



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Laboratory of Movement Analysis

Optional project in Communication Systems

A graphical user interface for physical activity analysis using various sensors configurations

Author:
Matteo Ciprian

Professor:
Prof. Kamiar Aminian
Supervisor:
Dr. Anisoara Ionescu

20 December 2017

Abstract

The possibility of monitoring in daily life the movement behavior of children affected by movement disorders is nowadays possible thanks to the development of wearable and precise inertial sensors, including accelerometers and gyroscopes (i.e. Physilog). The signals recorded by these sensors fixed on specific body parts, are processed with sophisticated signal processing techniques allowing the computation of important movement parameters, subsequently used by clinicians/therapists to make a diagnosis on the patient or to choose the proper therapy. Although most of these routines are just implemented, no graphical interface were developed for the final users, making difficult the usage by the clinicians. The goal of this project consists in creating an application which integrates all movement analysis algorithms in a new graphical user interface which helps the final user to interact with the program. This report describes the approach followed for developing this software, and provide also a detailed explanation of the facilities included.

1 Introduction

Over the last decades inertial sensors have been widely exploited for human movement activity analysis, more specifically for physical activity and gait analysis. These devices can be easily worn and applied in a specific part of the body, making possible a quantitative assessment of physical activity/movement features. The miniaturization and commercialization of these sensors as well as the reduction of the cost production has enabled, during the last years, the possibility to monitor people affected by movement disorders also in daily life. Despite a wide variety of movement monitoring devices (monitors) have been designed so far, it is possible to divide them in four main categories: 3D-accelerometer, for measuring the acceleration along the Frontal, Vertical and Lateral axis , 3D-gyroscopes for measuring the angular velocity along the Yaw, Roll, and Pitch axis, magnetometers for the magnetic field, and barometers for the pressure (allowing to measure the altitude). All these sensors are usually packed in a single device which can be directly applied in the interested part as depicted in figure 1. The recorded signals in binary format are synchronized (if monitoring includes more than 1 device), then data is saved in a appropriate format (e.g., .csv or .mat), ready to be processed by the algorithms. This approach has been used only in laboratory context where, after an initial phase of calibration (functional calibration), the patient had to move for an enough long period in monitored mode and according to a specific protocol. Despite the numerous validations of this method also in clinical environment, some limitations can be underlined. At first the data collected with this technique are referred to a limited time frame; secondly, evaluating the patient in supervised circumstances could affect the results since the patient is stimulated in improving his/her movement performance furnishing to the analyst a clinical picture not in agreement with the real conditions.

The possibility of monitoring the patient in daily life context could be a possible solution to these problems giving the opportunity to process more realistic data in reference to a longer period of movement. Following this approach in the last years, LMAM (Laboratory of Movement Analysis and Measurement, EPFL) has developed several routines based on different signal processing techniques to extract some important movement parameters [SRV⁺04], [PIBR⁺04]. These functions are implemented in Matlab but are not integrated one to the other in a unique application with a user-friendly graphical interface. The user has no possibility to directly interact with the process choosing the configuration of the sensors or selecting the correct axis. All these aspects limit/make difficult the usage of these new tools in clinical environment and, for this reason, the work in this master project is a first tentative of developing a graphical user interface to solve the cited problems. This platform is also able to manage all the data generated by the routines and visualizing the results on different kind of reports, automatically generated. In addition the software gives to the user the opportunity of storing the results in a database structure to allow the management of several patients.

This application is directed to clinicians, therapist and human movement scientists that will be referred simply as “user” from now on.

As stated in the paper, all the project choices have been done in agreement with the clinicians of CHUV and all the routines present in the current version of the application are updated to the new requirements. The report is structured in the following way: in 2 the whole structure of the application is explained in relation to the initial problematics. In the sections from 3 to 5 each part of the application is presented in detail while in 6.2, the main results and the performances of the software are discussed in relation to the future works and the improvements that can be done.

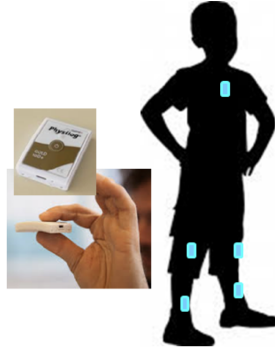


Figure 1: Left: Example of a monitor: “Physilog”; Right: example of a possible configuration of different monitors in a child body

At the end, the report includes two appendixes: *Manual for the user*, intended as a guide for the movement experts who will use this application, *Manual for the developers*, a guideline for the developers who will have to update and improve the software in the future.

2 Structure of the application

2.1 Routines developed and main problematics

The routines developed so far consist in a group of functions written in matlab which take as input the “.mat” files produced by the sensors. Each file is labeled as the body part on which it has been applied (i.e. “trunk.mat”) and contains the samples of every recorded signals: acceleration, angular velocity, pressure and magnetic field (in the context of this study magnetometers were not used). Acceleration and angular velocity are 3-D signals and are composed by three different channels corresponding to different *sensitive axis*.

A specific routine can be classified according to two main aspects: the kind *output results* and the *input sensor configuration*.

According to the first criteria it is possible to divide the routines in *Physical Activity Analysis* and *Gait Analysis*. In the first group all the parameters about the posture classification and the intensity of physical activity are computed. They can be summed up as following:

- *Daily posture allocation*: the percentage of time in which the subject is “standing”, “sitting/lying”, “walking” as overall;
- *Posture allocation per hour*: the percentage of time in which the subject is “standing”, “sitting/lying”, “walking” in one hour;
- *Intensity Pattern*: Variation in the intensity of the physical activity over the period considered;
- *Pattern of behavior*: Variation in the type of movement in relation to the variation of ”altitude” (walking down stairs, walking upstairs etc.);
- *Duration of the periods*: The duration of each of the single period of “standing”, “sitting/lying”, “walking”;

For *Gait Analysis* the following parameters are considered:

- *Gait Cycle Time*: The period of time from one event (usually initial contact) of one foot to the following occurrence of the same event with the same foot
- *Cadence*: Rate at which a person walk, expressed expressed in *steps/min*
- *Double Support*: Duration of the phase in which the body weight is supported by both legs.

