TALLINN UNIVERSITY OF TECHNOLOGY
Faculty of Information Technology
Department of Software Science

Anna Krajuškina

AN ALTERNATIVE APPROACH FOR GAIT ANALYSIS OF PARKNISON'S DISEASE PATIENTS

Bachelor's thesis

Supervisor: Sven Nõmm
PhD

Tallinn 2017

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond
Tarkvarateaduse instituut

Anna Krajuškina

ALTERNATIIVNE MEETOD PARKINSONI PATSIENTIDE KÕNNAKU ANALÜÜSIMISEKS

Bakalaureusetöö

Juhendaja: Sven Nõmm
PhD

Tallinn 2017

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Anna Krajuškina

22.05.2017

Abstract

The primary goal of this study is to provide a low-cost, easy-to-use solution for capturing and analysis of human gait. The targeted application area is diagnostics and modelling of motion impairments common for neurological, diseases such as Parkinson's or Alzheimer.

In this study, a simple application for recording and computing gait parameters has been developed as a part of the present studies. To fulfil the requirement of low-cost solution, motion capture is performed by the Kinect motion sensor, whereas the analysing module is agnostic to data origin and may be used work with data obtained using other systems.

Neurological diseases may severely affect walking patterns of the patients. This makes it impossible to apply commercially available tools for the purposes of step detection. To overcome this problem, a unique algorithm for individual step detection was developed within the framework of present research.

To demonstrate applicability of the approach for neurology-related motion analysis two samples of captured gait data were analysed: one representing movements of Parkinson's disease (PD) patients, the other containing recordings of controls of matching age and sex to PD patients. Analysing and comparing kinetic parameters and walking patterns revealed a statistically significant difference for a number of parameters and joints.

This thesis is written in English and is 25 pages long, including 7 chapters, 13 figures and 5 tables.

Annotatsioon

Alternatiivne meetod Parkinsoni patsientide kõnnaku analüüsimiseks

Käesoleva töö põhieesmärgiks on arendada odav ja kasutajasõbralik süsteem, mis võimaldaks inimese kõnnaku salvestamist ja analüüsimist. Süsteemi rakendusvaldkonnaks on selliste neuroloogiliste haiguste nagu Parkinson ja Alzheimer diagnoosimine.

Neuroloogilised haigused võivad tunduvalt mõjutada patsientide kõnnaku iseärasusi. Seepärast on kommertsiliselt saadavad lahendused sammude tuvastamiseks kasutuskõlbmatud. Selle töö raames valmis algoritm, mis suudab antud tingimustes üksikud sammud tuvastada.

Kõnnaku salvestamiseks ja tunnussuuruste arvutamiseks arendati tarkvarasüsteem. Saavutamaks süsteemi madalat hinda teostatakse liigutuste tuvastamine Kinect sensori abil. Teiste lahendustega võrreldes võtab Kinecti häälestamine vähem aega ja see ei ole patsiendi jaoks häiriv. Samas on süsteemi analüüsiv tarkvara andmete allikast sõltumatu.

Iga tuvastatud sammu kohta eraldati hulk liigutust kirjeldavaid parameetreid. Selleks et näidata töös arendatud raamistiku võimekust asjakohaselt analüüsida inimese liikumist rakendati see kahele salvestuste valimile. Ühte katserühma kuulusid Parkinsoni patsiendid, teine oli aga sama vanuse ja sooga inimeste kontrollrühm.

Kinemaatiliste parameetrite analüüsi ja võrdlemise tulemused näitavad statistiliselt olulist erinevust mitme parameetri ja liigese korral.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 25 leheküljel, 7 peatükki, 13 joonist, 5 tabelit.

List of abbreviations and terms

PD	Parkinson’s disease
EMG	Electromyography
RGB-D	Red-Green-Blue-Depth
RGB	Red-Green-Blue
IR	Infrared
IMU	Inertial Measurement Units
WSN	Wearable Sensor Nodes
MM	Motion Mass
XAML	Extensible Application Markup Language
RAM	Random Access Memory
3D	Three-dimensional
2D	Two-dimensional
WPF	Windows Presentation Foundation
SDK	Software Development Kit
API	Application Programming Interface
CSV	Comma Separated Values
E	Euclidean distance
Tm	Trajectory Mass
Vm	Velocity Mass
Am	Acceleration Mass
Jm	Jerk Mass

Table of contents

List of figures

List of tables

1 Introduction

Numerous areas of medicine use human gait analysis for disease diagnostics. It assists decision-making, but requires presence of a human specialist. Modern laboratories are equipped with an expensive set of tools to make the process of gait evaluation more objective. Apart from being over-priced, this solution is time-consuming and obtrusive for patients.

The purpose of this work is to develop an assisting application that will facilitate motion capture and recording, and provide basic tools for the analysis of individual steps. The step recognition algorithm must take gait abnormalities of people with neurodegenerative illnesses into consideration. One of the major tasks of this study is to extract the kinetic and angle parameters from raw data. To verify the results, it is important to determine if the values of parameters describing gait during forward walking motion differ significantly between the groups of healthy individuals (controls) and individuals with Parkinson's disease (PD) patients. The current approach focuses on studying the parameters for each step detected.

Apart from the functional requirements described earlier, there are critical non-functional requirements that may influence the technology and the entire approach. One of the most important features of the system under development is being low cost and easy-to-use. The system also must be convenient for patients.

To meet the requirements, the Kinect sensor was chosen to implement the data gathering system. The data processing and analysis model is platform independent and can be used with data from any system as long as it meets the structure requirements. The analysis unit consumes pre-recorded data files of human movements, visualises them, extracts steps and calculates fundamental parameters that can are used to analyse the subject's health.

Secondary goal is to evaluate the applicability of Whittle's gait analysis [1] for Parkinson's disease patients using Kinect-based solution. A positive result would allow to investigate the influence of neurodegenerative diseases on gait compared to healthy people and extract features that differ for two groups mentioned.

Lots of research has been done lately on the subject of gait analysis. Majority of the results available are either devoted to accurate step detection in the case of regular walking patterns or concentrate their attention on the extraction of certain features without proper detection of the beginning and ending points of the step. The core novel component of the present work is a unique algorithm for individual step detection, suitable for capturing gait patterns even at later stages of the disease.

The present thesis is organized as follows: Section 2 gives an overview of existing solutions for gait analysis and discusses related work. Section 3 provides background information on software, hardware and algorithms used in this work. Section 4 describes the current solution. Section 5 presents the results achieved and discusses their validation. Conclusions are drawn in the last section.

This work was a part of the B37 program.

2 Related work

Traditional gait analysis is a manual process that relies mostly on therapists that visually observe patients. To reduce human errors, specialists are supported with measurement systems that evaluate the biomechanics of the gait [2].

A modern gait laboratory [2] has a motion capture system that consists of a special purpose computer, marker-based high-speed motion capture cameras, pressure sensitive plates and electromyography (EMG) system. To obtain three-dimensional kinematics of the whole human body, around 30 markers are required [2]. The accuracy of experiments in such laboratories depends on the precision of marker placement.

This setup is expensive and can be operated only by specially trained staff. Moreover, the experiments are invasive and time consuming due to marker placement.

The ponderous solution used in laboratories encourages research in the area of gait analysis and step recognition. Among a large diversity of solutions three most popular subsets can be differentiated: image processing, wearables and Kinect based solutions.

