MSc. Thesis in Artificial Intelligence
Department of Information and Computing Sciences
University of Utrecht

Automatic Analysis of Synchrony in Dyadic Interviews

Sven Luehof

August 19, 2019

Supervisors:
dr. ir. R.W. Poppe
dr. A.A. Salah

Abstract

Nonverbal synchrony has received a great deal of attention from many different scientific areas for its relatedness to the quality of interaction and interpersonal relationships, functions in early infancy, and ability to be used as a predictor for variables such as therapy outcome. This motivates a need for automated synchrony analysis in order to exclude the possibility of human error and subjectivity. In this study the different methodologies used to extract movement data from video, as well as the methodologies to measure synchrony in movement data have been investigated. The goal of this study is to find the methodologies and settings that allow for the best quantification of synchrony in dyads. Synchrony is operationalized as the ability to distinguish rapport-building trained interviewers from interviewers that did not receive this training. For motion energy time series creation OpenPose and motion energy analysis (MEA) have been compared. Using the motion energy time series generated by MEA, the ability to measure synchrony of windowed cross-lagged correlation (WCLC), windowed cross-lagged regression (WCLR) and recurrence quantification analysis (RQA) have been investigated. The parameters of each of these methods have been tweaked to investigate their influence on the output score and find optimal values. The results show that MEA provides the best motion energy time series and that WCLR most accurately quantifies synchrony. Furthermore, the results show that the output score of WCLR is not robust against frame skip, therefore frame skip should not be used.

Contents

Chapter 1

Introduction

Nonverbal synchrony is usually an unconscious mutual behavior between humans that can often occur spontaneously and is closely related to the quality of the interaction. Synchrony can be used as an indicator of interpersonal relationships, has several functions during early infancy and adulthood and is also closely related to how dyadic partners are perceived [17]. Synchrony has also been used as a predictor, in [49] it was shown that by analysing nonverbal synchrony it is possible to effectively predict therapy outcome, as well as distinguish genuine interactions from pseudointeractions. Over the course of the last years, synchrony has received a great deal of attention from many different scientific areas, such as psychology, anthropology, sociology, linguistics psychotherapy, medicine, education, computational neurosciences and child psychiatry. Synchrony has come to influence these fields, because it can be used as a proxy for relevant concepts such as affection.

Before the arrival of computational methods, synchrony had to be manually measured by trained observers. These measurements suffered from drawbacks, such as the relatively long time required to do manual labeling and the subjectivity of the labeling. In consequence, a need for automatic analysis methods that can reliably quantify synchrony to eliminate the need for human experts and exclude possibilities of human error arose. In order to create such a method, synchrony must first be defined in a way that allows us to use it in a computational method. However, despite all the attention synchrony has received, a well-defined quantitative definition is still not available. Most literature refers to synchrony as individuals' temporal coordination during social interactions [17].

In this thesis we will analyse human body movement to create an estimate for synchrony between dyads. A measure of synchrony will be obtained by analysing the coordination of body movement between two dyadic partners. There are several methods to retrieve body movement from video, such as motion energy analysis, motion tracking devices and human pose estimators. Using one of these methods, a time series corresponding to the amount of body movement in between successive frames can be created for each individual. By using time series analysis methods, analysing the similarity between segments of the two time series can be used to obtain a measure of synchrony. This thesis aims to provide the best method to measure synchrony by comparing the body movement extraction techniques, the time series analysis methods, the settings of the time series analysis methods and also investigate practical considerations when creating an algorithm to measure synchrony.

The remainder of introduction is split up in three sections. Section 1.1 will describe the data that we will use to create and test our algorithm. Section 1.2 will define the focus of this thesis project, and will introduce the research questions. Finally, section 1.3 will provide an overview of the outline of the thesis.

1.1 Data

To create an algorithm that can automatically analyse synchrony, data displaying synchrony is required to test the algorithm. This data is provided by Wright et al. and consists of videos recorded in a comprehensive study conducted at Goldsmiths University of London [71]. The data consists of videos in which a pair of individuals is shown seated in a small room. Most interviews were recorded from two stationary viewpoints, however some were recorded from only one stationary viewpoint. The data shows one individual conducting the interview whilst the other answers the posed questions.

In their experiment, interviews were held in two waves. For the first wave, all interviewers received an initial interview training. In the second wave, some of the interviewers were also trained to use 9 rapport-building techniques. The 9 rapport-building techniques are: (1) using the preferred name, (2) self disclosure/reciprocity, (3) smiling, (4) conversational tone of voice, (5) open body posture, (6) eye-contact, (7) head-nodding, (8) active-listening, (9) empathy. After the interview concluded, the information disclosure was examined as well as several other variables, among which is rapport. For the purpose of this thesis, rapport will be used as a proxy for synchrony, because it has been shown that synchrony is closely related to rapport [64, 6]. Example frames extracted from the data are shown in Figure 1.1.

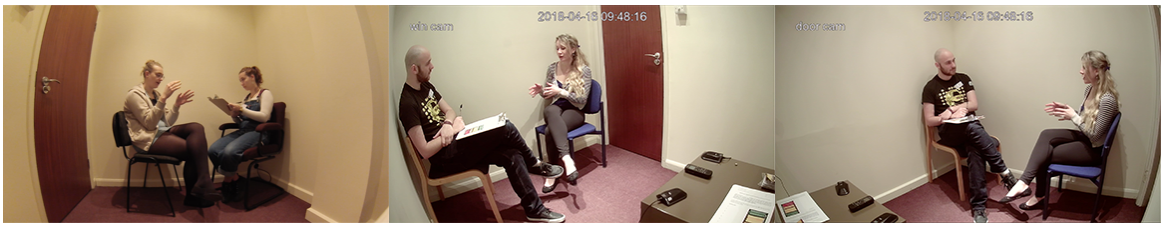

Figure 1.1: Example interview video fragments shown from the three different stationary viewpoints.

Before this data could be used, several adjustments had to be made. Some of the recordings still contained the instructional part, in which the researcher stands in the frame to talk to the participants, often even occluding one of the participants. These parts are not relevant for synchrony detection and were thus trimmed from the video. The videos also contained speech, which has also been shown to have a relation with synchrony [32]. However, since the focus of this thesis is on nonverbal synchrony, the audio was removed from the video to reduce the size of the data. Furthermore, as can be seen in Figure 1.1, the camera caused a slight fisheye distortion, which had to be undistorted using the right camera matrix. From the adjusted data we would like to extract the motion energy of the dyadic partners.

1.2 Research Objective

The vast interdisciplinary interest in synchrony and the need for an accurate automated approach for measuring synchrony contribute to the motivation for this work. This thesis' overall objective is to investigate the feasibility of creating an automated analysis algorithm for measuring synchrony from video and find the methodologies and settings that allow for the best quantification of synchrony. In order to achieve this objective, several challenges need to be solved.

Before an algorithm that automatically analyses synchrony can be created, a set of questions must be considered. Three theoretical challenges with respect to conducting an interaction study have been formulated by Capella: 'what to observe, how to represent observations and when and how frequently to make observations' [13].

Besides these three questions, the use of human pose estimators also introduces several challenges. Despite the simple poses of the dyadic partners in the data, human pose estimators still produce error in their estimations. One source of error is missing observations, which may be caused by events as self-occlusion or insufficient color difference in foreground and background. Another source of error are incorrect measurements, which may happen when people occlude each other, causing the human pose estimator to incorrectly assign body parts to individuals. A final possible cause of error is inaccurate measurements, where the human pose estimator's found body parts are not on the correct point in space, which may be caused by events as lighting changes. Therefore, with respect to pose estimation, one problem that needs to be solved is how to filter the error caused by signal-distortion.

Furthermore, there exists a more practical problem with respect to making the algorithm computationally feasible. Since human pose estimation from video is a computationally costly task, performance enhancing methods are desirable. For example, since humans generally do not move fast enough to significantly move in between successive frames, we may choose to skip frames in order to reduce computational cost.

The solutions to these problems will be explored in the attempt to create an accurate algorithm capable of automatically analysing synchrony in dyadic communication. The research questions of this thesis are:

1. Which synchrony measuring method most accurately measures synchrony?
2. What are the optimal parameter settings for each synchrony measurement method?
3. Do time series created by human motion analysis provide better synchrony measurements for the best synchrony measuring method than time series created with motion energy analysis?
4. What is the ideal frame rate for measuring interpersonal synchrony in dyadic interactions?

To answer these questions, this thesis aims to design, implement and evaluate an algorithm that automatically analyses synchrony in dyadic communication. To answer the first research question, the three most promising methods, windowed cross-lagged correlation, windowed cross-lagged regression

and recurrence analysis are implemented and their accuracy is investigated. The accuracy is defined as the ability to distinguish rapport-trained interviewers from control interviewers. This distinction is made by assigning higher output scores for wave 2 than for wave 1 for dyads that did receive the rapport-building training, whilst assigning similar output scores amongst waves for dyads that did not receive the rapport-building training.

The second question is answered by running each synchrony method with different values for each parameter. How changes in parameter settings influences the synchrony measurement and the ability of the synchrony measurement method to distinguish rapport-trained interviewers from control interviewers is investigated. The optimal parameter settings are defined as the set of parameter values that allow for the best distinction between rapport-trained interviewers and control interviewers.

To answer the third question, the ability of both methods to create time series, human motion analysis and motion energy analysis, to accurately represent movement is investigated. The best synchrony measuring algorithm is tested on the time series created by human motion analysis and the time series created by motion energy analysis. The accuracy of each time series creation method is determined by comparing the synchrony output score of each dyad per wave. The accuracy is determined by how well the synchrony measuring algorithm assigns a higher score in the second wave than in the first wave to dyads that have received the rapport-building training. On the other hand, the synchrony score per wave should be similar for dyads that did not receive the rapport-building training. The method that provides the time series which results in the most accurate synchrony measurement is deemed the better technique.

The fourth question is answered by testing several different numbers of frames to skip and look at the impact it has on the synchrony measurement for each of the three synchrony measuring methods. Greater frame skips will generally result in a faster running algorithm, but may also cause the algorithm to miss small movement. Therefore, we want the frame skip to be as large as possible without compromising the synchrony measurement. For fair comparison, the time series creation method will remain constant.

1.3 Thesis structure

The structure of the thesis is as follows. In Chapter 2 the related work will be discussed in order to provide the reader with background information required to understand the methods and results of this thesis. Chapter 3 provides the implemented approach to measure synchrony and a description of the algorithm's pipeline. In Chapter 4 the implemented approach will be evaluated and the results of the evaluation will be discussed. Finally, the conclusion of this thesis and a direction for future work is presented in Chapter 5.

Chapter 2

Related Work

This chapter is dedicated to providing an overview and explanation of the current literature. Section 2.1 is dedicated to defining synchrony and its most relevant properties. By using a definition that allows for computation, synchrony can be quantified. A general introduction on how synchrony can be measured is provided in Section 2.1.3. The frame-based movement measurement methods, motion energy analysis, motion capture and human pose estimation, to create movement time series are discussed in Section 2.2. These time series will contain error, Section 2.3 discusses the different types of error and how to remove them. The time series analysis methods are discussed in Section 2.4. This section will provide a description of one of the standardized analysis methods, called windowed cross-lagged correlation in Section 2.4.1, its altered version, called windowed cross-lagged regression in Section 2.4.2, and another method called recurrence analysis in Section 2.4.3. Finally, a method to determine whether detected synchrony is significant is given in Section 2.5.

2.1 Synchrony

Despite all the multidisciplinary attention synchrony has received, synchrony remains difficult to define and delimit. Synchrony has been defined using multiple terms and conceptualizations, many of which are synonymous or to some extent overlapping. For example, Schoenherr et al. identified synchrony as a suitable overarching term encompassing different conceptualisations such as facial imitation, movement synchrony or speech convergence [58].

In [17], Delaherche et al. stated that several synonyms for synchrony have been used throughout literature to describe the interdependence of dyadic partners' behaviors, such as mimicry, social resonance, coordination, attunement, chameleon effect, etc. Therefore, in order to define synchrony, they tried to first study its relation to similar concepts and define synchrony in terms of its physical manifestation as "the dynamic and reciprocal adaptation of the temporal structure of behaviors between interactive partners". They argued that synchrony is inextricably related to the study of communicative interaction and language. They refer to [14], in which Clark defines a conversation to be a joint activity that requires coordination at two levels: content and process. At the content level, coordination of what is being said is required for conversational partners to reach a common understanding. At the

process level, conversational partners can accurately predict when conversation phases start and end. By predicting the ending of the speaker's turn, can the listener begin his turn at the correct time, thereby achieving synchrony between the conversational partners.

On the other hand, in [20], Feldman explored to what extent synchrony influences the emergence of complex social behavior and higher-order cognitive capacities. Feldman offers a construct in which synchrony is posed as an "overarching, biologically based, micro-level behavioral framework that coordinates the ongoing exchanges of sensory, hormonal, and physiological stimuli between parent and child during social interactions". Along the lines of this conceptualisation, synchrony in terms of its underlying processes was defined as "the temporal coordination of micro-level social behavior".

Bernieri et al. defined behavioral entrainment, or synchrony, as "the adjustment or moderation of behavior to coordinate or synchronize with another, similar to the synchronization occurring between members of an orchestra" [8]. Furthermore, they suggest that the definitions of synchrony may be classified in three broad categories: biological rhythms, simultaneous behaviour, and perceived synchrony. The biological rhythms category is based on biological sciences, in which human behavior occurs rhythmically and can be described in cycles. Therefore, this category describes synchrony as the degree of conformity between the behavioral cycles of two or more people. The simultaneous behavior category is related to behavioral mirroring or mimicry. Along these lines, synchrony is defined as the quantity in which one person directly imitates or mimics another person's behavior. The perceived synchrony category defines synchrony as a perceptual social phenomenon. The essential feature in this definition is that the apparent synchronous events can be combined to create a perceptual unit, described as a 'whole'.

Finally, Harrist and Waugh view synchrony as a type of dyadic interaction, displaying an observable pattern that is mutually regulated, reciprocal, and harmonious [25]. They found that, in relation to caregiver-infant synchrony, synchrony is primarily achieved via attunement of the caregiver. They argued that caregiver-infant synchrony has three critical prerequisites. The first prerequisite is maintained engagement, synchrony can only occur in prolonged engagement with mutual attention and shared focus to track each other. The second prerequisite is temporal coordination, synchrony requires matching each other's activity level and finding a rhythm in their interaction. The final requirement is contingency, which represents the relationship between events, in which the occurrence of one event increases the likelihood of another event. Caregiver-toddler synchrony differs in two ways from caregiver-infant synchrony. The first difference is that, rather than one-directional attunement from the caregiver towards the infant, mutual affiliation is now required. Furthermore, since the communicative capabilities of the child have improved, interactions with the child now also require more variability in who leads and who follows. The second change is in the array of information and behaviors used by the caregiver. During early childhood, synchrony differs in two ways from caregiver-toddler synchrony. Firstly, involvement of the caregiver and the child has become equal or near-equal, resulting in a balance in turn-taking. Initiating has now become a critical characteristic of synchrony. Secondly, they argued that during early childhood synchronous exchanges should only occur with non-negative affect. They argued that interactions that are both synchronous and mutually negative may function in a particularly destructive way.

The previously mentioned definitions are but some of the many definitions used in literature and many definitions share common characteristics. The first commonality of the definitions is that synchrony has a temporal nature. Besides its temporal nature, another common characteristic the definitions of synchrony share is that synchrony must involve some notion of behavioral entrainment or adjustment to another, which can be simultaneous, time delayed or converging.

2.1.1 Properties of Synchrony

Previously, the definitions of synchrony throughout literature have been provided as well as the commonalities between the definitions. What has not been discussed yet are the properties of synchrony that will aid us in its measurements.

Time delay For behavior to be considered synchronous, each behavior produced by one partner must be reciprocated by the coordinated behavior of the other partner within a limited window of time [17]. Altmann suggested that, with respect to time delay, synchronous phenomena can be grouped into three categories: simultaneous, time delayed or converging [3]. Simultaneous synchrony means there is no time lag between movements of a dyad. Converging synchrony refers to the phenomenon that movements of a dyad become more similar over time.

However, there is no consensus over what the size of the limited window should be. Robinson et al. defined this range to be at most 7 seconds [52], whereas Bilakhia et al. defined this range to be 0.04 seconds up to 4 seconds [9]. Currently, the selection of the appropriate range is largely left up to the researcher. However, the findings of Sonnby-Borgström et al. may be used as an indication for a lower bound. They have shown that humans do not display facial mimicking at the 17ms level, but high-empathy participants do display significant facial mimicking at the 56ms level [60].

Orientation Orientation of synchrony refers to the leader-follower relationship within an interaction. Usually a conversation will be led by a person, who is driving the interaction and sets the pace, and a person that follows along. In social interaction this relation is often dynamic and will not remain constant throughout the entirety of the conversation [16]. One way to determine the orientation of synchrony is by looking at the time lag. A positive lag between partner 1's features and partner 2's features accounts for "partner 1 is leading the interaction", a negative lag between partner 1's features and partner 2's features accounts for "partner 2 leading the interaction". A zero lag between each partner's features accounts for mutual synchrony.

Mirroring Unlike mirroring or mimicry, synchrony is dynamic in the sense that the important element is the timing, rather than the nature of the behaviors. For example, dyadic partners both sitting cross-legged exhibit mimicry, however only when one person uncrosses their legs and the other follows by also uncrossing their legs, do they display synchrony [17].

2.1.2 Importance of Synchrony

As social organisms, synchrony influences our lives in many ways. A functional aspect of synchrony is its relevance in the development of rapport and interpersonal relationships. Firstly, Stel and Vonk argued that mimicry is beneficial for people in social interactions [63]. They showed that mimicry caused mimickers and mimickees to became more affectively attuned to one another, form a stronger bond with each other and rate the interaction as smoother. Secondly, LaFrance has shown that there exists a positive correlation between posture sharing and rapport [31]. Similarly, Bernieri et al. have shown that interactant rapport reported by women is positively correlated with synchrony [7]. On top of this, Tickel-Degnen and Rosenthal have shown that coordination, or interactional synchrony, is one of the key components that make up rapport, along with positivity and mutual attention [64]. Furthermore, Valdesolo et al. have shown that synchrony leads people to perceive those with whom they engage in synchronous behavior as more similar to themselves [66]. They also showed that people are more willing to help those with whom they had engaged in synchronous behavior and will do so for longer periods of time in comparison to unsynchronized individuals.

In addition to being an important component that substantiates rapport and influences interpersonal relationships, synchrony also plays an important role in children's development. The effect of synchrony with respect to children has been thoroughly studied in relation to parent-infant synchrony. Rocissano et al. have shown that in the toddler stage, children were more likely to comply with instructions of synchronous caregivers than with instructions of asynchronous caregivers. Furthermore, it was shown that children who did not participate in synchronous communication with their mother were least likely to comply with instructions [53]. Synchrony also seems to be related to the co-regulation of parent-infant affective states. In a study by Feldman, the co-regulation of affective states and synchrony were examined in video tapes of couples interacting with their first-born child. They found that mothers use co-regulation and synchrony to maintain and regulate the exchanges with their infant during face-to-face interaction. Through these synchronized exchanges, the mother can smoothly move the infant from one affective state to another [19]. Finally, for the curious reader we refer to [25], in which Harrist and Waugh provide an extensive review of empirical and theoretical work on the influence of dyadic synchrony on children's development during infancy, toddlerhood and early childhood.

On top of this, a relationship between synchrony and psychotherapy outcome has also been found. Ramseyer and Tschacher found an association between nonverbal synchrony and the patient's view of the process as well as with therapy outcome. They showed that higher levels of nonverbal synchrony resulted in better symptom reduction [49]. A similar relation was found by Paulicker et al. They found the highest level of synchrony in patients with non-improvement and consensual termination, improved patients showed a medium level of synchrony, and non-improved patients with drop-out showed the lowest level of synchrony at the beginning of therapy, even when controlling for the therapeutic relationship [45].

Vinciarelli et al. argue for the indispensability of social intelligence and the role it has in achieving success in life [67]. Therefore, they investigated the possibility of bringing social intelligence to computers by using Social Signal Processing (SSP). They argue that although the first steps towards artificial so-

cial intelligence and socially-aware computing have been taken, the road is still long as four issues still need to be addressed. The first issue relates to the required collaboration between engineers and social scientists. No automatic analysis of social interactions is possible without accounting for the basics of social behaviours such as interactional synchrony. Therefore, engineers need to include the social sciences in their reflection, while social scientists need to formulate their findings in a form useful for engineers. The second issue relates to need of implementing multi-cue, multi-modal approaches, since nonverbal behaviours may correspond to different interpretations depending on context and culture and are therefore ambiguous. The third issue relates to the need to use real-world data. Data is often acquired in an artificial setting, which causes a simplification of the investigated situation and may influence the assessment of the automatic approaches. The final issue relates to finding applications that will benefit from SSP as applications have the advantage that they link the effectiveness of SSP to reality.

2.1.3 Measuring Synchrony

Now that the common aspects between the definitions of synchrony and its properties have been investigated, the techniques used to measure synchrony can be investigated. Considering the properties of synchrony, to measure synchrony, the algorithm should analyse the co-regulation within a limited time window of body movement between two people and consider varying orientation.

In earlier days, this had to be done manually by trained observers. One proposed manual rating method by Bernieri et al. is the judgement method, in which synchrony raters represent subjective ratings for three aspects of synchrony on a 9-point Likert scale [8]. Firstly, simultaneous movement, which is the quantity in which the intaractant's movement begin and end at the same time. Secondly, tempo, or rhythm, similarity, which represents "the degree to which the two people in the clip seem to be 'marching to the beat of the same drummer'". And finally, coordination and smoothness, representing how smoothly the interactant's flow of behavior intertwines. An alternative way of measuring synchrony within time windows is by focusing on discrete movements (e.g. pose shift, touch of the face) and then correlating the occurrences over time [62]. However, it can be argued that the occurrence of these discrete movements is relatively rare and is therefore not suitable as a reliable source for synchrony. Such non-computational methods suffered from drawbacks, such as the required time to do manual labeling and the subjectivity of the labeling. Often, the annotator must make a trade-off, because no label exactly describes the observation. The judges' reliability in assessing such a subjective and complex construct is also questionable, and no general framework for synchrony assessment has been accepted to date [17]. On top of this, it is not possible to closely examine how various aspects of behavior (e.g. speed, body part, orientation, etc) affect synchrony.

Nowadays, the developments within the field of computer vision brought along new computational methods for measuring synchrony. The benefit of using computational methods is that we can avoid the drawbacks of non-computational methods. Even though coding is still required to create a computational method (for training or testing purposes), after the algorithm is completed future uses will no longer require raters to do behavioral coding. Furthermore, the issue of subjectivity of the judges will be solved, because an algorithm's output will be deterministic by nature.

However, before considering computational methods, we must define what movement should be considered. Schmais and Felber split synchronous body movement into three categories: (1) rhythmic synchrony, which considers movement rhythms in some body part between people (not necessarily the same body part), (2) effort synchrony, quantifying similar effort quality or dynamics between people, and (3) spatial synchrony, a measure of how much all body parts move in the same relative direction [56]. In more recent work, we see that some focused on movement of specific body parts, such as eyes [51], legs [57] or fingers [43], whilst others chose to focus on more global features, such as posture [40].

Even though many existing computational methods consider all general movement within a predefined region, when opting to look at a single body part another aspect to consider is whether mirrored synchronous movement is also considered to be synchrony. Mirrored synchronous movement in this context refers to the event when two people are standing opposite to each other and move the same body part, but on the opposite side of the body (e.g. person 1 raising their left arm and person 2 raising their right arm).

2.2 Frame-based Movement Measuring

Before it is possible to measure synchrony, it is necessary to retrieve the movement of a dyad from video. Several acquisition techniques capable of accomplishing this task are prominent in the literature: motion energy analysis, for analysing general movement within a region, motion capture, to measure motion of specific body parts, and human pose extraction, which estimates the human pose within an image.

2.2.1 Motion Energy Analysis

In [23], Grammer et al. proposed a new computational method to analyse behavior they called automatic movie analysis (AMA). AMA is an approach based on automatic analysis of changes in body contours between successive frames in a video. AMA quantifies the amount of motion energy by subtracting pixel color values of successive frames. This results in a measure of the total amount of movement within a certain time span.

The method described above is often referred to as motion energy analysis (MEA) [58]. In most literature MEA was applied to a region of interest (ROI) to create time series representing the total amount of movement between frames for that ROI. MEA is therefore an objective method that quantifies the intensity of videotaped movements within a region of interest in a frame-wise manner [49]. In relation to synchrony, useful time series were obtained by enveloping each person in their own ROI.

However, a ROI does not mitigate the error of measuring non-movement related pixel changes as movement. The pixel color changes may occur due to events as lighting changes or a change in camera position. To reduce the noise caused by events like these, pixel changes are only considered to be movement related if the change exceeds a threshold, as is shown in Figure 2.1. In general, setting the threshold too low will inadequately remove noise, on the other hand, setting the threshold too high will make the system unable to pick up small movement.

Figure 2.1: Removing noise from the measured movement by applying a threshold. The left image has a too low threshold, the center image has a good threshold and the right image has a too high threshold. This Figure is provided by [49].

Another issue is scale. If one person appears closer to the camera, then even small movement will result in a lot of pixel value changes. If the person farther from the camera recreates the same movement, then MEA would still quantify this as less movement, because fewer pixels changes. This issue cannot be solved by using a threshold, however some accounted for the different size ROIs by z-transforming the data [45].

Furthermore, the measure is heavily affected by the color of the clothing and the viewpoint under which the persons are recorded. For example, if the color of the clothing is similar to the background then movement may be overlooked, because the pixel color change did not exceed the threshold. The viewpoint determines what movement can be seen by the camera, if the viewpoint causes self-occlusion then the movement of the occluded body part will mostly be overlooked.

