**Motion capture data processing and analysis**

Table of Contents

[List of Abbreviations - 0 -](#_Toc521354400)

[1 Introduction - 0 -](#_Toc521354401)

[2 Theory. Analysis. Aim. - 3 -](#_Toc521354402)

[2.1 Motion Capture - 3 -](#_Toc521354403)

[2.1.1 History of motion capture - 3 -](#_Toc521354404)

[2.1.2 Optical-based motion capture system VICON - 4 -](#_Toc521354405)

[2.2 Sign language analysis - 6 -](#_Toc521354406)

[2.2.1 Hamburg Notation System - 7 -](#_Toc521354407)

[2.3 Optical-based motion capture for sign language synthesis - 13 -](#_Toc521354408)

[2.4 Segmentation of SL - 13 -](#_Toc521354409)

[2.4.1 Method for segmentation - 14 -](#_Toc521354410)

[2.4.2 Conclusion - 14 -](#_Toc521354411)

[2.5 Conclusion - 15 -](#_Toc521354412)

[3 System Architecture - 16 -](#_Toc521354413)

[3.1 Python - 16 -](#_Toc521354414)

[3.1.1 NumPy - 17 -](#_Toc521354415)

[3.1.2 SciPy - 17 -](#_Toc521354416)

[3.1.3 Matplotlib - 17 -](#_Toc521354417)

[3.1.4 C3D - 18 -](#_Toc521354418)

[3.2 Sublime Text code editor - 18 -](#_Toc521354419)

[3.3 Autodesk MotionBuilder - 18 -](#_Toc521354420)

[3.4 C3D file format - 18 -](#_Toc521354421)

[3.5 Activity flow - 19 -](#_Toc521354422)

[3.6 Input data - 21 -](#_Toc521354423)

[3.7 Marker set - 21 -](#_Toc521354424)

[3.8 Segmentation - 22 -](#_Toc521354425)

[3.9 Co-ordinate system - 25 -](#_Toc521354426)

[3.10 Noise and methods for filtering - 25 -](#_Toc521354427)

[3.11 Sign properties - 1 -](#_Toc521354428)

[3.12 SVM - 1 -](#_Toc521354429)

[4 Testing - 2 -](#_Toc521354430)

[5 Conclusion - 3 -](#_Toc521354431)

[6 Bibliography - 1 -](#_Toc521354432)

Table of Figures

[Figure 1: "The horse of motion" by Eadwerd Muybridge - 2 -](#_Toc521354800)

[Figure 2: General marker set for human body - 4 -](#_Toc521354801)

[Figure 3: General structure of HamNoSys notation - 7 -](#_Toc521354802)

[Figure 4: Handshapes - 8 -](file:///D:\Radi\MyProjects\MOCAP\Motion%20capture%20data%20processing%20and%20analysis.docx#_Toc521354803)

[Figure 5: Wrist bending - 8 -](file:///D:\Radi\MyProjects\MOCAP\Motion%20capture%20data%20processing%20and%20analysis.docx#_Toc521354804)

[Figure 6: Extended finger orientation and Palm orientation - 9 -](file:///D:\Radi\MyProjects\MOCAP\Motion%20capture%20data%20processing%20and%20analysis.docx#_Toc521354805)

[Figure 7: Body parts - 10 -](#_Toc521354806)

[Figure 8: Six zones of signing space and their symbols - 10 -](#_Toc521354807)

[Figure 9: General activity diagram - 18 -](file:///D:\Radi\MyProjects\MOCAP\Motion%20capture%20data%20processing%20and%20analysis.docx#_Toc521354808)

[Figure 10:Sign analysis - 19 -](#_Toc521354809)

[Figure 11: Rest-pose: on the left MoCap data; on the right - real signer during recording - 20 -](file:///D:\Radi\MyProjects\MOCAP\Motion%20capture%20data%20processing%20and%20analysis.docx#_Toc521354810)

[Figure 12: Hand markers - 20 -](file:///D:\Radi\MyProjects\MOCAP\Motion%20capture%20data%20processing%20and%20analysis.docx#_Toc521354811)

[Figure 13: Hand displacement - 21 -](#_Toc521354812)

[Figure 14: Hand velocity - 21 -](#_Toc521354813)

[Figure 15: Hand acceleration - 22 -](#_Toc521354814)

[Figure 16:Hand velocity and points of interest - 23 -](#_Toc521354815)

[Figure 17: Division of frequency domain - 24 -](file:///D:\Radi\MyProjects\MOCAP\Motion%20capture%20data%20processing%20and%20analysis.docx#_Toc521354816)

[Figure 18: Cutoff frequency = 2 - 25 -](#_Toc521354817)

[Figure 19: : Cutoff frequency = 4 - 26 -](#_Toc521354818)

[Figure 20: Cutoff frequency = 10 - 26 -](#_Toc521354819)

[Figure 21: Cutoff frequency = 12 - 26 -](#_Toc521354820)

# List of Abbreviations

3D Three-dimensional

MoCap Motion Capture

fps frames per second

SL Sign language

HamNoSys Hamburg Notation System

RP Rest Pose

# Introduction

Motion capturing is a modern, fast developing data acquiring method capable to record movement in 3D. The detailed data that is retrieved from such recording is very useful not only for movie and game industry, but also in other fields as military, medicine and for validation and control of computer vision and robotics. The benefits of this method attract scientist to utilize it for linguistic analysis. Sign language is the primary alternative to a spoken language. Unfortunately, there are people for whom this is the only alternative that gives them the ability to communicate and share thoughts directly. Sign language uses manual movements and body language to communicate thoughts with others. The basic component of a sign language includes hand gestures, movements, orientation of fingers and hands, hand shapes and facial expressions to communicate certain feelings. Every region in the world has a unique spoken language and similarly, every region has a unique sign language. Thus, sign language varies from culture to culture and from region to region. People with speech and/or hearing impairment find it difficult to communicate with other individuals via sign language due to the inability of most of the people to understand sign language.

The purpose of this paper is to show the methods of studying and understanding the properties of signs from motion point of view and developing a tool for processing sign language data base. It is focused on hand movement and gestures analysis. It is cultural and regional independent, because it uses kinematic and statistical methods for processing the data.