2.1 Vision-based solutions

In contrast to expensive cameras that are typically used for gait analysis, Saner et al. [3] suggested a more affordable alternative. The system consists of a simple two-dimensional web camera and is claimed to be easy-to-use. However, it is marker-based, which leaves the issue with correct marker placement unaddressed. In addition, the system is limited to analysing the lower part of the body, as it can detect only the hip and knee points. [4]

Reha@Home system proposed by Natarajan et al. [5] is a vision-based solution using an RGB-D camera to calculate angles and gait parameters. This system is portable, inexpensive and non-intrusive. Despite the advantages, it has several limitations. Reha@Home operates in the sagittal plane and consequently can recognise only 6 joints: head, hip, both knees and both ankles.

2.2 Wearables

Lately, Inertial Measurement Units (IMU)-based gait analysis systems have become increasingly popular due to being portable, inexpensive, and the potential for wireless data collection. However, the complexity of the sensor placement and issues with the consistency of the placement limits the usage of these settings to a laboratory. The effect of drift and noise in sensor measurements must be handled as well. Moreover, such systems are often intrusive, inconvenient for subjects of study, and experiments are time-consuming.

A wireless and wearable IMU system was proposed by Margiotta, Avitabile and Coviello [6]. Each of Wearable Sensor Nodes (WSN) used in the experiment contains a Bluetooth Low Energy Communication Module and a 6-axis IMU. The WSNs are dressed by means of stretchable on-body straps. The number and location of the sensors depend on the parameters that need to be measured.

Parisi et al. [7] introduced an IMU-based system that is time efficient and more convenient for subjects of study, yet is accurate. The main difference from other approaches was that only one IMU was used. The single IMU was placed on lower trunk.

Jarchi et al. [8] and Atallah et al. [9, 6] describe another approach that uses low-priced and non-intrusive ear-worn sensors. The advantage of this method is its potential for usage in everyday life. However, there are many limitations to its functionality. The most relevant of them is the system's inability to reliably detect right and left gaits and, consequently, the system is unable to calculate fundamental gait analysis parameters.

2.3 Kinect based solutions

The key benefits of using Kinect for motion analysis are its low cost, time efficiency, ability to operate in three-dimensional space and its non-invasive nature. The Kinect sensor provides a functionality to accurately track human skeleton without implementing complex algorithms.

Dao et al. [4] [10] developed a gait analysis solution for studying gait analysis based on Kinect technology. They used skeleton data to analyse of motion and depth frames to build a three-dimensional model of human body and visualise the movements. During the work steps are extracted, and the x-rotation and hip progression line calculated.

In 2013 Nomm et al. [10] proposed a set of parameters referred to as Motion Mass (MM) that could be used to measure the amount and smoothness of movement. MM of a moving point is a vector consisting of five variables: Euclidean distance, Trajectory Mass, Acceleration Mass, the ratio between Euclidean distance and Trajectory Mass, and the ratio between Euclidean distance and Acceleration Mass. In 2016 Nomm et al. [11] introduced one more parameter – Velocity Mass. Their work proved that those parameters are relevant to differentiating movements of PD patients and healthy individuals.

3 Background

3.1 Hardware

3.1.1 Kinect

Kinect is a motion sensing input device that contains a colour (RGB) camera, an infrared (IR) emitter and an IR depth sensor. The RGB camera allows capturing a colour image. The IR emitter emits infrared light beams and the depth sensor reads the reflected beams, which enables measuring the distance to an object. Kinect also contains microphones and a 3-axis accelerometer [12]. Kinect provides the functionality to track a human model - a so-called "skeleton".

The first generation of Kinects can track 20 points [13] that form a skeleton, whereas the second generation (Kinect 2.0) can track 25 points [14]. Apart from detecting a larger number of points, depth sensing improvements have influenced the stability and the quality of body tracking with Kinect 2.0 [15]. The points recognised by Kinect are called "joints". Joints are in fact structures that consist of the tracking state (Tracked, NotTracked or Inferred), a type (position of the joint within the skeleton) and a set of x, y and z coordinates that determine the joint position in 3D space [16]. Figure 1 illustrates the joint allocation in Kinect 1.0 (right) and Kinect 2.0 (left).

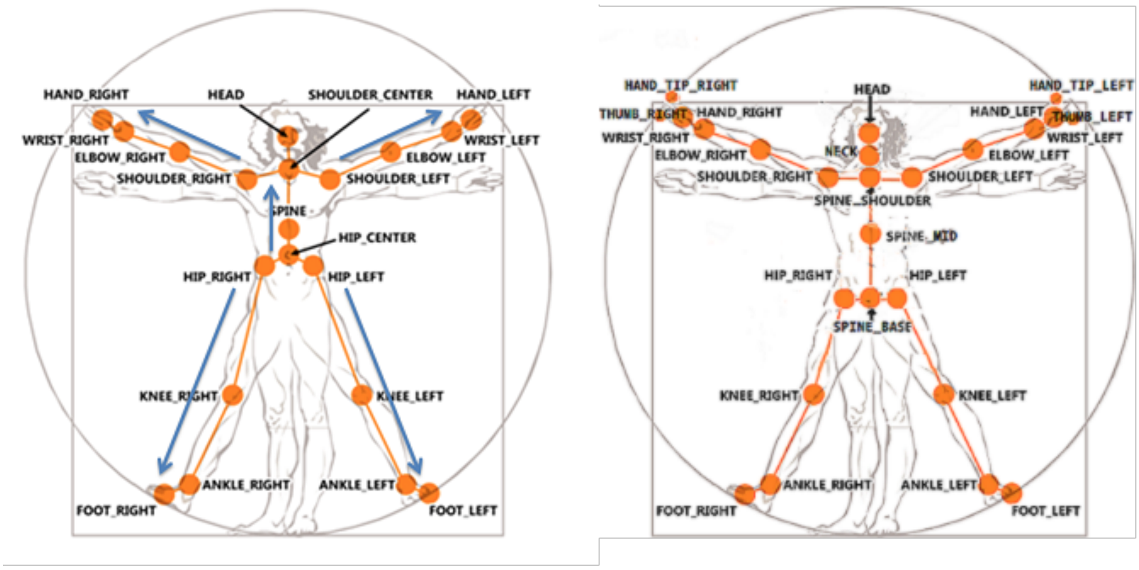

Figure 1. The human model captured by Kinect.

On the left: first generation captures 20 joints. On the right: Kinect 2.0 can track 25 joints. [13, 14]

3.2 Mathematical methods and algorithms

3.2.1 Simple Moving Average

Moving Average algorithms, such as the simple moving algorithm, are used to smooth noisy signal data. The simple moving algorithm has a number of advantages, being effective and easy to implement. It takes an array of noisy signal data as an input. The most important parameter that influences the outcome array is the frame length N: the larger the frame length is, the smoother the resultant signal will be (example shown in **Error! Reference source not found.** and **Error! Reference source not found.**). After the frame length is chosen, an average is calculated for the first N values in the array and saved into the smoothed values array. The frame is then moved one value further and the average is calculated again. These operations are repeated until the end of the noisy array. The output is a smoothed array with a length of the initial array – N + 1.