Overall, while the method is automatic, the constraints on the physical setup of the recordings need to be very strict in order to result in reliable measurements. One could argue that this reduces the ecological validity of the measurement.

2.2.2 Motion Capture

In an attempt to remedy the shortcomings of manual annotating synchrony ratings, researchers have started to investigate motion capture as a means to automatically measure human movement [47]. Advances in computing technology and the development of dedicated technologies have made it easier to record and analyse human nonverbal behavior. Motion capture can be used to create motion energy time series by tracking the position of sensors over time. Two distinctions can be made between the motion capture methods. The first distinction can be made based on whether the method relies on markers or sensors to record body movement. The second distinction is made based on whether the method offers full-body (global) or single body part (local) movement capture. The output of motion capture methods generally consists of a series of body parts, represented as shapes with a certain length, and joints, which is a single point in space.

Marker-based approaches captures the location of the markers worn on the body by triangulating the 3D position of a subject between two or more calibrated cameras. In order to avoid occlusion, usually many cameras are needed. There are two types of marker-based approaches: passive-marker

and active-marker approaches. Passive-marker are markers coated with retro-reflective material that reflect light that is generator near the camera, whereas active-markers transmit the light themselves. Passive-marker approaches ensure good visibility, but may confuse markers. Active-markers do not suffer from this, because each marker emits its own unique frequency by which they can be distinguished, but must be guaranteed power to avoid data loss.

Inertial systems employ a suit equipped with sensors to measure movement of the body. The sensors attached to this suit usually consist of 3D gyroscopes, accelerometers and magnetometers [54]. An example of this suit is shown in Figure 2.2. By combining the signal of each sensor, estimates about their position can be made. The accuracy of inertial systems is generally high, but may suffer from drift due to presence of sensor noise, sensor signal offset, or sensor orientation errors. Sensor noise may be the result of the presence of metal in the recording environment.

Figure 2.2: Xsens MVN consists of 17 inertial and magnetic sensor modules [54].

One drawback of motion capture systems is its intrusive nature, since it requires subjects to be equipped either with markers or with a suit equipped with sensors. It can be argued that this may cause subjects to be more conscious of their behavior than they would in a real-life setting, which would decrease the ecological validity of this method.

2.2.3 Human Pose Estimation

Due to the developments in computer vision have human pose estimation (HPE) techniques become a viable alternative to MEA and motion capture. HPE doesn't suffer from the same drawbacks as MEA and Motion Capture, as it doesn't require a ROI and will generally not count non-movement related pixel changes as movement, because those pixels do not belong to a human. Furthermore, HPE techniques do not require the participants to wear sensors and is therefore not as intrusive as motion capture. However, HPE techniques still suffer from scale and relies on contrasting colors to find humans in images, therefore clothing should preferably not be the same color as the background. Besides our application HPE has also been used in surveillance, animation, video games, athletic performance analysis and human-computer interaction.

In general, the goal of human pose estimators is to find 2D or 3D body part locations within an image, which can be connected to create a set of lines representing the human's pose. By comparing the location of a keypoint or line segment between keypoints in successive frames an estimate of motion can be calculated. The process of estimating poses over time is called human motion analysis. By comparing keypoints found by human motion analysis in successive frames, a motion energy time series can be created.

The process of human motion analysis has been summarised by Liu et al. in two main stages: preprocessing and body parts parsing. An illustration of their summarised common pipeline of human motion analysis is shown in Figure 2.3. The preprocessing stage includes feature extraction, camera calibration, body detection and foreground segmentation. It is necessary to do data calibration, because HPE is not always applied to images taken from the same camera viewpoint, for example camera calibration is often applied to alleviate the error caused by viewpoint changes.

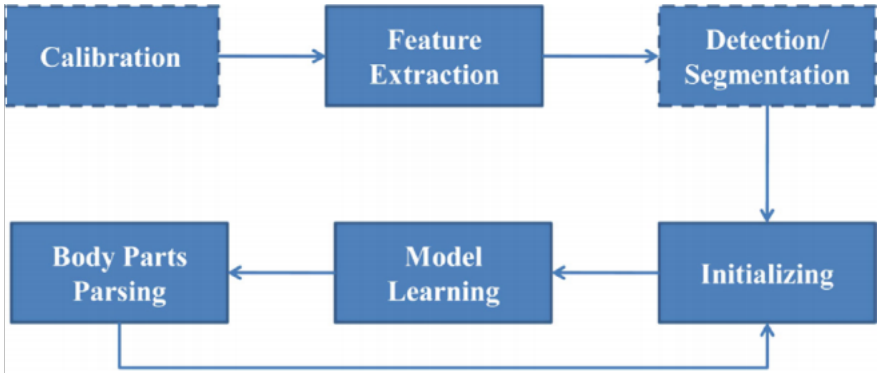

Figure 2.3: Model illustrating the common human motion analysis pipeline, provided by [34]. In this model, dashed-borders represent optional states.

The aim of the body parts parsing stage is to locate different body parts in the images. The body parsing methods can be split in methods estimating 2D body parts locations and methods estimating 3D body parts locations. 2D body parsing is done using feature extraction or feature learning. Feature extracting in images can be done with features such as histogram of oriented gradients and scale invariant feature transform [61]. Feature extraction in videos may be done with optical flow [55]. The alternative to feature extracting is feature learning, where the method is trained to recognise body parts. For example, Newell et al. proposed using a 'stacked hourglass' design for convolutional deep neural network to predict human poses from a single image [42]. The design is referred to as hourglass, because it consists of the steps of pooling followed by upsampling to get the final output of the network, which when visualised looks similar to an hourglass.

Many taxonomies have been proposed to categorize human motion analysis methods. In [46], Poppe discusses the characteristics and distinctions of human motion analysis methods used throughout literature. Poppe observes that methods fall in two main classes: model-based (or generative), which employ an a priori human body model, and model-free (or discriminative) approaches. These main classes can be further subdivided: model-based estimation can be split up in top-down and bottom-up, and model-free approaches can either be learning-based or example-based. On the other hand, Gavrilla assigned methods to one of three categories: 2D approaches with shape models, 2D approaches without shape models or 3D approaches [22]. Whereas Ji and Liu categorised view-invariant human motion

analysis methods in two classes: pose representation and estimation, and action representation and recognition [28]. The difference is that the former represents methods that estimate a 3D pose from an individual image in a sequence, the latter is focused on inferring and understanding human action patterns. However, the two types of methods are closely connected, because the view-invariant pose estimate can be used as input for the action recognition.

Model-based vs. model-free There are two main approaches for model-based estimation: top-down and bottom-up. However, recent work combines the approaches to benefit from the advantages of both [46]. Top-down approaches find the human pose within an image by searching for a projection of the human body. After an initial pose is found the estimate will in general be further improved with a local search around the pose. The high dimensionality of the pose space prevents a bruteforce local search, therefore local search is usually done with gradient descent on the cost surface. A drawback of top-down approaches is that it requires a (manually) specified initial pose estimation in the first frame of a sequence, because the initial estimate is often obtained from the estimate in the previous frame. This method also suffers from the computational cost of forward rendering the human body model and calculating the distance between this model and the image observation. On top of this, top-down approaches are sensitive to error caused by (self)occlusions. Moreover, errors are propagated through the kinematic chain. Error in the estimation for a body part at the beginning of the kinematic chain may cause errors in estimating the orientation of body parts lower in the kinematic chain.

Bottom-up approaches start by finding body parts throughout the entire image and then assembles these into a human body. These approaches find body parts by matching a 2D template. These approaches generally suffer from many false-positives, because there are many limb-like regions within an image. To find sufficient body parts in the image to construct a human, a part detector for most body parts is necessary. Body parts are assembled by taking physical constraints such as body part proximity into account. Bottom-up approaches are able to cope with occlusions by introducing temporal constraints. Furthermore, bottom-up approaches have the advantage that it does not require manual initialization.

Model-based approaches may choose to use appearance models or structure models to aid it in its search. Appearance models aim to parse each part of the body individually, whereas structure models also look at the relationship between different body parts. A popular appearance model is called poselet, whose purpose is to describe a particular part of the human pose under a given viewpoint [11]. Specifically, a poselet is a set of linear support vector machines, which bridges the gap between the body part appearance and configuration. Structure models represent the human body as a constrained tree model in which body parts are represented as nodes and each node is connected with its neighbouring body parts. The most popular structure model is the pictorial structure model, in which each node is modeled individually in a deformable form, and spring like connections are used to connect different parts [21], as is shown in Figure 2.4. This model is able to assume many different poses due to its special structure.

Model-free approaches aim to find a direct relation between the image observation and a pose. Two distinctions between model-free approaches can be made: learning-based and example-based. Learning-based approaches learn a function from image space to pose space from training data. Whereas

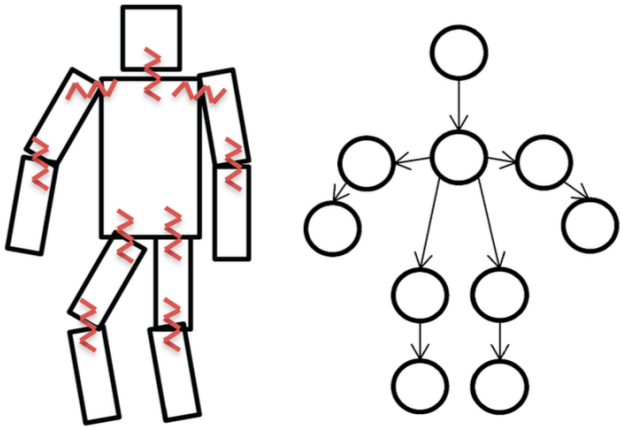

Figure 2.4: Illustration of the pictorial structure model with springs and as tree model, provided by [27].

example-based approaches do a similarity search to select candidate poses from a database containing exemplars and their pose descriptions. The pose estimate is obtained by interpolating the candidate poses.

Tracking Tracking is the process of estimating poses from frame to frame. Tracking is used to provide an initial pose estimate and to ensure temporal coherence between poses over time. When it is assumed that the time between subsequent frames is small, the difference in body poses is likely to be small as well. The body pose of the current frame be used to as a reasonable initialization for the next frame, because the difference in body pose between two successive frames is not significantly large. These body pose differences can be approximately linearly tracked, for example with particle filtering or a Kalman filter. Traditionally, tracking was aimed at maintaining a single hypothesis over time. Since this often causes the estimation to lose track, most recent work use multiple hypotheses to decrease the chance of losing track.

Benchmark Researchers have created a myriad of datasets to evaluate their proposed techniques for the specific task, which makes the fair comparison on the different algorithms even harder. Due to the large variations in different scenes, it is difficult to build a universal dataset to evaluate the human pose estimation. Therefore, datasets have been able to cover the entirety of the overall pose estimation challenges. In recent years, to compare the performance of each human pose estimators, several common sources created to evaluate, train and compare different models on have been developed, such as MPII Human Pose [5], PoseTrack [4] and COCO [33].

The MPII Human Pose benchmark provides a comprehensive dataset created from a wide range of human activites, such as recreational, occupational and householding activities, captured from a wide range of viewpoints. Furthermore, they provide labels of body joints positions, full 3D torso and head orientation, occlusion labels for joints and body parts, and activity labels. For each image they give adjacent video frames to enable the use of motion information.

PoseTrack is a benchmark aimed at video-based human pose estimation and articulated tracking.

They focus on three main tasks: (1) single-frame multi-person pose estimation, (2) multi-person pose estimation in videos, (3) multi-person articulated tracking. Their dataset features videos with multiple people labeled with person tracks and articulated pose.

The COCO benchmark dataset consists of images of complex everyday scenes containing common objects in their natural context. Objects are labeled using per-instance segmentations to aid in precise object localization. Their dataset contains photos of 91 objects types that would be easily recognizable by a 4 year old. The COCO dataset contains a total of 2.5 million labeled instances in 328k images.

Deep learning Recent advancements in artificial intelligence and the successes of deep learning for classification problems have made deep learning a strong contender for human pose estimation and has attracted a lot of research attention and brought forth many applications. For instance, Toshev and Szegedy [65] use a seven-layered convolutional deep neural network in their body joint regressor to represent the joint context and predict the body location. Also, Chen and Yuille [44] train deep convolutional neural networks on the image patches around the body joints to learn the probabilities for the absence and spatial relationship of different body parts. Newell et al. came up with a stacked hourglass design for their deep convolutional neural network that consists of successive steps of pooling and upsampling to produce a final set of predictions [42].

DensePose Güler et al. developed a promising deep learning open source multi-person human pose estimator, called DensePose [24]. DensePose aims to map all human pixels of an RGB image to the 3D surface of the human body without the need for depth information. They created a new fully convolutional network architecture comprised of the Dense Regression architecture [1] and the Mask-RCNN architecture [26]. To deal with scale differences, the DensePose architecture begins with extracting region-adapted features through region of interest pooling. These featues are then propagated to a region-specific branch, which is a fully-convolutional network that densely predicts discrete body part labels and continuous surface coordinates. This output is given to another DensePose network with a cross-cascading architecture for other specific tasks, such as keypoint detection. Once the predictions for the specific task have been found, they will be further improved by a refinement unit.

OpenPose Another promising open source multi-person 2D pose estimation approach has been proposed in [12]. Their approach, called OpenPose, estimates poses in a bottom-up manner by using Part Affinity Fields (PAF) to learn to associate body parts with individuals in the image. Where a PAF is defined as "a set of 2D vector fields that encode the location and orientation of limbs over the image domain". Due to the bottom-up nature of their approach is the algorithm's runtime not bound by the number of people in the image and does it not suffer from early commitment to perhaps faulty person detection like top-down approaches do. The OpenPose pipeline consists of the following steps: the entire image is used as input for a CNN, in which the body part confidence maps and the PAFs are jointly predicted. Afterwards, body part candidates are connected by performing bipartite matchings. Finally, all body parts are assembled into full body poses for all people in the image.

2.3 Signal Smoothing

Since all frame-based movement measuring techniques are prone to some form of error, it is necessary to correct the signal. In general will there be three types of noise in the data: (1) measurement noise, caused by variation in the image, (2) missing detection, which occurs when a body part could not be found in an image, and (3) wrong detection, for example when a body part is mistaken for another body part. These errors may be caused by events such as camera position changes, self-occlusion, lighting changes or tracking failure.

Measurement noise can be removed by applying a threshold, however this will also reduce the system's ability to pick up small movement. An alternative to applying a threshold is applying signal smoothing techniques. Throughout literature several signal smoothing techniques have been used, such a as moving average [45, 49] and a moving median [47].

Missing detections in data can be solved by interpolation, by replacing them with the mean/median of the entire dataset or by omitting them. Wrong detections are the hardest kind of noise to correct, because they are hard to detect and even if they are detected, assigning the data to the correct body part may be difficult because it is not always clear what the correct body part should be. This problem becomes increasingly difficult the longer the mistake persists. Since there is no single optimal solution to this problem, solutions are usually pragmatical. These errors may for example be solved by omitting all measurements within the timespan in which wrong detection errors were made.

However, opting to remove data introduces 'gaps' in the time series. This introduces another problem with respect to time series alignment. Usually this problem is solved pragmatically rather than optimally, because no one optimal solution exists.

Moving Average Paulick et al. [45] and Ramseyer and Fabian [49] used moving average with a window size of 0.4 seconds has been used to reduce noise caused by signal-distortion in motion energy analysis. Moving average can be mathematically defined as follows. Consider dataset $X = \{x_1, x_2, ..., x_n\}$ of n data points. The moving average filter generates an output dataset by sliding a window over every data point in X and taking the average of all elements within the window centered around the data point. The size of this window is defined by r. Formally, the output value of data point i will be the average of all elements in $\{x_{i-r}, x_{i-r+1}, ..., x_{i-1}, x_i, x_{i+1}, ..., x_{i+r-1}, x_{i+r}\}$.

Moving Median Poppe et al. used moving median to correct data distortions caused by measurement noise and longer-term inconsistencies in motion capture data due to equipment or transmission failure [47]. They used a modest window size, in the range of [0.25, 0.5] seconds. They noted that the window size is a trade-off between the ability to suppress inaccuracies in the output and the level of detail that is retained in the measurements. To emphasise their motivation for choosing to use moving median instead of moving average, they provide an illustration shown in Figure 2.5.

In [39], Moore and Jorgenson described the median filter mathematically as follows. The median filter takes as input dataset $X = \{x_1, x_2, ..., x_n\}$ of n data points, uses a filter rank r, where $n > r \geq 0$

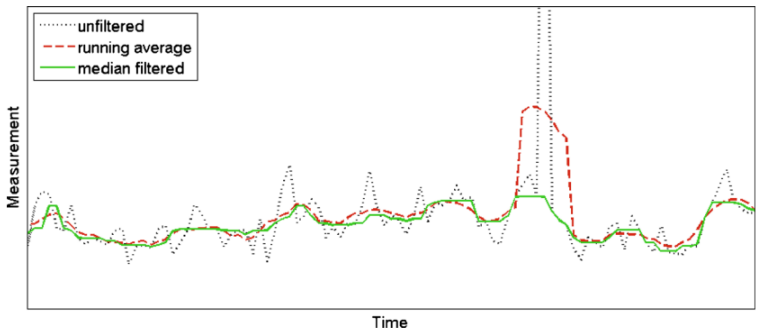

Figure 2.5: Illustration of the smoothing achieved by a moving median filter and a moving average filter, provided by [47].

and gives a filtered dataset Y with the same dimensions as X as output. Each point in Y is the median of a subset of $2r + 1$ data points centered on the corresponding point in X. Formally, the elements of Y are calculated by $y_i = median(J_i)$, for $i = 0, 1, 2, ..., n - 1$, where $J_i = \{x_{i-r}, x_{i-r+1}, ..., x_{i-1}, x_i, x_{i+1}, ..., x_{i+r-1}, x_{i+r}\}$. They note that the median filter preferentially removes sharper peaks and passes broader features, and its discrimination between sharp and broad is controlled by the value of the filter rank, r. Lower values of r give smaller windows and only remove the sharpest peaks, while higher values of r give larger filtering windows and can result in the removal of even relatively broad peaks from the input data.

2.4 Time Series Analysis Method

Once motion energy time series have been collected, an automated determination of synchrony using time series analysis methods (TSAMs) can be made. As the name suggests, TSAMs can measure the quantity and intensity of synchronous movement by analysing the similarity between time series. However, not every TSAM analyses time series in the same manner; there exist several distinct analysis methods [58].

The first distinction that can be made is between global and local analysis approaches, which depicts whether the time series will be analysed as a whole versus splitting up the time series and analysing it partially. For local methods another distinction can be made between overlapping windows and non-overlapping windows, which will decide how the time series' data is split. Thirdly, a distinction can be made between whether to correlate the time series or to use regression. Finally, a distinction can be made between the output scores of TSAMs. These distinctions should be carefully considered when implementing a TSAM, because not every TSAM provides the same output, rather they each make different assumptions about the manifestation of synchrony.

Global vs. local Regarding the length of the window sliding over the time series a distinction can be made between global and local. A global approach will set the window size equal to the length of the time series, which will result in an estimate using the time series as a whole. Two methods using the global approach are cross-lagged correlation (CLC) and cross-lagged regression (CLR). These

approaches calculate the Pearson correlation coefficient or regression, respectively, by looking at both entire time series, however the starting point of each time series may differ. If the starting points are not equal the difference is referred to as time lag. Currently no best amount of time lag has been found, and is up to the researcher to come up with an appropriate value. The advantage of using global approaches is that it is computationally less expensive than its local counterpart. However, one disadvantage is the necessity of the global stationarity assumption: the mean and variance of the time series have to remain constant. Furthermore, it requires that the person who leads the conversation and sets the pace remains constant as well, which is rarely the case in natural conversations. In order to eliminate the need for these assumptions local methods were created.

Local approaches calculate the Pearson correlation coefficient or regression between parts of the time series. These approaches use a sliding window to go over the time series part-wise rather than using the time series as a whole. Implementations of local approaches are windowed cross-correlation (WCC), windowed cross-lagged correlation (WCLC) [10], and windowed cross-lagged regression (WCLR) [2]. By using sliding windows the stationarity assumption no longer has to apply globally, but only locally. The size of the sliding window is referred to as the window size. Just as with finding the appropriate time lag, finding the appropriate window size is also up to the researcher. A too large window size will decrease the benefits of a local stationarity assumption, however settings window size too small may result in the inability to pick up on synchronous movement over a large time span.

Overlapping windows vs. non-overlapping windows When using local approaches another aspect that needs to be considered is whether to allow the sliding windows to overlap. Using overlapping windows will be computationally more expensive, however it has the ability to find synchrony everywhere within the time series. Opting for non-overlapping windows may result in synchronous movement being undetected if it happened over a splitting point. Therefore, overlapping windows are usually preferred.

Correlation vs. regression As mentioned before, TSAMs calculate either Pearson correlation coefficient or regression to determine the relationship between the time series. Even though correlation approaches can be viewed as single predictor regression, the advantage of using regression over correlation is the ability to also take autocorrelation into account. In CLR two predictors are used, the first predictor is autocorrelation and the second predictor is cross-correlation. If the model including autocorrelation and cross-correlation cannot explain the data significantly better than the model that only incorporates autocorrelation, the movement is categorized as non-synchronous.

Output score The final distinction can be made with respect to output scores. There are several distinct outcomes TSAMs can give. In general a TSAM's output score is average synchrony, maximum synchrony and/or frequency of synchrony. Which output score should be used as an indication of synchrony strength is dependent on the research question and is left for the researcher to decide. For example, in [2], Altmann used the frequency of synchrony, which is the summative length of all synchronous sequences of an episode in proportion to the episode length, as an output score to measure the strength of synchrony between children playing a game. Even though the type of output score

could serve as an indication to guide the definition of synchrony, this notion is largely lacking from any working definition of synchrony.

2.4.1 Windowed Cross-Lagged Correlation

In [10], Boker at al. created the windowed cross-lagged correlation (WCLC) method to analyse the association between two variables without need for the stationarity assumption. Eliminating the need for stationarity is interesting when trying to understand how adaptable creatures such as ourselves behave. Especially when considering correlations in exchanges between two individuals, because their behavior will not remain constant throughout a conversation due to adaptation to each other. WCLC is categorized as a local TSAM and employs overlapping windows to ensure correlation can be detected in either direction and at any moment in the time series. The output of WCLC are estimates of both the strength of peak association and the time lag when the peak association occurred. WCLC has become a standard method to analyse the linear relationship between two time series [2], and is used in [10] and [48].

To measure how the relationship between variables change over time WCLC calculates the Pearson product moment correlations between the two windowed slices of the time series. The algorithm has been created to work on time series with an equal interval of time between observation, however it could be adapted to work on unequal intervals. The advantage of using partial data obtained denoted with the windows on the time series is that the global stationarity assumption is reduced to a local stationarity assumption. Furthermore, by allowing windows to overlap and by splitting the time lags WCLC is able to calculate a moving estimate of association and lag without favoring one variable over the other.

WCLC is able to do so in the following manner: suppose we have two time series, each with N data points and equal interval between subsequent observations, $X = \{x_1, x_2, x_3..., x_N\}$ and $Y = \{y_1, y_2, y_3..., y_N\}$. Further suppose a window size w_{max}, a time lag τ on the integer interval $-\tau_{max} \leq \tau \leq \tau_{max}$ and an index i denoting the time within the time series. For each $i = \{\tau_{max} + 1, \tau_{max} + 2, ..., N - \tau_{max} - w_{max}\}$. A pair of windows W_x and W_y can be selected from X and Y respectively as follows:

$$W_x = \begin{cases} \{x_i, x_{i+1}, x_{i+2}, ..., x_{i+w_{max}}\}, & \text{if } \tau \leq 0 \\ \{x_{i-\tau}, x_{i+1-\tau}, x_{i+2-\tau}, ..., x_{i+w_{max}-\tau}\}, & \text{otherwise} \end{cases} \tag{2.1}$$

$$W_y = \begin{cases} \{y_{i+\tau}, y_{i+1+\tau}, y_{i+2+\tau}, ..., y_{i+w_{max}+\tau}\}, & \text{if } \tau \leq 0 \\ \{y_i, y_{i+1}, x_{i+2}, ..., y_{i+w_{max}}\}, & \text{otherwise} \end{cases} \tag{2.2}$$

The cross-correlation between W_x and W_y can now be defined as:

$$r(W_x, W_y) = \frac{1}{w_{max}} \sum_{i=1}^{w_{max}} \frac{(W_{x_i} - \mu(W_x))(W_{y_i} - \mu(W_y))}{\sigma(W_x)\sigma(W_y)}, \tag{2.3}$$

By differentiating the items within W_x and W_y based on τ, the overall strength and lag of the correlation will not be biased. By selecting the windows according to Equations 2.1 and 2.2 mirror synchrony is guaranteed, if the variables X and Y were to be swapped, then the cross-correlations remain the same, but in reverse order. A visualisation of how the windows slide over the data as well as the resulting cross-correlation matrix is given in Figure 2.6. The resulting matrix will have a number of columns equal to $(\tau_{max} * 2) + 1$ and number of rows equal to the largest integer less than $(N - w_{max} - \tau_{max})/w_{inc}$.

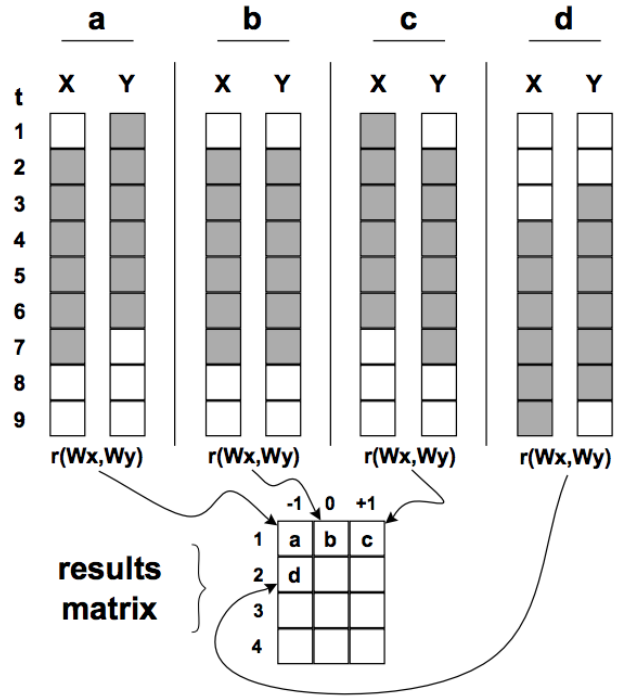

Figure 2.6: Illustration of window sliding over the dataset, provided by [10]. In this example a maximum time lag $\tau_{max} = 1$, the time increment $\tau_{inc} = 1$, the window size $w_{max} = 6$ and the window increment $w_{inc} = 2$ have been selected.