Such database is set of dictionary (lexical items separated by default stance //not sure if it should be here or later in the main part) files with motion capture data. The tool will perform raw segmentation (in the first step) and fine segmentation (using acquired parameters from data) of dictionary items to extract the meaningful information for each sign which will be used for further analysis. It will show the challenges to determine the exact beginning and ending of the sign. The significant problem with the nature of the data – the containing of noise and methods for isolating that noise. Each sign will be processed for extracting its properties. Such as: if the sign is one handed or two, which is the dominant hand, hand location and orientation, finger orientation.

Later with the help of computer learning methods such as SVM the extracted information will be used to cluster signs for further processing.

# Theory. Analysis. Aim.

## Motion Capture

### History of motion capture

In general, the term motion capture (MoCap) is understood as the process of recording the movement of object, people or even animals. It is not specifically related to any device or approach. Today’s motion capture systems are product of many years of tinkering and innovation.

At the beginning MoCap analysis originate as gait analysis and animal locomotion around the year of 1872 and work of the photographer Eadwerd Muybridge. He used multiple cameras, triggered by strings to take pictures of moving bodies and animals. Capturing what human eye could not distinguish as a separate movement. His work “The horse in motion” was the first work recognized as a motion capture analysis. With a series of photographs of a galloping horse (see figure 1) he proves that horses do have all four hooves of the ground during their running stride [[3](Muybridge's#_E._Muybridge,_)].

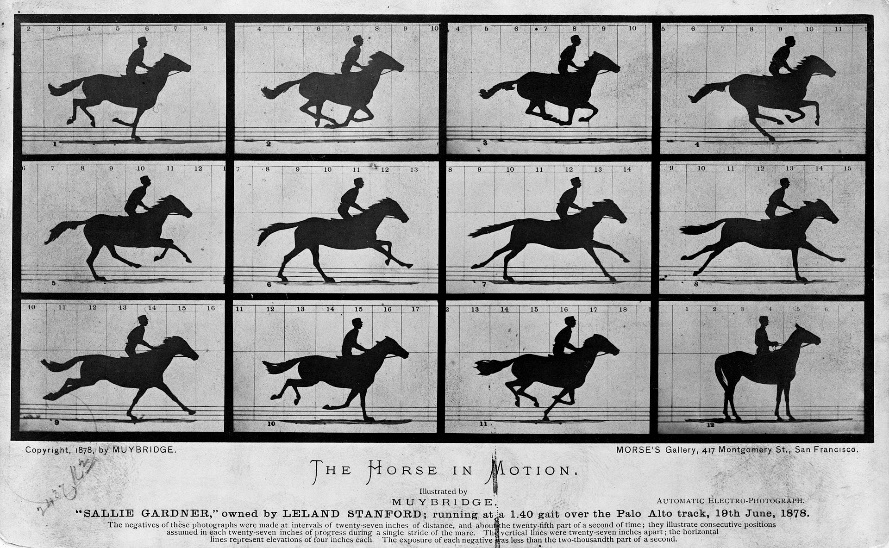


Figure 1: "The horse of motion" by Eadwerd Muybridge

Since then the technology has developed to the form of devices designed for direct 3D recording. The biggest success and from where it gains its popularity is the usage in the movie industry. Jar Jar Binks is the first fully digital created character with the help of MoCap. It was performed by Ahmed Best in Georges Lucas’ Star Wars Saga in 1990.

Motion capturing techniques started as biomechanics research tool, gained its popularity with its usage in entertainment industry and developed and expanded into education, training, sports, and robotics.

There are different approaches for motion tracking. The most advanced technologies are capable of tracking motion with high precision at very high sample rates. Optical passive system is considered as the most accurate and flexible technique and thus most common in the industry. On the other end of the scale are low-cost sensor technologies that most people use daily in their mobile phones, laptops, game controllers and so on. Every system has its advantages and disadvantages depending on its purpose.

The technologies presented in this section include acoustic, mechanical, magnetic, inertial and optical tracking.

**Acoustic tracking** systems calculate position upon the wavelength of an acoustic signal and the speed of sound. This approach is vulnerable due to the fact that sound speed varies with air pressure and temperature.

**Mechanical tracking** systems are usually based on some mechanical construction which measures angles or lengths between mechanical parts. These systems may be worn on the body, for example implementing sensor in exoskeleton or a glove. Mechanical tracking has been popular in music-related work, particularly for the purpose of developing new musical interfaces.

Magnetic tracking systems use the magnetic field around a sensor. Passive magnetometers can measure the direction and strength of the surrounding magnetic field. More advanced magnetic systems use an active electromagnetic source and a sensor with multiple coils. The position and orientation of each sensor can be calculated as a function of the strength of the induced signal in each sensor coil. Magnetic trackers are able to operate at high sampling rates (more than 200 Hz) with high theoretical accuracy. However, the systems are sensitive to disturbances from ferromagnetic objects in the tracking area.

**Inertial tracking** systems include those based on accelerometers and gyroscopes. These sensors are based on the physical principle of inertia. Inertial tracking systems have certain strong advantages. Firstly, they are completely self-contained, meaning that they do not rely on external sources such as acoustic ultrasound sensors or cameras which require line-of-sight. Secondly, the sensors rely on physical laws that are not affected by external factors such as ferromagnetic objects or light conditions. Thirdly, the sensors are very small and lightweight, meaning that they are very useful in portable devices; and finally, the systems have low latencies and can be sampled at very high sampling rates.

Various types of video camera are used in **optical motion tracking**. In principle, any digital video camera can be used — in fact, one of the most affordable sensors for conducting motion tracking is a simple web camera. Cameras used in optical motion tracking are either (1) regular video cameras, (2) infrared (IR) video cameras, or (3) depth cameras.

Ordinary video cameras sense light in the visible part of the electromagnetic spectrum. Each pixel in the camera image contains a value corresponding to the amount of light sensed in that particular part of the image.

Infrared cameras sense light in the infrared part of the electromagnetic spectrum. Some infrared cameras can capture heat radiation, e.g., from humans, but the most common use of infrared cameras in tracking technologies is in a slightly higher frequency range. This is achieved by using some active infrared light source, and either capturing the light from this source directly or as reflections on the tracked objects. Typical implementations consist of a group of infrared light-emitting diodes (LEDs) positioned near the infrared camera and capturing the reflection of this light as it is reflected from small spherical markers.

Depth cameras provide a layer of depth information in addition to the regular two-dimensional image. These cameras use some technology in addition to the regular video camera. When not provided by the camera itself depth information can be gained through the use of stereo cameras. This involves two cameras mounted next to each other, providing two similar images. Depth information is found as a correlation function of sideways shifting of the images.