3.2.2 Welch's test for unequal variances (Welch's t-test)

Welch's test is one of the statistical hypothesis tests. "Welch's t-test, unlike Student's t-test, does not have the assumption of equal variance (however, both tests have the assumption of normality)" [17]. Welch's test tends to give more reliable results in comparison to more frequently used Student's test [17]. The formula for t-statistics calculation is shown below.

$$t = \frac{\overline{X_1} - \overline{X_2}}{\sqrt{\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2}}}$$

where N_1 and N_2 are sizes of samples, s_1 and s_2 are sample variances and $\overline{X_1}$ and $\overline{X_2}$ are sample means (expected values).

3.3 Software

3.3.1 Windows Presentation Foundation

Kinect for Windows Software Development Kit (SDK) 2.0 amongst other provides a .NET Application Programming Interface (API) that allows developing Windows Presentation Foundation (WPF) applications. WPF is a unified programming model for building Windows desktop applications [18]. It is a subset of .NET Framework. “WPF provides support for an Extensible Application Markup Language (XAML), vector graphics, 2D and 3D graphics, and data binding” [19].

4 Methodology

4.1 Hardware requirements

The application developed in the current work runs with the first generation of the Kinect sensor and with Kinect 2.0. The system requirements [20, 21] that a computer must meet are brought in Table 1.

Table 1. Kinect hardware requirements

Kinect 1.5, 1.6, 1.7, 1.8	**Kinect 2.0**
32-bit (x86) or 64-bit (x64) processors	64-bit (x64) processor
Dual-core, 2.66-GHz or faster processor	Physical dual-core 3.1 GHz (2 logical cores per physical) or faster processor
USB 2.0 bus dedicated to the Kinect	USB 3.0 controller dedicated to the Kinect for Windows v2 sensor or the Kinect Adapter for Windows for use with the Kinect for Xbox One sensor
2 GB of RAM	4 GB of RAM
Graphics card that supports DirectX 9.0c	Graphics card that supports DirectX 11
Windows 7 or 8, Windows Embedded Standard 7 or 8	Windows 8 or 8.1, Windows Embedded 8, or Windows 10

4.2 Mathematical methods and algorithms

4.2.1 Step detection

The distance between ankles d is calculated using the following formula.

$$d = z_2 - z_1$$

here z_2 and z_1 denote the z-axis coordinates of human ankles: left and right correspondingly.

If one or both ankles are untracked, distance between them is set to be 0. Based on this formula, right and left steps can be differentiated – negative distance means that the current step has been done with the left leg; positive distance implies it is a right step (Figure 5). In this work steps are not divided into left and right, and absolute values of distances are used instead.

The distance between ankles changes over time and forms a sinus-like chart (Figure 5), which enables determination of individual steps and calculation of their length. Moving Average algorithm is used to remove the noise. Steps are recognised by finding local minima on the chart. A value is recognized as a local minima in case of the following conditions met: the value falls under a certain threshold – 5 cm in this work –, and the two previous and two following values are greater the current value.

If the step length is lower than a specified threshold – one seventh of the maximum step length –, it is considered as a part of the previous step. It is necessary in case when a person always starts his steps with one leg, but moves it a bit further than the other leg is. An exemplary visual representation of detected steps is shown in Figure 6.

4.2.2 Angles

Angles are calculated as angles between two vectors in three-dimensional space. The full set of angles measured by the system is shown in Figure 2. Each vector is fully described with coordinates of two points, using the following formula.

$$(x, y, z) = (x_2 - x_1, y_2 - y_1, z_2 - z_1)$$

where (x_1, y_1, z_1) and (x_2, y_2, z_2) are the coordinates of the first and second points, respectively.

The angle calculation is performed by a built-in C# function that is shown below

```
Vector3D.AngleBetween(Vector3D v1, Vector3D v2)
```

The application calculates angles for each frame processed, groups the angles by steps and calculates the following values regarding the angle magnitude observed in one step period: the maximum, the minimum and the difference between them. The magnitude of an angle at the start and the end of a step is calculated as well.

Figure 2. Schematic representation of a human and the angles calculated by the application.

Depicted are the following angles: 1, 2 - between the ankle-knee and knee-hip lines. 3 – between the imaginary right knee-spine base and left knee-spine base lines. 4, 5 - between the spine base-hip and knee-hip lines. 6 - between the head-spine shoulder and spine mid-spine shoulder lines. 7 - between the spine shoulder-spine mid and spine base-spine mid lines. 8 – the between head-spine mid and spine mid-spine base.

4.2.3 Motion Mass parameters

The set of human body joints is represented as a set J, so that each joint j_i represents one joint. Refer to Figure 1. The set of joints is described as follows.

$$J = \{J_1, \dots, J_n\}$$

where n is the number of joints.

A set of Motion Mass (MM) parameters is defined for each joint. These are Euclidean distance (E), Trajectory Mass (Tm), Velocity Mass (Vm), Acceleration Mass (Am), Jerk Mass (Jm) and time (t). Motion Mass for a joint is defined as a set of average MM parameters across the steps recognised.

$$M_j = \{E, Tm, Vm, Am, Jm, t\}$$

Average MM parameters across steps are defined as follows.

$$E = \frac{\sum_{i=1}^{n} E_i}{n}$$

$$T_m = \frac{\sum_{i=1}^{n} T_{m_i}}{n}$$

$$V_m = \frac{\sum_{i=1}^{n} V_{m_i}}{n}$$

$$A_m = \frac{\sum_{i=1}^{n} A_{m_i}}{n}$$

$$J_m = \frac{\sum_{i=1}^{n} J_{m_i}}{n}$$

$$t = \frac{\sum_{i=1}^{n} t_i}{n}$$

where n is the number of steps.

The parameters for each step are calculated as shown below and in Figure 3.

$$E = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$$

where (x_1, y_1, z_1) and (x_2, y_2, z_2) are the coordinates of a joint in the beginning and at the end of a step, respectively.

$$T_{m_i} = \sum_{i=1}^{n} E_i$$

$$V_{m_i} = \sum_{i=1}^{n} \frac{E_i}{t_i}$$

$$A_{m_i} = \sum_{i=1}^{n} \frac{V_{m_i}}{t_i}$$

$$J_{m_i} = \sum_{i=1}^{n} \frac{A_{m_i}}{t_i}$$

$$t_i = t_{end} - t_{start}$$

where n is the number of intermediate points captured by the Kinect sensor.