Selecting the four parameters of WCLC is no trivial task. There are two variables related to the sliding window: window size and window increment. The window size, w_{max}, depicts the number of samples that fit within a window. By setting the window size too small, the reliability of the correlation estimate will be reduced. However it should be small enough to ensure little change in who leads the conversation within the window's time frame. The window increment, w_{inc}, is the number of samples the window slides after the correlation for each time lag has been calculated. If the window increment is too small, there may be little change in successive rows in the result matrix. Short window increments also lead to large numbers of rows in the results matrix. On the other hand, if the window increment is too big, there may be so much change that successive rows in the results matrix will appear to be unrelated. Therefore, the window increment should be big, but still small enough that the relation between successive rows in the results matrix persists.

Besides the window parameters, there are also two time lag related variables: maximum time lag and time lag increment. The maximum time lag, τ_{max}, is the maximum time interval between the selected windows. A large maximum time lag will allow us to analyse synchrony with longer delays, however a large maximum time lag will also result in a large number of columns in the result matrix.

The time lag increment, τ_{inc}, is the number of data points a window slides after the correlation has been calculated. Small lag increments lead to little difference between successive columns in the result matrix and will result in many columns in the results matrix. On the other hand, long lag increments lead to seemingly unrelated successive columns.

Peak-picking To estimate the time lag between the association of two time series WCLC is extended with peak-picking. The goal of peak-picking is to find the difference in starting points of an event in W_x and a similar event in W_y. By using peaks as representations for events, events can be compared and deemed related to each other. The time lag can then be estimated by investigating the interval in between the two similar events.

The peak-picking algorithm estimates the time lag by finding the peak cross-correlation that is closest to a time lag of zero. Considering the results matrix in Figure 2.6, the search starts from column 0 and moves outwards, because peaks with a low time lag are most likely to be related. They defined a peak to be "a maximum value of cross-correlation centered in a local region in which values are monotonically decreasing on each side of the peak". It is up to the researcher to define the size of the local region. Peak-picking takes as input one row of the WCLC results matrix and finds a peak by starting with the element with a lag of zero. Once a peak has been found will the peak-picking algorithm return two numbers, the lag of the selected peak relative to the element with zero lag and the value of the crosscorrelation at that peak. These numbers are found by incrementally increasing the search region, until a search region is centered over a peak. Then the lag value, which is derived from the index of the element centered at the peak, and the cross-correlation value are returned.

2.4.2 Windowed Cross-Lagged Regression

Altmann argued that windowed cross-lagged correlation (WCLC) does not take into account the possibility to get a significant cross-correlation of two time series which are independent from each other [2]. Such spurious cross-correlations could be arised if both time series are auto-correlated (cyclic). In other words, WCLC could be biased with auto-correlation and consequently will the conclusions about the occurrence of movement synchrony be biased too. Therefore did they develop a new method called windowed cross-lagged regression (WCLR), which tackles the auto-correlation problem by using regression rather than correlation.

First a window size (W_{max}) and range of time lags (τ_{max}) is defined and the regression is computed in the same window-wise manner as in WCLC. To measure synchrony for each position t in the time series and relative time lag τ, WCLR uses two models:

$$\text{Model 1: } X_{t+\tau} = \beta_0 + \beta_1 X_{1t} + \epsilon_{1t} \tag{2.4}$$

$$\text{Model 2: } X_{t+\tau} = \beta_0 + \beta_1 X_{1t} + \beta_2 X_{2t} + \epsilon_{1t} \tag{2.5}$$

Where model 1 only keeps track of the auto-correlation and model 2 keeps track of both the auto-correlation and cross-correlation. The variance explained by cross-correlation can now be defined

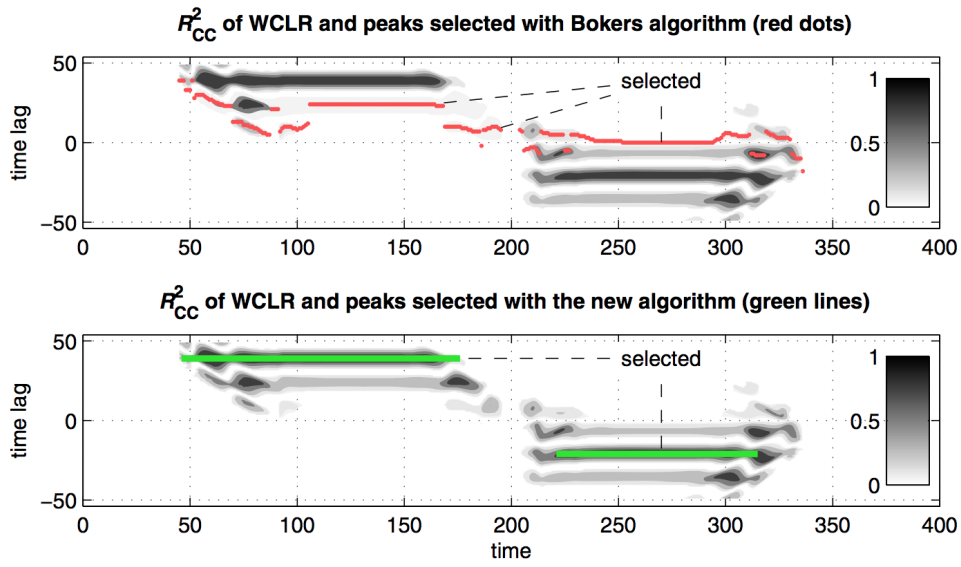

Figure 2.7: Peaks selected by the original peak-picking algorithm by Boker et al. [10] and peaks selected by the adjusted peak-picking algorithm, provided by [2].

by:

$$R^2_{CC} = R^2_{Model2} - R^2_{Model1} \tag{2.6}$$

Where R^2_{Model2} and R^2_{Model1} are the coefficient of determination of models 2 and 1 respectively. If $R^2_{CC} > 0$, then the model with cross-correlation fits better for the given t and τ than the model with just auto-correlation.

Peak-picking Altmann implemented his own peak-picking algorithm to find intervals of synchrony. For each R^2_{CC} peak, the neighbouring peaks are identified and combined into a line, as can be seen in Figure 2.7. Where neighbouring peaks are said to form a line if their time lags are within a certain distance from the time lag of the original peak. It is up to the researcher to set the distance threshold. If there are multiple lines spanning the same time frame, then the line with the highest average R^2_{CC} is selected to best represent the synchronous interval, where lines span the same time frame if both lines contain share one or more peaks. The beginning and end of a line represents the beginning and end of synchronous movement.

2.4.3 Recurrence Quantification Analysis

In [50], an alternative method to measure synchrony is proposed, called recurrence analysis (also known as recurrence quantification analysis, and cross recurrence quantification analysis). Recurrence analysis is originally designed to find recurring patterns in datasets and the main benefits of recurrence analysis are its ability to take auto-correlation into account and its unboundedness by the stationarity constraint. The unboundedness of the stationarity contraint allows recurrence analysis to work with

varying orientation of synchrony. This allows us to analyse human social interaction in a more realistic way, since human interaction usually has more variation in who leads and who follows throughout a conversation.

In recurrence analysis the degree of synchrony depends on how often two time series are in similar states, where a state refers to a subset of observations and are considered to be similar if the observations display a similar pattern. If the time series are in a similar state, a recurrence point for that point of time is created. Next, the method quantifies those recurrences and outputs a two-dimensional plot of the recurrence points matrix. It does so as follows: consider a time series $X = \{x_1, x_2, x_3, ..., x_N\}$ of N numerical measurements. By using a window with a window size, w_{max}, a set of time-delayed vectors from these time series can be constructed, referred to as an "embedded" time series $\mathcal{E}\{X\}$.

$$\mathcal{E}\{X\} = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, ..., \mathbf{x}_{N-w_{max}+1}\}, \text{ where } \mathbf{x}_i = (x_i, x_{i+1}, ..., x_{i+w_{max}-1}) \tag{2.7}$$

For example, consider the time series $X = \{1, 2, 3, 4, 5\}$, by using $w_{max} = 3$, the following embedded time series can be created: $\mathcal{E}\{X\} = \{(1, 2, 3), (2, 3, 4), (3, 4, 5)\}$. Once embedded time series are created, a recurrence plot (RP) can be created by creating recurrence points for vectors in the embedded time series that lie below threshold ϵ according to distance measure d.

$$RP = \{(i, j) \mid (d(\mathbf{x}_i, \mathbf{x}_j) < \epsilon\}, \text{ where } \mathbf{x}_i, \mathbf{x}_j \in \mathcal{E}\{X\} \tag{2.8}$$

An RP is therefore a visualisation of points in time where the time series moved in a similar trajectory. An example RP is given in Figure 2.8 and is provided by Webber et al. [70] where recurrence points are denoted by darkened pixels located at specific i, j coordinates. These RPs were created by analysing breathing patterns of unrestrained rats, where (A) shows quiet breathing and (B) shows active breathing.

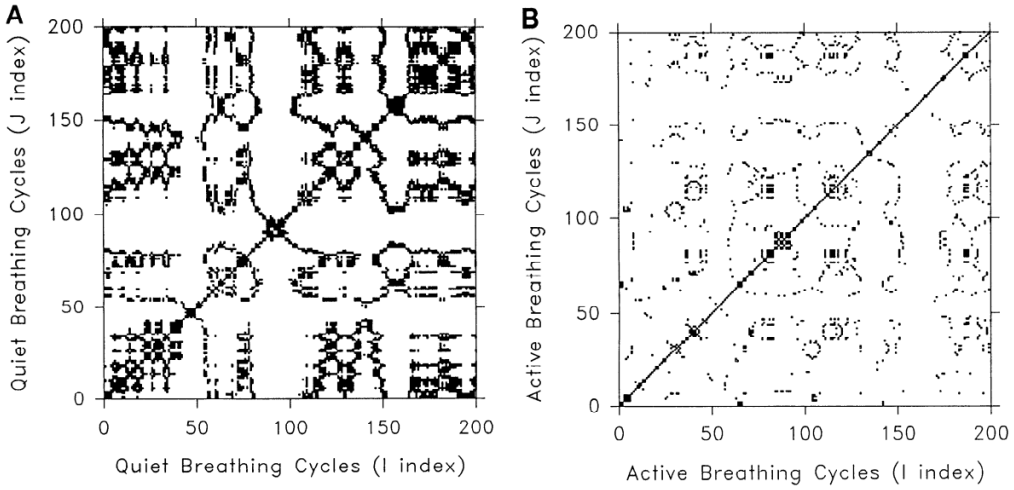

Figure 2.8: Visualisation of recurrence plots for (A) quiet breathing patterns and (B) active breathing patterns of unrestrained rats where recurrence points are denoted as darkened points. Recurrence plots reveal dramatic qualitative difference between quiet breathing (more complex) and active breathing (less complex). Visualisation provided by [70].

This method can be extended to be able to compare time series with each other by comparing recurrent vectors of each time series, rather than comparing recurrent vectors within just one time series. The output of the extended method is referred to as cross-recurrence plots (CRP).

$$CRP = \{(i,j) \mid (d(\mathbf{x}_i, \mathbf{y}_j) < \epsilon\}, \text{ where } \mathbf{x}_i \in \mathcal{E}\{X\}, \mathbf{y}_j \in \mathcal{E}\{Y\} \tag{2.9}$$

Another way to define recurrence and cross-recurrence plots is by using a matrix notation rather than a set notation [37, 35, 69]. In these works they defined the recurrence plot R and cross-recurrence plot CR as follows:

$$R_{i,j} = \Theta(\epsilon - ||\mathbf{x}_i - \mathbf{x}_j||), \qquad i,j = 1, ..., N, \tag{2.10}$$

$$CR_{i,j}^{\mathcal{E}\{X\},\mathcal{E}\{Y\}}(\epsilon) = \Theta(\epsilon - ||\mathbf{x}_i - \mathbf{y}_j||), \qquad i = 1, ..., N,\ j = 1, ..., M, \tag{2.11}$$

where $\mathbf{x}_i \in \mathcal{E}\{X\}$, $\mathbf{y}_j \in \mathcal{E}\{Y\}$, N is the number of data points in $\mathcal{E}\{X\}$, M is the number of data points in $\mathcal{E}\{Y\}$, and Θ is the Heaviside function.

To quantify synchrony Webber and Zbilut proposed five variables to depict similarity in structure of a recurrence plot [70] and their mathematical definitions using matrix notations are given in [37, 35, 69]. The first variable is recurrence rate (RR) or percent recurrence, which quantifies the relative number of recurrence points in relation to the total amount of points in the RP. Data displaying a pattern will in general result in a higher percent recurrence than random data. Mathematically, recurrence rate is defined as:

$$RR(\epsilon) = \frac{1}{N^2} \sum_{i,j=1}^{N} R_{i,j}(\epsilon) \tag{2.12}$$

The next variables depend on the histogram $P(\epsilon, l)$ of diagonal lines of length l, defined as follows:

$$P(\epsilon, l) = \sum_{i,j=1}^{N} (1 - R_{i-1,j-1}(\epsilon))(1 - R_{i+l,j+l}(\epsilon)) \prod_{k=0}^{l-1} R_{i+k,j+k}(\epsilon) \tag{2.13}$$

Where N represents the length of the data series, and ϵ may be omitted for readability. The second variable, percent determinism (DET), quantifies the percentage of recurrent points that form diagonal structures. Random data tends to display only short diagonal structures, whereas data displaying a pattern will display long diagonal line segments in the RP. Diagonal structures in the CRP indicate periods of time in which similar phase space behaviour occurred in both time series [36].

$$DET = \frac{\sum_{l=l_{min}}^{N} lP(l)}{\sum_{l=1}^{N} lP(l)} \tag{2.14}$$

The third variable is entropy (ENTR), which quantifies the complexity of recurrence plots by constructing a histogram of diagonal line segment lengths.

$$ENTR = -\sum_{l=l_{min}}^{N} p(l) \ln p(l), \tag{2.15}$$

where $p(l)$ is the probability to find a diagonal line of length l in the RP or CRP, defined as $p(l) = P(l)/N_l$, where N_l is the number of diagonal lines.

The fourth variable is ratio and represents the ratio between percent determinism and percent recurrence.

$$RATIO = N^2 \frac{\sum_{l=l_{min}}^{N} lP(l)}{(\sum_{l=1}^{N} lP(l))^2} \tag{2.16}$$

Finally, the fifth variable, trend, is defined as the slope of the best fitted drift, where drift is the percentage of recurrence points in long diagonals parallel to the central line and is plotted as a function of distance away from the central diagonal. Trend provides information about the stationarity versus nonstationarity in the process. The downside of the trend variable is that it is very sensitive to the window size and small changes in the window size can reveal even contrary results [35]. Trend is based on the τ-recurrence rate for those diagonal lines with distance τ from the line of identity. The τ-recurrence rate RR_τ is defined as follows:

$$RR_\tau = \frac{1}{N-\tau} \sum_{l=1}^{N-\tau} lP_\tau(l) \tag{2.17}$$

Now trend can be defined as:

$$TREND = \frac{\sum_{\tau=1}^{\tilde{N}} (\tau - \tilde{N}/2)(RR_\tau - \langle RR_\tau \rangle)}{\sum_{\tau=1}^{\tilde{N}} (\tau - \tilde{N}/2)^2}, \tag{2.18}$$

where $\langle x \rangle$ is the mean of x, and $\tilde{N}$ is defined by the researcher and depends on the studied process. The trend variable is greatly affected by the window size [35].

Besides measures based on diagonal line structures, several measures have been proposed for vertical line structures. Vertical and horizontal lines indicate that the phase within the phase space did not change for some time [38]. In [37] three measures based on vertical structures have been used. First histogram P of vertical line segment length v is created.

$$P(v) = \sum_{i,j=1}^{N} (1 - R_{i,j})(1 - R_{i,j+v}) \prod_{k=0}^{v-1} R_{i,j+k} \tag{2.19}$$

The first measure of vertical line segments is similar to percent determinism variable, and is called laminarity (LAM). Laminarity is the ratio between the recurrence points forming vertical lines in an RP and the entire set of recurrence points. LAM will decrease if the RP consists of more individual recurrence points than vertical structures.

$$LAM = \frac{\sum_{v=v_{min}}^{N} vP(v)}{\sum_{v=1}^{N} vP(v)} \tag{2.20}$$

The second measure is called trapping time (TT) and represents the average vertical line segment length. TT estimates the mean time that the system will remain in a specific state or how long the state will be trapped.

$$TT = \frac{\sum_{v=v_{min}}^{N} vP(v)}{\sum_{v=v_{min}}^{N} P(v)} \tag{2.21}$$

Finally, the third measurement representing the maximum vertical line segment length (V_{max}).

$$V_{max} = max(\{v_l\}_{l=1}^{N_v}), \tag{2.22}$$

where N_v is the absolute number of vertical lines.

2.5 Surrogate Testing

A critical question when attempting to measure synchrony is where the boundary between scores indicating significant and insignificant synchrony should be [17]. Ramseyer and Wolfgang proposed a method based on pseudo-interactions to create this distinction, called surrogate testing [48]. With respect to dyadic movement time series, the method consists of generating surrogate data by isolating each person from a video and randomly combining them with isolated persons from another video, thereby creating pseudo-interactions. The scores assigned to the pseudo-interaction therefore represent coincidental synchrony and can be used as a baseline for judging scores of original interaction.

The output of all surrogate data methods is 'new' data created by rearranging an already available original dataset. Several methods exist to rearrange data, such as jackknife, randomisation, permutation, shuffling and bootstrap. An overview of the difference between these methods is provided in Table 2.1.

- Jackknife: a resampling technique especially useful for variance and bias estimation. It recomputes the statistical estimates after drawing a subset from the available dataset. The method does not use replacement and therefore removes the sampled data from the available dataset after it has been drawn.

		Sample Size (for 1 dataset)	
		Subsample	*Full Sample*
Sampling Method	*Without Replacement*	Jackknife	Randomization Permutation Shuffling
	With Replacement		Bootstrap

Table 2.1: A 2x2 taxonomy of data rearranging methods used to create a surrogate database from [48]

- Randomization, permutation, shuffling: resampling techniques that alter or restructures the available dataset.
- Bootstrap: a resampling technique that randomly draws sample data with replacement, therefore the same data sample may be drawn more than once.

Which sampling method should be used, depends on the research application at hand. When it comes to resampling data of social interaction, choosing a technique with replacement may result in a new interaction in which the same individual was chosen twice. On the other hand, allowing replacement generally produces better results when the available dataset is small and its distribution and characteristics are identical to the bootstrap data [48].

Chapter 3

Method

This chapter provides a description of the implemented approach to measure synchrony as well as an in-depth description of the algorithm's pipeline. The goal of the algorithm is to quantify synchrony between individuals of a dyadic interview. An overview of the algorithm's general pipeline is provided in Figure 3.1.

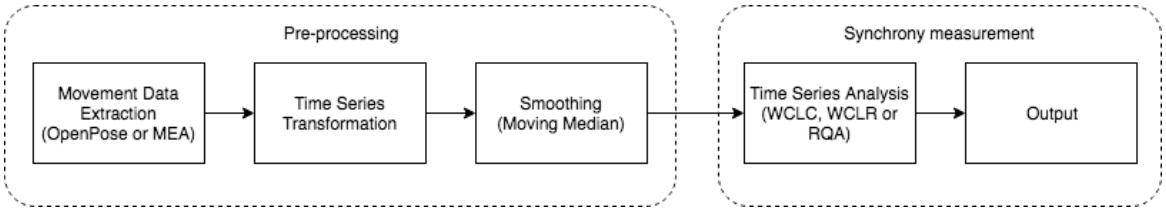

Figure 3.1: Schematic of the algorithm's general pipeline.

First of all, the algorithm begins with a pre-processing phase in which movement data is obtained from the video and transformed into motion energy time series, as is explained in Section 3.1. The motion energy time series are required by the time series analysis method in order to quantify synchrony. Two methods are used to extract from videos: OpenPose [12] and Motion Energy Analysis (MEA) [23, 58], these methods are further elaborated in Section 3.1.1 and Section 3.1.2, respectively. MEA and OpenPose are used, because they are unobtrusive and require no specialized equipment to extract movement data. They can also be applied in a post-processing fashion, allowing us to extract movement data from the video data that came without extra information about the location or movement of the subjects. Several human pose estimators have been tested on our dataset, such as DensePose [24], AlphaPose [18, 72], Associative Embedding [41] and OpenPose [12]. Their outputs were compared based on their ability to place keypoints at the correct location within the image and on their level of noise. OpenPose is used, rather than one of its alternatives, because it is easy to use and generally provides good pose estimations in our data and has relatively little noise. After motion energy has been obtained, the data is transformed into time series so that they can be compared using a time series analysis method. Afterwards, the time series will be smoothed using a moving median filter to reduce noise which may be caused by lighting changes or camera position changes, as is explained in Section 3.1.3, and will be corrected for the difference in body sizes per individual. The moving median filter is chosen rather than the moving average filter, because the moving median will be less

influenced by outliers.

The second phase of the algorithm quantifies synchrony by analysing the movement time series of each individual. The time series obtained in the pre-processing phase will be used as input for one of three time series analysis methods: windowed cross-lagged correlation (WCLC) [10], windowed cross-lagged regression (WCLR) [2], and recurrence quantification analysis (RQA) [50]. These three time series analysis method are chosen, because they have been used in synchrony research before and a comparison between the three will help to understand the relative advantages and disadvantages. Furthermore, because they are not bound by a global assumption of stationarity are they well suited for measuring synchrony. Since they do not require the leader and follower of synchronous behaviour to be constant throughout the video [58], can they deal with the orientation and temporal aspects of synchrony. Descriptions of how each method analyses the time series and quantifies synchrony are given in Sections 3.2.1, 3.2.2, and 3.2.3, respectively.

3.1 Pre-processing

In this first step, the input videos of dyadic interviews is passed through either OpenPose or MEA to extract movement data, resulting in a list of 2D keypoint positions or in a list of motion energy scalars, respectively. From these lists, movement time series are created by parsing the list by a method tailored to the chosen movement extraction method, as is described in their respective section below.

3.1.1 OpenPose

OpenPose is a 2D human pose estimator that is able to jointly detect keypoints in the human body, hand, face, and foot in images. By passing a frame through the CNN can OpenPose find and output the 2D location per keypoint. Which keypoints OpenPose tries to find depends on the supplied body model.

Body model For the purpose of this thesis we used the 25-keypoint body model, containing only keypoints for the body and feet, thereby excluding keypoints for the hand and face models. This model is sufficient, because we are only interested in relatively large movements, which will not be significantly be influenced by facial movement or the movement of individual fingers. On top of this, due to the viewpoint and way the individuals face the camera, hands and face are often occluded. An illustration of the 25-keypoint body model is provided in Figure 3.2. By tracking these keypoints is OpenPose able to capture the movement of behaviours that involve movement of the torso, arms, legs, feet and head, such as crossing of the legs or head scratching. However, it will not be able to pick up movement of finer behaviours, such as smiling or finger tapping.

Filtering OpenPose occasionally assigns some keypoints of a single person to another non-existent person, resulting in two partial sets of keypoints, which is the first problem that has to be solved.

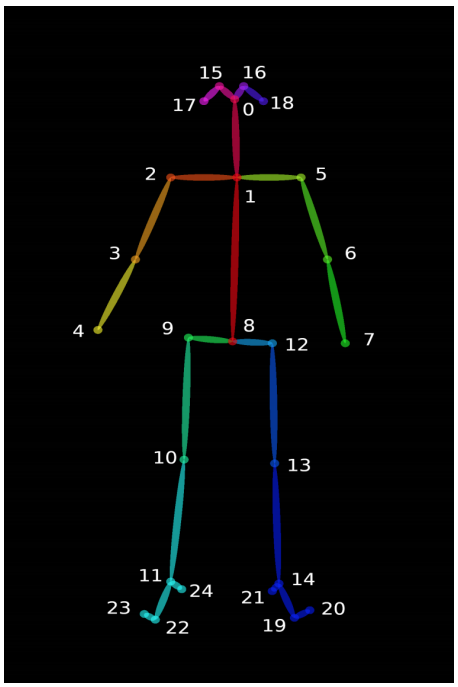

Figure 3.2: OpenPose 25 keypoint body and feet model, provided by [12]

However, since there is no way of knowing whether these keypoints belong to the same person or whether they belong to a third person that happens to be in the same general location, only the two persons with the highest number of keypoints found will be used for synchrony estimation. The missing set of keypoints are linearly interpolated in a later stage of the filtering pipeline. This method also allows us to remove the keypoints of the researcher, that were sometimes found at the beginning of the video, at the end of the instructional phase. Since the researcher is always only partially in view, the number of keypoints found for the researcher is always lower than the number of keypoints found per participant, therefore the researcher is always excluded from the synchrony measurement.

Another problem is that, as of this moment, OpenPose does not have the ability to track people over frames, but rather finds people from scratch in every single frame without using information from previous frames. Therefore, it is not guaranteed that OpenPose finds people in the same order throughout the video. If the order is mixed up, the movement of one person will be attributed to the movement of the other person. These temporary confusions cause outliers in the motion energy time series for both participants, since all of their keypoint locations made considerable jumps for the duration of confusion. To remedy this, for each frame, the distance between successive keypoints in original order and in mixed order are compared. If the keypoint distance is smaller in the mixed order, then a vote is cast in favour of switching. After all keypoints have been checked and the number of votes cast in favour of switching is greater than half the number of keypoints, then all keypoints in that frame will be switched between the participants.

