
LLFs	Amplitude Envelope	The maximum values of amplitude for the time-domain representation of the signal.	MirToolbox

LLFs	(Log) Attack Time	The duration of the attach phase of a note onset; it provides information on the timbre.	MirToolbox, VAMP
LLFs	Attack Slope	The slope of the attack phase; it is another way to estimate the attack.	MirToolbox
LLFs	Attack Leap	The amplitude difference between the beginning and the end of the attack phase.	MirToolbox
LLFs	Decrease Slope	The slope of the deceased phase.	MirToolbox
LLFs	Onset Duration	Duration of a note, from the attack to the release phase.	MirToolbox
LLFs	Onset Duration	Duration of a note, from the attack to the release phase.	MirToolbox
LLFs	Spectral Centroid	The center of energy of the distribution of the spectrum over the frequency bins.	Librosa, VAMP
LLFs	Spectral Spread, Skewness, Kurtosis, Bandwidth	Several statistical moments applied to the distribution of the magnitude of the spectrum over the frequencies.	Librosa, VAMP
LLFs	Spectral Flatness	The degree of resemblance between the distribution of the magnitude of the spectrum and a flat spectrum; it is an estimator of the noisiness of the sound.	MirToolbox
LLFs	Spectral Entropy	The amount of information encoded in the distribution of the magnitude of the spectrum; it is another estimator for the noisiness.	MirToolbox
LLFs	Spectral Contrast	Difference between peaks and valleys in the magnitude spectrum distribution.	Librosa, VAMP
LLFs	Spectral Roll-off	The frequency below which the 85% of total energy is contained.	Librosa, MirToolbox, VAMP
LLFs	Spectral Brigthness	The amount of energy above a certain frequency (given as a parameter).	MirToolbox
LLFs	Spectral Roughness	An estimation of the roughness depending on the frequency ration of pair of sinusoids closed in frequency (beating phenomenon).	MirToolbox
LLFs	Spectral Inharmonicity	Amount of energy outside the ideal harmonic series (fundamental frequency and its harmonics).	MirToolbox
LLFs	Spectral Irregularity	Degree of variation of two successive peaks in the spectrum.	MirToolbox
LLFs	Polynomial Features	The coefficients of a polynomial fitting over the distribution of the magnitude spectrum.	Librosa
LLFs	Fluctuations	Spectrum computed over the different frequency bins of the real spectrum; it estimates rhythmic periodicities.	MirToolbox
LLFs	Mel Spectrogram	The spectrogram mapped in the mel scale.	Librosa

LLFs	Mel-Frequency Cepstral Coefficients (MFCCs)	The coefficients computed from the mel-spectrogram by means of the DCT.	Librosa, MirToolbox VAMP
LLFs	Similarity Matrix	Self-similarity among the representation of different frames of the signal	MirToolbox
MLFs	Chromagram	Distribution of the pitches over time, usually regardless to the original octave (pitch class profile).	Librosa, MirToolbox, VAMP
MLFs	Tonal centroid (Tonnetz)	Projection of the chords along circles of fifths, of minor thirds and of major thirds.	Librosa, MirToolbox
MLFs	Tempogram	Local autocorrelation of the onset strength envelope, to estimate the distribution of onsets periodicities.	Librosa, VAMP
MLFs	Beats (beat tracking)	Instants when the musical beats occur.	Librosa, VAMP
MLFs	Tempo	The speed of a performance in beats per minute.	Librosa, MirToolbox
MLFs	Note onset	Instants where a musical onset occurs.	MirToolbox, VAMP
MLFs	Event Density (Onset Rate)	Number of note onsets per second.	MirToolbox
MLFs	Metre	Hierarchical metrical structure of the onsets.	MirToolbox
MLFs	Metroid	Centroid of metre.	MirToolbox
MLFs	Pulse Clarity	Strength of the estimated beats.	MirToolbox
MLFs	Pitch	Pitch curves or note events of a melody.	MirToolbox, VAMP
MLFs	Key	Harmonic key.	MirToolbox, VAMP
MLFs	Mode	Modality of the key, i.e., major vs minor.	MirToolbox, VAMP
MLFs	Chords	The chords played during a musical piece.	VAMP