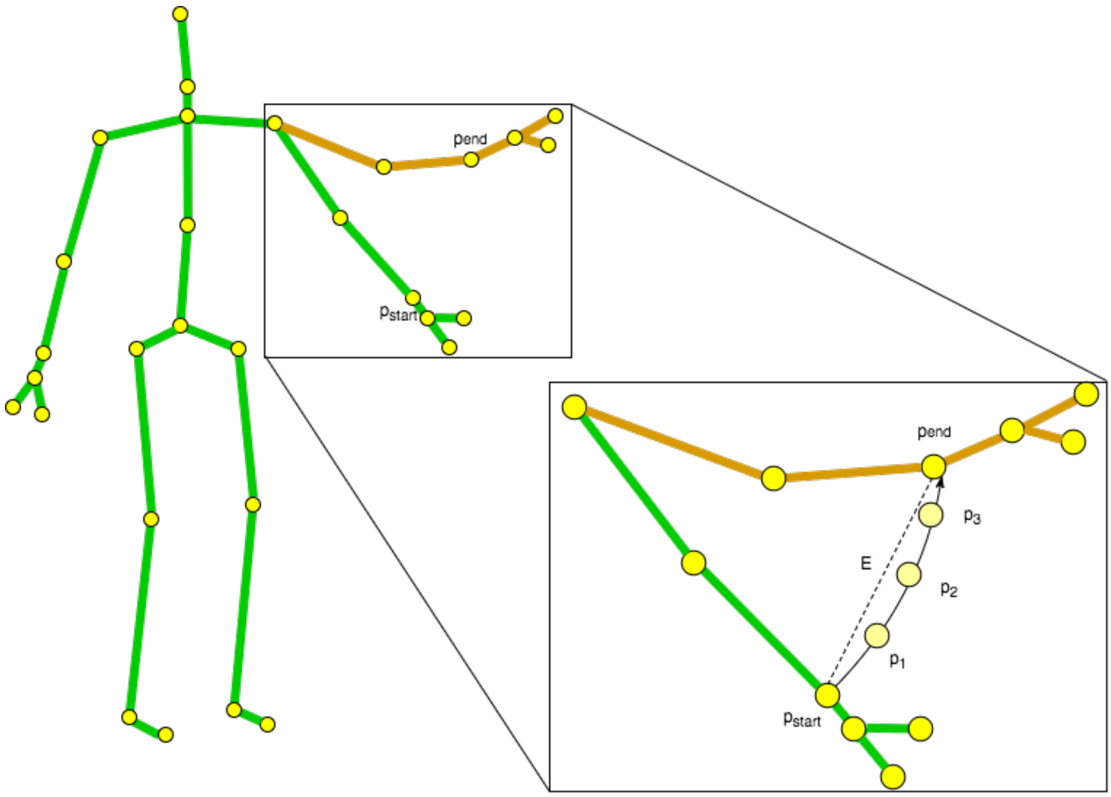

Figure 3. Schematic representation of a Kinect skeleton and method for parameter determination.

Yellow circles depict joints (25). p_{start} denotes the position of a joint at the start of a step. p_{end} indicates the final position of the joint (marked in orange). On the right: in the close-up on the moving part of the skeleton. p_i show the intermediate positions of the joint captured by Kinect. Parameters except E are determined at each intermediate point and summed. E is the distance between p_{start} and p_{end}.

4.2.4 Statistical t-test

Statistical data analysis was implemented in Python using the SciPy library and the function that calculates the t-test for two independent sample scores (scipy.stats.ttest_ind). This method takes four parameters as an input: two arrays of independent samples, the axis over which to operate on arrays, and a Boolean value showing whether the two variances are equal. In case when the variances are non-equal, the method performs Welch's test, and Student's test otherwise. The method returns the calculated the t-statistics value and the two-tailed p-value [22].

4.3 Software and implementation

4.3.1 Visualisation

The application has three views, showing a person in three planes: sagittal (x, z coordinates), traverse (y, z coordinates) and frontal (x and y coordinates). Observation of movements in the sagittal plane gives the most valuable information about a person's gait. A model of a human reconstructed from a recording in the sagittal plane view is shown in Figure 4. Other views can be found in Appendix 1 – Frontal and traverse planes.

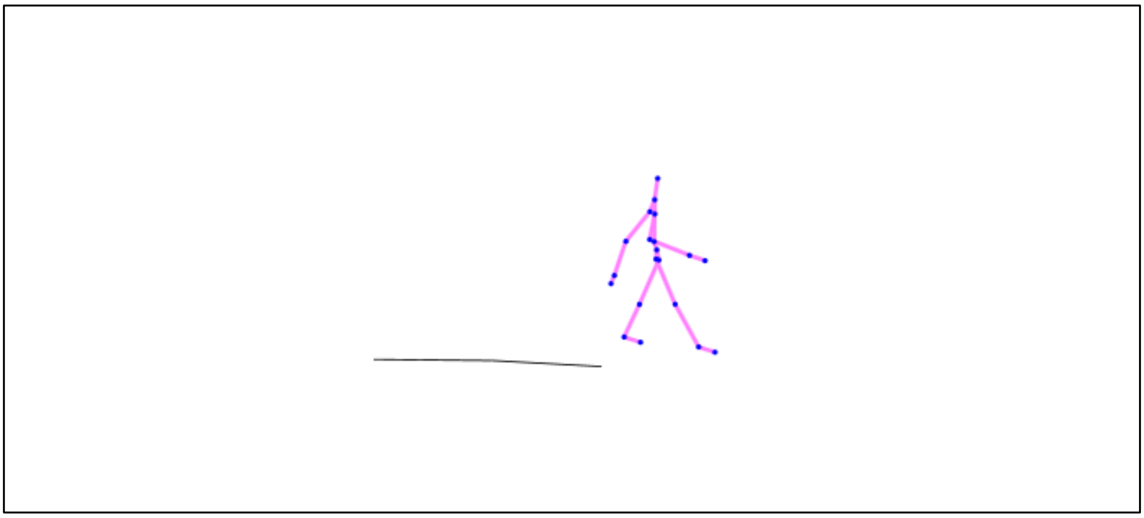

Figure 4. Sagittal plane (image colours inverted).

An example of human body reconstructed based on pre-recorded Kinect data files.

The program recognises steps and visualises the start of each step by changing the colour of a human model - skeleton. After the walking part of the recording is finished, the application plots the change in distances between ankles over time in two charts. The first chart shows the raw data collected during the replay (example in Figure 5), the other chart represents the absolute values of smoothed distances (example in Figure 6). Vertical lines denote points where one step ends and another one begins.

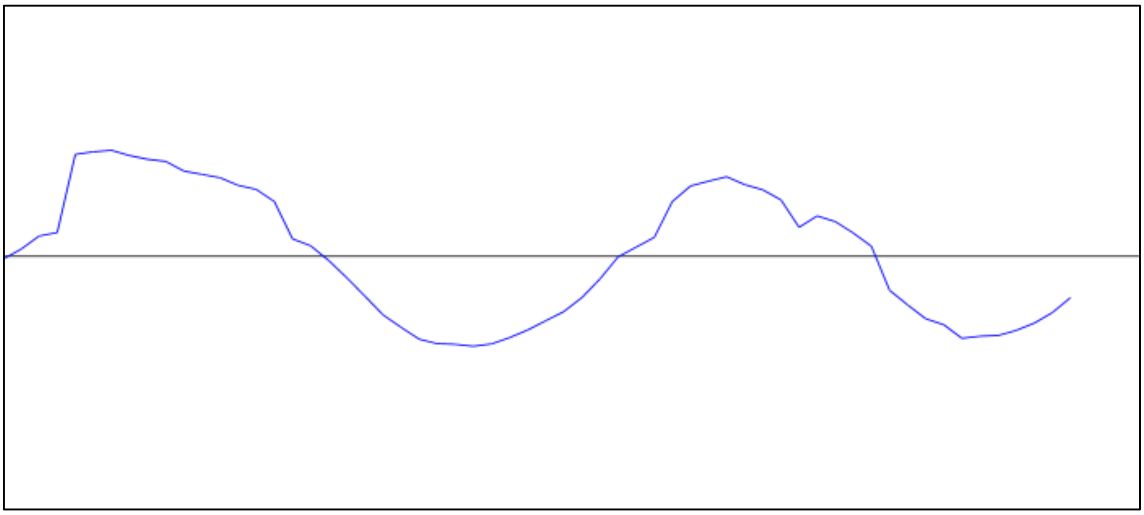

Figure 5. Visualisation of distances between ankles based on raw data (image colours inverted).