The keypoint estimations detected by OpenPose are also prone to noise, which may be caused by lack of contrast between clothing and background or due to poor lighting. Therefore several filtering techniques are applied to remedy these spurious keypoint locations after each keypoint has been assigned to the correct person. First of all, keypoints which are rarely found throughout the video

are excluded from the motion energy calculation, because these keypoints cannot be reliably used to calculate movement between frames. Often the keypoints 19, 20, 21, 22, 23, and 24 located in the feet of the 25 keypoint body model shown in Figure 3.2 could not be reliably found and are not excluded from motion energy calculation.

Furthermore, the confidence score OpenPose assigns to keypoints makes it possible to exclude keypoint locations that OpenPose did not confidently find. OpenPose assigns low confidence scores when it is uncertain whether the keypoint location is accurate or if it is noise. Removing these keypoints can be beneficial, because keypoints with a low confidence score are most likely noisy estimations. However, OpenPose still occasionally wrongly estimates a keypoint location with high confidence. These locations are often quite far off from the correct keypoint location, resulting in incorrect movement vectors. These incorrect keypoint locations are removed by applying a confidence threshold and extreme movement threshold, respectively. Thereby removing all keypoint locations whose confidence score does not exceed the confidence threshold or whose distance in successive frames exceeds the extreme movement threshold.

As a final step, linear interpolation is used to fill the gaps in keypoint positions caused by keypoint removal or by failure to track the keypoint. Linear interpolation suffices, because when the frame rate is sufficiently high, then humans will not be able to make large complex movement in between frames. If not enough keypoint locations have been found and interpolation cannot be done, then the entire list of keypoint locations is ignored in the motion energy calculation. If there are locations missing at the beginning or at the end of the keypoint location sequence, then the closest found location is copied over the gaps. Thereby not influencing the motion energy, since the distance between the successive locations in these gaps will then be 0. Extrapolation could not used for this, because occasionally the number of missing entries at the beginning or end of the keypoint location sequence is large enough that values will be set beyond the size of the frame.

Time series The time series are created by transforming the original keypoint locations after they have been filtered. Time series are created by calculating movement vectors between frames. Where movement vectors are calculated by taking the Euclidean distance between keypoint locations in successive frames, resulting in a list of movement vectors per keypoint. Afterwards, the sum of the length of all movement vectors of a frame is used to define motion energy scalars per frame, resulting in a motion energy time series.

However, these motion energy scalars still have to be normalized between participants. Motion energy may be exaggerated or underestimated depending on how tall an individual is and on the individual's distance to the camera [47]. For example, if both individuals are the same height, yet person 1 is closer to the camera than person 2, then even if they make the same movement, person 1's estimated movement will be greater than person 2's. Therefore, the pose is normalized by z-transforming all keypoint positions after smoothing. By calculating the mean and standard deviation of the motion energy scalar distribution, each motion energy scalar can be replaced with the amount of standard deviations it is from the mean.

3.1.2 Motion Energy Analysis

An alternative to using OpenPose for human movement extraction in video is Motion Energy Analysis (MEA) [58, 23]. MEA measures movement by counting the pixels, within a predefined region of interest, whose color change in successive frames exceeded a threshold. Where the region of interest usually is a bounding box surrounding the subject.

Person distinguishment Unlike OpenPose is MEA not able to find the number of persons in a frame and their respective location automatically. The researcher will have to define the number of participants and their location beforehand. A naive approach would be to split the frame in the center between individuals, but this approach is susceptible to attributing irrelevant background noise to movement of an individual. This problem can be reduced by introducing a region of interest by defining a bounding box.

A bounding box defines the region of interest per person in order to reduce the influence of background noise by reducing the background considered in the movement estimation process. In general, the bounding box will be centered around the person of interest and be large enough to completely envelop the person whilst leaving room for movement, yet still small enough to not include unnecessary background information. Usually this bounding box will be set manually the researcher. However, due to our access to the OpenPose keypoint locations is it possible to use the extremes of these locations to define a bounding box without manually going over every video. By sorting the keypoint locations and taking the element located with an index of 95% of the length of the keypoint location list, we can make sure that one of the highest or lowest keypoint locations is chosen, whilst making sure no outlier is chosen. The last few element of the list should not be chosen, because these may be outliers and therefore do not reliably represent the true extreme keypoint locations. Afterwards, a padding is added to account for the keypoints not being located at the edges of the subject. An illustration of the bounding boxes this method created is shown in Figure 3.3

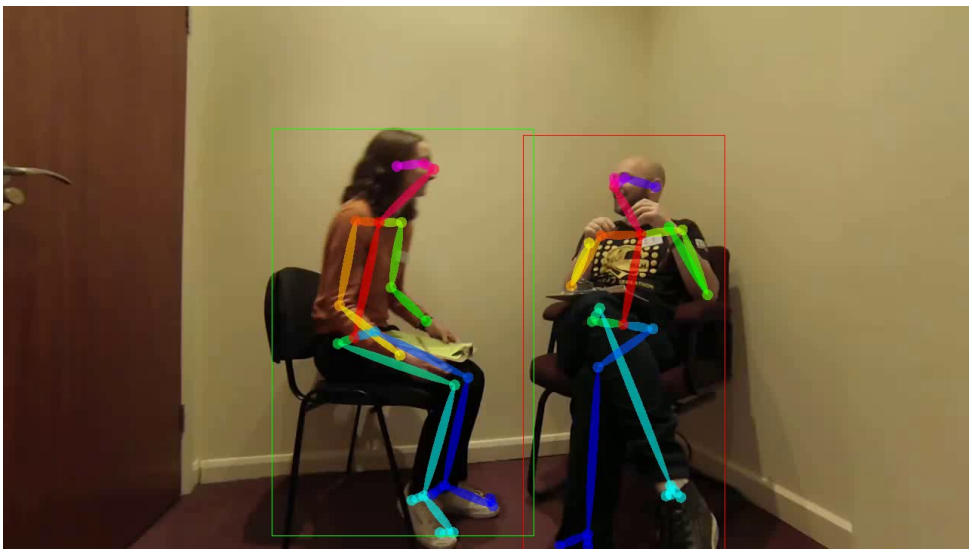

Figure 3.3: Bounding box created by taking the extremes of OpenPose keypoint average locations and adding a padding.

Movement threshold MEA uses a movement threshold in order to make distinctions between pixel color changes that were caused by movement and pixel color changes that are noise. This threshold is usually set by the researcher and is largely dependant on the environment. The threshold should be high enough to exclude noise, but still low enough to accept actual movement.

Time series MEA calculates motion energy scalars by counting the number of pixels whose color value change in successive frames exceed the movement threshold. To reduce the influence light has on the color changes are all pixel color values transformed into grayscale values before they are compared. After the movement time series is created, it is smoothed using a moving median filter to remove incidental outliers from the time series.

Since MEA does not account for the people sizes or the distance of the individual to the camera. Therefore the time series is normalized by dividing the number of changed pixels by the total number of pixels in the bounding box. Thereby preventing these variables from influencing the perceived motion energy and make comparison between the two individuals more equitable.

3.1.3 Smoothing

Since both movement measuring methods are susceptible to noise is there a need to reduce the influence of noise without deforming the underlying patterns. In MEA the movement threshold already reduces general noise present in the data, however this will often not be enough to fully filter out background noise. OpenPose is less susceptible to changes in the background. However, it is still prone to noise caused by occlusion. These erroneous keypoints can be partially corrected by removing keypoint locations with a low confidence score, removing keypoint locations that are too far away from the previous location, and finally interpolation. Afterwards, smoothing is applied to further reduce the noise. Smoothing attempts to capture important structures in data while ignoring noise [59]. Throughout literature, two smoothing methods have been prominently used to smooth time series: moving average and moving median. Both smoothing methods require a window size to be set. One important thing to keep in mind when skipping frames is that the range of the window increases.

Moving Median Moving median was chosen, rather than moving average, because the moving median smoothing method is less prone to outliers, because the median will not be as influenced by outliers, given that the ratio between outliers and regular data points is small. The moving median smoothing method smoothes the time series by sliding a window over the entirety of the time series. The smoothed value is calculated by taking all values within the window and take the median. The value at the center of the window is then replaced with the smoothed value.

3.2 Synchrony Measurement

After the time series have been created and smoothed, they will be passed to the second phase of the algorithm: synchrony measurement, in which synchrony will be quantified. Three synchrony

measurement methods have been implemented: windowed cross-lagged correlation (WCLC), windowed cross-lagged regression (WCLR), and recurrence quantification analysis (RQA).

3.2.1 Windowed Cross-Lagged Correlation

WCLC measures synchrony by sliding a window over both time series, which can at most be a maximum time lag apart. It then computes a correlation matrix by calculating the Pearson correlation between the items in both windows for each window and time lag step. To provide insight in the dynamical structures of the correlation matrix, it is usually displayed as a heatmap, as is shown in Figure 3.4.

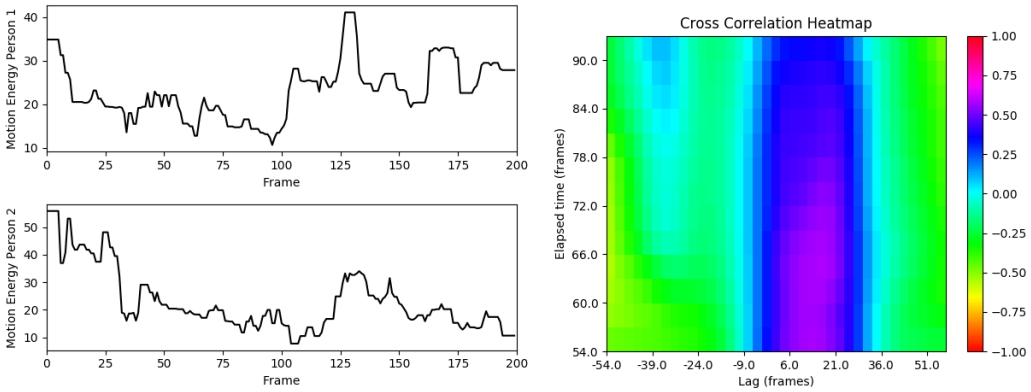

Figure 3.4: On the left are two time series 200 frames long. On the right is the corresponding correlation heatmap. Window size = 108, window increment = 3, max lag = 54, lag increment = 3.

This correlation matrix is then reduced to an estimate of synchrony by subjecting the rows of the correlation matrix to a peak-picking algorithm.

Peak-picking The peak-picking algorithm [10] iterates over every row of the correlation matrix and finds the peak correlation value and its corresponding time lag, where the search starts in the center of the matrix, at a time lag of 0, and moves outwards towards the maximum time lag in both directions. A peak is defined as a data point whose neighbouring data points, within a local region, are monotonically decreasing on both sides of the peak. The number of neighbouring data points that the local region covers is usually defined by the researcher. Before peaks are identified, the correlation matrix is first subjected to linear loess smoothing in an effort to reduce high frequency noise [15]. Loess smoothing smoothes values in a window-wise fashion using locally weighted regression. It fits a linear or quadratic fitting function through the portion of the data that is inside the window, whose size is defined by the span. The window slides over the dataset in a similar fashion as the moving median filter does. Linear spline fitting may be applied if intermediate results are required, which may be the case if the sampling rate is low. However, in our case, there is no need for linear spline fitting, since our dataset has a sufficiently high sampling rate. Although frame skipping decreases the number of samples per second, linear spline splitting was not used. Linear spline fitting remained unnecessary because the parameters are defined in seconds and tuned to ensure that enough samples will be covered.

One problem that may occur is that a peak with a smaller value is chosen than a peak with a larger value on the opposite sign, because the smaller peak is closer to a lag of 0. To reduce this problem, a peak will not be selected if values of the opposite sign are monotonically increasing. However, if no peak is found, because there is no clear pattern in the data, then a peak with correlation 0 and time lag 0 is returned. A peak with 0 lag and 0 correlation is returned, rather than not returning any peak, to make sure not only relevant time is considered. If the rows in which no peak is found are ignored, then the peak distribution will be skewed in favor of synchronous time sequences. Such a skewed peak distribution will result in a higher synchrony quantification, despite there not being more evidence suggesting that the synchrony quantification should indeed be higher. After all peaks have been found, a peak distribution is created and an output can be formulated. The output of the peak-picking algorithm is the average, standard deviation, and max peak value of the peak correlation and peak time lag distributions.

3.2.2 Windowed Cross-Lagged Regression

It can be argued that WCLC may provide biased output, because it does not take auto-correlation into account when quantifying synchrony. If auto-correlation is to be taken into account the time series can be passed to WCLR. WCLR investigates both time series in the same window-wise manner as WCLC, however fits two models and compares their coefficients of determination rather than calculating the Pearson correlation. Therefore, making this approach more suitable in situation where people move randomly or move completely unaware of each other. In these situations the difference between coefficients of determination will be small. On the other hand, it will still be possible for WCLC to assign a high Pearson correlation value to these cases.

Model fitting WCLR quantifies synchrony within a time frame by comparing the coefficient of determination of two fitted models over the data points within the time frame. One model only considers auto-correlation and thus only uses the previous movement of the individual as a dependent variable. Whilst the other model takes into account auto-correlation as well as the previous movement of the other individual. Synchrony within the time frame is then quantified as the difference between the coefficient of determination of the model taking both persons' movement into account and the coefficient of determination of the model taking only auto-correlation into account.

Peak-picking The peak-picking algorithm of WCLR is similar to the one of WCLC, however has been slightly adjusted by the authors of WCLR, because it could not correctly identify peaks in the regression matrix rows. The main difference between the two peak-picking algorithms is that the adjusted version takes structures over time into account, whereas the original version does not. It was adjusted to select successive peaks that have the same time interval, a succession of peaks with a similar time lag is displayed as a black line if the regression matrix were to be shown as a heatmap, as is shown in Figure 2.7. Peaks are said to be similar if the the difference between their time lags does not exceed a threshold. If there are multiple lines within the same time frame, then the line with the highest average regression value is chosen. Lines are said to span the same time frame if one or

more of their points are in both lines. The start point and end point of each line define a period of synchrony. The output of the adjusted peak-picking algorithm is the ratio between the sum of the length of all synchronous time frames and the total time.

3.2.3 Recurrence Quantification Analysis

The final synchrony measurement method is recurrence quantification analysis (RQA). It differs from the previous two synchrony measurement methods in that it does not consider the time series in the same window-wise manner, but rather looks at how often a system revisits states in any point of time. Due to this approach does RQA not assume a linear relationship between the two time series. This method can be extended by comparing states of one system to states of another system, allowing it to be used to analyse how often the state of one time series corresponds with the state of another time series. However, one drawback of this method is that it considers many points in time that are not close enough to each other, thereby ignoring the temporal aspect of synchrony.

To analyse how often a system revisits states, states must first be defined first. The set of all states a system has visited over time is called an embedded time series. An embedded time series is the time series transformed to contain only sets of data points that together form a state. How many data points are combined into a state depends on a window size.

After both embedded time series are created, RQA creates a recurrence matrix, in which moments in time where both systems were in similar states are represented with a 1 and 0 otherwise. States are deemed similar if the distance, according to some distance measure, between the states is smaller than a threshold set by the researcher. The distance measure used is Euclidean distance between the states that are represented as vectors.

From the recurrence matrix a histogram of diagonal line segments and vertical line segments can be computed. These diagonal structures are interesting, because they represent time intervals in which both systems transitioned into similar states in the same order. Vertical structures represent points in time where the system stayed in the same state. Using the lengths and frequencies of these histograms, several measurements can be made, such as recurrence rate, determinism, entropy, trend, laminarity, trapping time, maximum vertical line segment, maximum diagonal line segment. The mathematical definitions as well as further elaboration on these variables is given in Section 3.2.3. Exemplary histograms of dyads achieving low synchrony output scores and high synchrony output scores are displayed in Figure 3.2 and 3.1, respectively. The figures show that the histogram with a higher synchrony output score contains a higher number of diagonal structures, as well as longer diagonals than the histogram with a low synchrony output score.

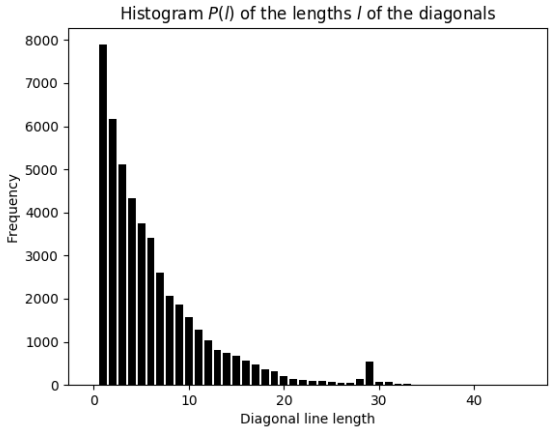

Table 3.1: Histogram of a dyad assigned a low synchrony output score (percent determinism = 0.4552).

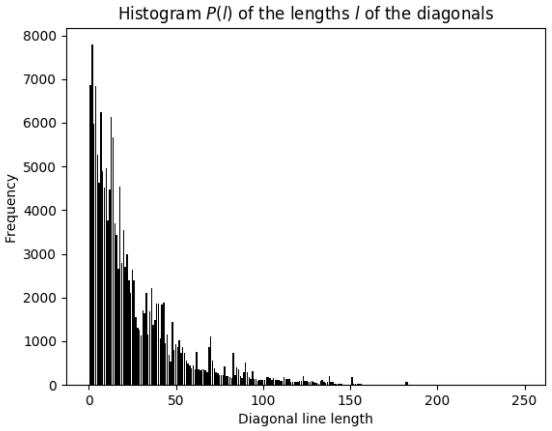

Table 3.2: Histogram of a dyad assigned a high synchrony output score (percent determinism = 0.9344).

Output Since we are interested in how two systems transitioned similarly in their respective state phases are only diagonal line segment measures included. The computation of vertical structure measurements is unnecessary, because the duration in which a system stayed in the same phase is not relevant for measuring synchrony. The measurements the algorithm takes into account are: recurrence rate, percent determinism, entropy, average diagonal line segment length, and longest diagonal line segment length. The recurrence rate represents the relative number of recurrence points in relation to the total amount of points in the cross-recurrence plot. Percent determinism quantifies the percentage of recurrence points that form diagonal structures. Entropy measures the complexity of recurrence plots by constructing a histogram of diagonal line segment lengths. The average diagonal line segment length represents the average length of all diagonal segments in the recurrence plot. Finally, the longest diagonal line segment length represents the length of the longest diagonal line in the recurrence plot.

Chapter 4

Evaluation

This chapter first describes the dataset that was used to run the experiments on, then provides a setup of the experiments and an overview of the used settings, and finally gives a discussion of the results using the methods explained in Chapter 3.

4.1 Experiments

Since rapport is closely related to synchrony, as is explained in Section 2.1, rapport can be used as a proxy for synchrony. Therefore rapport is used in the assessment of a synchrony estimation. Therefore, the assumption is made that synchrony should increase between dyads after rapport-building training has been received. This assumption is made to accommodate for the lack of ground-truth. How well a synchrony measurement method performs will therefore be measured by its ability to assign increasing scores for dyads that did receive rapport-building training, yet assign similar scores to dyads that did not receive rapport-building training. Scores are deemed similar if their respective difference does not exceed a threshold, which is uniquely set per output score for each synchrony measurement method.

The method's ability to make distinction between trained and untrained interviewers is quantified as the average F-score of the rapport-building trained interviewers and the interviewers that did not receive the training. If no distinction is made and all interviewers are said to have received the rapport-building training, then the average F-score will be 0.4359, because the F-score for trained interviewers will then be 0.8718, however the F-score for the control interviewers will be 0. On the other hand, if all interviewers are labelled as control, the average F-score will be 0.1852, because the F-score assigned to trained interviewers will be 0 and the F-score assigned to control interviewers will be 0.3704. The boundary delimiting similar output scores from distinct enough output scores is set using a threshold. This threshold is set for each experiment setting and is found by testing every value between 0 and 1 with a step size of 0.001. The threshold that corresponds to the highest average F-score is used.

To investigate how well methods generalise, leave-one-out cross-validation is used. A total of 5 groups are used to investigate the ability of a method to distinguish trained interviewers from control interviewers. Each group contains 2 or 3 rapport interviewers and 1 control interviewer. Several values per

setting will be tested resulting in an average F-score per setting per set of groups. Afterwards, the average F-score for each setting will be taken; the highest average F-score will correspond with the optimal value for that setting.

The hardware used for all experiments has been provided in the Lisa computer cluster from SurfSara. An overview of the hardware used is listed on the following website: `https://userinfo.surfsara.nl/systems/lisa/description`.

Despite OpenPose's general good results in finding keypoints, several keypoints were still hard to find mostly due to dark same colored clothing resulting in a lack of contrast. These keypoints were mostly located in the feet. Because feet were rarely moved on their own and therefore barely influence the movement vector were they removed from the calculation, because their positions are not reliably detected. These set of keypoints that are ignored from the body model shown in Figure 2.4 are: 19, 20, 21, 22, 23, and 24.

The first experiment that will be performed will investigate the influence of frame skipping on the output score, the setup of the experiment is described in Section 4.1.2 and its results are shown in Section 4.2.1. The second experiment investigates how the parameter settings of each synchrony measurement method influences the output score and its ability to distinguish rapport-building trained interviewers from control interviewers. The settings of the second experiment are shown in Section 4.1.3 and its results are shown in Section 4.2.2. The third experiment compares the three synchrony measurement methods with each other, the settings are shown in Section 4.1.4 and its results are shown in Section 4.2.3. The fourth experiment investigates how time series generated by MEA and OpenPose influence the output score of WLCR, the settings are described in Section 4.1.5 and its results are shown in Section 4.2.4.

WCLR has only one output score that can be considered, which is the ratio between the synchronous time fragments and the total time. WCLC on the other hand, has two output scores, the average and the standard deviation of the peak correlation distribution. When estimating the ability of WCLC to distinguish trained interviewers from control interviewers, the average of the peak distribution output score will be used, because this provided better F-scores. The percent determinism output score of RQA is used, because it is strongly related to the diagonal structures present in the cross-recurrence plot. These diagonal structures represent points in time in which both systems transitioned between states in a similar manner and is therefore able to capture the temporal aspect of synchrony. Furthermore, percent determinism is chosen, because it has been used throughout literature to study synchrony [29, 30, 68].

For all experiments which only require the use of a single synchrony measurement method, WCLR is used, because it gives a single output variable which can easily be compared across experiments. The WCLR settings are tailored to the values used in [10] and [2]: window size of 4 seconds, window increment of 1/10 second, maximum time lag of 2 seconds and a lag increment of 1/10 second. However, greater durations in seconds are used to compensate for the lower frame rate of 27 frames per second in our data, versus the frame rate of 80 frames per second in the original papers, to make sure enough samples are taken into consideration. Another benefit of increasing parameter values is that the algorithm will be able to run faster, due to the increased window increment and lag increment.

The recurrence threshold used in RQA is set along the lines of [73], in which the threshold is set in such a way that the recurrence rate is approximately 1%. The goal of this guideline is to set a threshold in such a way that it is big enough to not only consider noise, yet small enough to only capture values that reoccur. Following this guideline the threshold was set so that the average recurrence rate of all videos processed with a frame skip of 0 is approximately 1%.

The default settings for experiments are as follows:

- MEA
  - Grey value difference threshold: 10
  - Frame skip: 0
- WCLR
  - Window size: 12 seconds
  - Window increment: 0.3 second
  - Maximum time lag: 6 seconds
  - Lag increment: 0.3 second
  - Local region size: 3
  - Minimum synchrony line length: 0.5 second
  - Allowed lag difference: 2
- Moving Median Filter
  - Window size: 0.5 second

4.1.1 Data

To run the experiments a subset of the dataset provided by Wright et al. [71] is used. The original dataset contained videos from different viewpoints as is shown in Figure 1.1. However, only the videos taken from the center viewpoint have been used, because this viewpoint captures both participants in the fairest way. Excluding participants that did not partake in both waves, a total of 87 videos are used. Before these videos were used, they had to be transformed. Because the videos from this viewpoint have been captured using a GoPro Hero 4, they suffer from a fisheye distortion. To remove this distortion from the frame it was multiplied using the GoPro's corresponding camera matrix.

In the original dataset all videos were split up into a number of different files. These files were reconstructed into a single video. Another problem was the presence of the researcher in the beginning the videos, because the instructional part of the experiment is not relevant for measuring synchrony it was trimmed from the videos.

Because the hardware used for these experiments was powerful enough, no frames were skipped and no resizing was applied. The size of the videos is 1280x720 and the frame rate is 27 frames per second.

4.1.2 Frame Skip

The first experiment will investigate the effect frame skip has on the synchrony estimation. This experiment is done first, because its result may be used to speed up other experiments. To analyze the effect frame skip has on synchrony estimation the following frame skips were used on the dataset for each synchrony measurement method: 0, 1, 2, 3, and 4. Since the frame rate of the data is 27 frames per second, the investigated frame rates are: 27, 14, 9, 7, and 5 frames per second. Trying values beyond a frame skip of 4 are not considered, because it would skip many small movements and decrease the number of samples required to detect synchrony resulting in many spurious synchrony detections. The influence of frame skip on the synchrony measure will be tested on all 87 videos for each synchrony measurement method.

Increasing the frame skip decreases the number of frames per second, thereby increasing the range of the window and time lag variables. To account for the affect of frame skip on window and time lag span, are the variables declared in seconds and their exact value is calculated according to the following formula: number of seconds * 27 frames per second / (frame skip + 1).