- *Stride length*: the linear distance in the plane of progression between two successive points of foot floor contact of the same foot;
- *Step length*: Linear distance in the plane of progression between two successive points of foot floor contact of the opposite feet;
- *Swing duration*: The period of time when the foot is not in contact with the floor;
- *Stance duration*: The period of time when the foot is in contact with the floor;
- *Speed*
- *Limp*: percentage of limp in the walking period;
- *Steps per walking period*
- *Knee Angle*
- *Thigh Angle*
- *Shank Angle*
- *Kinematic Curves*: Variation in time of the angles cited above;
- *Symmetry Indexes*;

Considering different *input sensors configuration*, it is possible to devise algorithms that work with a *full configuration* or with *reduced configuration*. In the full configuration a sensor is applied to each part of the body, for instances: *Trunk, Left Thigh, Right Thigh, Left Arm, Right Arm, Left Shank, Right Shank*. In this study the arms are not considered and the full configuration is defined as following: *Trunk, Left Thigh, Right Thigh, Left Shank, Right Shank*.

In the current version of the software two main routines are included:

- *Physical_Activity_Classification*, full configuration;
- *Extract Gait Parameters(Sh,Th)*, reduced configuration: Left Thigh, Right Thigh, Left Shank, Right Shank;

Other routines which consider different type of configurations are completed but they will be available on this platform only in the future versions. For example, in Gait Analysis another routine has been developed for estimating gait parameters only with data coming from sensors attached to the two shanks [CP17]. As stated in first sentences, each routine needs to process a certain set of signals which came from a specific *sensitive axis* in order to compute properly the movement parameters. In relation to the 3-D accelerometer it is possible to consider 3 axis: Frontal axis, Vertical axis and Lateral axis as partially shown in 2, while in relation to the gyroscope the sensitive axis are named as in 3: Roll axis, sensitive to the rotation on frontal axis, Yaw axis, sensitive to the rotation on vertical axis and Pitch sensitive to the rotation on lateral axis. Only as example, a specific routine could require the signal of the vertical acceleration recorded by the accelerometer on the trunk (“Trunk vertical acceleration”) or/and the signal of the angular velocity on pitch axis that comes from gyroscope positioned on the Left Shank (a typical pattern is depicted on 4) and so on. This report will not go through the rationale below the signal processing methods used to extract the output parameters but useful informations can be found in the following studies: [ALKY11], [SRV+04], [SRJGV+08].

The selection of the correct signal raises a very important problem that has to be considered in the development of the application: *the axis selection*. This issue will be explained with an example. Considering the 3D-accelerometer, for a given placement of the sensor, let the signals be recorded in the following way: channel-1 records the acceleration on the frontal axis (frontal acceleration), channel-2 records acceleration on the vertical axis and channel-3 records acceleration on the lateral axis; this correspondence is true for a specific placement of the monitor but this is not true if the device is positioned in a different way. The same problematic concern the gyroscope. The fixation of wearable sensors on body segments of children affected

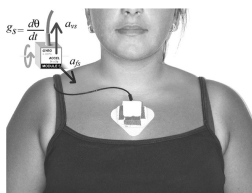


Figure 2: Sensor positioned on the trunk with vertical acceleration axis and frontal acceleration axis underlined

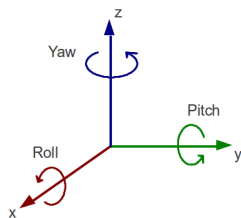


Figure 3: Reference axis configuration for measuring angular velocity

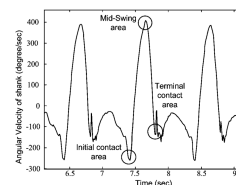


Figure 4: Typical pattern of pitch angular velocity of the shank

by cerebral palsy (sometimes affected also by limbs deformation) is often very difficult and the position of the monitors has to be changed according to his/her specific physical structure. Therefore, data collected from different subjects can vary relatively to the defined measurement protocol, but has to be corrected before the use by analysis algorithms/routines. When measurements are performed in laboratory context, under the supervision of the clinical investigator, this issue can be solved by a functional calibration procedure (sequence of specific body movements and data processing procedure). However, this approach is not possible when the measurements are taken in daily-life context and/or in children with severe movement impairments due to the difficulties to accurately perform the sequence of necessary movements. Solving this problematic is a very important point and the software has to be designed in such way to allow the user to choose what is the correct configuration of the axis. In other terms, the analyst has to be able to detect the correct channel for a specific signal required by the routine. This goal can be pursued in two ways. The first one is programming a graphical user interface which enable the user to manually select the channels according to the pattern of the signals visualized. Another solution would consist in developing an algorithmic procedure which performs these computations in an automatic way without user intervention.

Another milestone of this project concerns the “*automatic report generator*”. The existing routines only display at standard output the results of the analysis, saving them without a proper organization. Clinicians need an efficient mode to visualize the results, which have to be easily exported and exchanged. Saving all the plots and tables in document format seems to be the more suitable solution for this problem. Each report, has to be identified by the ID of the patient and by the ID of the single ‘perform’¹ and it should contain useful informations on the patient and about the context like the date of analysis, weather conditions. This can be done only with the presence of a *database* which allows the application to manage and save all the data in an ordered way.

Another specification for the development of this software concern the *updatability*. The routines developed could be updated in the future by adding new functions with different input configurations of the sensors or other type of results. For this reason modularity is essential and have to be considered as one of the priority for the project development.

¹In this context the term “perform” is used to define a single analysis session which usually correspond to a continuous monitoring session, for example from one day

2.2 Project choices and scheme of the application

Project choices All the points evaluated in the previous paragraph have to be considered as a baseline for the software. After a planning phase some important choices had to be done especially considering the problem of *Axis selection*. Although automatizing the process would be a very elegant way to solve the problem, this approach could be very challenging. Assigning the correct channel to a specific axis would be possible by exploiting pattern matching techniques combined with a machine learning approach, but the non-regularity of the signals (the signals recorded from children affected by movement disorders) could jeopardize the efficiency of this approach. Moreover, considering the time constrains, the development of this advanced functionality could have compromised the implementation of other important parts of the software (for example the report generator). After a meeting with the clinical collaborators from CHUV (“Centre hospitalier universitaire vaudois” of Lausanne) it was decided to give priority to the realization of the project in direction of fully integration of the available routines and manual axis selection, letting the implementation of *automatic axis selection* routines as future work; some useful guidelines are still given in section 6.2. Therefore, the current version, the application provides a graphic tool for selecting manually the correct channels as better explained in section 4.1.