3.2 SW Library for Emotion analysis from full-body movement and multimodal data

3.2.1 Synchronization between Video and motion Capture

During the initial phases of the WhoLodancE project a large dataset of recordings has been created, consisting of different formats and streams, such as motion-capture, video and audio. The large size of the dataset, which counts approximately 18 hours of recorded material, highlights the need of automating the processes of matching and synchronizing data streams from different formats and modalities. Such techniques could be used as pre-processing step for analysis algorithms that use multimodal data, for example the joint music-

motion analysis. Polimi and UniGe developed two different methods and tools to manage the synchronization between different input data streams.

A technique for synchronizing video and motion capture data has been developed at Polimi, using LLFs extracted from the two streams. The method extracts the optical flow of the video data in the horizontal and vertical directions. Similar information is extracted from the motion capture by differentiating successive positions of all the joints. The offset between the two 2d streams is obtained as the lag that maximizes the cross-correlation between such streams, and the maximum value is used as an indicator of the reliability of the estimate. A visual representation of the approach is shown in Figure 4, where it is clear the correlation between the couples of streams (optical flow in the horizontal and vertical directions). This technique has been shown to achieve an accuracy of 97% from our experiments on the Greek and Flamenco performances, which include 230 video and motion-capture sequences.

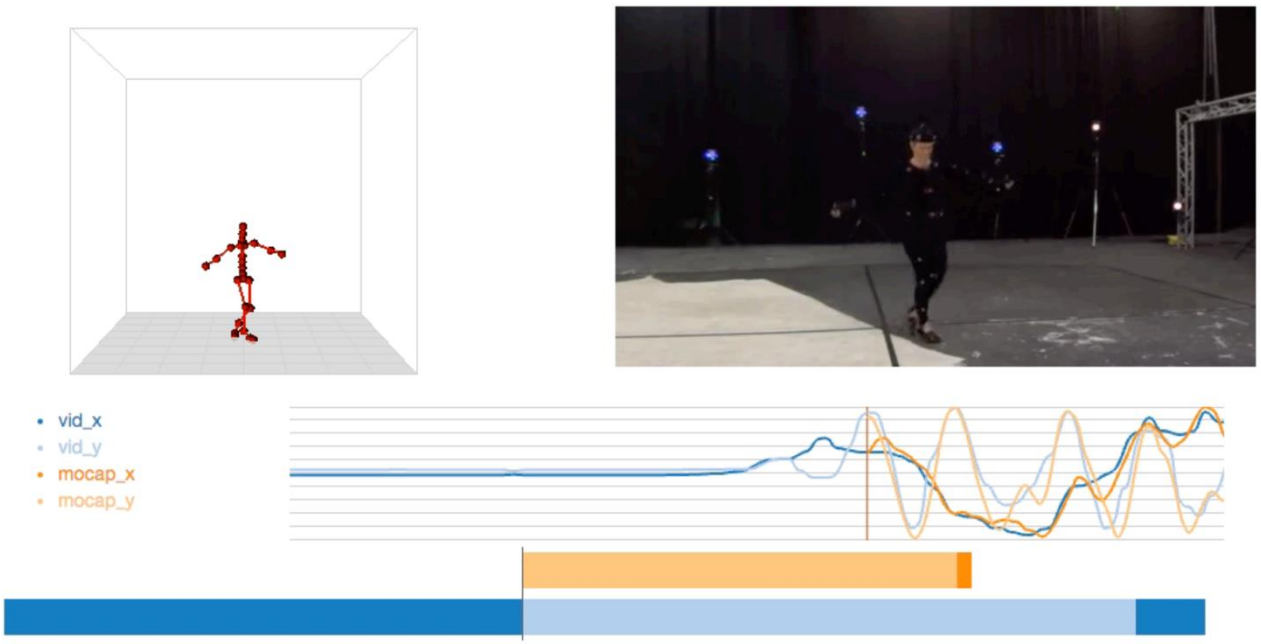


Figure 4 An example of the mocap-video synchronization, with the optical flow in the horizontal (x) and vertical (y) directions of the two streams

Multimodal Synchronization, based on SMPTE, has been developed by UniGe by exploiting the EyesWeb platform. In the recording system, EyesWeb is used to generate the reference SMPTE clock used by all other EyesWeb recording modules. The SMPTE is shared by the recording modules of all multimodal channels (a Qualisys Motion Capture system, two broadcast video-cameras, IMU sensors, and a professional microphone) to obtain a fine-grained multimodal synchronization.