The graph produced by the application shows the change of distance (vertical axis) in time (horizontal axis) and has a sine-like shape. The values above the horizontal axis are positive (right step), whereas the ones below are negative (left step).

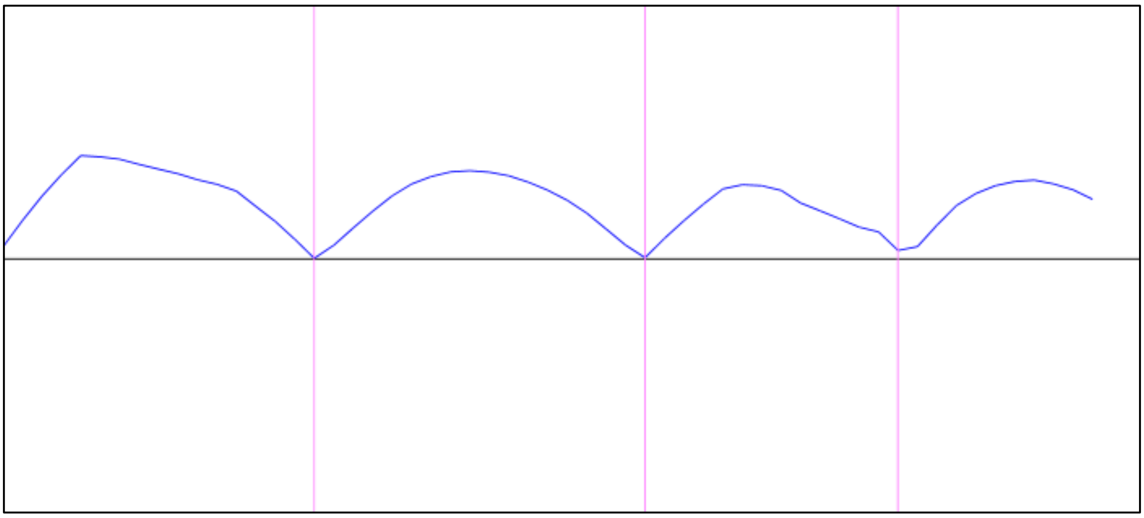

Figure 6. Visualisation of processed data (image colours inverted).

The raw data was smoothed using Simple Moving Algorithm and the absolute value was taken to aid step detection. The vertical lines denote the ankles' superposition in the sagittal plane.

4.3.2 Input data

The application consumes a file with pre-recorded data in a Comma Separated Values (CSV) format (refer to Figure 7). Each row in the file contains a frame number, a timestamp and a set of x, y and z coordinates for every point of human body Kinect can recognize. Sample rate of Kinect is approximately 25 frames per second, so the file contains approximately 25 rows per second of recordings. The application was built having Kinect data in mind, but can be used for processing any CSV file that meets the structure requirements. The program works with data from Kinect 2.0 as well as with data from previous versions of the sensor.

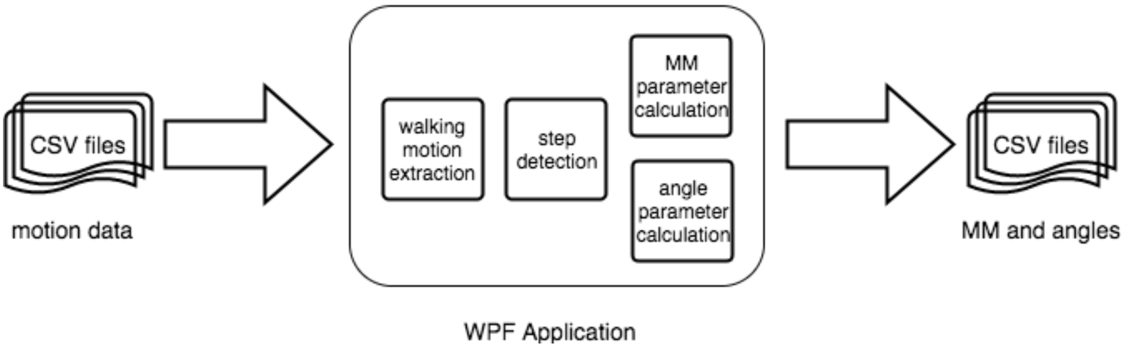

Figure 7. Scheme of parameter extraction application

4.3.3 Output

For each file processed the application produces a set of CSV files (Figure 7). It saves the parameters for each step to one file and the angles to another, and the calculated average values into a third (source code in Appendix 2 – Source code). An example of average parameters can be found in Table 2 and Table 3.

Table 2. Example of average angle parameters

Angle Type	Max	Min	Delta	Start	Stop
RIGHT_KNEE	177.324	154.164	23.160	154.698	171.956
LEFT_KNEE	174.757	145.745	29.012	173.631	153.756
RIGHT_HIP	173.249	162.884	10.365	168.938	173.122
LEFT_HIP	171.127	162.919	8.207	167.013	164.386
KNEE_HIP_KNEE	30.346	14.637	15.709	20.754	19.473
HEAD_SPINESHOULDER_SPINEMID	175.135	165.711	9.423	166.751	168.983
SPINESHOULDER_SPINEMID_SPINEBASE	176.826	161.117	15.709	176.258	161.117
HEAD_SPINESHOULDER_SPINEBASE	174.384	164.591	9.792	166.326	165.960

Table 3. Example of average parameters

Joint Type	E	Tm	Vm	Am	Jm	E/Tm	E/Am	t
Head	0.516	0.524	5.363	68.255	1687.986	0.985	0.008	0.597
SpineShoulder	0.520	0.533	6.616	104.461	2798.674	0.974	0.005	0.597
SpineMid	0.525	0.531	2.919	48.646	1459.660	0.989	0.012	0.597
SpineBase	0.541	0.547	2.715	50.840	1470.363	0.989	0.011	0.597
ShoulderLeft	0.521	0.531	5.194	85.666	2071.121	0.982	0.007	0.597
ElbowLeft	0.506	0.519	5.478	85.747	1811.179	0.975	0.006	0.597
WristLeft	0.503	0.547	5.085	106.466	2749.819	0.920	0.005	0.597
HandLeft	0.503	0.570	3.462	91.463	2145.103	0.874	0.005	0.597
ShoulderRight	0.527	0.535	4.113	65.814	1430.123	0.986	0.009	0.597
ElbowRight	0.564	0.576	4.147	74.552	1926.588	0.980	0.008	0.597
WristRight	0.577	0.592	5.339	112.093	2794.377	0.975	0.006	0.597
HandRight	0.586	0.615	5.979	115.338	3062.093	0.955	0.005	0.597
HipLeft	0.547	0.554	3.958	69.339	1731.270	0.987	0.008	0.597
KneeLeft	0.572	0.636	9.597	206.903	6689.562	0.891	0.003	0.597
AnkleLeft	0.461	0.675	11.389	293.336	7837.832	0.689	0.002	0.597
FootLeft	0.456	0.832	17.513	505.504	13958.585	0.583	0.001	0.597
HipRight	0.545	0.551	3.891	64.354	1627.596	0.989	0.009	0.597
KneeRight	0.544	0.592	7.973	153.943	3685.416	0.919	0.004	0.597
AnkleRight	0.679	0.731	4.981	107.870	2436.502	0.918	0.006	0.597
FootRight	0.676	0.849	10.938	262.128	7892.950	0.798	0.003	0.597

4.3.4 Python scripts

To validate the new application, a set of previous Kinect recordings was used. The recordings were in MATLAB file format (.mat extension), and a Python script was produced to convert those files into CSV format (Figure 8 and Appendix 2 – Source code). The script recursively iterates over all files with a .mat extension. For each file, the script iterates over data in the file and extracts coordinates, as well as the data concerning intervals of the experiments – stand up, walking forward, sit down – into two separate CSV files.