After obtaining the video data various processing is applied to the video stream. The video processing that is performed in optical tracking systems is primarily dependent on two factors: (1) whether or not the tracking is based on markers and (2) the camera configuration. But in any case the first processing step is to remove unwanted information from the video, i.e. separate the foreground from the background.

### Optical-based motion capture system VICON

Motion capture technology was used to generate data files for both continuous and isolated utterances. The data used for this project was recorded at the University of West Bohemia with MX series device from VICON [[6](Go_Further_with#_)]. This is an overview of the basic principle of the technology on which the system is based. This system is based on optical- passive technology and was chosen for the sign language project because it suits the best for the purposes of the project.

The technology provides accurate data at fast sampling rates, and the same system can be used to capture the motion of a wide range of structures, including objects, animals, human bodies, fingers and faces. By using passive (reflective) markers, all processing is done externally, and the captured subject does not need to wear electrical equipment or wires. Which can be considered as advantage in capturing finger motion, because the presence of wires can impede the naturalness of movements. The system is comprised of eight specialized infra-red cameras along with computers and software for image analysis and processing. The cameras detect small markers placed on strategic locations on the captured subjects. Markers are treated as points without volume, and as such only their positon (not orientation) can be tracked On figure 2 can be seen the general marker placement for human body. For this project passive markers are used, they are coated with retroreflective material, and this requires the cameras to emit the light, which is reflected back and detected. The external source of light is part of the camera’s body.



Figure 2: General marker set for human body

The detected information is then processed in dedicated hardware using software provided by the manufacturer to triangulate the 3D locations of the markers. The exact process and algorithms are know-how of devices’ manufacturer, but it is based on stereoscopic vision.

After reconstructing the 3D points into a point cloud, the system needs to determine which point is which and label each point with a marker id. This process is commonly referred to as marker labelling. The marker data is then used to estimate the kinematic motion of a model of a human skeleton. The estimation in form of bone lengths and joint angles can be used for further processing or animation. This process is called solving. The mentioned software is able to provide semi-automatic calibration of human subject for general movement.

Although this technology comes with highly accurate results and it is very flexible there are some disadvantages which may result in poor data quality or extensive costs in manual post-processing. As every vision-based technology it needs clear line of sight and occlusions may cause serious challenges. Markers placed on fingers are especially problematic and often suffer from self-occlusions when the fingers are bent or the hands are facing towards the body or with palm-up [[5](#_N._Wheatland,_Y.)]. Occlusions do not only cause problems with missing data, they also make the labeling process more difficult as this reduces the available information for inference. Further challenges arise in situations when several markers come in close contact (such as clapping hands).

To summarize, optical motion capture provides highly accurate data, but may require a large amount of manual post-processing.

## Sign language analysis

Sign language is used by millions of people around the world. It is used to facilitate communications with people with speech or hearing impairments. There are different sing languages as there are different spoken languages. Also there is common misconception that sign language is dependent on spoken language, that it is spoken language expressed in signs [[8](#_Perlmutter,_David_M.,)]. Linguists has studied and proved that sign languages exhibit the fundamental properties that exist in all languages and there are similarities between both forms, but there are some basic differences. The linguistic mechanism in both is different and therefore it causes difficulties for people with such impairments, especially for those who are born this way, to use even the written form of a spoken language [[9](#_Sandler,_Wendy;_&)]. In spoken language units are organized sequentially (it is not possible to say two different words at the same time), but in sign language the meaning of one unit may be carried by the shape of the hands and their position or/and by the position and movement of head and mimics. These two components can be carried simultaneously.

All these circumstances not only put a communication barrier between people using sign language and majority of hearing community, but also restrict them from most sources of information. Another difference between both is sign languages does not have its own system. This led scientist to develop writing system to represent signs. The pioneer in sign language analysis was W. Stoke. As it is explained in M. Kato paper [[3](A#_Mihoko_Kato,_)] Stoke proves that each sign in American sign language has tree elements that distinguish it from all other signs:

1. Hand Configuration (the distinctive configuration of the hand or hands making a sign),
2. Place of Articulation (the place where a sign is made),
3. Movement (the action of the hand or hands).

Stokoe decided to call the active hand the “designator” or “dez”; the place, the “tabula” or “tab”; and the action, the “signation” or “sig.” A sign is produced by a combination of these three aspects.

Nowadays most of the notion systems for sign languages are based on his study and notion system.

### Hamburg Notation System

Such system is the Hamburg Notation System (HamNoSys) [[1](Hamnosys-representing#_T._Hanke,_)]. It is a work of scientists from University of Hamburg. It is an alphabetic system that decompose signs to phonetic level and describes their sub lexical parameters location, configuration and movement. It is based on Stokoe’s notation system [[2](#_Stokoe,_William_C.)]. It is designed to be usable in variety of context with the following goals in mind:

* International use - HamNoSys transcriptions to be possible for virtually all sign languages in the world.
* Iconicity – because of the large number of parameters variations, newly created glyphs should be designed to be easy to memorize or even deduct the meaning
* Economy – notation of signs should make use of principles as symmetry conditions, thus resulting in shorter notions.
* Integration with standard computer tools – It should be usable for computer –supported transcription, standard text processing and database applications.
* Formal syntax – The notation language should have well-defined syntax.
* Extensibility – As SLs are developing and differ from each other, HamNoSys should allow both for a general evolution and specializations. [[1](Hamnosys-representing#_T._Hanke,_)]

Because of all described goals of HamNoSys it seems that it is preferred from scientific community working in the domain of language analysis and synthesis, although it is not very accepted from deaf community.

This notation system was chosen for sign language project because it suits the best for its purposes. I used it as guidelines in my work for describing the properties of signs from motion point of view.