3.2.2 Expressive features

In this section, we provide an introduction to the expressive features and propose a few introductory examples of feature extraction algorithms we are developing for specific movement analysis in the context of the project.

In WhoLoDancE, the extraction of expressive features is carried out according to the multi-layered conceptual framework developed by Camurri and colleagues [Camurri, Antonio, et al. 2016]. Existing techniques for the automated analysis of non-verbal expressive movements have been extended according to specific needs of WhoLoDancE, i.e., investigating qualities of movement and dance principles in the dance learning process faced in the first phase of the project, focusing on “space” and “orientation”.

Movement expressive features are ranging from physical signals to high-level qualities and address several aspects of movement analysis with different spatial and temporal scales. Features computed at lower layers contribute to the computation of features at higher levels, which are related to more abstract concepts. In more details, the movement analysis is organized in the following layers:

- **points: physical data** that can be detected by exploiting sensors in real-time (for example, position/orientation of the body joints);
- **frames: physical and sensorial data**, not subject to interpretation, detected uniquely starting from instantaneous physical data on the shortest time span needed for their definition and depending on the characteristics of the human sensorial channels;
- **qualities: perceptual features, interpretable and predictable**, with a given error and correction, starting from different constellations of physical and sensory data, on the time span needed for their perception;
- **affects: perceptual and contextual features, interpretable and predictable**, with a given error and correction, starting from a narration of different qualities, on a large time span needed for their introjection.

3.2.3 Graph-restricted game approach for investigating human behavior

A novel computational method for the analysis of expressive full-body movement qualities is implemented in Matlab and Eyesweb as a software module included in the WhoLoDancE software library, which exploits concepts and tools from graph theory and game theory. Our method is based on the idea that “important” joints during a specific movement are those that separate parts of the body characterized by different motion behaviors. For instance, if an arm is moving and the corresponding shoulder and the other parts of the body are still, then that shoulder may be considered as an “important” joint because, in a certain sense, it “controls” the motion of the arm, and is a point where movement propagation between joints is interrupted.

The human skeletal structure is modeled as an *undirected graph*, where the joints are the vertices and the edge set contains both *physical* and *nonphysical* links. *Physical* links correspond to connections between adjacent physical body joints (e.g., the forearm, which connects the elbow to the wrist). *Nonphysical* links act as “bridges” between parts of the body not directly connected by the skeletal structure, but sharing very similar feature values. The edge weights depend on features obtained by using Motion Capture data. Then, a *game* theory approach is constructed over the graph structure [Myerson, 1977], where the vertices represent the players and the edges represent communication channels between them. Hence, the body movement is modeled in terms of a game built on the graph structure. Since the vertices and the edges contribute to the overall quality of the movement, the adopted game-theoretical model is of cooperative nature. A game-theoretical concept, called Shapley value [Tijs, 2003], is exploited as a centrality index to estimate the contribution of each vertex to a shared goal (e.g., to the way a particular movement quality is transferred among the vertices). We use Shapley value to rank the joints [Michalak et al., 2013]. The “most important” joint in each specific frame is defined as the one with the largest rank, if there is only one joint with that property. The proposed method is applied to a data set of Motion Capture data of subjects performing expressive movements, recorded in the framework of the WhoLoDancE project.

For each frame, we compute the Shapley values of the 20 joints (reduced number of joints computed from the position of 64 mocap markers), using the method described above. Then, we extract the “most important” joint, according to the proposed method. Then we visualize them, highlighting in red, frame-by-frame, the most important joint extracted by the method inside the 20-markers skeletal graph (see Figure 4 for an example of visualization of the results).