Focusing now on *report generator*, it is clear that only a database structure can support this facility. Considering this aspect the application has to be developed in several levels and has to be organized in two main blocks. One part, *Analysis tool* is the core of the software and it is dedicated on the management of the routines for parameters extraction. It is separated from the rest of the application whose task is the management of the generated data in relation to the database.

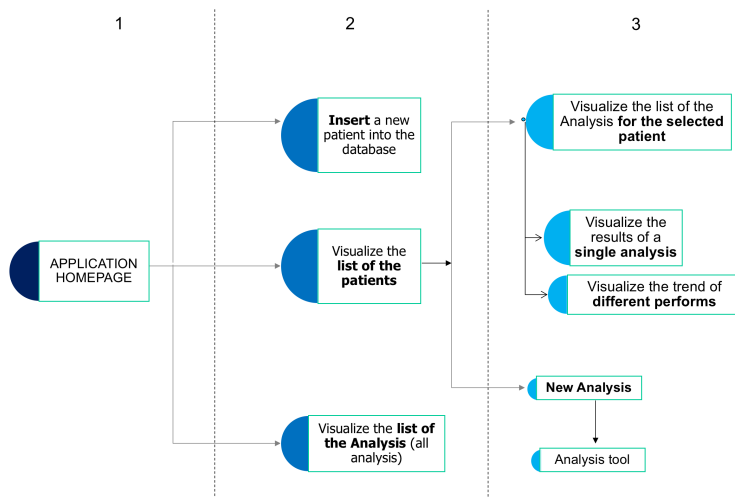


Figure 5: General scheme of the application

Scheme of the application In figure 5 is depicted the basic structure of the software. In a first step the user has the possibility of choosing what is the main operation he/she wants to execute. At the current version, the application provides three main options: *Inserting a new patient*, *Visualizing the list of the patients*, *Visualizing the list of the analysis*. By choosing the second option it is possible to enter in the core of the software. Once the user selects a patient from the list, he has the opportunity to visualize the list of the analysis already done for a *single* patient or, to start a new analysis. The first option allows the user to see the results and the report of one analysis performed. In this context the clinician can also consider a set of different analyses chosen from the list and can to compare their results in a dedicated interface which enable the visualization of a temporal trend. By choosing *New Analysis* the user can access the *Analysis Tool*, described in Figure 5. This tool is divided in two main parts corresponding to the two different types of routines: Physical Activity Analysis and Gait Analysis. At this point it is possible to choose the correct sensor configuration and after several steps the selected routine is run. The results presented as plots/graphs and tables are visualized at standard output with a dedicated interface, or on the report automatically generated. After this phase all the results are saved in a local folder closing the loop and allowing the user

to interact with new data through the database software at level 2 and 3.

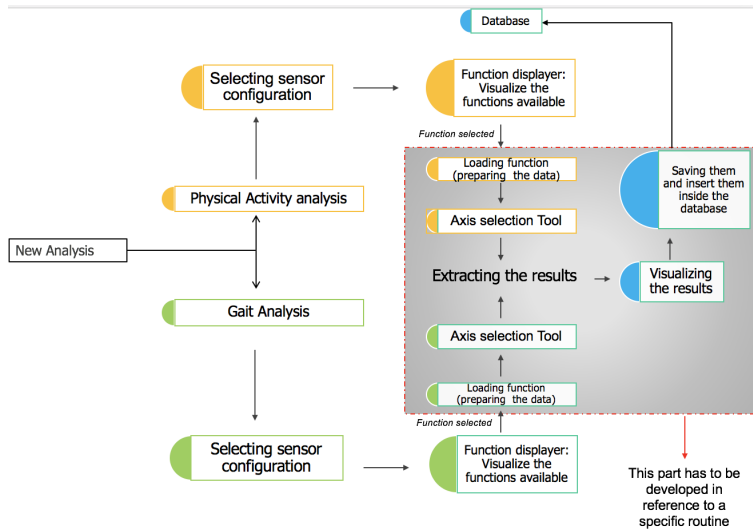


Figure 6: General scheme of the “Analysis Tool”

2.3 Tools used and work-flow organization

All the previous routines being written in Matlab, for compatibility, all this software has been developed using Matlab as well. This programming environment is particularly powerful for signal processing and provide also a tool for programming graphical interface: GUIDE. Despite the advantages of this choice some drawbacks exist and are underlined in section 6.1. For creating a so-called “Standalone” application which work without the use of Matlab, the *Deploy tool* can be used, enabling the creation of the executable starting from the matlab functions. After this project, a first trial of application in Matlab will be furnished to the clinicians at CHUV for a first period of testing. Once the feed-back of the clinicians will be received it will be possible to start with the phase of deployment. More informations about the deployment can be founded on the Appendix “Manual for Developers”.

3 Database

Database structure is one of the most innovative aspect of this application. It gives the user the opportunity to manage the results and saving them in an structured manner. The database provided is very simple and it is essentially composed by two main entities: the patient and the analysis. As suggested by the name, “*Patient entity*” identifies uniquely the patient with his/her personal and clinical informations. To insert a new patient into the database there is a specific function runnable from the home page. Here the user has to fill a form with the following details: *ID*, *Name*, *Surname*, *Date of birth*, *Gender*, *CP Subtype*, *Height*, *Weight*, *GMFCS level*, *Other pathologies*. These parameters are established in agreement with the clinicians but in order to make the application as portable as possible, no mandatory field have been set except the *ID* which is unique for each patient and it can be chosen by the user. Some specific functions controls the validity of the ID assuring its consistency and uniqueness.