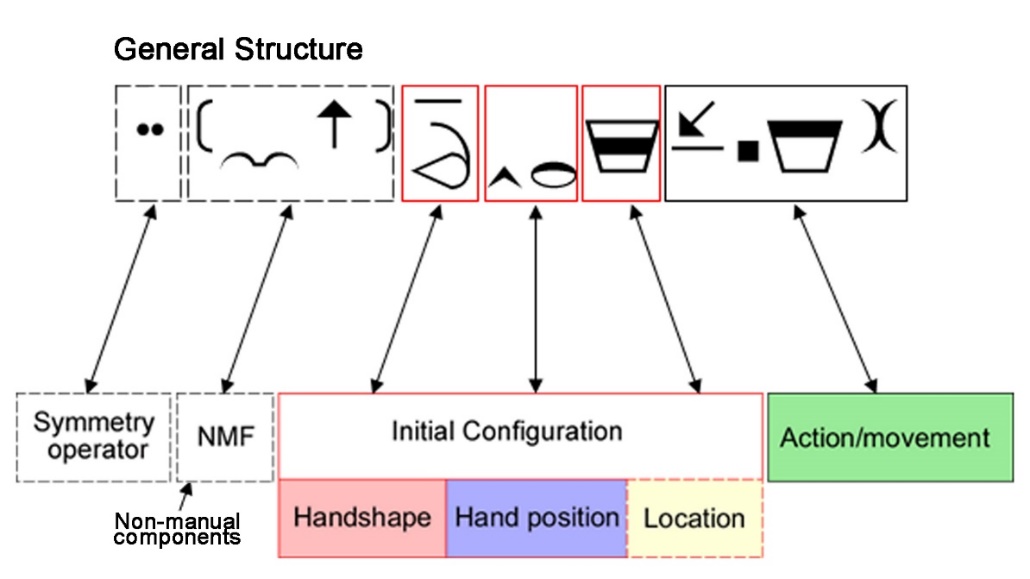


Figure 3: General structure of HamNoSys notation

In general, a sign notation consists of description of non-manual features, handshape, hand orientation and location, plus the actions changing this posture. If the sign is two handed at the beginning of notation is added an operator to show how the description of dominant hand is copied to the non-dominant hand. The signs are realized in signing space and terms expressing directions are determined from the signer’s perspective. Figure 3 shows example of sign notation and its general structure with its mandatory (the ones with solid border) and optional components (boxes with dashed border).

#### Handshapes

Handshape is expressed by symbols for basic forms – **Fist, Flat hand, Separated fingers, Thumb combination** and bending (see Figure 4)

The configuration of the thumb alters the structural arrangement of the entire hand and thereby define a new group of handshapes, where shapes are derived from other basic shapes.

In addition to the general description of the sign, information for different fingers and finger parts (in respect to the fingers involved), may be included.

Figure 4: Handshapes

#### Hand orientation

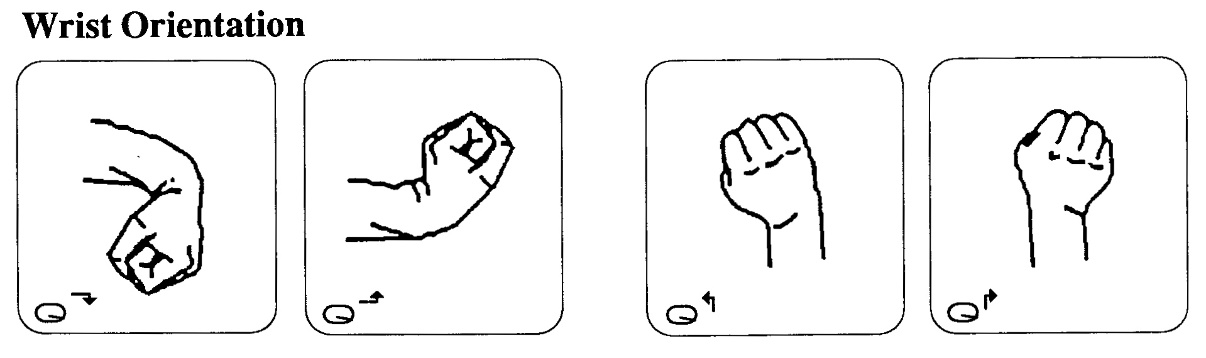
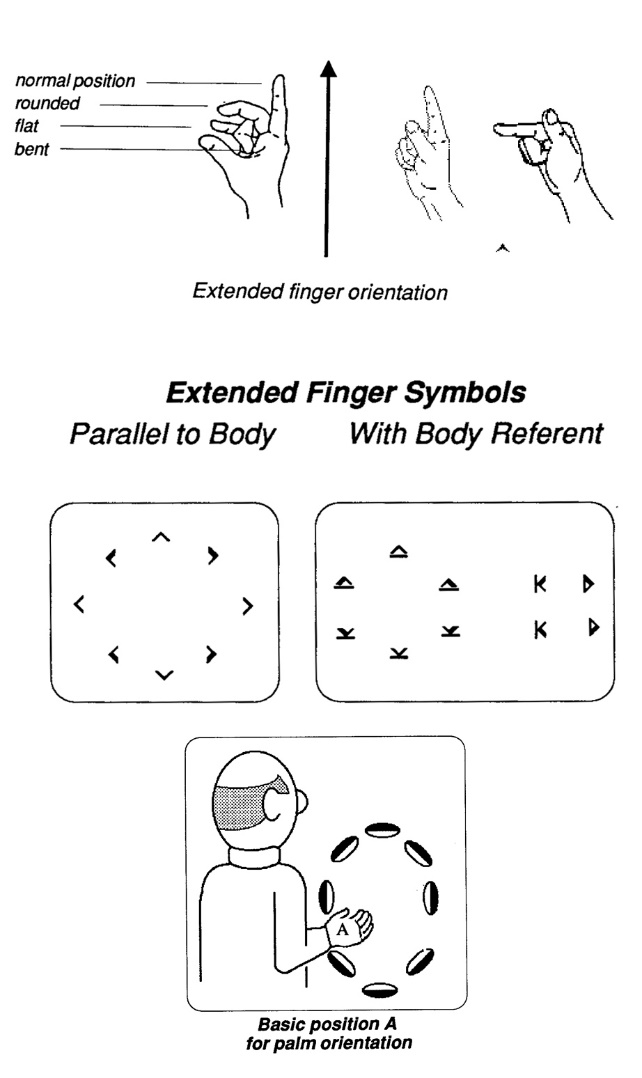
The orientation of hand is described by **Wrist orientation, Extended finger and Palm orientation**. Where wrist orientation is characterized by bending of the wrist toward the pulse or back of the arm, toward the thumb or little finger (see Figure 5). It is omitted when the bending is natural consequence of palm orientation with movement.

Figure 5: Wrist bending

The vector originating at wrist, running along the back of the hand and pointing to the direction pointed by the fully extended finger, shows the orientation of the Extended finger. This orientation may be difficult to be defined when all fingers are bend in some way. As it can be seen on Figure 4, the two hand shapes on the right are noted with same symbol. Fingers point in different direction, but the base of the finger (knuckle) points in the same. If finger orientation is not parallel to the body plane, the notation must contain the body referent symbol (Figure 4).