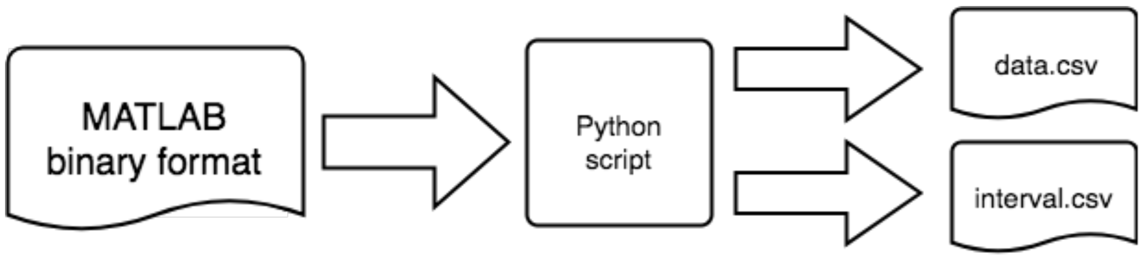

Figure 8. Schematic representation of file conversion

As it was described in Section 4.3.3, the analysis module produces one file with average parameters and another with average angles. Several recordings are usually produced per person, so there was a need for a script that would calculate average values per person across those files. To simplify the process, another Python script was developed to achieve this in a single step (Appendix 2 – Source code). The structure of output files is similar to examples brought in Table 2 and Table 3.

Statistical hypothesis test was also implemented as a script (Appendix 2 – Source code), performing the t-test for each joint-parameter and joint-angle pair. The script runs on previously described script's output files. Results are discussed in Section 5.3.1.

5 Results and method validation

5.1 Motion Mass parameters

In this study, the set of parameters proposed by Nomm et al. was complimented with one more parameter – Jerk Mass (Jm). Furthermore, parameters are calculated not for an entire walk, but as an average across the steps.

The application calculates 8 motion mass parameters for each step: time of every step (t), Euclidean distance (E) – distance between the starting and the final positions of a joint; trajectory length (Tm), velocities (Vm), accelerations (Am) and jerks (Jm), all calculated as a sum of the corresponding parameter taken in every point. The ratio between Euclidean distance and Trajectory Mass, and the ratio between Euclidean distance and Acceleration Mass are determined as well.

5.2 Angles

Angles are calculated as angles between so called bones [23] in Kinect skeleton in 3D space. The application calculates 8 angles (Figure 2):

- knee, spine base, knee
- hip, knee, ankle (for both legs)
- head, spine base, knee (for both legs)
- head, spine shoulder, spine mid
- spine shoulder, spine mid, spine base
- head, spine shoulder, spine base

There are several angle parameters calculated for each step and each angle group. These are the maximum and minimum angles during a step, the difference between the maximum and minimum, angle at the start and at the end of a step.

5.3 Method validation

Previous studies using other methods have proven a significant difference in kinematic parameters between PD patients and healthy individuals [11]. Consequently, the ability to distinguish between them would demonstrate the validity of our approach. The system was tested on a dataset of Kinect recordings from 23 PD patients and 19 controls (Sven Nomm, unpublished data). The subjects of study were asked to stand up, make a few steps, turn around, return to the chair, turn around and sit down again [11]. The data files contain three iterations of the experiment, and the start and the end of each phase is marked. The results confirmed the method's applicability in a clinical setting.

5.3.1 Statistical hypothesis test

The goal of the test was to check whether the MM parameters of Parkinson disease patients vary significantly different from those of the control group. Therefore, the null hypothesis (H0) was: "The parameter P for joint J is the same for Parkinson disease patients and the control group". The alternative hypothesis (H1) stated, that the parameter P for joint J was different between the two groups studied. A difference in angle parameters was detected for 5 angle-parameter pairs (significance level 0.05). At the level of significance 0.01, the results indicate that Euclidean distance is distinct for all joints, while Trajectory Mass differs for the majority. Therefore, although a larger dataset should be analysed for higher confidence, preliminary testing has confirmed the validity of the method implemented. The complete results of the tests are brought in Table 4 and Table 5.

Table 4. Hypothesis chosen (H0 or H1) with level of significance 0.05

	Delta	**Max**	**Min**	**Start**	**Stop**
HEAD_SPINESHOULDER_SPINEBASE	0	0	0	0	0
HEAD_SPINESHOULDER_SPINEMID	0	0	0	0	0
KNEE_HIP_KNEE	1	1	0	0	0
LEFT_HIP	0	0	0	0	0
LEFT_KNEE	0	0	0	0	1
RIGHT_HIP	1	0	0	0	0
RIGHT_KNEE	0	0	0	0	0
SPINESHOULDER_SPINEMID_SPINEBASE	1	0	0	0	0

Table 5. Hypothesis chosen (H0 or H1) with level of significance 0.01

	Am	**E**	**E/Am**	**E/Tm**	**Jm**	**Tm**	**Vm**	**t**
AnkleLeft	0	1	0	0	0	0	0	0
AnkleRight	0	1	1	0	0	0	0	0
ElbowLeft	0	1	0	0	0	1	0	0
ElbowRight	0	1	0	0	0	1	0	0
FootLeft	0	1	0	0	0	0	0	0
FootRight	0	1	0	0	0	0	0	0
HandLeft	0	1	0	0	0	0	0	0
HandRight	0	1	0	0	0	0	0	0
Head	0	1	0	0	0	1	0	0
HipLeft	0	1	0	0	0	1	0	0
HipRight	0	1	0	0	0	1	0	0
KneeLeft	0	1	0	0	0	1	0	0
KneeRight	0	1	0	0	0	1	0	0
ShoulderLeft	0	1	0	0	0	1	0	0
ShoulderRight	0	1	0	0	0	1	0	0
SpineBase	0	1	0	0	0	1	0	0
SpineMid	0	1	0	0	0	1	0	0
SpineShoulder	0	1	0	0	0	1	0	0
WristLeft	0	1	0	0	0	0	0	0
WristRight	0	1	0	0	0	1	0	0

6 Discussion

6.1 Whittle's step phases

As a result of the present research, Whittle's step phases could not be reliably recognised based on Kinect data only. According to Whittle, the start and the end of each phase is defined by positions of toes and heels relative to the floor [1]. This approach is only applicable for normal gait analysis or a gait with small deviations. The system was not precise enough to reliably detect the position of heels on the floor due to the limited accuracy of Kinect, which is "within a centimetre range" [24]. In addition, the distance to Kinect sensor has a major impact on the precision of detection. During the research, the foot tracking was also noticed to be less stable than tracking of other joints. Heels are not detected by the Kinect system, so their position had to be determined theoretically (refer to Figure 10). Hence, tracking could only be done assuming that a person used the full area of their foot for walking. In the end, all the calculations were too imprecise and the measurement error strongly affected the result, making determination of step phases impossible. In Figure 9 an example of an application detecting Whittle's step phases based on Kinect files is brought.