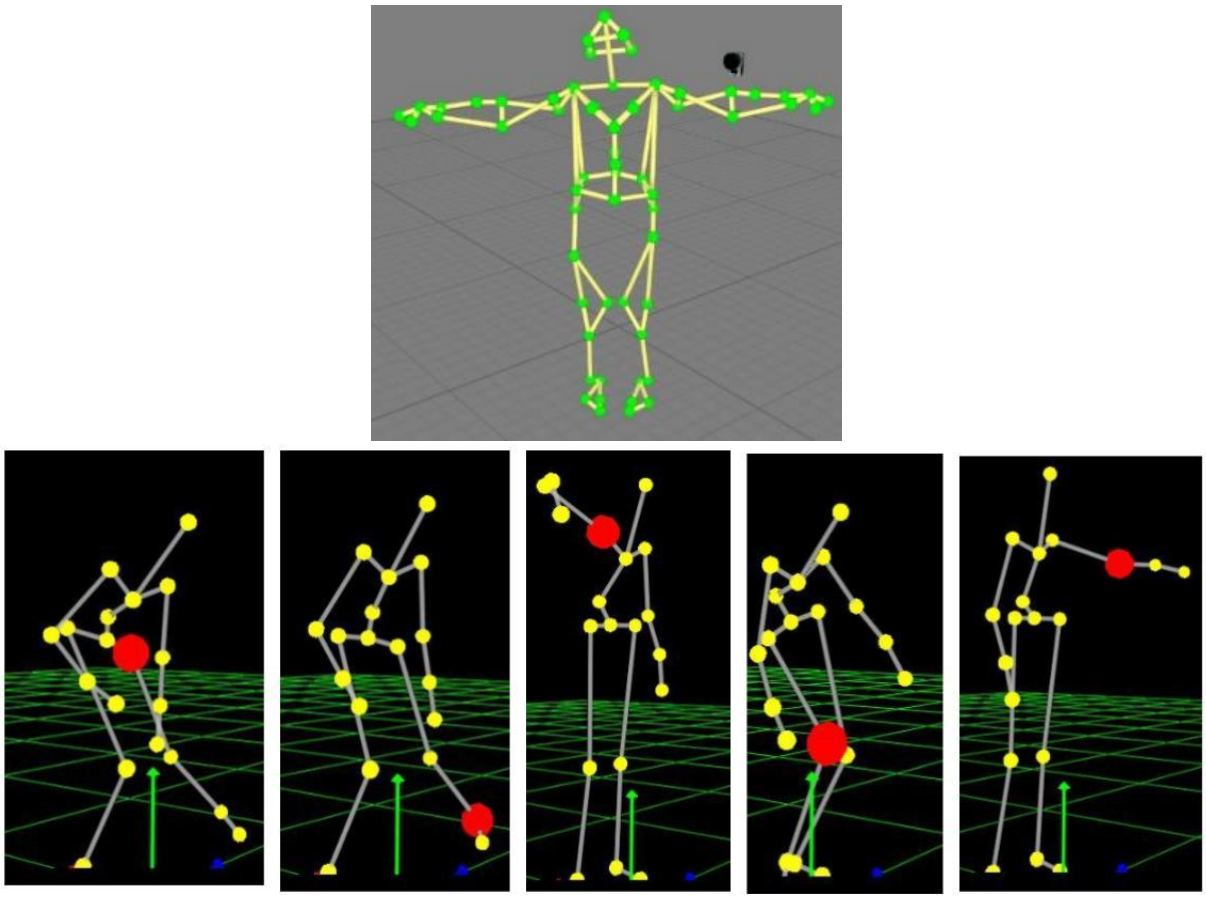


Figure 5 A representation of the approach described in Section 3.2.3

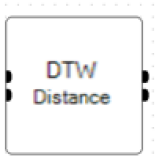
4 Software Library for Non-Verbal Social Signal Analysis

Synchronization

Synchronization (to not be confused with synchronization at the software/hardware level, see 3.2.1), is a high-level feature that deals with other user's movement features and that can be computed both on a single user (intra-personal features) and on multiple users (inter-personal features). For example, synchronization can be used to measure the degree of coordination between the joints' velocities of a dancer or to detect the level of entrainment between multiple users to measure collaboration and coalition between them.