Palm orientation is always noted after the Wrist and Extended finger orientations are. To define it, first Basic Position of the palm must be determined, then the orientation is determined by the orientation of the palm around the shaft of the hand (Figure 6, bottom). There two Basic Positions. When finger orientation is away from the body is Basic Position A, otherwise is Basic Position B.

Figure 6: Extended finger orientation and Palm orientation

#### Hand location

Hand location is noted only if it is very specific on the body or in the signing space. If not noted it is understood that sign takes place in the natural space, in front of the upper part of the body. There are different sets of location signs. One is expressing locations on body (Figure 7), another one describes relative position to the body part. Figure 8 shows the division of the signing space into six horizontal zones, each zone is noted with different symbol. In two-handed signs location information concerning hands and arms, refers to locations on non-dominant hand and characterize the relation between the two hands. Again all determinations of left and right are made from signer’s point of view.

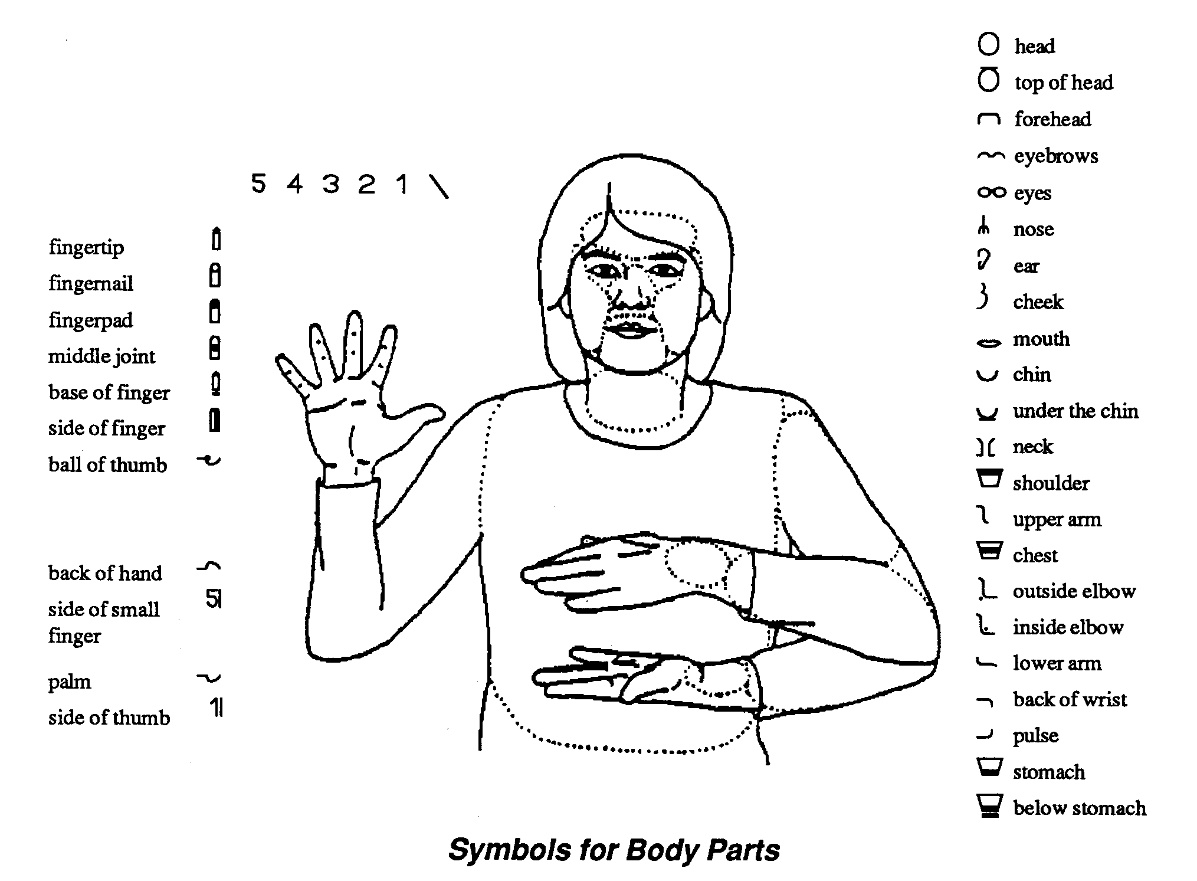


Figure 7: Body parts

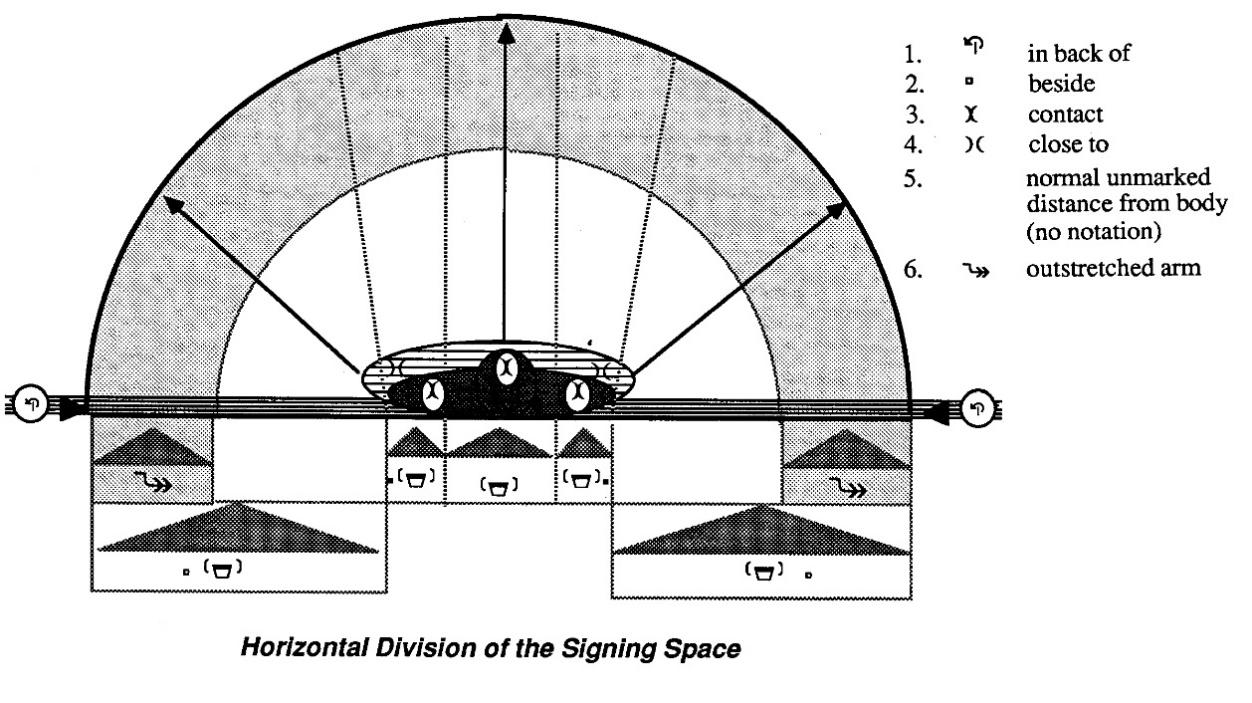


Figure 8: Six zones of signing space and their symbols

#### Movement

There are few aspects of the movement that have to be considered when describing movement in signs: **Movement type, Manner of movement, Repetitions, Order of movement symbols**. The distinguished types of movement are: straight, curved, wavy, zigzag, circular and spiral.

The manner of movement aims do describe size, speed and intensity of the movement. There are three degrees of size: large, normal and small and only large and small size are explicitly notated. The speed and intensity are described as: fast movement, slow movement, tense, hold or rest (usually on onset or offset of the movement), abrupt halt at the end of the movement.

Two types of repetitions are distinguished. Ones which lead back to the beginning of the movement and those whose initial position continuously change. The number of repetitions is indicated by the number of repeating symbols.

The order of notation of these aspects is strongly defined as follows:

1. Circling movement or direct movement
2. Type of movement:
3. Manner of movement
4. Repetition

#### Two handed signs

Two handed signs are divided into symmetrical and nonsymmetrical signs. In the symmetrical ones both hands show the same handshape and therefore only the dominant hand is notated. But the coordination of both hands should be noted either as mirrored or parallel. Also should be noted if the movement in both hands occurs simultaneously or alternately. In case of nonsymmetrical signs, the movement occurs only with the dominant hand and the handshapes are often different, therefore notation for both hands is needed. Always the notation of the dominant hand should precede the non-dominant.

#### Conclusion

Sign language and spoken language developed separately and this led to different linguistic structures, restricting people with haring and speaking impairments from vast information and free communication. Linguist have been working on developing good sign representation as it is important from many aspects. Although it is not very accepted by deaf community, because of its complexity, the Hamburg Notation System seems to be a good choice for representing signs in computer, as it was developed with the idea to be available for international use and easy for computer implementation.

## Optical-based motion capture for sign language synthesis

There have been several research projects for developing assistive technology for Deaf population. They are based on different techniques, such as key frame techniques and procedural synthesis. As it is explained in [[13](#_L._Naert,_C.)] they allow fine control over the movement of the signing avatar, but it might be poorly accepted by people, because of lack of human-like movements. Scientists as McDonald et al [[4](#_J._McDonald,_R.)] work on resolving this problem by analyzing noise in motion capture data and adding the human-specific noise in the key-frame driven motion. Other approach is data-driven synthesis. When used with MoCap data from real signer the motion is preserved and therefore the animation is more human-like.