The settings differ from the default settings as follows:

- MEA
  - Frame skip: 0, 1, 2, 3, and 4

4.1.3 Parameter Settings

This experiment investigates the influence of the parameter settings of each synchrony method on the output score and its ability to distinguish rapport-building trained interviewers from control interviewers. To investigate how each parameter influences the output score and the ability to distinguish interviewers, several values will be tested for each of the parameters.

The following parameter settings will be tested:

- WCLR
  - Window size: 8, 10, 12, 14, and 16 seconds
  - Window increment: 1/10, 2/10, 3/10, 4/10, and 5/10 second
  - Maximum time lag: 2, 4, 6, 8, and 10 seconds
  - Lag increment: 1/10, 2/10, 3/10, 4/10, and 5/10 second
  - Minimum synchronous line length: 3/10, 4/10, 5/10, 6/10, and 7/10 second
- WCLC
  - Window size: 8, 10, 12, 14, and 16 seconds
  - Window increment: 1/10, 2/10, 3/10, 4/10, and 5/10 second

  - Maximum time lag: 2, 4, 6, 8, and 10 seconds
  - Lag increment: 1/10, 2/10, 3/10, 4/10, and 5/10 second

- RQA
  - Embedding dimension: 1/10, 2/10, 3/10, 4/10, and 5/10 second
  - Recurrence threshold: embedding dimension * 0.00001, 0.00003, 0.00005, 0.00007, and 0.00009
  - Diagonal line length threshold: 1/10, 2/10, 3/10, 4/10, and 5/10 second

4.1.4 Synchrony Measurement Method

This experiment investigates the ability of each synchrony measurement method to distinguish interviewers that did receive rapport-building training from interviewers that did not receive this training. To measure this ability the generalised F-scores of each synchrony measurement method are compared. Furthermore, the correlation of the output scores of each synchrony measurement method is investigated.

4.1.5 Motion Energy Time Series

Using the synchrony measurement method that is best at distinguishing rapport-building trained interviewers from control interviewers with its optimal settings, the influence the movement estimator on the ability to distinguish rapport-building trained interviewers from control interviewers is investigated. To find out how movement estimators influences the synchrony measure, both methods: MEA and OpenPose, are used to create motion energy time series of the entire dataset without skipping frames. The resulting time series are analysed using WCLR and their output scores and ability to distinguish rapport-building trained interviewers from control interviewers are evaluated.

The settings of MEA and OpenPose are as follows:

- MEA
  - Grey value difference threshold: 10
  - Frame skip: 0
- OpenPose
  - Confidence threshold: 0.3
  - Extreme movement threshold: 270 per second
  - Frame skip: 0

4.2 Results

This section will provide an overview of the obtained results per research question. The results are shown by providing an overview of the distribution of output scores and the correlation between them. If the experiment is about the ability to distinguish rapport-building trained interviewers from control interviewers, then the F-score representing how well this distinction is made is listed alongside the threshold used to make the distinction. For all experiments that investigate parameter settings the changes per parameter value will also be visualised in a line graph.

4.2.1 Frame skip

Using the settings as described in Section 4.1.2, the influence of frame skip on the synchrony output score is investigated for each synchrony measurement method. The distribution of the output score per frame skip as well as the correlation between the output scores per frame skip are given.

WCLR The synchrony output score of WCLR is the ratio between synchronous time and total time. The WCLR output score distribution for each frame skip are shown in Table 4.3 and in Figure 4.1. The results show that frame skip does influence the output of WCLR. The average synchrony ratio increases alongside the frame skip, however the standard deviation decreases. The correlation between the output scores with frame skip and the output scores without frame skip is shown in Table 4.2. As the frame skip increases, the correlation between the output with the original output decreases. The average F-score per frame skip is shown in Table 4.1.

	Frame Skip				
Leave-out fold	0	1	2	3	4
1	0.8500	0.5333	0.7619	0.9068	0.5500
2	0.7917	0.6606	0.8500	0.9068	0.6400
3	0.8295	0.5333	0.7847	0.7917	0.6591
4	0.8295	0.7222	0.7847	0.9206	0.6400
5	0.7257	0.6761	0.7681	0.7949	0.6444
μ	0.8053	0.6251	0.7899	0.8642	0.6267

Table 4.1: Average F-score achieved by WCLR per fold per frame skip. The bottom row depicts the average score of all leave-out folds.

Frame Skip	Correlation
1	0.8118
2	0.5424
3	0.5287
4	0.5577

Table 4.2: Pearson correlation between output scores using frame skips and output scores without a frame skip

Frame Skip	Avg.	Std. dev.
0	0.4375	0.0744
1	0.6413	0.0546
3	0.7664	0.0359
2	0.8310	0.0306
4	0.8054	0.0315

Table 4.3: Output score distribution of WCLR per frame skip

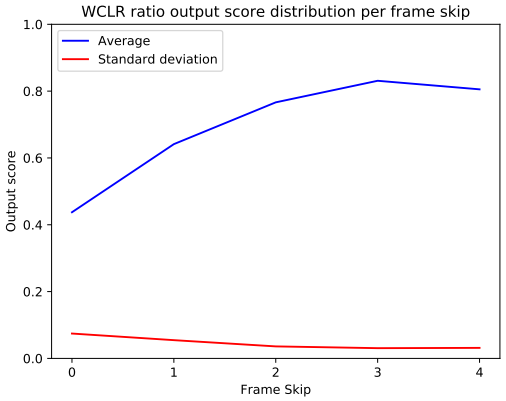

Figure 4.1: WCLR output score distribution per frame skip

WCLC WCLC has two synchrony output scores, the average and the standard deviation of the peak distribution. The Pearson correlation between output scores of WCLC using frame skips and output scores without frame skip is shown in Table 4.5. The average F-score per frame skip using the average peak correlation output score is shown in Table 4.4.

	Frame Skip				
Leave-out fold	0	1	2	3	4
1	0.7205	0.7917	0.6411	0.7917	0.6411
2	0.7205	0.7917	0.6411	0.7917	0.6411
3	0.6400	0.6400	0.7000	0.6400	0.6411
4	0.6591	0.6591	0.7619	0.6591	0.7000
5	0.7257	0.7949	0.7922	0.7949	0.7333
μ	0.6932	0.7355	0.7073	0.7355	0.6714

Table 4.4: Average F-score achieved by WCLC per fold per frame skip using the average peak correlation output score. The bottom row depicts the average score of all leave-out folds.

Frame Skip	Correlation of Avg.	Correlation of Std. dev.
1	0.1776	0.1969
2	0.4479	0.2622
3	0.1413	0.0674
4	0.1024	0.1538

Table 4.5: Pearson correlation between WCLC output scores using frame skips and WCLC output scores without frame skip.

The WCLC peak correlation output score distribution for each frame skip is shown in Table 4.6 and in Figure 4.2. The WCLC peak standard deviation output score distribution for each frame skip is shown in Table 4.7 and in Figure 4.3. The results show that the average and standard deviation of the

output scores are barely affected by frame skip. However, the correlation between the output scores per frame skip is at most 0.4479.

Frame Skip	Avg.	Std. dev.
0	0.2109	0.0462
1	0.2108	0.0463
2	0.2223	0.0473
3	0.2103	0.0468
4	0.2106	0.0466

Table 4.6: Peak correlation output score distribution per frame skip using WCLC

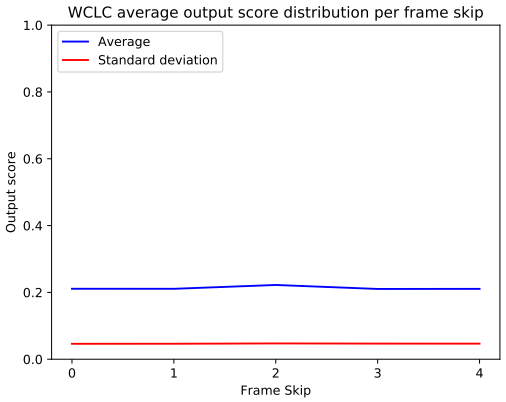

Figure 4.2: Average peak correlation output score distribution of WCLC per frame skip.

Frame Skip	Avg.	Std. dev.
0	0.1992	0.0272
1	0.1993	0.0270
2	0.2056	0.0279
3	0.1998	0.0271
4	0.1997	0.0273

Table 4.7: Peak standard deviation output score distribution per frame skip using WCLC.

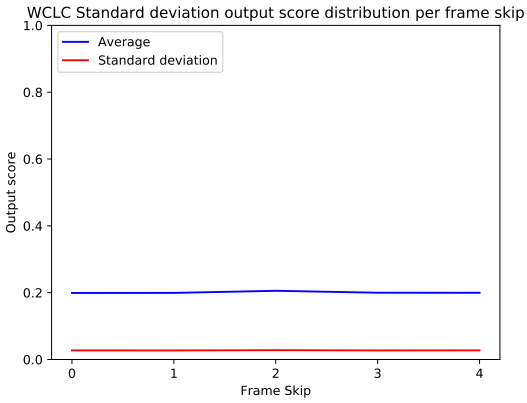

Figure 4.3: Standard deviation output score distribution of WCLC per frame skip.

RQA The output scores of RQA are percent recurrence (%REC), percent determinism (%DET), entropy (ENTR), ratio and average diagonal line length. Although the comparison between average diagonal line length is unfair, because the minimal amount of points in the CRP required to form a line decreases due to frame skip, thereby decreasing the average diagonal line length. The correlation between frame skips for each variable is shown in Table 4.9. The average and standard deviation per frame skip for %REC, %DET, ENTR, ratio and average diagonal line length are shown in Tables 4.10, 4.11, 4.12, 4.13, and 4.14 and Figures 4.4, 4.5, 4.6, 4.7, and 4.8, respectively. The average F-score per frame skip using the percent determinism output score is shown in Table 4.8.

Leave-out fold	Frame Skip 0	1	2	3	4
1	0.5833	0.6032	0.6400	0.5500	0.6400
2	0.4643	0.4976	0.4886	0.4444	0.4444
3	0.5342	0.5833	0.6400	0.5249	0.6400
4	0.5833	0.5833	0.6400	0.4886	0.6400
5	0.5429	0.5897	0.6444	0.5000	0.6444
μ	0.5416	0.5714	0.6106	0.5016	0.6018

Table 4.8: Average F-score achieved by RQA per fold per frame skip using the percent determinism output score. The bottom row depicts the average score of all leave-out folds.

Frame Skip	%REC.	%DET	ENTR	Ratio	Avg. diagonal length
1	0.9927	0.9802	0.9537	0.9658	0.9988
2	0.9839	0.9623	0.9287	0.9771	0.9957
3	0.9778	0.9452	0.8877	0.8770	0.9920
4	0.9738	0.9258	0.8621	0.9445	0.9927

Table 4.9: Pearson correlation between RQA output scores with frame skips and RQA output scores without frame skip.

Frame Skip	Avg	Std. dev.
0	0.0096	0.0119
1	0.0070	0.0091
2	0.0059	0.0080
3	0.0058	0.0076
4	0.0052	0.0071

Table 4.10: Recurrence rate output score distribution per frame skip using RQA.

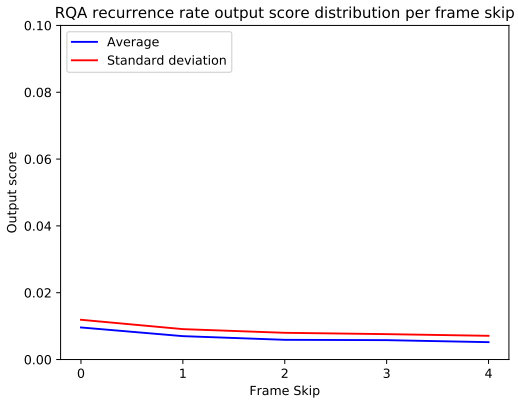

Figure 4.4: Recurrence rate output score distribution of RQA per frame skip.

Frame Skip	Avg.	Std. dev.
0	0.6857	0.1668
1	0.7028	0.1675
2	0.6714	0.1859
3	0.7265	0.1695
4	0.6761	0.1847

Table 4.11: Percent determinism output score distribution per frame skip using RQA.

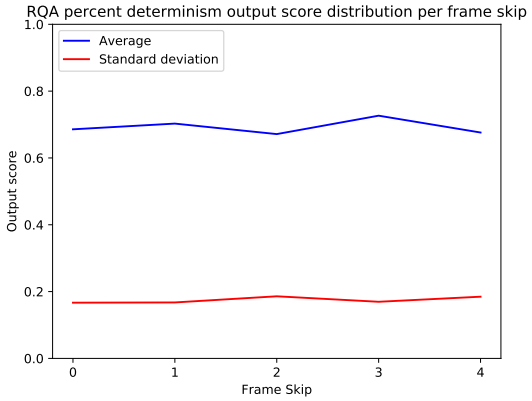

Figure 4.5: Percent determinism output score distribution of RQA per frame skip.

Frame Skip	Avg.	Std. dev.
0	1.4793	0.5237
1	1.3032	0.4894
2	1.1272	0.4590
3	1.1134	0.4249
4	0.9820	0.4069

Table 4.12: Entropy output score distribution per frame skip using RQA.

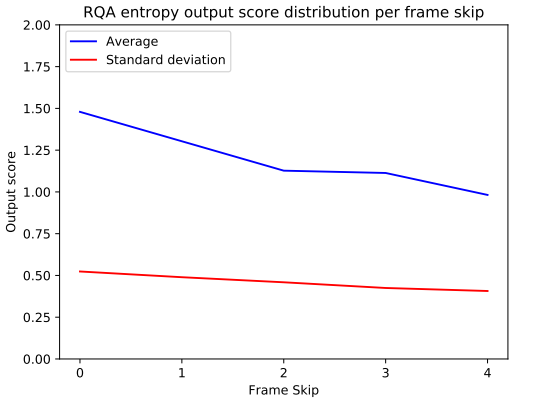

Figure 4.6: Entropy output score distribution of RQA per frame skip.

Frame Skip	Avg.	Std. dev.
0	314.7114	547.8369
1	679.6937	1291.5200
2	956.8003	1901.2531
3	1536.0934	2831.7570
4	1845.1801	261.2299

Table 4.13: Ratio output score distribution per frame skip using RQA.

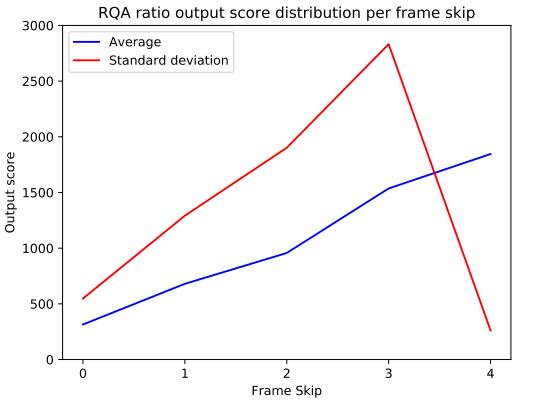

Figure 4.7: Ratio output score distribution of RQA per frame skip.

Frame Skip	Avg.	Std. dev.
0	20.0671	20.1171
1	8.6405	10.2209
2	6.3492	11.2968
3	4.6630	12.4794
4	4.1387	10.0922

Table 4.14: Average diagonal line length output score distribution per frame skip using RQA.

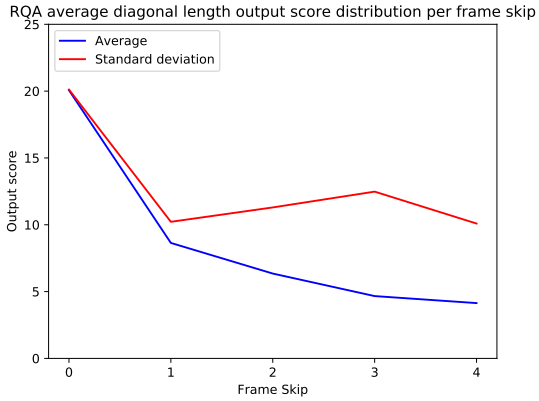

Figure 4.8: Average diagonal line length output score distribution of RQA per frame skip.

4.2.2 Parameter setting

A total of five values will be tested for each parameter settings in order to find the optimal value for the parameters of each synchrony measure method individually. Per synchrony measurement method an overview of the influence of the parameter on the output score distribution and its influence on the F-score are given.

WCLR The influence of the window size (w_{max}), window increment (w_{inc}), maximum time lag (τ_{max}), lag increment (τ_{inc}), and minimum synchronous line length (s_{min}) on the ratio output score distribution is shown in Tables 4.16, 4.18, 4.20, 4.22, and 4.24, and Figures 4.10, 4.12, 4.14, 4.16, and 4.18, respectively. Furthermore, the influence of these parameters on the F-score is shown in Tables 4.15, 4.17, 4.19, 4.21, and 4.23, and in Figures 4.9, 4.11, 4.13, 4.15, and 4.17, respectively.

	w_{max}				
Leave-out fold	8	10	12	14	16
1	0.4886	0.4976	0.8500	0.6400	0.7917
2	0.4444	0.4231	0.7917	0.4231	0.5833
3	0.4976	0.4886	0.8295	0.6400	0.7205
4	0.5500	0.5342	0.8295	0.6400	0.7205
5	0.5636	0.5429	0.7257	0.6444	0.6444
μ	0.5089	0.4973	0.8053	0.5975	0.6921

Table 4.15: Average F-score achieved by WCLR per fold per window size (w_{max}). The bottom row depicts the average score across all folds.

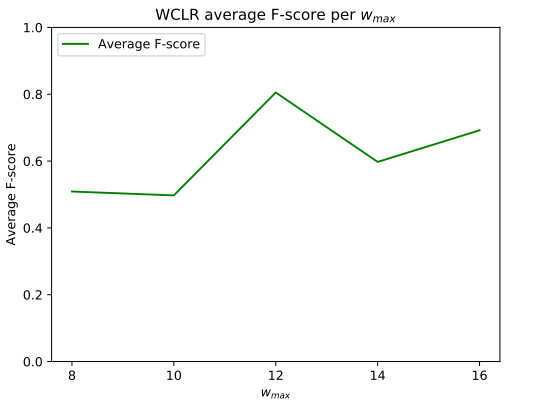

Figure 4.9: The average F-score of WCLR per window size (w_{max}).

w_{max}	Avg.	Std. dev.
8	0.3182	0.0617
10	0.3825	0.0770
12	0.4375	0.0744
14	0.4693	0.0838
16	0.5028	0.0807

Table 4.16: The ratio output score distribution of WCLR per window size (w_{max}) in seconds.

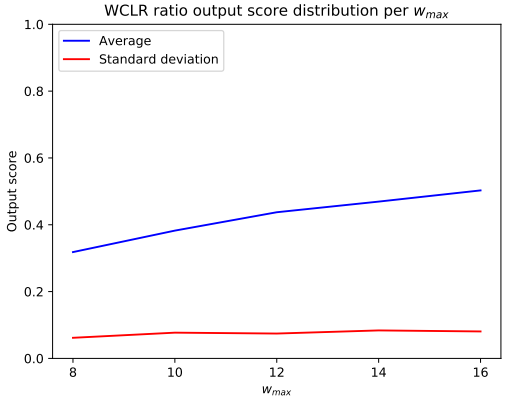

Figure 4.10: The ratio output score distribution of WCLR per window size (w_{max}) in seconds.

Leave-out fold	w_{inc} 0.1	0.2	0.3	0.4	0.5
1	0.5500	0.6591	0.8500	0.6032	0.5342
2	0.5982	0.7917	0.7917	0.5342	0.4586
3	0.5333	0.6591	0.8295	0.6032	0.4586
4	0.6606	0.7619	0.8295	0.5500	0.4231
5	0.6537	0.7681	0.7257	0.6667	0.5000
μ	0.5992	0.7280	0.8053	0.5914	0.4569

Table 4.17: Average F-score achieved by WCLR per fold per window increment (w_{inc}). The bottom row depicts the average score across all folds.

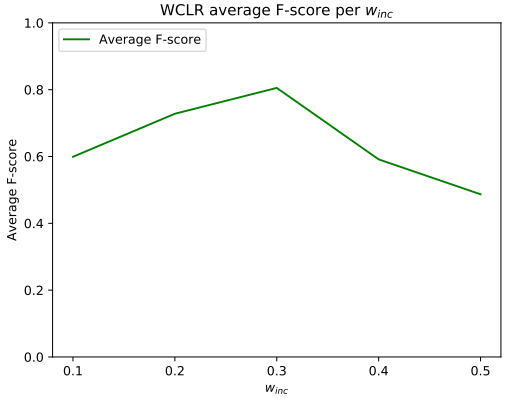

Figure 4.11: The average F-score of WCLR per window increment (w_{inc}).

w_{inc}	Avg.	Std. dev.
0.1	0.6961	0.0449
0.2	0.5682	0.0652
0.3	0.4375	0.0744
0.4	0.3341	0.0859
0.5	0.2543	0.0922

Table 4.18: The ratio output score distribution of WCLR per window increment (w_{inc}) in seconds.

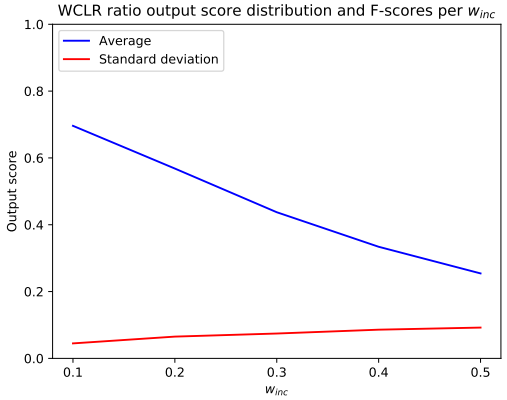

Figure 4.12: The ratio output score distribution of WCLR per window increment (w_{inc}) in seconds.

Leave-out fold	τ_{max} = 2	4	6	8	10
1	0.6032	0.7847	0.8500	0.8295	0.6591
2	0.6591	0.8295	0.7917	0.7847	0.7619
3	0.5342	0.7847	0.8295	0.7917	0.7000
4	0.5500	0.7000	0.8295	0.7847	0.6591
5	0.5152	0.7091	0.7257	0.8333	0.7922
μ	0.5723	0.7616	0.8053	0.8048	0.7145

Table 4.19: Average F-score achieved by WCLR per fold per maximum time lag (τ_{max}). The bottom row depicts the average score across all folds.

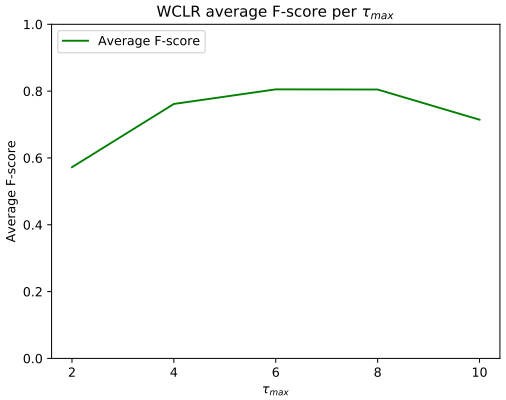

Figure 4.13: The average F-score of WCLR per maximum time lag (τ_{max}).

τ_{max}	Avg.	Std. dev.
2	0.4554	0.0693
4	0.4428	0.0798
6	0.4375	0.0744
8	0.4520	0.0696
10	0.4368	0.0720

Table 4.20: The ratio output score distribution of WCLR per maximum time lag (τ_{max}) in seconds.

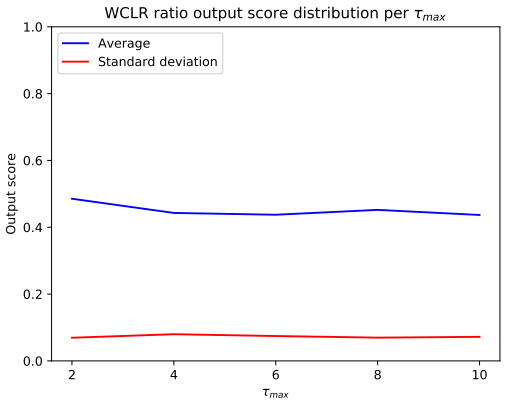

Figure 4.14: The ratio output score distribution of WCLR per maximum time lag (τ_{max}) in seconds.

Leave-out fold	τ_{inc} = 0.1	0.2	0.3	0.4	0.5
1	0.7205	0.5500	0.8500	0.5833	0.7222
2	0.7917	0.6591	0.7917	0.6032	0.5833
3	0.7205	0.5500	0.8295	0.6032	0.5249
4	0.7000	0.5333	0.8295	0.7000	0.6411
5	0.6537	0.5466	0.7257	0.7091	0.6000
μ	0.7173	0.5678	0.8053	0.6398	0.6143

Table 4.21: Average F-score achieved by WCLR per fold per lag increment (τ_{inc}). The bottom row depicts the average score across all folds.

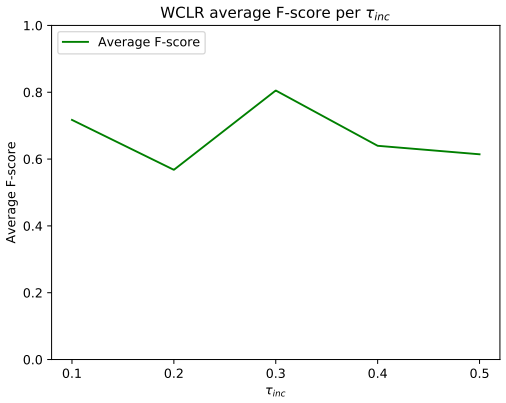

Figure 4.15: The average F-score of WCLR per lag increment (τ_{inc}).

τ_{inc}	Avg.	Std. dev.
0.1	0.3408	0.0746
0.2	0.3840	0.0745
0.3	0.4375	0.0744
0.4	0.4586	0.0718
0.5	0.5084	0.0813

Table 4.22: The ratio output score distribution of WCLR per lag increment (τ_{inc}) in seconds.