For a single patient it is possible to have more analyses that may correspond to successive monitoring sessions. The ID of a single analysis is automatically generated by the algorithm in the following way:

$$“Patient” + ID_{patient} + \text{Type of Analysis} + “ID” + N_{analysis}$$

where $ID_{patient}$ is the ID of the patient and $N_{analysis}$ is a consecutive number incremented each time a new analysis is added to the database. For instance *Patient_CP05_Gait_Analysis_ID-2* identifies the second analysis of the patient *CP05* which is a Gait Analysis.

Before starting each perform, the user can insert the following details: *Date* in which the measurements have been taken, the weather, and also the *advancement of the analysis*. This is a very important parameter because it identifies the analysis itself in a chronological context. According to the protocol defined by the clinicians (specific to the current clinical study), one perform, intended as a pair *Physical Activity Analysis + Gait Analysis*, can be classified in four degrees of advancement: *Baseline* (the first), *Pre* (or *Follow-up1*), *Post*(or *Follow-up2*), *Follow-up* (or *Follow-up3*). These four advancements compose an entire cycle of analysis and can be used to visualize the trend and the variation of the clinical conditions over a certain period. The structure of the application assures that a new analysis can be initialized only if it is connected to a valid patient inserted in the database. The visualization of the analysis is possible in two different ways. The first way consist in opening, from the home-page, the entire list of the analysis ordered according to alphabetical criteria (regardless the patient). The second way consist in acceding before to the list of the patients and, once an ID is selected, visualizing the list of the analysis only for that specific patient. This distinction can be considered trivial but it is very important to assure the final utilizer the best user experience.

4 Analysis tool

The Analysis tool is the core of the application because it provides an interface between the user and the scripts written to extract the movement parameters. When a new analysis is started there are seven steps to follow:

- *Analysis Selection*: selecting the type of analysis between Physical Activity Analysis and Gait Analysis.
- *Sensor Configuration Selection*: according to the input data available the user has the possibility of selecting the configuration of the sensors.
- *Function visualizer*: When the selection is done, another interface visualizes at standard output the routines available for the chosen configuration. At the current version only the routines with the full are included in the application but in the future it will be possible to update the software inserting new functions for different sensor configuration. A detailed explanation of this part is provided in section 4.3.
- *Adding analysis details*: in this section the user can insert advanced details on the starting analysis as for example the advancements. All these details will be stored in the database and will reported in a second moment in the report.
- *Loading function*: the raw signals recorded by the sensors need to be preprocessed before being analyzed. According to the standard defined in this application all the pre-processing have to be done in this part.
- *Axis selection*: when pre-processing is done all input data is passed to this “Axis selection user interface”. This interface is programmed to show at standard output the signals recorded for each channel and allow the user to select the appropriate channels as described in 4.1.
- *Routine processing*: when the correct signals are identified it is possible to progress with the next steps of analysis. The function selected on the “function visualizer” interface is run to extract the output parameters. To maintain the code as simple as possible, this part have been split in several functions.
- *Results visualization*: when the parameters are extracted and the plots generated, all the results are displayed with a dedicated graphic interface. At this step it is also possible to generate the report.

4.1 Axis selection Tool

As stated above this tool has been created to solve the problem of selecting the correct channels in reference to the required signals. This tool provides at first a graphic interface to show all the signals recorded in the measurements, allowing the user to visualize them according to the label (Trunk, Shank, ecc..) and the channel. The functioning of this tool will be here explained with an example considering a routine developed

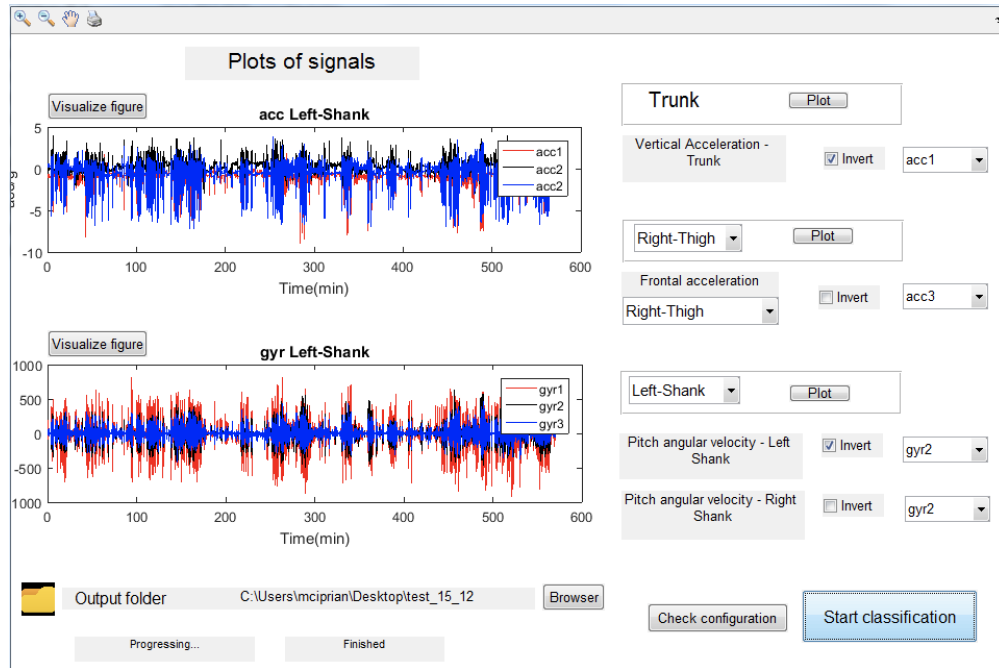


Figure 7: Visualization tool

for “Gait Analysis”. It works with a reduced configuration (Right Thigh, Left Thigh, Right Shank, Left Shank) and it need to process four specific signals to complete its computation: Angular velocity of the Right and Left Thigh on the Pitch Axis and Angular velocity of the Right and Left Shank on the Pitch Axis. The user can plot all the signals referred to a specific monitors as shown in figure 7 (in the figure Left Shank is considered). Here it is possible to focus on the three angular velocities pattern recorded by the gyroscope and visualize them separately according to the channel. Zooming the signal on a specific region, the user can recognize the pattern of the signal interested (in this case gyr2) and select the correct channel as depicted on figure 7. All this procedure has been agreed with the clinicians of CHUV and practical guide can be found in *Appendix 1*. In section “Axis Selection: Protocol Data Analysis” the user can find a standard procedure for recognizing the correct pattern for all the specifics signals required by the routines present so far in the software.