On the other hand, this approach needs a rich MoCap dataset for sign language, which faces other challenges. As mentioned before SL is a complex composition of simultaneously movements of different parts of human body. The difficulty is to record and synchronize all these components, because every utterance can be done with time and special variance even if performed by one signer. At university of West Bohemia face this problem by using the state-of-the-art device for motion capturing [[12](#_P._Jedli_cka,_Z.)]. The system they use is combination of different cameras and retro-reflexive markers with different sizes for different body parts. As this is a new approach, the data set is not that rich yet.

Datasets for SL may differ in their content. Some contain isolated utterances other whole sentences or phrases with continuous utterances. For the purpose of my work I have used the dataset with isolated utterance.

## Segmentation of SL

The benefits of data-driven methods for SL synthesis were discussed in the previous section, but in order to have good and valid results the data must reliable. Before proceeding to any kind of analysis the data need to be structured in forms that suits the final purposes

Segmentation is the process of breaking a continuous sequence of movement data into smaller and meaningful components. The process consists of determining the exact beginning and ending frames of this meaningful component. Identifying of segments is challenging task due to multidimensional nature of SL.

There is a study over French sign language [[13](#_L._Naert,_C.)] which is based on manually annotated data, segmentation is performed by expert annotators (deaf signers). As it is explained in the paper manual segmentation and annotation is laborious and time-consuming process. Identifying the exact start and end frame made by human is subject to variability, even if made by experts. The focus of this study is transition movement. This is movement have no linguistic meaning but it is important for making the animation more human-like.

There are several studies researching the issue of segmentation, but as far as I am aware, there still have not been developed a fully automatic segmentation of continuous utterance with good precision. The approach for identifying segment boundaries by detecting changes in kinematic features for general motion and in the case of isolated utterance seems to be promising.

### Method for segmentation

In my work the segmentation is designed to be made over sets of isolated utterance and is done in two steps. In these data sets signs are separated with one exact pose – rest-pose (RP). The identification of segments is based on kinematic analysis of hands movement.

At first step start and end tag define these frames where hands leave and enter, respectively, rest-pose. Knowing this frames helps me to analyze hands behavior during rest pose and thus refine threshold variables for finer segmentation on next step.

At second step the frames that I am interested in are those where the meaningful part of sign begins and ends. Meaningful part is the one that have linguistic meaning and is described by HamNoSys notation. This motion subsequence will be further analyzed.

The main idea behind kinematic analysis is that significant changes in trajectory of a hand can be indicated by changes in velocity and acceleration. Indeed, human body is not a simple mechanism and hand movement (or in our case change in the trajectory of the marker) is related to the kinesthetic muscle response, resulting in changes in velocity and acceleration. The correlation between this points of interest and changes in kinematic features was studied at the university of California for the for a project for mapping music to a dancer’s motion [[16](#_F._Bevilacqua,_J.)].

### Conclusion

In this paper SL is reviewed as continuous stream of motion, with no consideration of linguistic meaning of the signs. My purpose is to extract the properties of each sign based on HamNoSys notation (described in section 2.2.1). But in order to have valid results I had to do proper segmentation, to extract the meaningful parts from long stream of data. To achieve that I analyzed the kinematics of the motion as position, velocity and acceleration.