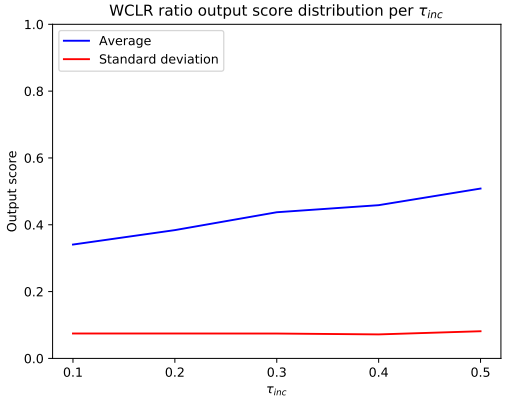

Figure 4.16: The ratio output score distribution of WCLR per lag increment (τ_{inc}) in seconds.

Leave-out fold	s_{min} 0.3	0.4	0.5	0.6	0.7
1	0.5333	0.6591	0.8500	0.5833	0.5342
2	0.6606	0.6591	0.7917	0.7000	0.5833
3	0.5333	0.6032	0.8295	0.6606	0.5342
4	0.7222	0.6411	0.8295	0.6411	0.5333
5	0.6761	0.7257	0.7257	0.6135	0.5429
μ	0.6251	0.6576	0.8053	0.6397	0.5456

Table 4.23: Average F-score achieved by WCLR per fold per minimum synchronous line segment length (s_{min}). The bottom row depicts the average score across all folds.

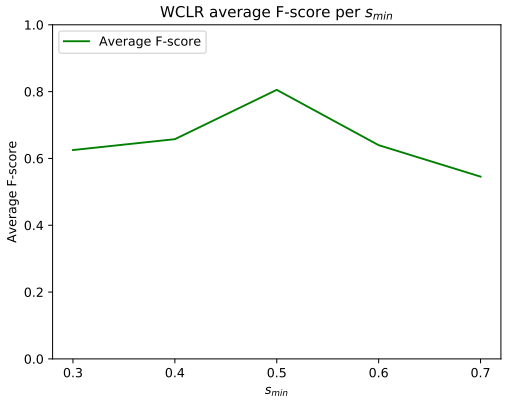

Figure 4.17: The average F-score of WCLR per minimum synchronous line segment length (s_{min}).

s_{min}	Avg.	Std. dev.
0.3	0.6045	0.0622
0.4	0.5116	0.0698
0.5	0.4375	0.0744
0.6	0.3910	0.0785
0.7	0.3304	0.0854

Table 4.24: The ratio output score distribution of WCLR per minimum synchronous line segment length (s_{min}) in seconds.

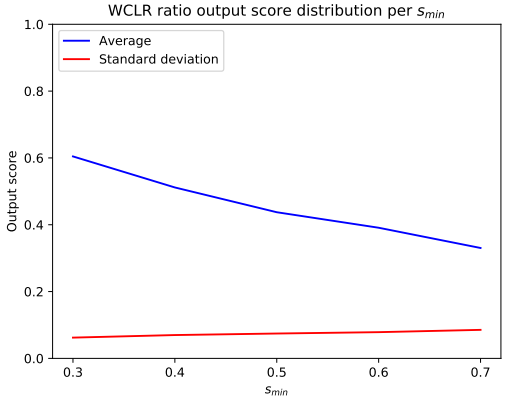

Figure 4.18: The ratio output score distribution of WCLR per minimum synchronous line segment length (s_{min}) in seconds.

WCLC The influence of the window size (w_{max}), window increment (w_{inc}), maximum time lag (τ_{max}), and lag increment (τ_{inc}) on the average peak distribution output score distribution is shown in Tables 4.26, 4.28, 4.30, and 4.32 and in Figures 4.20, 4.22, 4.24, and 4.26, respectively. Furthermore, the influence of these parameters on the F-score is shown in Tables 4.25, 4.27, 4.29, and 4.31, and in Figures 4.19, 4.21, 4.23, and 4.25, respectively.

Leave-out fold	w_{max} = 8	10	12	14	16
1	0.6032	0.7000	0.7205	0.6400	0.6400
2	0.5249	0.7000	0.7205	0.7917	0.6411
3	0.6032	0.7000	0.6400	0.7917	0.7000
4	0.6032	0.8295	0.6591	0.7000	0.7222
5	0.7091	0.8545	0.7257	0.7949	0.7333
μ	0.6087	0.7568	0.6932	0.7436	0.6873

Table 4.25: Average F-score achieved by WCLC per fold per window size (w_{max}). The bottom row depicts the average score across all folds.

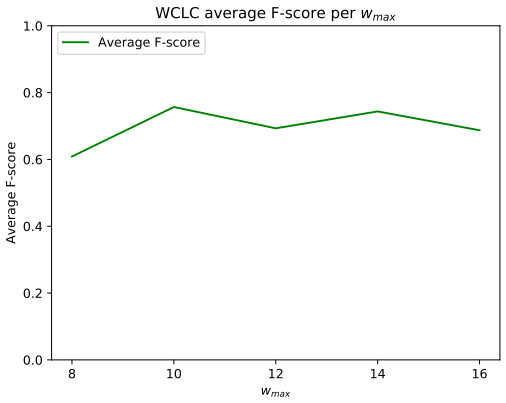

Figure 4.19: The average F-score of WCLC per window size (w_{max}).

w_{max}	Avg.	Std. dev.
8	0.2522	0.0379
10	0.2299	0.0413
12	0.2109	0.0462
14	0.1947	0.0513
16	0.1809	0.0559

Table 4.26: The average of the peak distribution output score of WCLC per window size (w_{max}) in seconds.

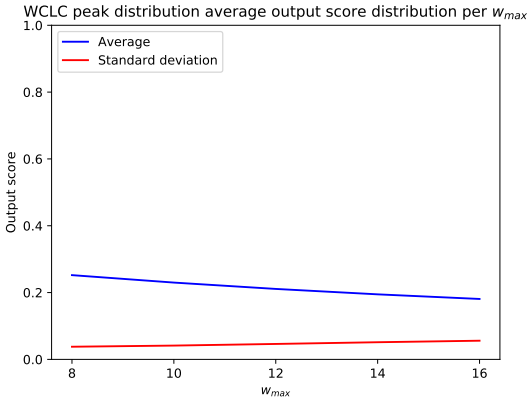

Figure 4.20: The average of the peak distribution output score distribution of WCLC per window size (w_{max}) in seconds.

Leave-out fold	w_{inc} 0.1	0.2	0.3	0.4	0.5
1	0.7205	0.7205	0.7205	0.7205	0.6591
2	0.7205	0.7205	0.7205	0.7205	0.6591
3	0.6400	0.6400	0.6400	0.6400	0.6032
4	0.6591	0.6591	0.6591	0.6591	0.6591
5	0.7257	0.7257	0.7257	0.7257	0.7091
μ	0.6932	0.6932	0.6932	0.6932	0.6579

Table 4.27: Average F-score achieved by WCLC per fold per window increment (w_{inc}). The bottom row depicts the average score across all folds.

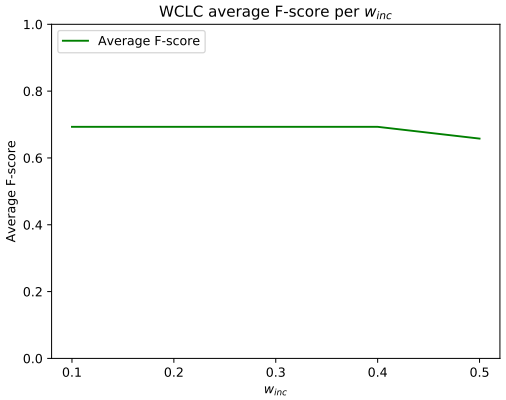

Figure 4.21: The average F-score of WCLC per window increment (w_{inc}).

w_{inc}	Avg.	Std. dev.
0.1	0.2107	0.0462
0.2	0.2107	0.0461
0.3	0.2109	0.0462
0.4	0.2106	0.0462
0.5	0.2109	0.0463

Table 4.28: The average of the peak distribution output score distribution of WCLC per window increment (w_{inc}) in seconds.

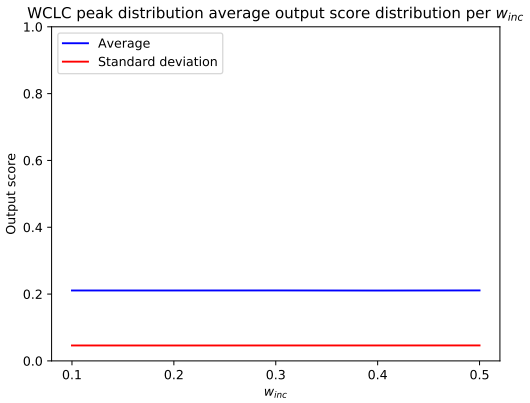

Figure 4.22: The average of the peak distribution output score distribution of WCLC per window increment (w_{inc}) in seconds.

Leave-out fold	τ_{max} 2	4	6	8	10
1	0.6411	0.7000	0.7205	0.5982	0.6411
2	0.6606	0.8500	0.7205	0.5833	0.6032
3	0.6400	0.7619	0.6400	0.5500	0.5500
4	0.5249	0.7847	0.6591	0.6400	0.5342
5	0.5897	0.7333	0.7257	0.6537	0.6537
μ	0.6113	0.7660	0.6932	0.6050	0.5964

Table 4.29: Average F-score achieved by WCLC per fold per maximum time lag (τ_{max}). The bottom row depicts the average score across all folds.

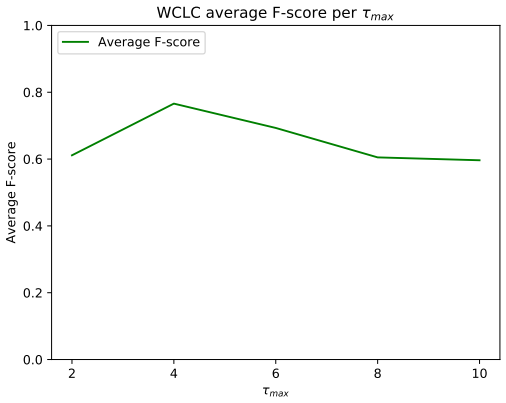

Figure 4.23: The average F-score of WCLC per maximum time lag (τ_{max}).

τ_{max}	Avg.	Std. dev.
2	0.2234	0.0663
4	0.2253	0.0537
6	0.2109	0.0462
8	0.2109	0.0425
10	0.1922	0.0372

Table 4.30: The average of the peak distribution output score distribution of WCLC per maximum time lag (τ_{max}) in seconds.

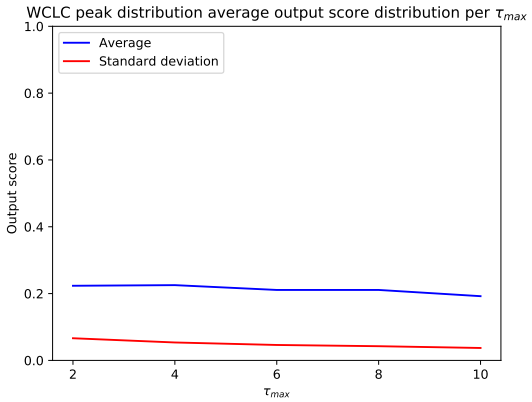

Figure 4.24: The average of the peak distribution output score distribution of WCLC per maximum time lag (τ_{max}) in seconds.

Leave-out fold	τ_{inc} 0.1	0.2	0.3	0.4	0.5
1	0.6591	0.6032	0.7205	0.7205	0.7917
2	0.6591	0.6411	0.7205	0.7619	0.7917
3	0.6411	0.6032	0.6400	0.7205	0.6400
4	0.7000	0.6591	0.6591	0.7205	0.6400
5	0.7333	0.7091	0.7257	0.7681	0.7949
μ	0.6785	0.6431	0.6932	0.7383	0.7316

Table 4.31: Average F-score achieved by WCLC per fold per lag increment (τ_{inc}). The bottom row depicts the average score across all folds.

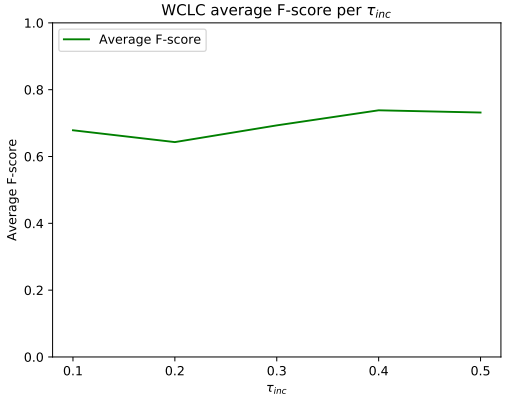

Figure 4.25: The average F-score of WCLC per lag increment (τ_{inc}).

τ_{inc}	Avg.	Std. dev.
0.1	0.2139	0.0466
0.2	0.2105	0.0463
0.3	0.2109	0.0462
0.4	0.2304	0.0484
0.5	0.2088	0.0451

Table 4.32: The average of the peak distribution output score distribution of WCLC per lag increment (τ_{inc}) in seconds.

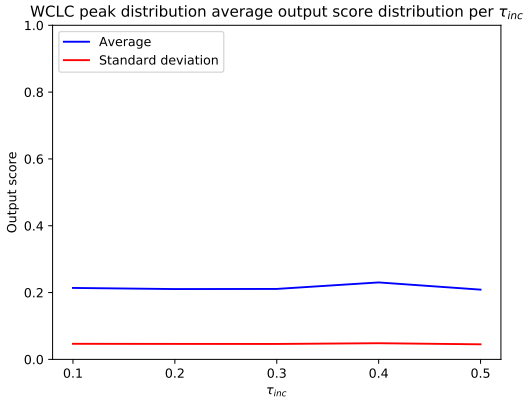

Figure 4.26: The average of the peak distribution output score distribution of WCLC per lag increment (τ_{inc}) in seconds.

RQA The influence of the embedding size, the minimum diagonal line segment length (diagonal length), and the recurrence threshold on the percent determinism (%DET) output score distribution is shown in Tables 4.34, 4.36, 4.38, and 4.34 and Figures 4.30, and 4.32, respectively. Furthermore, the influence of these parameters on the F-score is shown in Tables 4.33, 4.35, and 4.37, and in Figures 4.27, 4.29, and 4.31, respectively.

	Embedding Dimension				
Leave-out fold	0.1	0.2	0.3	0.4	0.5
1	0.6400	0.5833	0.5833	0.6400	0.5833
2	0.4231	0.4444	0.4643	0.4976	0.5342
3	0.6400	0.5342	0.5342	0.6400	0.5833
4	0.6400	0.5342	0.5833	0.6400	0.5833
5	0.6444	0.5429	0.5429	0.6444	0.5897
μ	0.5975	0.5278	0.5416	0.6124	0.5748

Table 4.33: Average F-score achieved by RQA per fold per embedding dimension. The bottom row depicts the average score across all folds.

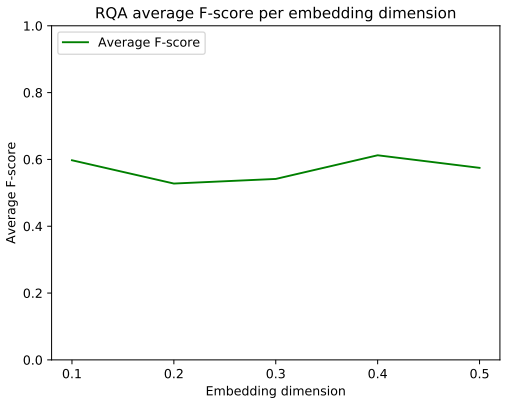

Figure 4.27: Average F-score achieved by RQA per fold per embedding dimension.

Embedding Dimension	Avg.	Std. dev.
0.1	0.4903	0.2042
0.2	0.5878	0.1856
0.3	0.6682	0.1702
0.4	0.7126	0.1677
0.5	0.7468	0.1666

Table 4.34: The percent determinism (%DET) output score distribution of RQA per embedding dimension in seconds.

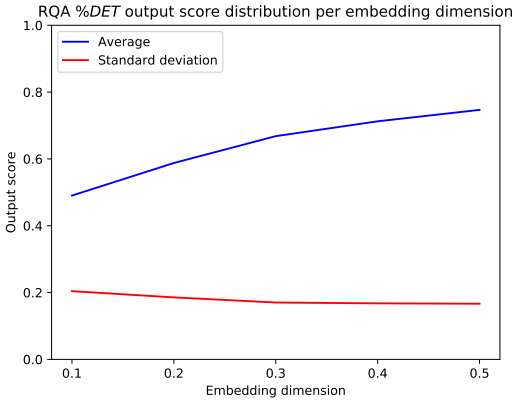

Figure 4.28: The percent determinism (%DET) output score distribution of RQA per embedding dimension in seconds.

Leave-out fold	Diagonal Threshold 0.1	0.2	0.3	0.4	0.5
1	0.6400	0.5342	0.5833	0.5833	0.5500
2	0.4444	0.4976	0.4643	0.4976	0.4643
3	0.5833	0.4976	0.5342	0.5833	0.5342
4	0.5833	0.5342	0.5833	0.6400	0.5833
5	0.5897	0.5152	0.5429	0.5897	0.5429
μ	0.5682	0.5157	0.5416	0.5788	0.5349

Table 4.35: Average F-score achieved by RQA per fold per diagonal line length threshold. The bottom row depicts the average score across all folds.

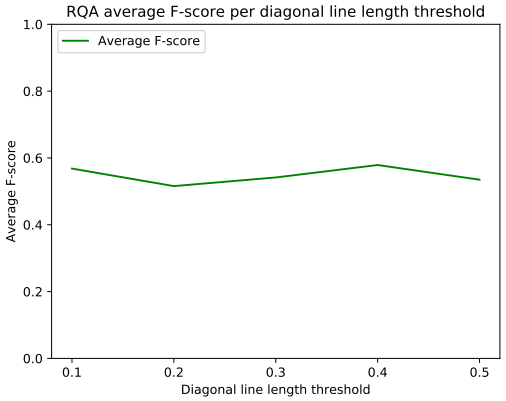

Figure 4.29: Average F-score achieved by RQA per fold per diagonal line length threshold.

Diagonal Threshold	Avg.	Std. dev.
0.1	0.9533	0.0331
0.2	0.8716	0.0834
0.3	0.7293	0.1572
0.4	0.6051	0.1835
0.5	0.5049	0.1893

Table 4.36: The percent determinism (%DET) output score distribution of RQA per diagonal line length threshold in seconds.

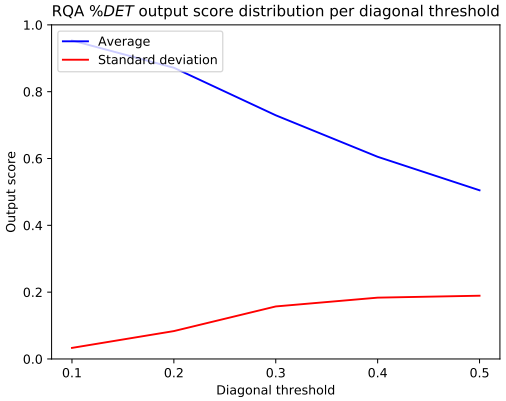

Figure 4.30: The percent determinism (%DET) output score distribution of RQA per diagonal line length threshold in seconds.

Leave-out fold	Recurrence Threshold 1e-05	3e-05	5e-05	7e-05	9e-05
1	0.5833	0.7917	0.7205	0.6591	0.7205
2	0.4643	0.5833	0.5833	0.6400	0.6400
3	0.5342	0.7205	0.6591	0.6591	0.7205
4	0.5833	0.5833	0.5833	0.6400	0.6400
5	0.5429	0.7257	0.6667	0.6667	0.7257
μ	0.5416	0.6809	0.6426	0.6530	0.6893

Table 4.37: Average F-score achieved by RQA per fold per recurrence threshold. The bottom row depicts the average score across all folds.

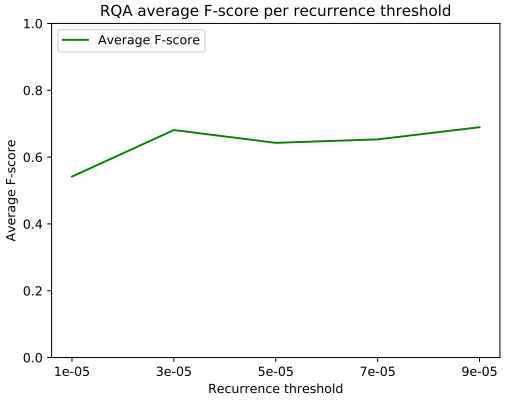

Figure 4.31: Average F-score achieved by RQA per fold per recurrence threshold.

Recurrence Threshold	Avg.	Std. dev.
1e-05	0.6857	0.1668
3e-05	0.7692	0.1153
5e-05	0.7997	0.1033
7e-05	0.8179	0.0957
9e-05	0.8325	0.0899

Table 4.38: The percent determinism (%DET) output score distribution of RQA per recurrence threshold in seconds.

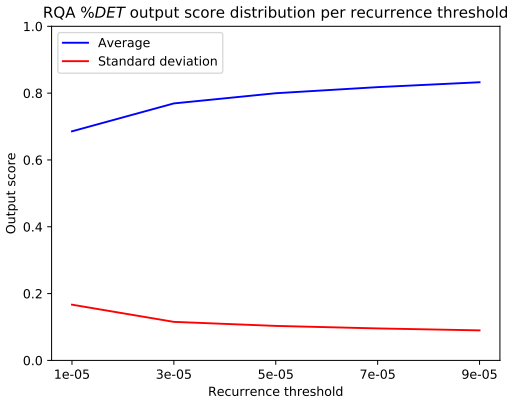

Figure 4.32: The percent determinism (%DET) output score distribution of RQA per recurrence threshold in seconds.

4.2.3 Synchrony Measurement Method

The settings of the three synchrony measurement methods are set at the values that allow the synchrony measurement method to achieve the highest generalised F-score. The settings for this experiment are as follows:

- WCLR
  - Window size: 12 seconds
  - Window increment: 0.3 seconds
  - Maximum time lag: 6 seconds
  - Lag increment: 0.3 seconds
  - Local region size: 3
  - Minimum synchrony line length: 0.5 seconds
  - Allowed lag difference: 2
- WCLC
  - Window size: 10 seconds
  - Window increment: 0.3 seconds
  - Maximum time lag: 4 seconds
  - Lag increment: 0.4 seconds
  - Local region size: 3
  - Loess smoothing span: 0.25
- RQA
  - Embedding dimension: 0.4 seconds

- Recurrence threshold: embedding dimension * 0.00009
- Diagonal line length threshold: 0.4 seconds

Three of these parameters have not been investigated, because they are not defined in seconds or are not varied throughout literature. The first parameter is the local region size of WCLC and WCLR, which defines the number of neighbours must be monotonically decreasing on both sides of a peak. The second parameter is the allowed lag difference, which defines the maximum time lag difference successive peaks may have before they are deemed unrelated to the same synchronous time segment. Finally, the loess smoothing span of WCLC defines the range of loess smoothing applied to the rows of the correlation matrix before peaks are found.

Using the optimal settings for each of the synchrony measurement methods found in the previous experiment, their ability to distinguish rapport-building trained interviewers from control interviewers is investigated. The output scores of each method and their respective generalised F-score achieved when making distinctions based on the output score are shown in Table 4.39. The Pearson correlation between the set of output scores obtained by running each synchrony measurement method on all videos is shown in Table 4.40.

		Synchrony Method		
		WCLR	WCLC	RQA
Leave-out fold	1	0.8500	0.7000	0.7205
	2	0.7917	0.8500	0.5833
	3	0.8295	0.7619	0.7205
	4	0.8295	0.7619	0.6400
	5	0.7257	0.7091	0.7257
	μ	0.8053	0.7566	0.6780

Table 4.39: Generalised F-scores per synchrony measurement method. Using the ratio output score of WCLR, the peak distribution average output score of WCLC, and the percent determinism output score of RQA.

	WCLR	WCLC	RQA
WCLR	1	-0.0875	-0.0133
WCLC	-0.0875	1	-0.0179
RQA	-0.0133	-0.0179	1

Table 4.40: Correlation between all output scores. Using the ratio output score of WCLR, the average of the peak distribution output score of WCLC, and the percent determinism output score of RQA.

4.2.4 Motion Energy Time Series

The distribution of motion energy per person of MEA and OpenPose shown in Tables 4.41 and 4.42 and their respective time series are illustrated in Figures 4.33 and 4.34, respectively. The plot at

the top of the figure represents the motion energy time series of one individual, whereas the plot at the bottom of the figure represents the motion energy time series of the other individual. A visual comparison of the motion energy per time series generated with both methods shows that OpenPose has more fluctuations than MEA.

An overview of the output score distribution using both movement estimators are shown in Table 4.43. The generalised F-scores per movement estimator is shown in Table 4.44. The correlation between the sets of ratio output scores obtained using both movement estimators on all videos is -0.0242. This means, that there is no relationship between the output scores, indicating that the movement estimator's generated motion energy time series greatly influences the output scores. The average correlation between the motion energy scalars assigned to person 1 by MEA and the motion energy scalars assigned to person 1 using OpenPose is 0.4207. The average correlation between the motion energy scalars assigned to person 2 by MEA and the motion energy scalars assigned to person 2 using OpenPose is 0.3857.

Person	Avg.	Std. dev.
1	0.0026	0.0053
2	0.0017	0.0030

Table 4.41: Motion energy distribution per person for time series generated with MEA.

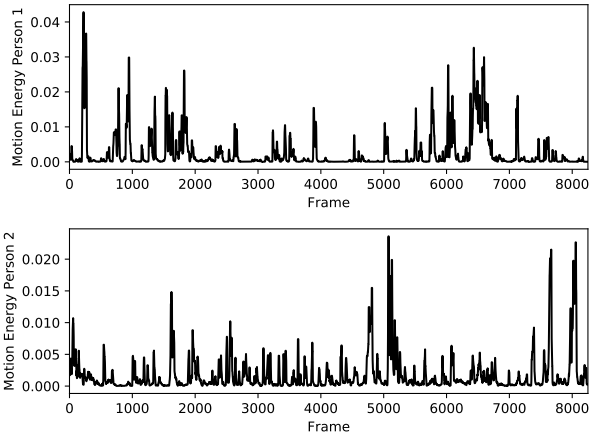

Figure 4.33: Motion energy distribution per person for time series generated with MEA.