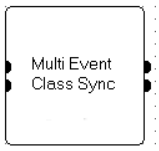
In this first version of the WhoLoDancE Software Library we present a collection of algorithms that are implemented as a software module and integrated in the EyesWeb platform, in order to be used for future experiments and for the annotation of recorded movement segments.

Dynamic Time Warping – Real Time



Dynamic Time Warping (DTW) measures the degree of similarity between two time series that vary their structure over time. A time series can present a pattern that had an evolution in a certain range of time. The same pattern can be contained as well in the second time series presenting, for example, a speed or amplitude variation. This particular implementation of the algorithm has been introduced in the WhoLoDancE platform with the characteristic of being computed in the real time.

Multi Event Class Synchronization



The Multi-Event-Class Synchronization (MECS) [Alborno et al. 2017, in preparation], computes the degree of synchronization between several time-series containing several types of events. Since events are no longer of the same typology, they belong to different *classes of events*. The algorithm evaluates the synchronization degree for each class of event. MECS relies its computations on the temporal distances between the timings at which events belonging to the different classes appear. Moreover, the algorithm allows to establish time constraints between events and to define event hierarchies. In the context of the WhoLoDancE project, the original implementation has been modified by introducing several kernel functions of the algorithm, (the original version provided only a linear computation). The design of different kernel functions allows to better adapt the calculus of the synchronization degree to different application contexts.

4.2 Dynamic Symmetry

UNIGE developed a computational model and software module to compute Symmetry [Camurri et al 2016]. *Symmetry* is a movement quality that is considered important in dance teaching [Wholodance Partners 2016], and in general in learning movement. Symmetry can be considered at two different levels of our conceptual framework, represented in Figure 6 [Camurri et al 2016, MOCO]: low-level *Postural Symmetry* and high-level *Dynamic Symmetry*.

Postural Symmetry - Layer 2 in our conceptual framework - can be described in terms of the shape of the body silhouette with respect to an axis or plane (e.g., vertical). It can be computed from the dancer's silhouette in 2D or in 3D (including depth measurement from RGB-D sensors) or in terms of a cluster of body markers in a motion capture setting. This measure has been implemented and is available in the Wholodance EyesWeb library.

In order to take dynamic and temporal dimensions into consideration, UNIGE developed analysis modules of *Dynamic Symmetry* as a higher-level feature - Layer 3 in our conceptual framework [Camurri et al MOCO16] by considering the coordination and dynamics of parts of the body. *Dynamic Symmetry* is an important aspect of several physical activities for example in sport (e.g., synchronized swimming, rowing), and in motoric rehabilitation. The ability to maintain the dynamic symmetry (but also dynamic asymmetry, i.e.,

independency of movement of different parts of the body) is very important especially in contemporary dance.

Details on the implementation of the software module included in the WhoLoDancE software library, and adopted for the automated annotation of the WhoLoDancE repository, are described in [Camurri et al 2016].