## Conclusion

Sing language is important for certain part of our community. It is complex and by its nature very different of the spoken language that most of us are familiar with. There have been different researches and projects for developing signing avatar with increasing quality over time. New approaches include data-driven methods, using devices for recording 3D movements in order to save human-like motion. But the issue of automatic processing and computer understanding are still challenging tasks. My purpose is to develop a tool helping dealing with such issues.

# System Architecture

After analyzing the theory behind motion capture technology, sign language and its properties and different approaches for sign language synthesis is possible to proceed with actual describing of the system’s architecture. The tool will perform two step segmentation over input data. The input data is long stream of 3D recorded signs performed by professional signer. After that the motion data will be analyzed to extract motion features of each sign.

## Python

Python is a high-level, interpreted and general-purpose dynamic programming language that focuses on code readability. The syntax in Python helps the programmers to do coding in fewer steps as compared to other programming languages. It is very flexible language supporting multiple programming paradigms, including object-oriented, imperative, functional and procedural. It has a large and comprehensive standard library that has automatic memory management and dynamic features. The programmers of big companies use Python as it has created a mark for itself in the software development with characteristic features like-

* Interactive
* Interpreted
* Modular
* Dynamic
* Object-oriented
* Portable
* High level
* Extensible in C++ & C

One of its biggest advantages is its Extensive Support Libraries that include wide range of areas such as string operations, Internet, web service tools, operating system interfaces, protocols, data-analysis and more. These libraries make Python more useful for specific purposes. Additionally, Python has become a go-to language for data analysis. With data-focused libraries like NumPy, and matplotlib, making it powerful tool for processing, manipulating, and visualizing data [[14](#_Python_documentation_-)].

### **NumPy**

The most fundamental package, around which the scientific computation stack is built, is NumPy (stands for Numerical Python). It provides an abundance of useful features for operations on n-arrays and matrices in Python. The library provides vectorization of mathematical operations on the NumPy array type, which ameliorates performance and accordingly speeds up the execution.

### SciPy

SciPy is a collection of mathematical algorithms and convenience functions built on the NumPy extension of Python. Its different submodules correspond to different applications, such as interpolation, integration, optimization, image processing, statistics, special functions, etc.

### Matplotlib

Another SciPy Stack core package and another Python Library that is tailored for the generation of simple and powerful visualizations with ease is Matplotlib. It is a top-notch piece of software which is making Python (with some help of NumPy, SciPy) a cognizant competitor to such scientific tools as MatLab or Mathematica.

However, the library is pretty low-level, but with a bit of effort, you can make just about any visualizations:

* Line plots;
* Scatter plots;
* Bar charts and Histograms;
* Pie charts;
* Stem plots;
* Contour plots;
* Quiver plots;
* Spectrograms.

There are also facilities for creating labels, grids, legends, and many other formatting entities with Matplotlib. Basically, everything is customizable.

### C3D

This is a small library for reading and writing C3D binary files. C3D files are a standard format for recording 3-dimensional time sequence data, especially data recorded by a 3D motion tracking apparatus.

## Sublime Text code editor

Sublime Text 3 is a lightweight, cross-platform code editor known for its speed, ease of use, and strong community support. It’s an incredible editor right out of the box, but the real power comes from the ability to enhance its functionality using Package Control and creating custom settings. There are many plug-ins accessible that make Python development extremely smooth and pleasant.

**Anaconda** is an extremely powerful Python package for Sublime. It offers:

* Python code auto completion
* Python lining (highlights both syntax errors and PEP8 violations)
* Python documentation

## Autodesk MotionBuilder

should I include it?

## C3D file format

C3D is a biomechanics and motion capture file format. It stores raw 3D coordinates and analog sample data, together with information that describes the stored data. The C3D format treats information as if it belongs to one of two classes: **Physical Measurements, Parameter Information**.

Physical Measurements - The C3D specification expects physical measurements to be one of two types, either positional information (3D coordinates) or numeric data (analog information).

Parameter Information - contain information about the data such as measurement units and data point labels, database information such as the subjects name, diagnosis and other items that may be specific.[[15](The_C3D_file#_)]

For this project C3D file format is used for the input data. It contains the raw output data of the MoCap system.

## Activity flow

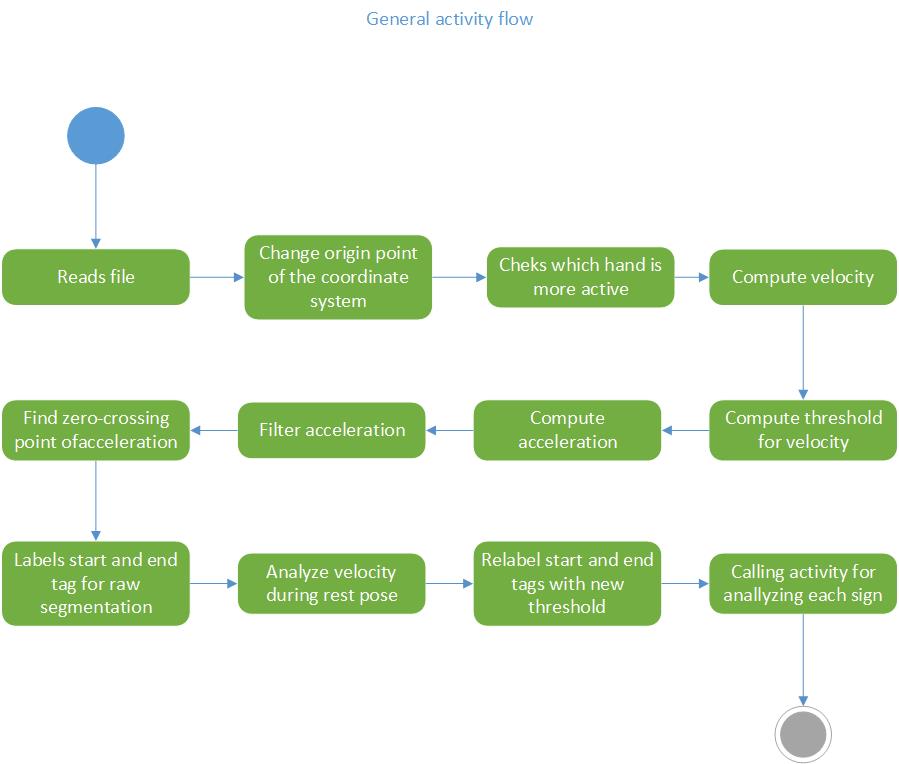
The dynamic behavior of the system is shown in the following UML activity diagrams.