Person	Avg.	Std. dev.
1	0.6938	0.6979
2	0.8504	0.8386

Table 4.42: Motion energy distribution per person for time series generated with OpenPose.

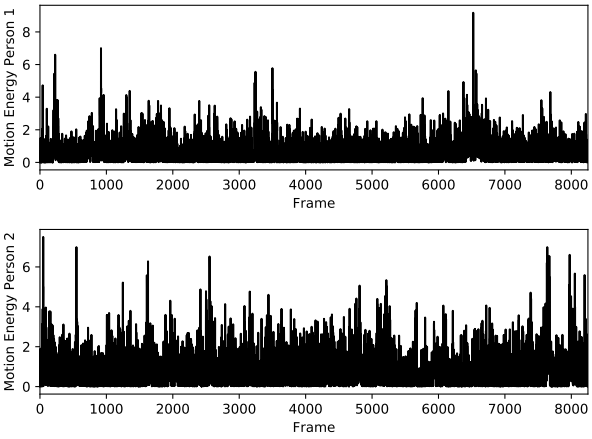

Figure 4.34: Motion energy distribution per person for time series generated with OpenPose.

Movement Estimator	Avg.	Std. dev.
MEA	0.4375	0.0744
OpenPose	0.6304	0.0722

Table 4.43: WCLR ratio output score distribution using time series generated with MEA and OpenPose.

		Movement Estimator	
		MEA	OpenPose
Leave-out fold	1	0.8500	0.6606
	2	0.7917	0.7847
	3	0.8295	0.5982
	4	0.8295	0.6606
	5	0.7257	0.6761
	μ	0.8053	0.6761

Table 4.44: Generalised F-scores per movement estimator. Using the ratio output score of WCLR.

4.3 Discussion

In this section the results shown in the previous sections will be discussed. First, Section 4.3.1 will discuss the input data and its transformations. Afterwards, the results of the experiments will be discussed. Section 4.3.2 discusses the results of the frame skip experiment. Section 4.3.5 discusses the influence of the movement time series on the synchrony measurement. In Section 4.3.3 the influence of the parameter settings on the synchrony measurement is discussed. Finally, in Section 4.3.4 the results of the comparison between the synchrony measurement methods is discussed.

4.3.1 Data

The videos provided by Wright et al. [71] originally suffered from lens distortion. To correct this distortion the frames were multiplied with the GoPro's camera matrix, however this matrix was not given, but had to be manually found. Although the corrected videos look good, the distortion was most likely not completely corrected, which may influence the ability of OpenPose to correctly estimate keypoint positions.

Furthermore, in the evaluation of the ability of synchrony measurement methods to distinguish rapport-building trained interviewers from control interviewers it should be taken into account that even the interviewers that did not receive the rapport-building training would perform better in the second wave. This is because they will already have learnt from their experience during the first wave. Furthermore, the duration of the interviews in the second wave are shorter than those of the first wave, on average the duration of the videos in wave 1 is 7:40 versus an average duration of 6:33 in wave 2.

This suggests that in the second wave the interviewers were more adapt at acquiring the necessary information. However, this did not greatly influence the synchrony measurements, as only the output score of WCLR became slightly higher in wave 2 than in the wave 1. WCLR achieved an average output score of 0.4303 in wave 1 and an average output score of 0.4445 in wave 2. The average output scores for RQA and WCLC remain similar across waves. RQA achieves an average output of 0.8069 in wave 1 and an average output of 0.7992 in wave 2. Finally, WCLC achieves an average output of 0.2695 in wave 1 and an average output of 0.2655 in wave 2.

On top of this, the clipboard held by the interviewer also restricted their hand movement as well as the ability of OpenPose to correctly estimate the location of the hand occluded by the clipboard. The overall implication of this is that the interviewer will most likely not move this hand as they would have done in a more natural setting in which no clipboard is present. The implication for movement estimation is that OpenPose will attribute noisy movement to the hand, due to not being able to find the hand. On the other hand, MEA will attribute movement of the clipboard towards movement of the interviewer, because the pixel changes caused by the clipboard lie within the bounding box.

4.3.2 Frame Skip

Overall, the results show that for WCLR and for WCLC the correlation between the output scores with frame skip and the output scores without frame skip decreases as the frame skip increases. However, the results show that the output scores of RQA show a greater resilience to frame skip. Therefore, frame skip should only be considered when opting to use RQA if deviation from the original output scores is unwanted.

WCLR The results of the frame skip experiment for WCLR show that as the frame skip increases, the average of the ratio output scores distribution increases and its standard deviation decreases. The correlation between the output scores hit a low point at a frame skip of three with a correlation of 0.5287. The correlation shows that there is a linear relationship between the output scores across frame skips. However, the relatively low correlation between the output scores suggests that frame skipping does compromise the synchrony measure. Furthermore, the average F-scores obtained per frame skip, shown in Table 4.1, show that the highest average F-score is achieved at a frame skip of 0, corresponding to a frame rate of 27.

The sensitivity of the output scores to frame skip may be caused by skipping over small movement, as well as by the influence of frame skip on its parameters. As the frame skip increases, the number of frames per second decreases, thereby also decreasing the number of samples contained in windows whose range is defined in seconds. This is most likely caused by the decrease in checked time lags, this results in fewer columns in the regression matrix, making it more likely that peaks found for each row of the regression matrix are found in roughly the same column. This results in longer periods of perceived synchrony, resulting in a higher ratio output score.

WCLC The results of WCLC show that the average and standard deviation distributions both remain similar despite a change in frame skip. However, despite the similarity in the output score distribution, the correlation between the output scores is low. Therefore, it can be said that frame skip also compromises both output scores of WCLC. Furthermore, the average F-scores obtained per frame skip, shown in Table 4.4, show that the highest average F-score is achieved at a frame skip of 3, corresponding to a frame rate of 9. The same average F-score is obtained at a frame skip of 1, corresponding to a frame rate of 14. However, since a greater frame skip is preferred, as this reduces the required runtime, the frame skip of 3 is preferable.

WCLC is influenced by the frame skip in a similar manner as WCLR is. By increasing the frame skip, the number of frames in a second decreases. Thereby also decreasing the number of samples that its window covers and the number of possible time lags that will be considered, since these parameters are defined in number of seconds. It appears that the range of values the output scores can attain is very limited, which reduces the impact the frame skip can have on the output score distribution, because the average remains roughly 0.21 and the standard deviation remains close to 0.46. However, it does assign different output scores when looking at individual videos. This explains the similarity in distributions, despite the low correlation.

RQA Finally, the results of RQA show that its output scores are more robust to changes in frame skip than WCLR and WCLC. With a high correlation between the output scores of all tested frame skips and the output scores without frame skip. We see that the distribution of the %REC, %DET, and the entropy output scores remains similar across frame skips. On the other hand, the ratio and the average diagonal line length output score distribution are more sensitive to changes in frame skip.

The %REC output score average and standard deviation slightly decrease as the frame skip increases. The results show that initially it is 1% and at a frame skip of 4, it is still at 0.52%. This variable is strongly dependent on the recurrence threshold parameter, which has been set in such a way to ensure that %REC will be roughly 1%. Since the frame skip influences the embedding dimension and the recurrence rate threshold, the %REC will also be influenced. An increase in frame skip leads to a decrease in the embedding dimension, delimiting the number of samples that together form states. Decreasing the embedding dimension should increase the %REC, because fewer samples have to be similar across states, leading to an increase in the number of recurrence points. However, an increase in frame skip also decreases the recurrence threshold. The recurrence threshold sets the boundary how far apart two states may be to still be considered similar. By decreasing this threshold, the states have to be more similar in order to be considered a recurrence point. The decreased threshold culls more recurrent points than the decreased embedding dimension adds, hereby decreasing the %REC.

The %DET output score is less affected by the frame skip than the %REC output score. The average and the standard deviation remain roughly the same despite the increase in frame skip. %DET measures is the ratio between recurrence point in the CRP that form diagonal structures and the total number of recurrent points. An increase in frame skip leads to fewer points in the CRP, however this does not fundamentally change the diagonal structures within the CRP. Therefore, the diagonal structures remain the same, only their length decreases. Therefore, the ratio between recurrent points that form diagonal structures and all recurrent points remains similar.

The results show that the entropy output score average and standard deviation decrease as the frame skip increases. Entropy measures the complexity of deterministic structures in the CRP and depends greatly on the number of bins in the diagonal line segment length histogram. Therefore, this behaviour can be explained by the influence frame skip has on the number of bins in the histogram. As the frame skip increases, the number of recurrent points in the CRP decreases, thereby decreasing the length of the diagonals. A decrease in diagonal line segment length results in a decrease in the number of bins in the histogram.

The ratio output score average increases per frame skip, whereas its standard deviation reaches its peak at a frame skip of 3 and then decreases. The ratio depicts the ratio between %DET and %REC. The %DET remains roughly the same no matter the frame skip, however the %REC decreases as the frame skip increases, thereby increasing the ratio.

The average diagonal line length distribution decreases per frame skip, because the number of points in the cross-recurrence plot also decreases. Thereby decreasing the number of points that can be present in diagonal structures. This results in a decrease in the average diagonal line segment length.

4.3.3 Parameter Settings

Due to the time constraints, it was not possible to try every unique combination of parameter settings in order to find the globally optimal parameter settings. However, the influence each parameter individually has on the generalised F-score and the output score distribution has been investigated in order to find optimal settings for each synchrony measurement method, whilst using default settings for the other parameters.

WCLR The investigated parameters of WCLR are the window size (w_{max}), window increment (w_{inc}, maximum time lag (τ_{max}), lag increment (τ_{inc}), and the minimum synchronous line segment length (s_{min}). The results of the WCLR parameter tuning show that the highest generalised F-score is achieved for $w_{max} = 12$, $w_{inc} = 0.3$, $\tau_{max} = 6$, $\tau_{inc} = 0.3$, and $s_{min} = 0.5$. Furthermore, the results show that parameters do influence the average of the ratio output score distribution, however the standard distribution remains relatively unaffected.

Firstly, an increase in the w_{max} leads to an increase in the average of the output score distribution. Increasing the w_{max} results in larger sliding windows, containing more samples, which influences the rows in the regression matrix. The higher number of samples contained within the window, the more similar two consecutive windows will be. The similarity increases, because the number of shared samples also does, given that the w_{inc} is smaller than the w_{max}. The increased similarity propagates to the rows in the regression matrix, thereby making it more likely that the time lag corresponding to the peak of those rows is similar as well. Synchronous time fragments are defined by the consecutive rows in in which the time lag of the peaks is similar. Therefore, the total amount of synchronous time is increased, which results in a higher ratio between the synchronous time and the total time.

On the other hand, an increase in the w_{inc} leads to a decrease in the average of the output score distribution, yet also leads to a slight increase in its standard deviation. A larger w_{inc} leads to fewer

window comparisons, since the steps size the window uses to slide over the data increases. Therefore, the number of rows in the regression matrix decreases. The output score of WCLR is the ratio between the synchronous time and the total time. It defines synchronous time as a period of time in which the peak regression values are all at a similar time lag. Since window increment causes the window to skip over movement, the difference between consecutive windows, and therefore rows in the regression matrix, will be greater. The greater difference in consecutive rows in the regression matrix makes it less likely that their respective peaks will be at a similar time lag. The greater difference in consecutive peak time lags decreases the amount of synchronous time and thereby the ratio between the synchronous time and the total time.

The output score distribution's average and standard deviation are not influenced by a change in the τ_{max} and remain similar throughout. Increasing τ_{max} leads to an increase in the number of columns in the regression matrix. However, the added columns hardly influence the output score distribution, because the peaks found within the rows of the regression matrix are usually found close to the column corresponding to a time lag of 0, which is located at the center column, where the search starts. Therefore increasing the number of columns does not influence the selection of peaks, because the peak will be found before the added columns are considered.

Unlike increasing the w_{inc}, increasing the τ_{inc} also increases the average of the output score distribution. Similarly to τ_{max}, increasing the τ_{inc} also decreases the number of columns found in the correlation matrix. However, rather than decreasing the range of time lags that are considered, as is done when τ_{max} is increased, it decreases the number of columns by skipping samples. A reduction in the number of columns in the regression matrix makes it more likely that peaks in consecutive rows of the regression matrix will be found in a similar column, at a similar time lag, because the range of possible time lags has been decreased. Rows with peaks at a similar time lag are defined to be synchronous time and an increase in total synchronous time will increase the ratio between the synchronous time and the total time.

Finally, an increase in the s_{min} parameter leads to a decrease in the average of the output score distribution. As the threshold increases, fewer periods in time will be considered synchronous, because the threshold delimits the number of consecutive peaks in the regression matrix that must have a similar time lag. This causes the total synchronous time to decrease, because an increase in this threshold means that fewer selection of consecutive rows will fulfill this criteria. The decrease in total synchronous time leads to the decrease of the ratio between the synchronous time and the total time.

Overall, the results show that all parameters influence the average of the ratio output score distribution, whereas the standard deviation remains relatively unaffected. The influence on the output scores is also visible in the generalised F-score, where clear optimal values can be seen.

WCLC The investigated parameters of WCLC are the window size (w_{max}), window increment (w_{inc}), maximum time lag (τ_{max}), and lag increment (τ_{inc}). The results show that the highest generalised F-scores are obtained using $w_{max} = 10$, w_{inc} can be 0.1, 0.2 or 0.3, $\tau_{max} = 4$, and $\tau_{inc} = 0.4$.

The results show that an increase in w_{max} decreases the average of the peak distribution. An increase

in the w_{max} leads to an increase in the number of samples contained within the window. An increase in the number of samples within the window makes it less likely that all samples within the two windows transform in a similar fashion, resulting in a lower correlation between the two windows.

The influence of w_{inc} on the output score distribution is minimal. Increasing the w_{inc} decreases the number of rows in the correlation matrix, because some combinations of windows will be skipped. However, this does not influence the output score distribution, because the correlation assigned to windows throughout the dataset remains similarly spread around the same average. Therefore a decrease in the number of windows that are considered for synchrony does not significantly influence the output score distribution, since the average remains unaffected.

An increase in τ_{max} results in a slight decrease in the average of the output score distribution. Increasing the τ_{max} also increases the number of columns in the correlation matrix. However, this barely influences the output score distribution, because the peaks found within the rows of the correlation matrix are usually found close to the column corresponding to a 0 lag, located at the center of the row, where the search starts. Therefore increasing the number of columns does not influence the peak-picking algorithm, because the peak will be found before the columns corresponding to the greater time lag are considered.

Finally, similarly to the w_{inc}, does the τ_{inc} barely influence the output score distribution. An increase in the τ_{inc} leads to fewer time lags being considered, thereby decreasing the number of columns in the correlation matrix. Despite this, the correlation of the peaks found in the rows with fewer columns does not differ much from the correlation of the peaks found in the original columns. The correlations are similar, because in most cases it is close to the original average. Therefore, the output score distribution remains relatively unaffected.

Overall, the results show that none of the parameters greatly influence the output score distribution, as all output scores retain a similar mean and standard distribution. This effect is also visible when investigating the generalised F-scores achieved by WCLC. The influence of the τ_{max} is minimal, because peaks are often found close to a time lag of zero. Furthermore, the w_{inc} and the τ_{inc}also barely influence the output and F-score as the outputs throughout the video remain close to the mean, therefore it does not matter if samples are skipped. Finally, the w_{max} parameter shows the greatest influence on the generalised F-score, as this is the only parameter that influences the output score distribution.

RQA The parameters of RQA that are investigated are the embedding dimension, diagonal line segment length threshold (diagonal threshold), and the recurrence threshold. The results show that the highest generalised F-scores are attained using embedding dimension = 0.4, diagonal threshold = 0.00009, and the recurrence threshold = 0.4.

Firstly, as the embedding dimension increases, so does the percent determinism output score. The embedding dimension defines the number of samples that together form a state. Increasing this number leads to fewer recurrence points within the cross-recurrence plot, because it becomes less likely that similar states will be found throughout the system, as more samples are now required to be similar. However, the diagonal structures within the cross-recurrence plot remain relatively unaffected, only

the number of samples they consist of decreases. The percent determinism output score increases, because the number of recurrent points that do not form diagonal structures decreases more than the number of recurrent points in diagonal structures.

On the other hand, as the diagonal threshold increases, the percent determinism output score distribution average decreases. The diagonal threshold delimits the minimal number of recurrent points in the cross-recurrence plot that a diagonal line structure must contain. Increasing this threshold, decreases the percent determinism as the diagonal structures that no longer contain the necessary number of recurrent points are now dropped.

Finally, an increase in the recurrence threshold leads to an increase in the average percent determinism. The recurrence threshold represents the maximal Euclidean distance the two states can be apart from each other for them to still be considered similar. Increasing this threshold, allows for the creation of more recurrent points in the cross-recurrence plot, thereby increasing the number of diagonal structures that may be found within the cross-recurrence plot, which leads to an increase in the percent determinism. Although a higher generalised F-score may have been achieved with an even higher recurrence threshold, increasing the recurrence threshold further also increases the number of recurrence points in the cross-recurrent plot. The increase in recurrence points comes at a computational cost and it was not possible to test greater values for the recurrence threshold due to memory limitations.

Overall, the results show that although changes in the embedding dimension, recurrence threshold, and the diagonal line length threshold influence the output score distribution, however the generalised F-score remains relatively unaffected. The setting of the parameters does not greatly increase nor diminish its ability to distinguish rapport-building trained interviewers from control interviewers. If memory or computational power is limited it is recommended to decrease the recurrence threshold, as this decreases the number of recurrence points in the cross-recurrence plot, thereby greatly reduce the required memory and runtime.

4.3.4 Synchrony Measurement Method

Overall, every synchrony measurement method is able to distinguish rapport-building trained interviewers from control interviewers better than when no distinction is made, as this would have resulted in a generalised F-score of 0.4242 if all interviewers are classified as trained, or 0.2084 if all interviewers are classified as control. The results show that WCLR achieves the highest generalised F-score of 0.8053, WCLC achieves the second highest generalised F-score of 0.7566, and RQA achieves the lowest generalised F-score of the three with a value of 0.6780.

Furthermore, no correlation is found between the outputs scores of the synchrony measurement methods, which is in line with the findings of Schoenherr et al. in [58], who found that there is no correlation between the WCLR ratio and WCLC average peak strength. The lack of correlation between the output scores may be caused by the underlying dataset. Since WCLR achieves the highest generalised F-score, it is plausible that auto-correlation has a strong presence throughout the dataset, which only WCLR capitalises on. On top of this, the lack of correlation may be caused because the synchrony

measurement methods measure different facets of synchrony. WCLR measures the ratio between synchronous time and the total time, WCLC measures the average strength of synchrony, and the percent determinism output score of RQA measures the frequency of synchrony.

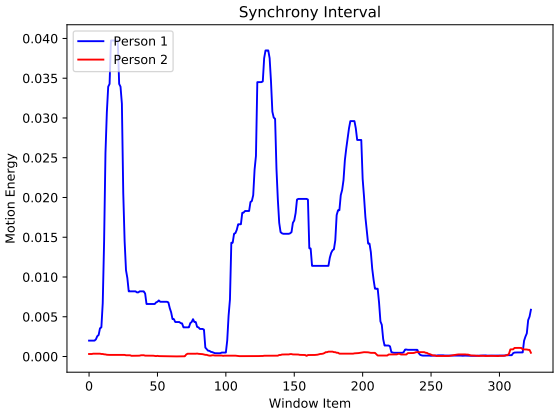

Table 4.45: Motion energy per person for windows that receive a low synchrony output score from WCLR (R^2_{CC} = 6.7188e−08), as well as a low synchrony output score from WCLC (average peak correlation = −0.0929).

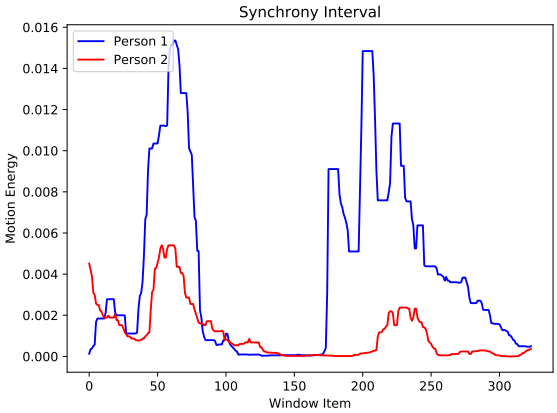

Table 4.46: Motion energy per person for windows that receive a high synchrony output score from WCLR (R^2_{CC} = 0.2507), as well as a high synchrony output score from WCLC (average peak correlation = 0.5995).

Exemplary windows of motion energy of both persons that achieve high synchrony output scores and low synchrony output scores by WCLR and WCLC are shown in Figures 4.45 and 4.46, respectively. The windows corresponding to the high synchrony output scores are of a time interval in which both persons move their hands to aid with their explanation. The windows corresponding to low synchrony output score are based on a time interval in which the interviewee used large arm movement to aid her explanation, however the interviewer only pays attention to his clipboard whilst barely moving. This suggests that both WCLR and WCLC are able to detect synchrony in behaviours where both individuals perform similar behaviour, without the need for motion energy to be of similar strength.

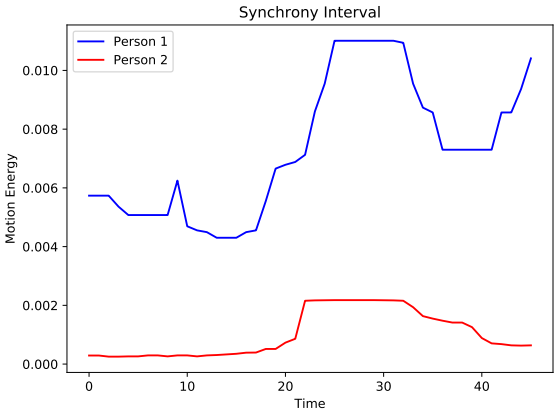

Table 4.47: Motion energy of both person which created no diagonal in the cross-recurrence plot.

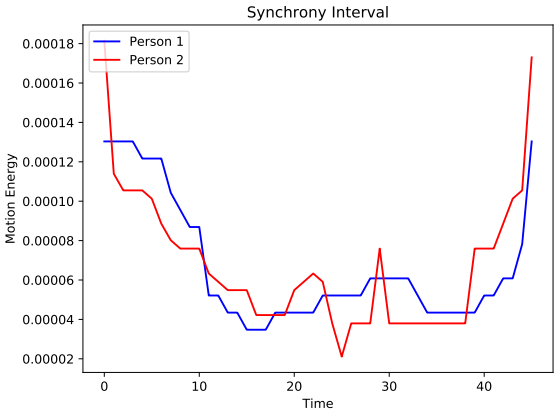

Table 4.48: Motion energy of both person which created the longest diagonal in the cross-recurrence plot.

Figures 4.47 and 4.48 show motion energy for time intervals of about 2 seconds to which RQA assigned a low synchrony output score and a high synchrony output score, respectively. The time interval that received the high synchrony output score is of both individuals moving just one hand up and down. The low synchrony output score is assigned to a time interval in which the interviewee made big hand movements to aid her explanation and the interviewer only made small hand movements to write on the clipboard. Overall, RQA only detects synchrony if the motion energy of both individuals is similar. Therefore, RQA cannot detect synchrony in events where the movement is copied, but executed on a different scale by one person than by the other. For example, if both individuals scratch their head, but one of them makes large hand movements whereas the other only slightly moves their fingers, then this will not be classified as synchronous. On the other hand, it does pick up on movement that is caused by a different behaviour but causes the same amount of motion energy. For example, if one individual moves their arms and the other individual moves their legs and both events occur with similar motion energy, then this event will be classified as synchronous.

4.3.5 Motion Energy Time Series

The low correlation between the output scores of all videos obtained using both movement estimators suggests that the movement time series greatly influences the WCLR output score and that MEA provides vastly different time series than OpenPose. On the other hand, the average correlation between the movement scalars assigned by MEA and the movement scalars of the same person assigned by OpenPose suggests that there is a relation between the assigned movement scalars. The average correlation between motion energy assigned to person 1 is 0.4207, and the average correlation between motion energy assigned to person 2 is 0.3857, indicating that there is a correlation, albeit low. Although there is a relation between the assigned motion energy scalars, the motion energy time series remain vastly different from each other. This difference is indicated by the lack of correlation between the output scores generated using the time series generated using OpenPose and MEA. Had there been more similarity between the motion energy time series, then there would also have been a greater similarity in the output scores of WCLR as its input would have been more alike. However, the low correlation of -0.0242 shows that no similarity in the output scores is present.

The mean of the movement distribution of both persons for both movement estimators is relatively close to zero, indicating that generally there is little movement throughout the video. The standard deviation is large in comparison to the mean of the distribution, which is required to capture the short bursts of relatively large movement. Furthermore, when looking at Figures 4.33 and 4.34 the graphs show that the time series created by OpenPose have stronger fluctuations than the time series created by MEA. These fluctuations suggest that, despite the filtering applied to the OpenPose keypoint estimations, some noise is still present in the OpenPose time series.

With respect to the output score distribution, the results show that the average of the ratio output score distribution is higher when OpenPose is used than if MEA is used, despite the stronger fluctuations present in the time series of OpenPose. On the other hand, the standard deviation of the ratio output score distribution of both MEA and OpenPose is similar. This suggests that, despite having a different average, the difference between the output scores assigned to videos may be similar. However, the low

correlation between the output scores and the different F-scores shows that this is not the case. It is plausible that the output scores generated using MEA time series better represent the true movement, as they allowed for a higher generalised F-score than time series generated with MEA.