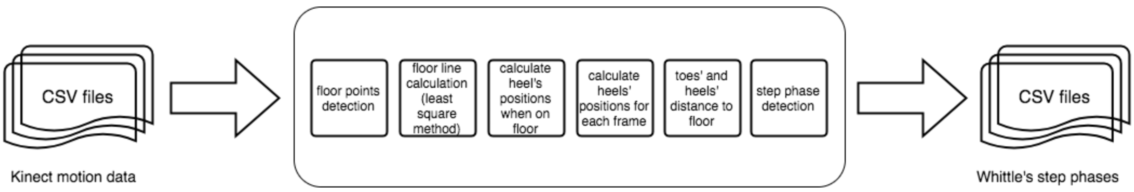

Figure 9. Scheme of Whittle's step phases extracting application.

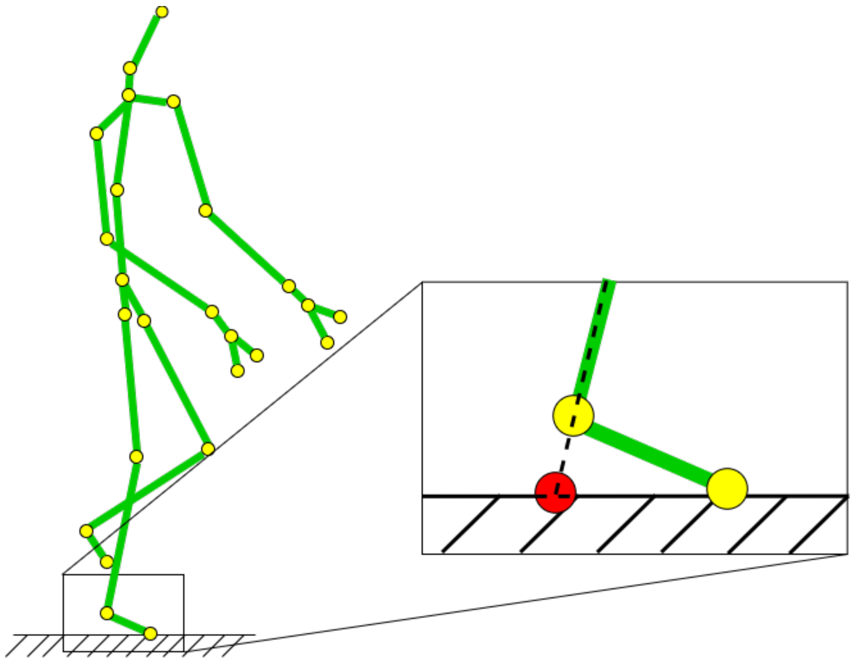

Figure 10. Schematic representation of the floor points and heels' positions determination.

The red dot depicts the heel as an intersection point of the floor line and the knee-ankle line. An assumption is made that the heel is on the floor if the other foot is in the air.

6.2 Future work

There are numerous ways to improve the reliability of method validation by statistical hypothesis testing. Statistical tests give robust result if applied to a larger sample, while the number of recordings available was limited. Consequently, to produce more trustworthy conclusions, it would be necessary to repeat the analysis on a larger dataset.

The developed system enables processing of files recorded with Kinect 2.0. As it was discussed previously, the second generation of Kinects has an enhanced tracking algorithm and detects more joints. The second Kinect was claimed to have more anatomically correct skeleton [15] meaning that the detected joints reflect the moving patterns of actual human joints better. To retrieve more accurate and reliable results, recordings with Kinect 2.0 are still to be done and to be evaluated.

Another enhancement would be to make the graphical interface more user-friendly. Although, the application was implemented having end users in mind, some usability issues might occur. Testing sessions and overviews by potential users should be organised to gather feedback and make the system more convenient.

This application was designed in a way that it functionality could be easily extended. An example of a modification that can be easily implemented is differentiation between left and right steps. As a result, parameters' averages could be calculated separately for those to evaluate the distribution between sides. Another enhancement could be to implement real-time computation with a connected Kinect sensor to make the whole process more time-efficient and easy.

7 Summary

The present work is devoted to the problem of gait capture and analysis. The major task was to differentiate steps for people with irregular gait, such as PD patients, and perform an analysis according to the steps detected. The secondary goal was to test the applicability of Whittle's gait analysis using the Kinect sensor to PD diagnosis.

The key benefits of the approach described are its low-cost compared to the current state-of-the-art systems, time-efficiency and non-obtrusiveness, which makes it suitable not only for research purposes, but also in a clinical setup. The Kinect skeleton tracking is precise and facilitates the human motion analysis in three-dimensional space. The system is easy to operate and it does not require any special skills.

The analysis module is self-sufficient and can be used to process data from different sources provided it fulfils the structure requirements. The main feature of this module is the ability to detect single steps even for patients with neurodegenerative diseases. It also calculates a set of Motion Mass and angle parameters that can be used to evaluate the person's health and aid in tracking the disease progression.

Statistical tests confirmed the existence of significant differences in PD patients' and controls' parameters, determined by the method developed in this work. The difference was discovered for a number of joint-parameter pairs, indicating the robustness of the approach. However, more studies should be conducted to improve the reliability of the outcome.

During the work, it was proven that with given setup Whittle's step phases cannot be reliably detected due to Kinect's limited accuracy and the absence of heel tracking. In addition, this method is not applicable on later stages of Parkinson's disease.

References

[1] M. W. Whittle, J. Richards and D. Levine, Whittle's Gait Analysis, 5th Edition ed., 2012, pp. 32-34.

[2] M. Alwan and R. Felder, Eldercare Technology for Clinical Practitioners, Humana Press, 2007, pp. 37-38.

[3] R. J. Saner, E. P. Washabaugh and C. Krishnan, “Reliable sagittal plane kinematic gait assessments are feasible using low-cost webcam technology,” *Gait & Posture,* vol. 56, pp. 19-23, July 2017.

[4] N.-L. Dao, Y. Zhang, J. Zheng and J. Cai, “Kinect-based non-intrusive human gait analysis and visualization,” in *2015 IEEE 17th International Workshop on Multimedia Signal Processing (MMSP)*, Xiamen, 2015.

[5] S. K. Natarajan, X. Wang, M. Sprangel and A. Graser, “Reha@Home - a vision based markerless gait analysis system for rehabilitation at home,” in *2017 13th IASTED International Conference on Biomedical Engineering (BioMed)*, Innsbruck, 2017.

[6] N. Margiotta, G. Avitabile and G. Coviello, “A Wearable Wireless System for Gait Analysis for Early Diagnosis of Alzheimer and Parkinson Disease,” in *2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA)*, Ras Al Khaimah, 2016.

[7] F. Parisi, G. Ferrari, A. Baricich, M. D’Innocenzo, C. Cisari and A. Mauro, “Accurate Gait Analysis in Post-Stroke Patients using a Single Inertial Measurement Unit,” in *2016 IEEE 13th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, San Francisco, 2016.

[8] D. Jarchi, B. Lo, C. Wong, E. Ieong, D. Nathwani and G.-Z. Yang, “Gait Analysis From a Single Ear-Worn Sensor: Reliability and Clinical Evaluation for Orthopaedic Patients,” in *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2016.