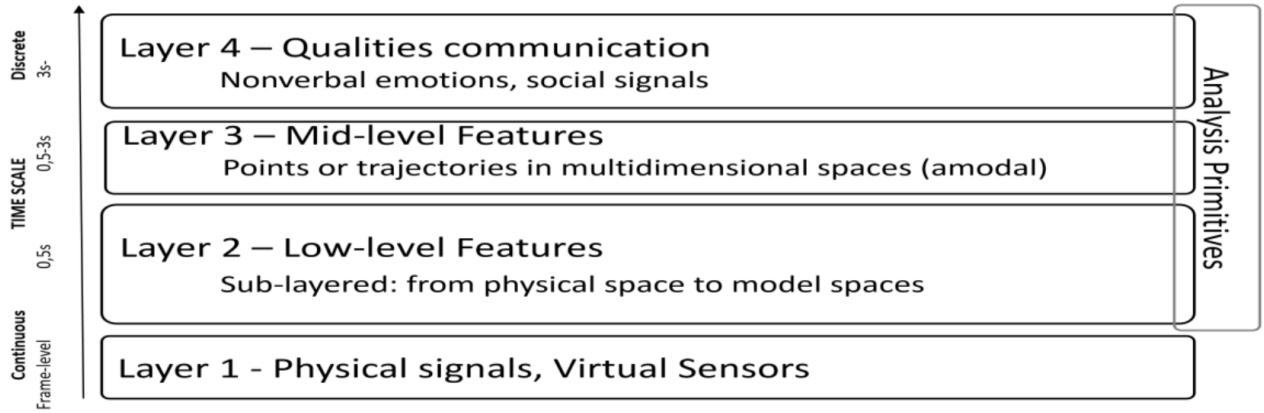


Figure 6 The conceptual framework on the layers of abstractness of motion features, from [Camurri et al, 2016]

5 Multimodal analysis of qualities in individual dance

UniGe developed a number of software prototypes to measure orientation. The *orientation* is the first movement principle to explore that has been chosen by the Wholodance consortium, and experimented with in the Workshop in Coventry. The multimodal approach was used to analyse data from Kinect, IMUs, and audio of respiration.

5.1 Software Prototype of teaching tool based on Laban's "Cube"

This software module, implemented in EyesWeb, measures the direction of body parts (arms, trunk and head) and shows in real-time the visualisation of dancer's orientation. The teacher can define the setup for the exercise, by choosing the number of target directions in the Cube (possibly random) and which parts of the body should be used (e.g., left or right forearm, head). The exercise is then presented to the student who has to perform it.

Figure 7 The chest mount harness adapted

A second, downscaled prototype of the exercise has been developed, this version uses IMUs instead of mocap sensors. In this prototype a number of IMUs are placed on the body of the dancer (using a simple wearable GoPro *chest mount harness,* shown in Figure 7). Three IMUs are located on the back of the dancer (hips, trunk, and shoulders planes), and other four IMUs are placed on wrists and ankles. The direction of each sensor is used to extract the direction of the corresponding body plane and is visualised by arrows. One of the many possible exercises can be similar to the previous one: the teacher specifies a number of different directions of the various body planes, then the dancer can try to orientate the hips, trunk, and shoulders towards those directions. Moreover, the configuration of the IMUs enables the measure of a number of further features that under development.

Both prototypes are a starting point for possible serious games to help the dancers to train their orientation towards different points, directions, and planes, both relative and absolute, and ultimately to enhance their directionality awareness.

In details, these software modules developed by UNIGE are based on their proposed "movement sketching" framework, aiming at enabling the user to build non-verbal movement queries to search the WhoLoDancE movement repository. This Movement Sketching paradigm is based on a downscaled version of the features available starting from the MoCap data, including features developed in DANCE (as previously mentioned), to build real-time WhoLoDancE exercises. In Figure 8 and Figure 9 we show the setup and the GUI of the described prototype, respectively.

Figure 8 Setup of the UniGe's kinect tool for teaching orientation presented and experienced at Coventry Workshop.

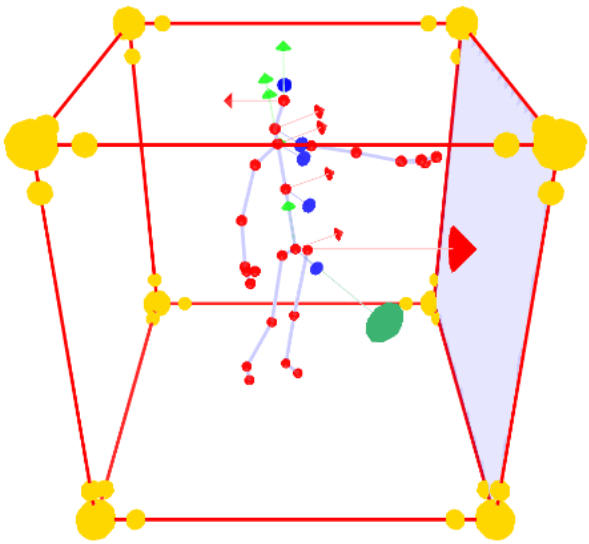


Figure 9 Visualization of the UniGe's kinect tool for teaching orientation

5.2 Movement segmentation

Movement segmentation is the process of identifying "motions of interest" from long sequences of human movement data and of reducing them to smaller components.