For what concern *Gait Analysis* at this step the user has to furnish to the interface some additional parameters as:

- Length of the right and left leg
- Length of tibia
- Length of femur
- Threshold of speed: Maximal speed of walking periods selected for evaluationn (e.g. comparison with control group, post-pre clinical intervention)
- Threshold of steps: Minimum number of steps required for a detected walking period to be considered for analysis (extraction of gait parameters)

As it can be seen in 6, the user has also the possibility of choosing an external folder to save a copy of the results. Anyway, for maintaining the application as robust as possible the output data generated by the routines are saved in a local folder always reachable by the software.

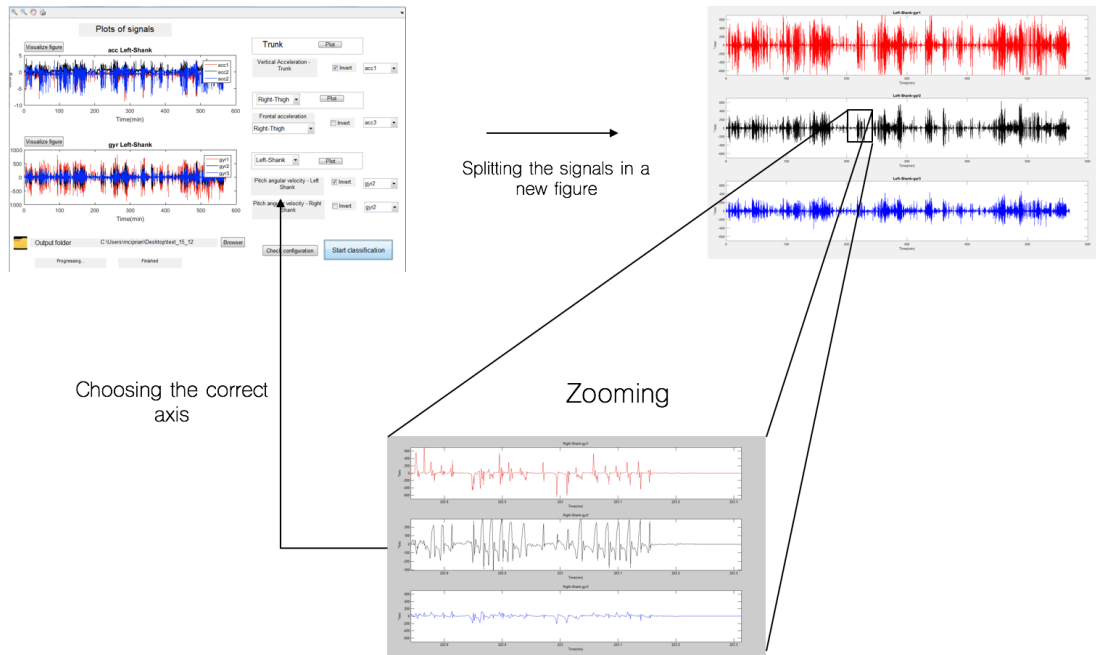


Figure 8: Procedure for detecting the correct channel for the required axis. In this case the interested signal is the Pitch angular velocity of Left Shank. As possible to see in the reference pattern 8 channel 2 is the correct one

4.2 Results Visualization

Once the correct configuration of input signals has been selected, the routine compute the parameters of interest. The results are summed up in some graphs and tables which are saved automatically in a local folder managed by the database software as soon as the computation is finished. To make the visualization easier, a specific interface has been developed for each of the two routines considered so far (*Physical_Activity_Classification(Full_Configuration)* and *Gait_Analysis_Sh.Th*); here the user has the possibility to look at each graph in an ordered manner directly on the interface 9 or, if he wants, on the report, automatically generated as explained in 5. The function which realizes this “result-interface” is called at the end of each routine but can also be called in a second moment once a specific analysis is selected from the database. 5.

Considering the feed-backs given by the clinicians in the last meetings, on the section dedicated to Gait Analysis it is also possible to generate and exports all the data generated in an excel table directly from the *Results Visualization interface*. This can be very useful for the clinicians to deeply inspect the results since in these tables all the parameters are displayed per single gait trial and per gait cycle, furnishing a detailed view of the Gait parameters.

From a graphical point of view, this interface is also equipped with a dedicated visualization tool which allow to plot the chosen graph in a new figure. This can be considered a trivial aspect but it can happen that in a not very performing computer the quality of the graphs can be strongly reduced making impossible a correct inspection. Thanks to this facility the problem has been partially solved.

As for *Axis selection* interface, “Results Visualization” has to be adapted according the routine considered according to some specifics underline in the next section.

4.3 Updating section

Updatability can be considered one of the most important requirements for this software. The possibility for future developers of adding new functions or changing the current ones without compromising the whole