[9] L. Atallah, B. Lo, G.-Z. Yang and O. Aziz, “Detecting Walking Gait Impairment with an Ear-worn Sensor,” in *2009 Sixth International Workshop on Wearable and Implantable Body Sensor Networks*, Berkeley, 2009.

[10] S. Nomm, A. Toomela and J. Borushko, “Alternative Approach to Model Changes of Human Motor Functions,” in *2013 European Modelling Symposium*, Manchester, 2013.

[11] S. Nomm, A. Toomela, M. Vaske, D. Uvarov and P. Taba, “An Alternative Approach to Distinguish Movements of Parkinson Disease Patients,” *IFAC-PapersOnLine,* vol. 49, no. 19, pp. 272-276, 2016.

[12] “Kinect for Windows Sensor Components and Specifications,” [Online]. Available: https://msdn.microsoft.com/en-us/library/jj131033.aspx. [Accessed 08 03 2017].

[13] “Tracking Users with Kinect Skeletal Tracking,” [Online]. Available: https://msdn.microsoft.com/en-gb/en-%20us/library/jj131025.aspx. [Accessed 08 03 2017].

[14] “JointType Enumeration,” [Online]. Available: https://msdn.microsoft.com/en-gb/en-%20us/library/microsoft.kinect.jointtype.aspx. [Accessed 08 03 2017].

[15] “Kinect harware,” [Online]. Available: https://developer.microsoft.com/en-us/windows/kinect/hardware. [Accessed 21 05 2017].

[16] “Joint Properties,” [Online]. Available: https://msdn.microsoft.com/en-us/library/microsoft.kinect.joint_properties.aspx. [Accessed 14 05 2017].

[17] “Welch’s Test for Unequal Variances,” [Online]. Available: http://www.statisticshowto.com/welchs-test-for-unequal-variances/. [Accessed 21 05 2017].

[18] “Getting Started with WPF,” [Online]. Available: https://msdn.microsoft.com/en-us/library/mt149845.aspx. [Accessed 25 05 2017].

[19] “Windows Presentation Foundation (WPF),” [Online]. Available: https://www.techopedia.com/definition/16623/windows-presentation-foundation-wpf. [Accessed 25 04 2017].

[20] “System Requirements,” [Online]. Available: https://msdn.microsoft.com/en-us/library/hh855359.aspx. [Accessed 17 05 2017].

[21] “System Requirements,” [Online]. Available: https://msdn.microsoft.com/en-us/library/dn782036.aspx. [Accessed 15 05 2017].

[22] “scipy.stats.ttest_ind SciPy v0.15.1,” [Online]. Available: https://docs.scipy.org/doc/scipy-0.15.1/reference/generated/scipy.stats.ttest_ind.html. [Accessed 27 05 2017].

[23] “Joint Orientation,” [Online]. Available: https://msdn.microsoft.com/en-us/library/hh973073.aspx. [Accessed 21 05 2017].

[24] “Skeletal Joint Smoothing White Paper,” [Online]. Available: https://msdn.microsoft.com/en-us/library/jj131429.aspx. [Accessed 27 04 2017].

Appendix 1 – Frontal and traverse planes

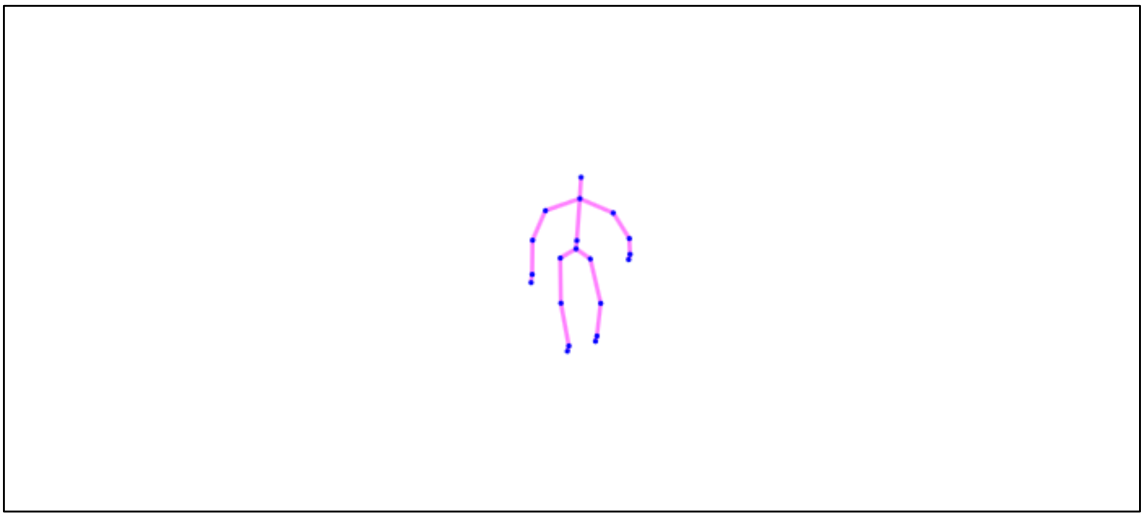

Figure 11. Frontal plane (image colours inverted)

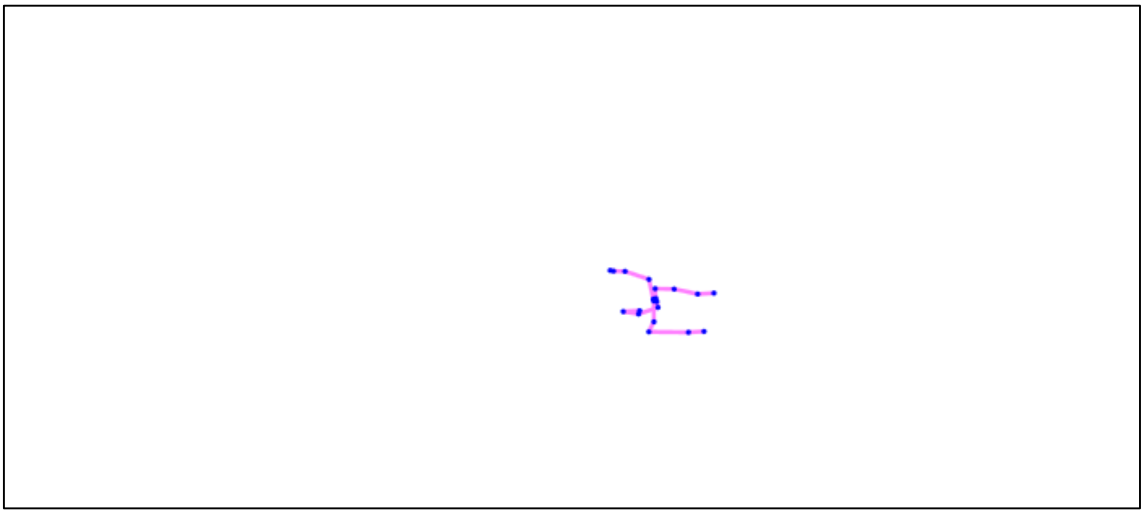

Figure 12. Traverse plane (image colours inverted)

Appendix 2 – Source code

WPF application’s (analysis module) source code:

- https://bitbucket.org/anna_kraj/gait-analysis

Python scripts’ source code:

- https://bitbucket.org/anna_kraj/python-scripts