To automatically segment motion is fundamental to improve both recognition and reproduction of movement, on the other hand it turns out to be a complex activity due to the dimensionality and the (spatial and temporal) variability of human motion.

To perform the movement segmentation, one of the most common approach is represented by zero-crossings detection of specific motion features (e.g. velocity, joint linear acceleration etc.). In fact, zero crossings, characterize movements' trajectory direction changes and can be considered as segment

boundaries (i.e., start or stop of a movement). The problem of these approaches is represented by the tendency to over-segmenting the sequences.

We developed a more sophisticated methodology that combines multi-scale analysis [Lindeberg, 1992] and event synchronization [Quiroga et al. , 2002] techniques [Alborno et al., submitted to MOCO 2017]. Such methodology aims at being completely independent from any assumptions on the input data: i.e., it is be independent of length, speed and movements shape. This makes our technique easily applicable to different categories of motion data.

Our methodology has been preliminarily tested on a movement dataset containing karate sequences [Kolykhalova et al. INTETAIN 2015] and will be used to segment dance sequences recorded for WhoLoDancE.

Given a set of input motion features (for example the acceleration of the performers' limbs), the algorithm firstly evaluates the entire signal, then detects and isolates relevant events and assign a rank to each of them, and finally computes the synchronization between events with the highest ranks. The events characterized by high enough synchronization values are considered significant (i.e., the movement presents a clear transition that may represent a start or end of the movement).

6 References

- **Bruno Di Giorgi, Massimiliano Zanoni, Sebastian Böck, Augusto Sarti, Multipath Beat Tracking, in Special Issue on Intelligent Audio Processing, Semantics, and Interaction, Journal of the Audio Engineering Society, vol.64, no.7/8, pp.493-502, 2016**
- **Camurri, Antonio et al. "The dancer in the eye: towards a multi-layered computational framework of qualities in movement." Proceedings of the 3rd International Symposium on Movement and Computing. ACM, 2016.**
- **R. B. Myerson. 1977. Graphs and cooperation in games. Mathematics of Operations Research 2, 3 (1977), 225–29**
- **S. Tijs. 2003. Introduction To Game Theory. Hindustan Book Agency, New Delhi**
- **T. P. Michalak, K. W. Aadithya, P. L. Szczepanski, B. Ravindran, and N. R. Jennings. ´2013. Efficient computation of the Shapley value for game-theory**
- **Antonio Camurri, Corrado Canepa, Nicola Ferrari, Maurizio Mancini, Radoslaw Niewiadomski, Stefano Piana, Gualtiero Volpe, Jean-Marc Matos, Pablo Palacio, Muriel Romero (2016) "A System to Support the Learning of Movement Qualities in Dance: a Case Study on Dynamic Symmetry", *Proc. Ubicomp/ISWC'16 Adjunct*, September 12-16, 2016, Heidelberg, Germany ACM 978-1-4503-4462-3/16/09. http://dx.doi.org/10.1145/2968219.2968261**
- **WhoLoDancE Partners. WhoLoDancE: Towards a methodology for selecting Motion Capture Data across different Dance Learning Practices. In *Proc. MOCO 2016 International Workshop*, ACM Press, 2016, Thessaloniki.**
- **Paolo Alborno, Nikolas de Giorgis, Enrico Puppo, Antonio Camurri, Limbs synchronisation as a measure of movement quality in karate (in review, MOCO 2017).**
- **Quian Quiroga, T Kreuz, and P Grassberger. 2002. Event synchronization: a simple and fast method to measure synchronicity and time delay patterns. Physical review E 66, 4 (2002), 041904.**
- **Tony Lindeberg. 1994. Scale-space theory in computer vision. Kluwer Academic, Boston.**
- **Ksenia Kolykhalova, Antonio Camurri, Gualtiero Volpe, Marcello Sanguineti, Enrico Puppo, and Radoslaw Niewiadomski. 2015. 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. IEEE, 74–78.**