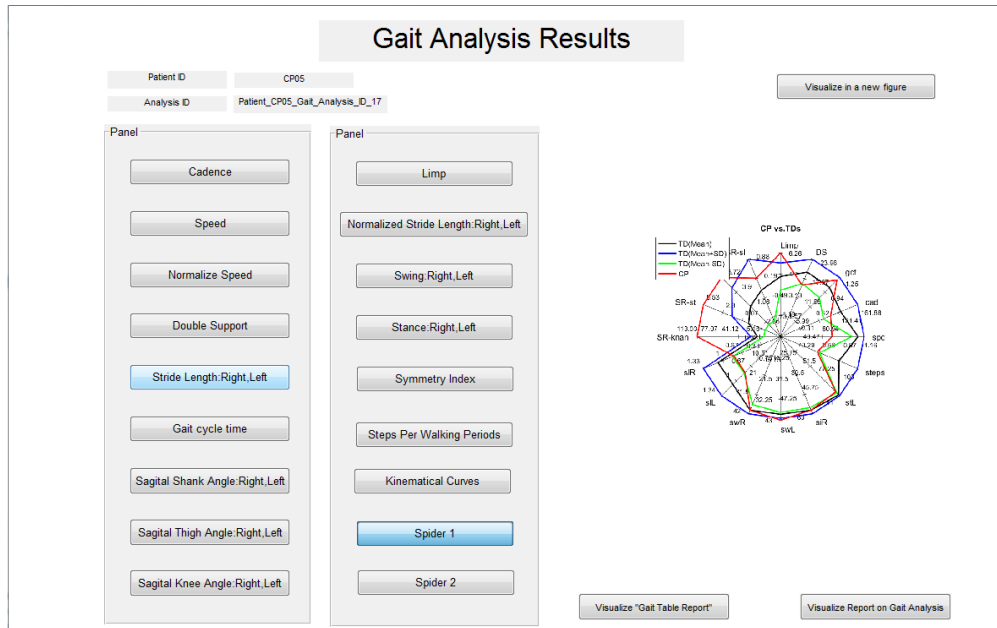


Figure 9: Example of the user interface for visualizing the results of Gait Analysis;

code is one of the most important aspects. The layered structure makes the updating easier because, as possible to see in figure 6, only the part included in the square has to be adapted on the specific routine. *Function displayer* interface is connected to a sort of “database of the routines”, basically a “.mat” structure containing the list of the routines available with indicated the name of function and the configuration of the sensors it needs to make the computation. With this scheme, adding a new function can be done following these steps:

- adding the name of the routine and its input configuration to the “.mat” structure corresponding to “database of the routines”
- updating *Function displayer* interface in such way the user can visualize and select the new routine;
- developing the user remaining part of the user interface for *pre-processing*, *axis selection* and *Result visualization* as defined on the standard shown in 6

More detailed informations can be found in Appendix 2, *Manual for developers* in the section “How to update the application” .

5 Report Generator

All the output data produced by the routines have to be summed up in a document format to help the user in the inspection and to the results make exportable in an simple manner allowing a rapid exchange of informations. For this reason a dedicated interface has been created to export in automatic way all the results in a Word document. The report is generated at at the end of each analysis and can be inspected from ”Results Visualization” interface.

All the routines programmed at this goal uses “Matlab Report Generator” Toolbox which integrate functions to create a customizable report relying on some advanced facilities provided in Office Development Kit. Despite this is a very powerful and efficient tool some problematic is underlined in section 6.1. From a point of view of the updatability, the routines for report generator are sensitive on the output results and not the input configuration. Considering this aspect it is possible to reuse the same interface for different functions as long as the same output data are furnished.

At the current version there are three kind of reports available: “Physical Activity Analysis Report”, “Gait Analysis Report” and “Multi-perform Report”:

- *Physical Activity Analysis Report*: here are schemed the output parameters relative to the posture and physical activity classification cited in 2.1;
- *Gait Analysis Report*: here are schemed the output parameters relative to Gait analysis 2.1;

As it is possible to see in the figures 12 all the data are collected in graphs of different types to enable an easy visualization but are also tabulated in table format for a deep inspection. The preamble of the report shown in 10 and 11 contains also patient details and the informations about the considered analysis. *Multi-perform Report* is one of the newest facility provided in this application. A function collects a set of output data for a chosen sequence of performs and each parameter is compared with its counterparts on the other analysis, generating a new trend-plot. As stated before, according to the protocol, a cycle of analysis is composed by four performs: Baseline, Pre, Post and Follow-up. To create the multiple report it is asked to enter on the database and select a set of eight analysis (“Posture + Gait” for each step). The graphic interface helps the user in this operation visualizing important details of the selected analysis as of course “the advancement” or the date of monitoring.

This new facility is very important to allow the clinicians to investigate over a trend of different analysis, checking the variation of the movement performances and validating the efficiency of a specific therapy.

6 Results of the project

From an experimental point of view the application has been tested many time with different input data to check the consistence of the results and the quality of user experience. The data considered comes for *Physilog 4*, monitor produced by Gait-up but the application can be used to process data from other devices as long as the data furnished are in raw format “.mat”.

From the tests made in laboratory the application can be considered enough stable from a point of view of the computation performances although the hardware of the running device seems to have a considerable impact on the time required to complete a single analysis. In this regard, the presence of a dedicated unit for graphic processing seems to be the most impacting hardware feature since the computational time increase around 30-40% if the running terminal is not furnished of a GPU (Graphic Unit Processor). As stated before, this can also condition the quality of the figures but it is not a mandatory requirement for the correct working of the application. In this respect, based on the empirical tests, some minimum hardware and software requirements are here listed:

- *Operative System*: Microsoft Windows
- *External Software required*: Microsoft Office
- *RAM*: at least 4 GB
- *CPU*: at least 2.2 GHz
- *Matlab Version*: later than 2013

From a project point of view it is undeniable that, at current state, many advantages have been provided to the final user despite many improvements have to be done to get a complete software. At the current version the application is depending on Matlab platform runnable on Windows operative systems and it is ready to be tested by the clinicians of CHUV. As stated in the introduction, in order to deliver a product as complete as possible two manuals have been annexed in this project. The first manual is directed to the final user and can be considered as an “instruction sheet” of the application. The second manual is directed to the future developers who should be able to understand more in deep the structure of the interface. All the code has been commented in order to allow the future programmers to update the software in the easiest way.