The results show that both movement estimators achieve a higher F-score than when no distinction is made, which would have resulted in a generalised F-score of 0.4242 if all interviewers are classified as trained, or 0.2084 if all interviewers are classified as control. Furthermore, Table 4.44 shows that WCLR achieves the highest generalised F-score by using time series created with MEA. Therefore, when estimating synchrony, it is recommended that MEA is used to generate time series.

Relation between motion and synchrony The relation between the amount of movement of a dyad and the synchrony output score is investigated. To find this relation, the correlation between the average movement of the dyad and the output scores of WCLC, WCLR and RQA is used. The average movement of the dyad is defined as the average movement of both persons as measured by MEA. The output scores are obtained using the optimal settings per synchrony measurement method on MEA time series. The correlation between the output scores of WCLC and the average movement is -0.0332. The correlation between the output scores of WCLR and the average movement is 0.0252. The correlation between the output scores of RQA and the average movement is -0.3199. The correlations show that for WCLR and WCLC there is no significant relation between movement and synchrony output score. On the other hand, the output scores of RQA have a negative relation with the average movement, indicating that as the average movement decreases, the output score increases. This is plausible, because as the average movement decreases, the number of states containing little to no movement increases. Since the states with little to no movement will be similar to each other, will the number of recurrence points increase. The higher number of recurrence points allows for the creation of more diagonal structures that satisfy the minimal diagonal line length threshold, thereby increasing the percent determinism output score.

4.3.6 Research Questions Revisited

The first research question is "which synchrony measuring method most accurately measures synchrony?", and has been answered in the synchrony measurement method experiment in Section 4.2.3. In this experiment the accuracy of three synchrony measurement methods, WCLC, WCLR and RQA, using their optimal parameter settings on time series generated by MEA is investigated. The accuracy is determined by the ability to distinguish rapport-trained interviewers from control interviewers. This distinction is made by comparing the output scores assigned in wave 2 to a dyad to the output scores assigned to the same dyad in wave 1. If the output scores are sufficiently different, then the dyad is classified as having received the rapport-building training, or will be classified as control otherwise. The distance between output scores is sufficiently different if it exceeds a threshold. How well this distinction is made is quantified as the average F-score of how well rapport-trained interviewers have been classified correctly and how well control interviewers have been classified correctly. These F-scores are used in leave-one-out cross validation to obtain a generalised F-score representing the accuracy of the synchrony measurement. The results of the synchrony measurement method experi-

ment show that WCLR achieves a generalised F-score of 0.8053, WCLC a score of 0.7566 and RQA achieves a generalised F-score of 0.6780. Therefore, we can conclude that WCLR provides the best distinction between dyads that did receive rapport-building training and dyads that did not receive rapport-building training.

The second research question is "what are the optimal parameter settings for each synchrony measurement method?", and has been answered in the parameter settings experiment in Section 4.2.2. The results of this experiment show that the highest generalised F-score is achieved by WCLC for settings: $w_{max} = 10$, w_{inc} can be 0.1, 0.2 or 0.3, $\tau_{max} = 4$, and $\tau_{inc} = 0.4$. The w_{inc} and τ_{inc} parameters of WCLC barely influence the generalised F-score. However, the w_{max} and the τ_{max} do influence the generalised F-score, but also show a clear optimal value and are therefore easily set. The highest generalised F-score is achieved by WCLR for settings: $w_{max} = 12$, $w_{inc} = 0.3$, $\tau_{max} = 6$, $\tau_{inc} = 0.3$, and $s_{min} = 0.5$. All parameters of WCLR show a clear optimal value and are therefore easily set. The highest generalised F-score is achieved by RQA for settings: embedding dimension = 0.4, diagonal threshold = 0.00009, and the recurrence threshold = 0.4. The embedding dimension and diagonal threshold parameters barely influence the generalised F-score and therefore their exact setting is less critical. On the other hand, the recurrence threshold parameter shows a greater influence on the generalised F-score and the trend shows that as the recurrence threshold increases, so does the generalised F-score. Therefore, it is uncertain if the optimal value is 0.00009, as higher values could not been tested due to memory limitations.

The third research question is "do time series created by human motion analysis provide better synchrony measurements for the best synchrony measuring method than time series created with motion energy analysis?", and has been answered in the motion energy time series experiment in Section 4.2.4. In this experiment the influence of time series generated using OpenPose and MEA on the output scores of WCLR and its ability to distinguish rapport-trained interviewers from control interviewers is investigated. The results show that the mean output score using OpenPose time series is higher than the mean output score using MEA time series, however the variance of the output scores is similar. Furthermore, the highest generalised F-score is obtained using MEA time series, achieving a score of 0.8053, whereas the generalised F-score obtained using OpenPose time series is 0.6761. Therefore, time series created by human motion analysis do not provide better synchrony measurements for the best synchrony measuring method than time series created with motion energy analysis.

Finally, the fourth research question is "what is the ideal frame rate for measuring interpersonal synchrony in dyadic interactions?", and has been answered in the frame skip experiment in Section 4.2.1. In this experiment the influence of frame skip on the output score of each synchrony measurement method is investigated. The investigated frame skip values are: 0, 1, 2, 3, and 4. The results show that the output of WCLC and WCLR are not robust against frame skip, as the correlation between their output scores obtained without frame skip and their output scores obtained with frame skip quickly drops as the frame skip increases. On the other hand, the output scores of RQA show resilience against frame skip, as the correlation between the output scores obtained without frame skip and the output scores obtained with frame skip remain close to 1. Furthermore, based on the average F-scores achieved per frame skip, the results show that the ideal frame rate for measuring synchrony of WCLR is 27 frames per second, of WCLC is 7 frames per second, and of RQA is 9 frames per second.

Chapter 5

Conclusion

This chapter summarises the results of this thesis in Section 5.1, and suggests improvements to the methods used in this thesis in Section 5.2.

5.1 Summary of Thesis Achievements

Synchrony has received a great deal of attention from many different scientific areas for its relatedness to the quality of interaction and interpersonal relationships, functions in early infancy, and ability to be used as a predictor for variables such as therapy outcome. The multidisciplinary attention synchrony receives inspired a need for automated synchrony analysis in order to exclude the possibility of human error and subjectivity. In this thesis the different methodologies used to extract movement data from video, as well as the methodologies that measure synchrony in movement data have been investigated. The goal of the research of this thesis is to find the methodologies and settings that allow for the best quantification of synchrony. Where synchrony is operationalized as the ability to distinguish rapport-building trained interviewers from interviewers that did not receive this training. Therefore, the assumption is made that synchrony should increase between dyads after rapport-building training has been received. This assumption is made to accommodate for the lack of ground-truth. How well a synchrony measurement method performs will therefore be measured by its ability to assign increasing scores for dyads that did receive rapport-building training, yet assign similar scores to dyads that did not receive rapport-building training. Scores are deemed similar if their respective difference does not exceed a threshold, which is uniquely set per output score for each synchrony measurement method.

With the data provided by Wright et al. [71], motion energy time series from rapport-building trained and control interviewers and their subjects are extracted using Motion Energy Analysis (MEA) [23, 58] and OpenPose [12]. The resulting motion energy time series have been used as input for three synchrony measurement methods: windowed cross-lagged correlation (WCLC) [10], windowed cross-lagged regression (WCLR) [2], and recurrence quantification analysis (RQA) [50]. These methods have been chosen, because they have been used in synchrony research before and because from the myriad of time series analysis methods these are adapt at handling the temporal aspect of synchrony.

Firstly, the possibility of frame skipping to speed up the runtime of the algorithm was explored. The following frame skip values have been tested: 0, 1, 2, 3, and 4. The results show that the output scores of WCLC and WCLR are greatly influenced by frame skipping, as is indicated by the low correlation between the output scores obtained using a frame skip and the original output scores. On the other hand, RQA shows a great resilience against frame skip, as the correlation of four out of five output scores obtained with frame skip and the original output scores does not drop below 0.92 even at a frame skip of 4. Furthermore, the results show that for measuring synchrony the ideal frame rate of WCLR is 27 frames per second, of WCLC is 7 frames per second, and of RQA is 9 frames per second.

Since all synchrony measurement methods require several parameters to be set by the experimenter, the parameters of each synchrony measurement method that have been different across literature have been investigated. For each of these parameters multiple values have been tested on time series generated with MEA in order to find out how they influence the output scores and what values optimises the generalised F-score. The optimal values for the parameters of WCLC are $w_{max} = 10$, w_{inc} can be 0.1, 0.2 or 0.3, $\tau_{max} = 4$, and $\tau_{inc} = 0.4$. The optimal values for the parameters of WCLR are $w_{max} = 12$, $w_{inc} = 0.3$, $\tau_{max} = 6$, $\tau_{inc} = 0.3$, and $s_{min} = 0.5$. The optimal values for the parameters of RQA are embedding dimension = 0.4, diagonal threshold = 0.00009, and the recurrence threshold = 0.4.

The comparison between the three synchrony measurement methods show that WCLR most accurately quantifies synchrony, as it provides the best distinction between rapport-building trained interviewers and control interviewers, achieving a generalised F-score of 0.8053. WCLC is able to achieve a generalised F-score of 0.7566. Finally, RQA is able to achieve an F-score of 0.6780. All synchrony measurements method are able to measure synchrony, as they all achieve a higher F-scores than if no distinction is made between the interviewers. Failure to distinguish rapport-trained interviewers from control interviewers would have resulted in a generalised F-score of 0.4242 if all interviewers are classified as trained, or 0.2084 if all interviewers are classified as control. Furthermore, we found that the correlation of the output scores per video between the synchrony measurement methods is low. This indicates that despite all being able to distinguish rapport-building trained interviewers from control trainers relatively well, each method makes this distinction based on a different facet of synchrony. Therefore, each method assigns other pairs of people low and high rapport. Both WCLC and WCLR assign higher synchrony output scores to events where both individuals display similar behaviour without the need for similar strength in motion energy. For example, both WCLR and WCLC will assign high synchrony output scores to events where both individuals move their hands to aid their explanation, even if one of the individuals uses smaller movements. Despite WCLC and WCLR detecting synchrony in the same windows, the eventual output scores remain different because WCLC measures the strength of synchrony, whereas WCLR measures the frequency of synchrony. RQA detects synchrony when the amount of motion energy between two individuals remains similar over time, without considering how the motion energy transforms over time. For example, RQA will detect synchrony when one individual moves their arms with similar motion energy as the other individual moves their legs. On the other hand, if both individuals move their arms but one individual does so with smaller movements, then the difference in motion energy may cause RQA to classify this as not synchronous despite the similarity in behaviour.

Finally, movement energy time series created with OpenPose and movement energy time series created by MEA have both been used as input for WCLR to investigate their influence on the output scores and the generalised F-scores. A visual comparison between the time series created with OpenPose and the time series created by MEA show that the time series created with OpenPose contain more fluctuations. Furthermore, the results show that there is no correlation between the output scores of WCLR using MEA time series and the output scores of WCLR using OpenPose time series. However, there is a positive average correlation between the motion energy assigned to person 1 by MEA and the motion energy assigned to person 1. A slightly lower positive average correlation is also found between the motion energy assigned to person 2 by MEA and the motion energy assigned to person 2 by OpenPose. Finally, The highest generalised F-score is achieved using the time series of MEA.

Overall, the results show that it is feasible to automatically analyse synchrony. Furthermore, the recommended methodologies to apply in automatic synchrony analysis are WCLR with the following settings: $w_{max} = 12$, $w_{inc} = 0.3$, $\tau_{max} = 6$, $\tau_{inc} = 0.3$, and $s_{min} = 0.5$. It is recommended that MEA is used for the creation of the motion energy time series. Finally, the recommended frame rate for WCLR is 27, of WCLC is 7, and of RQA is 9.

5.2 Limitations And Future Work

Although the used dataset allowed us to investigate the accuracy of a synchrony measuring method by its ability to distinguish rapport-building trained interviewers from control interviewers, it requires the assumption that the rapport-building training also increased synchrony. This approach does not take into account that not all interviewers may have benefited from the training. Nor does it take into account that some interviewers may naturally be skillful at rapport-building, and therefore achieve higher rapport scores without the need for the rapport-building training. To gain more insight in the accuracy of the synchrony measurement methods the experiments may be repeated using data alongside human annotated synchrony values.

Furthermore, there is room for further improvement in the motion energy time series created by OpenPose. Although the pre-trained neural network of OpenPose was generally already able to adequately estimate the keypoint location in the setting of our data, training the neural network ourselves would most likely have resulted in an even higher accuracy. On top of this, OpenPose currently does not have the ability to track people over frames but finds persons from scratch for every frame without using previous results. Tracking may also have increased the accuracy of the keypoint estimations, which would result in more accurate time series. Additionally, OpenPose is not able to estimate the 3D location of the keypoints, but provides 2D keypoint locations within the image. Some information is lost in the translation from a real world 3D location to the 2D image coordinates. Being able to capture 3D keypoint locations would result in a more accurate representation of a person's real world location, allowing for the creation of more accurate movement energy time series.

Since OpenPose has the ability to measure movement of any body part individually, it may be interesting to investigate how the synchrony output score is influenced per body part. This will provide

insight in which body parts contribute to the synchrony measurement, so that body parts that do not contribute no longer have to be considered.

Finally, for synchrony measurement methods that provide multiple output scores, the output score that is predominantly used throughout literature is used in the experiments. It may be interesting to investigate what generalised F-scores are obtained when other output scores are considered.

Bibliography

[1] Riza Alp Guler, George Trigeorgis, Epameinondas Antonakos, Patrick Snape, Stefanos Zafeiriou, and Iasonas Kokkinos. Densereg: Fully convolutional dense shape regression in-the-wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6799–6808, 2017.

[2] Uwe Altmann. Investigation of movement synchrony using windowed cross-lagged regression. In *Analysis of Verbal and Nonverbal Communication and Enactment. The Processing Issues*, pages 335–345. Springer, 2011.

[3] Uwe Altmann. *Synchronisation nonverbalen Verhaltens: Weiterentwicklung und Anwendung zeitreihenanalytischer Identifikationsverfahren*. Springer-Verlag, 2012.

[4] M. Andriluka, U. Iqbal, E. Ensafutdinov, L. Pishchulin, A. Milan, J. Gall, and Schiele B. PoseTrack: A benchmark for human pose estimation and tracking. In *CVPR*, 2018.

[5] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[6] Frank J Bernieri. Coordinated movement and rapport in teacher-student interactions. *Journal of Nonverbal behavior*, 12(2):120–138, 1988.

[7] Frank J Bernieri, Janet M Davis, Robert Rosenthal, and C Raymond Knee. Interactional synchrony and rapport: Measuring synchrony in displays devoid of sound and facial affect. *Personality and social psychology bulletin*, 20(3):303–311, 1994.

[8] Frank J Bernieri, J Steven Reznick, and Robert Rosenthal. Synchrony, pseudosynchrony, and dissynchrony: Measuring the entrainment process in mother-infant interactions. *Journal of personality and social psychology*, 54(2):243, 1988.

[9] Sanjay Bilakhia, Stavros Petridis, Anton Nijholt, and Maja Pantic. The mahnob mimicry database: A database of naturalistic human interactions. *Pattern recognition letters*, 66:52–61, 2015.

[10] Steven M Boker, Jennifer L Rotondo, Minquan Xu, and Kadijah King. Windowed cross-correlation and peak picking for the analysis of variability in the association between behavioral time series. *Psychological methods*, 7(3):338, 2002.

[11] Lubomir Bourdev and Jitendra Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1365–1372. IEEE, 2009.

[12] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In *arXiv preprint arXiv:1812.08008*, 2018.

[13] Joseph N Cappella. Coding mutual adaptation in dyadic nonverbal interaction. *The sourcebook of nonverbal measures: Going beyond words*, pages 383–392, 2005.

[14] Herbert H. Clark. *Using Language*. 'Using' Linguistic Books. Cambridge University Press, 1996.

[15] William S Cleveland and Susan J Devlin. Locally weighted regression: an approach to regression analysis by local fitting. *Journal of the American statistical association*, 83(403):596–610, 1988.

[16] Emilie Delaherche and Mohamed Chetouani. Multimodal coordination: exploring relevant features and measures. In *Proceedings of the 2nd international workshop on Social signal processing*, pages 47–52. ACM, 2010.

[17] 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.

[18] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. RMPE: Regional multi-person pose estimation. In *ICCV*, 2017.

[19] Ruth Feldman. Infant–mother and infant–father synchrony: The coregulation of positive arousal. *Infant Mental Health Journal: Official Publication of The World Association for Infant Mental Health*, 24(1):1–23, 2003.

[20] Ruth Feldman. Parent–infant synchrony: Biological foundations and developmental outcomes. *Current directions in psychological science*, 16(6):340–345, 2007.

[21] Martin A Fischler and Robert A Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on computers*, (1):67–92, 1973.

[22] Dariu M Gavrila. The visual analysis of human movement: A survey. *Computer vision and image understanding*, 73(1):82–98, 1999.

[23] Karl Grammer, Masanao Honda, Astrid Juette, and Alain Schmitt. Fuzziness of nonverbal courtship communication unblurred by motion energy detection. *Journal of personality and social psychology*, 77(3):487, 1999.

[24] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7297–7306, 2018.

[25] Amanda W Harrist and Ralph M Waugh. Dyadic synchrony: Its structure and function in childrens development. *Developmental Review*, 22(4):555–592, 2002.

[26] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[27] Zdravko Ivankovic, Milos Rackovic, and Miodrag Ivkovic. Automatic player position detection in basketball games. *Multimedia tools and applications*, 72(3):2741–2767, 2014.

[28] Xiaofei Ji and Honghai Liu. Advances in view-invariant human motion analysis: a review. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(1):13–24, 2010.

[29] Kentaro Kodama, Shintaro Tanaka, Daichi Shimizu, Kyoko Hori, and Hiroshi Matsui. Heart rate synchrony in psychological counseling: A case study. *Psychology*, 9(07):1858, 2018.

[30] Ivana Konvalinka, Dimitris Xygalatas, Joseph Bulbulia, Uffe Schjødt, Else-Marie Jegindø, Sebastian Wallot, Guy Van Orden, and Andreas Roepstorff. Synchronized arousal between performers and related spectators in a fire-walking ritual. *Proceedings of the National Academy of Sciences*, 108(20):8514–8519, 2011.

[31] Marianne LaFrance. Nonverbal synchrony and rapport: Analysis by the cross-lag panel technique. *Social Psychology Quarterly*, pages 66–70, 1979.

[32] Rivka Levitan and Julia Hirschberg. Measuring acoustic-prosodic entrainment with respect to multiple levels and dimensions. In *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[33] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[34] Zhao Liu, Jianke Zhu, Jiajun Bu, and Chun Chen. A survey of human pose estimation: the body parts parsing based methods. *Journal of Visual Communication and Image Representation*, 32:10–19, 2015.

[35] Norbert Marwan. How to avoid potential pitfalls in recurrence plot based data analysis. *International Journal of Bifurcation and Chaos*, 21(04):1003–1017, 2011.

[36] Norbert Marwan and Jürgen Kurths. Nonlinear analysis of bivariate data with cross recurrence plots. *Physics Letters A*, 302(5-6):299–307, 2002.

[37] Norbert Marwan, M Carmen Romano, Marco Thiel, and Jürgen Kurths. Recurrence plots for the analysis of complex systems. *Physics reports*, 438(5-6):237–329, 2007.

[38] Norbert Marwan, Niels Wessel, Udo Meyerfeldt, Alexander Schirdewan, and Jürgen Kurths. Recurrence-plot-based measures of complexity and their application to heart-rate-variability data. *Physical review E*, 66(2):026702, 2002.

[39] Alvin W Moore and James W Jorgenson. Median filtering for removal of low-frequency background drift. *Analytical chemistry*, 65(2):188–191, 1993.

[40] Davida Navarre. Posture sharing in dyadic interaction. *American Journal of Dance Therapy*, 5(1):28–42, 1982.

[41] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in Neural Information Processing Systems*, pages 2277–2287, 2017.

[42] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, pages 483–499. Springer, 2016.

[43] 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.

[44] Wanli Ouyang, Xiao Chu, and Xiaogang Wang. Multi-source deep learning for human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2329–2336, 2014.

[45] Jane Paulick, Anne-Katharina Deisenhofer, Fabian Ramseyer, Wolfgang Tschacher, Kaitlyn Boyle, Julian Rubel, and Wolfgang Lutz. Nonverbal synchrony: A new approach to better understand psychotherapeutic processes and drop-out. *Journal of psychotherapy integration*, 28(3):367, 2018.

[46] Ronald Poppe. Vision-based human motion analysis: An overview. *Computer vision and image understanding*, 108(1-2):4–18, 2007.

[47] Ronald Poppe, Sophie Van Der Zee, Dirk KJ Heylen, and Paul J Taylor. Amab: Automated measurement and analysis of body motion. *Behavior research methods*, 46(3):625–633, 2014.

[48] Fabian Ramseyer and Wolfgang Tschacher. Nonverbal synchrony or random coincidence? how to tell the difference. In *Development of multimodal interfaces: Active listening and synchrony*, pages 182–196. Springer, 2010.

[49] Fabian Ramseyer and Wolfgang Tschacher. Nonverbal synchrony in psychotherapy: coordinated body movement reflects relationship quality and outcome. *Journal of consulting and clinical psychology*, 79(3):284, 2011.

[50] Daniel Richardson, Rick Dale, and Kevin Shockley. Synchrony and swing in conversation: Coordination, temporal dynamics, and communication. *Embodied communication in humans and machines*, pages 75–94, 2008.

[51] Daniel C Richardson and Rick Dale. Looking to understand: The coupling between speakers' and listeners' eye movements and its relationship to discourse comprehension. *Cognitive science*, 29(6):1045–1060, 2005.

[52] John W Robinson, Al Herman, and Bonnie J Kaplan. Autonomic responses correlate with counselor–client empathy. *Journal of Counseling Psychology*, 29(2):195, 1982.

[53] Lorraine Rocissano, Arietta Slade, and Victoria Lynch. Dyadic synchrony and toddler compliance. *Developmental Psychology*, 23(5):698, 1987.

[54] Daniel Roetenberg, Henk Luinge, and Per Slycke. Xsens mvn: full 6dof human motion tracking using miniature inertial sensors. *Xsens Motion Technologies BV, Tech. Rep*, 1, 2009.

[55] Benjamin Sapp, David Weiss, and Ben Taskar. Parsing human motion with stretchable models. In *CVPR 2011*, pages 1281–1288. IEEE, 2011.

[56] Claire Schmais and Diana Jacoff Felber. Dance therapy analysis: A method for observing and analyzing a dance therapy group. *American Journal of Dance Therapy*, 1(1):18–25, 1977.

[57] Richard C Schmidt, Claudia Carello, and Michael T Turvey. Phase transitions and critical fluctuations in the visual coordination of rhythmic movements between people. *Journal of experimental psychology: human perception and performance*, 16(2):227, 1990.

[58] Désirée Schoenherr, Jane Paulick, Susanne Worrack, Bernhard M Strauss, Julian A Rubel, Brian Schwartz, Anne-Katharina Deisenhofer, Wolfgang Lutz, Ulrich Stangier, and Uwe Altmann. Quantification of nonverbal synchrony using linear time series analysis methods: Lack of convergent validity and evidence for facets of synchrony. *Behavior research methods*, pages 1–23, 2018.

[59] Jeffrey S Simonoff. *Smoothing methods in statistics*. Springer Science & Business Media, 2012.

[60] Marianne Sonnby-Borgström, Peter Jönsson, and Owe Svensson. Emotional empathy as related to mimicry reactions at different levels of information processing. *Journal of Nonverbal behavior*, 27(1):3–23, 2003.

[61] David Stavens and Sebastian Thrun. Unsupervised learning of invariant features using video. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1649–1656. IEEE, 2010.

[62] Mariëlle Stel, Eric Van Dijk, and Einav Olivier. You want to know the truth? then don't mimic! *Psychological Science*, 20(6):693–699, 2009.

[63] Mariëlle Stel and Roos Vonk. Mimicry in social interaction: Benefits for mimickers, mimickees, and their interaction. *British Journal of Psychology*, 101(2):311–323, 2010.

[64] Linda Tickle-Degnen and Robert Rosenthal. The nature of rapport and its nonverbal correlates. *Psychological inquiry*, 1(4):285–293, 1990.

[65] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1653–1660, 2014.

[66] Piercarlo Valdesolo and David DeSteno. Synchrony and the social tuning of compassion. *Emotion*, 11(2):262, 2011.

[67] Alessandro Vinciarelli, Maja Pantic, and Hervé Bourlard. Social signal processing: Survey of an emerging domain. *Image and vision computing*, 27(12):1743–1759, 2009.

[68] Sebastian Wallot, Panagiotis Mitkidis, John J McGraw, and Andreas Roepstorff. Beyond synchrony: joint action in a complex production task reveals beneficial effects of decreased interpersonal synchrony. *PloS one*, 11(12):e0168306, 2016.

[69] Charles L Webber Jr and Norbert Marwan. Recurrence quantification analysis. *Theory and Best Practices*, 2015.

[70] Charles L Webber Jr and Joseph P Zbilut. Dynamical assessment of physiological systems and states using recurrence plot strategies. *Journal of applied physiology*, 76(2):965–973, 1994.

[71] Gordon Wright, Fiona Gabbert, Magdalene Ng, Serena Ventura, and Paraschiv Roxana. Quantifying the effectiveness of an evidence-based rapport-building training programme. 2018.

[72] Yuliang Xiu, Jiefeng Li, Haoyu Wang, Yinghong Fang, and Cewu Lu. Pose Flow: Efficient online pose tracking. In *BMVC*, 2018.

[73] Joseph P Zbilut, José-Manuel Zaldivar-Comenges, and Fernanda Strozzi. Recurrence quantification based liapunov exponents for monitoring divergence in experimental data. *Physics Letters A*, 297(3-4):173–181, 2002.

List of Figures

List of Tables