6.1 Main problematics

In programming this software many problems have been coped with. Many of them have been solved and others not. This section is intended to be a guideline for helping future developers to have a better

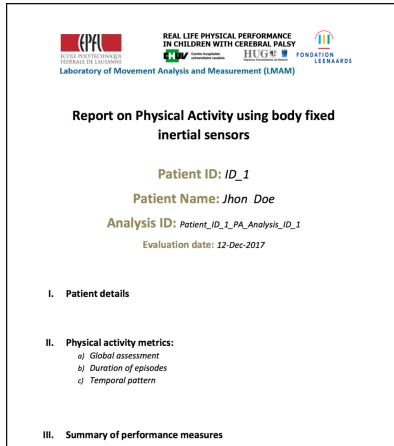


Figure 10: Preamble of the report for “Physical Activity Analysis”

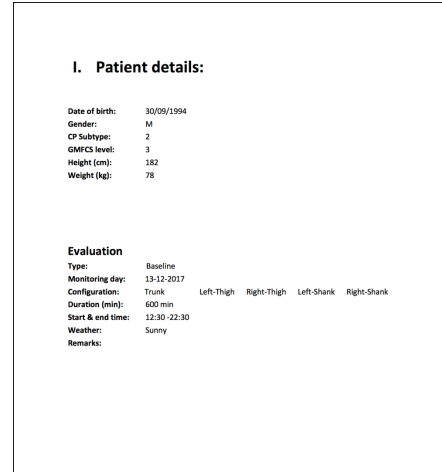


Figure 11: Patient details summed up

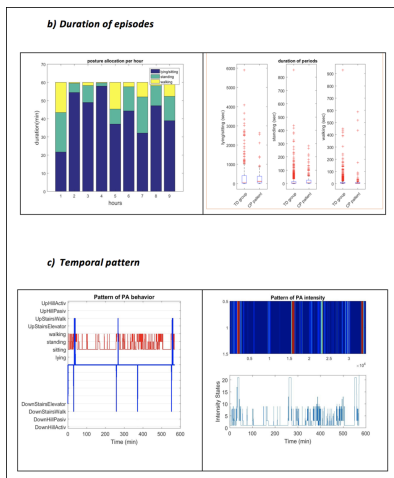


Figure 12: Example of the template used for the report. Here some graphs are shown.

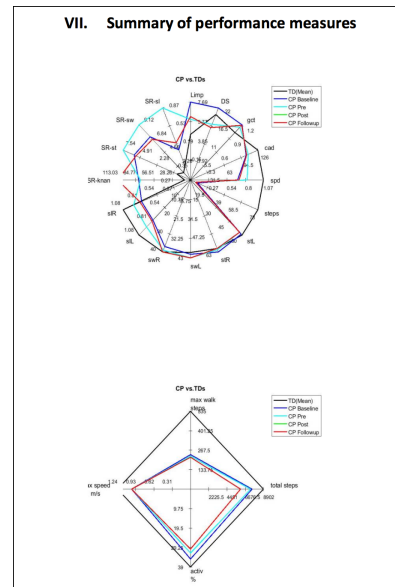


Figure 13: Summary of performances for comparing a complete cycle of analysis. The first is referred to Gait Parameters , the second to Posture Classification

understand of the potentialities and the weak points of this application. A first set of problematics concern directly Matlab programming platform. As known, it is very powerful for signal processing purposes and for this reason all the routines developed in the past for parameters extraction was entirely written in this language. Although the potentiality of this computing environment, some problems have been founded. A first issue concerns the “graphic renderer”. In computer graphic, “graphic rendering” is the automatic process of generating a photo realistic or non-photo-realistic image from 2D or 3D model and “graphic renderer” is the software or the hardware which has this task. Matlab does not provide specific functions for the management at low-level of “graphic rendering” and this functionality is completely delegated to external tools. If the computer is owned by a GPU, the task is carried out by this dedicated unit otherwise it is performed by software, usually OpenGL. In this case the version of the operative system as well as the version of OpenGL can affect deeply the quality of the graphic especially if the computer is not enough powerful. A possible solution in this direction could be to use another language as JavaScript to program the parts of the software which require particular efforts in these terms. On the other hand, it is not yet clear how to integrate functions written in different languages and some investigations have to be done. Another problem deals with the report generator. “Report generator toolbox” provided in Matlab is a very powerful but it relies on specific functions of “Microsoft Office” only available on Windows operative systems. This compromise the use of this application on MAC computers or UNIX operative system. Other possible problems can arise from the deployment tool. It is not clear how the relative path used in the database for files management will be treated when the application will be run from the executable. From a theoretical point of view this should not be a problem but some investigations have to be done to better understand the behavior of Matlab Compiler in this case.

6.2 Future works

As stated in the introduction, all the work done has to be intended as a baseline for the application. In order to furnish to the clinicians a product as complete as possible many improvements can be done. One of the main improvements concerns the “Axis Selection”. As widely explained in 4.1, the application include a tool for manually selecting the channels in relation to the sensitive axis. Programming an algorithm able to do this automatically could improve a lot the user task. This possibility has been theoretically investigated during the project and some possible solutions have been identified for solving this problem.

Many roads can be followed but, so far, two main approaches appear interesting. The first approach is based on supervised machine learning combined with pattern matching techniques. Following this method, the signal is divided in several windows of a specific duration and some informative features are extracted. All the features computed are exploited to characterize as well as possible the pattern of the signal and are used as input of a machine learning *classification* algorithm as, for example, “logistic regression” or “KNN”. The features are extracted firstly on a knowing *training set* of signals measured on patients with similar pathological background of the ones considered in these analysis (in this case children affected by cerebral palsy). These signals are already labeled with their sensitive axis (for instances “Trunk Vertical Acceleration” or “Pitch Angular Velocity”) and the algorithm fits the input data on the assigned label (in this case the label can be considered as the output of the process). Very useful parameters can be computed considering the signal in the time domain as for example: the mean, the variance, the autocorrelation, the mean distance between the consecutive peaks but also in the frequency domain: maximum frequency of the spectrum, mean of the spectrum etc. (all these features are of course referred to a single window). In some cases advanced pattern matching techniques can be used to compare the input signal with a reference pattern as shown in figure 15 and extrapolate a coefficient of similarity. Following this method, once a new signal is considered, the algorithm classifies each time-slot with a label and then the final classification is done just assigning to the signal the most frequent label. This approach can be very effective but a huge work of features engineering has to be done to find the most informative attributes to extract. Moreover there are many specific parameters to tune as for example the length of the window time slot and many cross-validation have to be done. Another problem, typical of machine learning approach, concerns the representativity of the training set in respect to the data set. As just written in the paper, signals coming from children affected by cerebral palsy can be very different one the other in reference to the subtype of pathologies. The method considered has to take into account this problematic.

All the problems connected to features extraction could be potentially solved by using advance machine

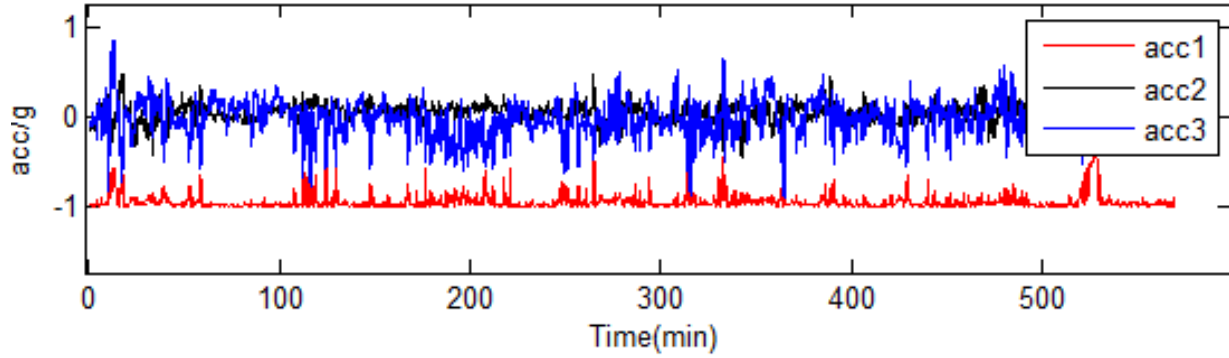


Figure 14: Pattern of the acceleration signals recorded on trunk. Here acceleration on the vertical axis is recorded on channel 1 and it is possible to notice that the absolute value of the mean of acc1 is near to g . In this case the mean is a very informative feature

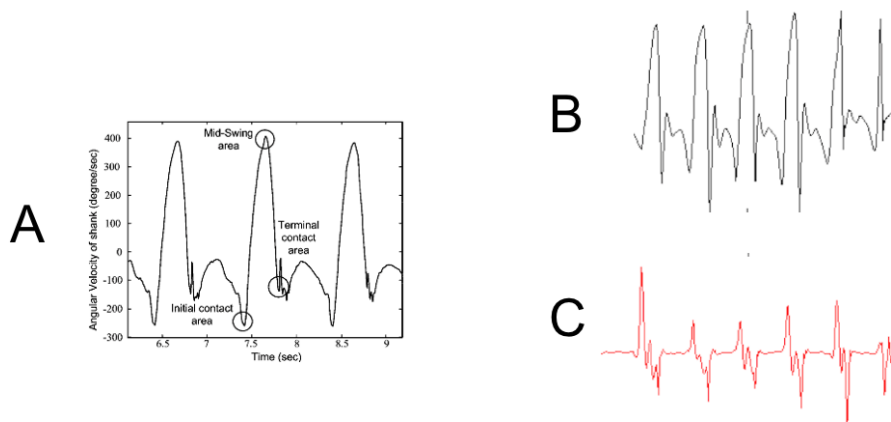


Figure 15: A: Pattern of the Pitch angular velocity recorded on shank. B is very similar to A differently from C. This visual operation made by human brain could be automatized with pattern matching function for calculating a “coefficient of similarity”.

learning methods, such as deep learning approach, more specifically by exploiting Convolutional Neural Networks (CNN). Over the last few years, CNN has been deeply utilized in computer vision and image processing for some specific tasks as image segmentation or object recognition. As suggested by the name, these computing systems perform multiple convolutions on the input signal and through a neural network structure can be used to do classification. This method can be considered particularly useful because no parameters have to be extracted/tuned and only the input signals (raw data) have to be provided to the algorithm. The application of CNN in movement analysis field has just been investigated in this paper [FJOR16] which considers similar objectives. Although it is possible to reach very high performances a huge computational effort is required to run these types of algorithm and it is not sure if standard computers available in usual work places are powerful enough to support these routines.

From a general point of view the application can also be updated adding new routines for the computation of new output parameters or by considering other sensor configurations.

7 Conclusion

In agreement with the goals of the project a new application for Movement Analysis of children patient affected by cerebral palsy has been developed and it is ready to be tested. All the specifics and facilities

provided have been agreed with the clinicians of CHUV in various meetings. The software is functioning at the moment on Matlab platform, and will be delivered in the next period to the clinicians for testing it on different terminals and considering a greater number of cases. Once a future developer will adapt the program according to the feedbacks received, it will be possible to perform a first deployment and standalone application can be released. To support the final users and the developers as much as possible two useful manuals have been drawn up and putted on the appendix of this paper.

References

- [ALKY11] L. Atallah, B. Lo, R. King, and G. Z. Yang. Sensor positioning for activity recognition using wearable accelerometers. *IEEE Transactions on Biomedical Circuits and Systems*, 5(4):320–329, Aug 2011.
- [CPIG⁺17] Lena Carcreff, A Paraschiv-Ionescu, Corinna Gerber, Geraldo Coulon, Kamiar Aminian, Christopher Newman, and Stéphanie Armand. Assessment of the spatiotemporal gait parameters of children with cerebral palsy in daily-life settings: comparison between wearable systems using different sensor location. 57, 07 2017.
- [FJOR16] Francisco Javier Francisco Javier Ordóñez and Daniel Roggen. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors*, 16(1), 2016.
- [PIBR⁺04] A. Paraschiv-Ionescu, E.E. Buchser, B. Rutschmann, B. Najafi, and K. Aminian. Ambulatory system for the quantitative and qualitative analysis of gait and posture in chronic pain patients treated with spinal cord stimulation. 2004.
- [SRJGV⁺08] Arash Salarian, Heike Russmann, François J G Vingerhoets, Pierre R Burkhard, and Kamiar Aminian. Ambulatory monitoring of physical activities in patients with parkinson’s disease. 54:2296–9, 01 2008.
- [SRV⁺04] A. Salarian, H. Russmann, F. J. G. Vingerhoets, C. Dehollain, Y. Blanc, P. R. Burkhard, and K. Aminian. Gait assessment in parkinson’s disease: toward an ambulatory system for long-term monitoring. *IEEE Transactions on Biomedical Engineering*, 51(8):1434–1443, Aug 2004.