Figure 9: General activity diagram

The general activity flow of the system is shown on Figure 9. First the input file is read and the information about markers positions is stored in multidimensional array (in form of: for each frame, for each marker, x ,y ,z, coordinates). Other extracted information is the frame rate and the set of labels of used markers (in an list structure). The last element of the list of markers is additionally added marker for the origin of the coordinate system.

Next activity is to check which hand is more active. The further analysis is performed over the movement of the dominant hand.

Next step is to recalculate 3D co-ordinates for each frame so that the center of the co-ordinate system to be in the middle of the body. This is done in order to make further calculations independent of the signer’s physicals characteristics or his position in the scene.

After that the hands velocity and acceleration are computed. Acceleration computation is made over the normalized velocity.

A crucial step in the processing of the data is the use of low pass filtering to eliminate jitter in the acceleration. It is important because of the next step – finding the zero-crossing points. If the function is too noisy there might be found false points. On the other hand, filtering cause loss of information and thus missing important points. As explained earlier the sudden changes in velocity indicates changes in trajectory. Therefore, this points are matter of interest, they can indicate a start or end of a sign.

Next step is called labeling. It uses a threshold function to filter the set of zero-crossing points. The result at the end of this step is list of start and end frames for the raw segmentation.

Knowing this start and end tags, gives the knowledge to analyze hand velocity during rest-pose and this way to choose better value for the threshold function.

Next step is just redoing the labeling with the new threshold value.

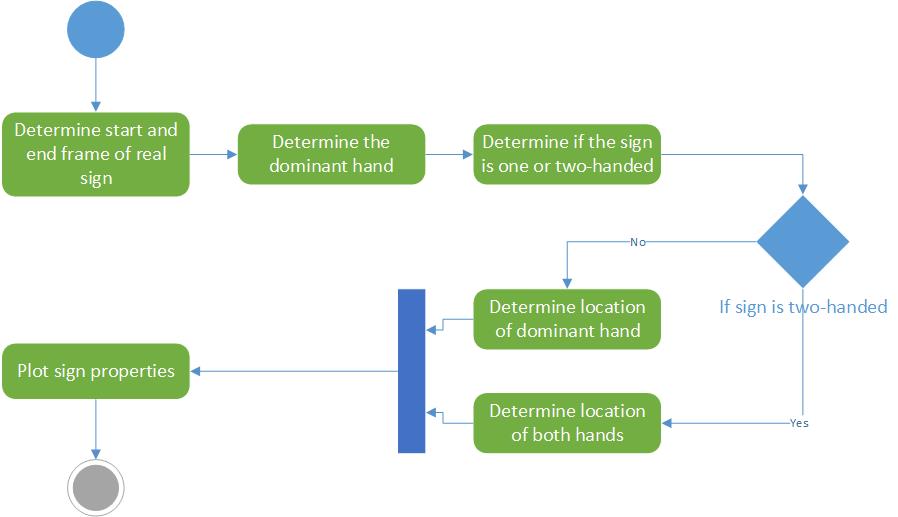


Figure 10:Sign analysis

Figure 10 shows the flow activity during analysis of the segmented data. First is determined the exact start and end frame of the real sign (the explanation what exactly is the real sign is given in section 2.4). By comparing hand displacement of left and right hand, can be determined the dominant hand and if the sign is one or two-handed. Based on this knowledge hand location can be examined (this is explained in more details in section 3.11).

## Input data

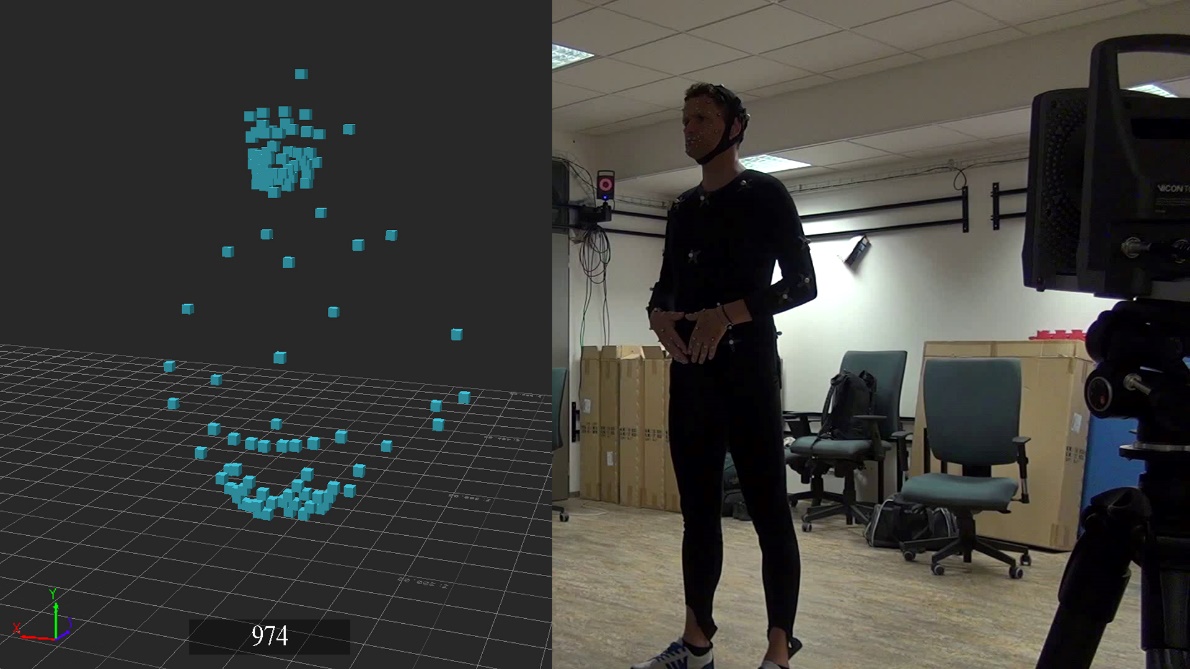
The input data is a so called dictionary file, it consists of sequence of lexical items, followed by a rest-pose (see Figure 11).

Figure 11: Rest-pose: on the left MoCap data; on the right - real signer during recording

The input data is direct output of a MoCap system in .c3d file format. Basically it is a “cloud” of points with labels and coordinates for each frame. The information that needs to be extracted from this file is the 3D information about markers position in each frame, the set of used labels and the frame rate at which the recording was performed.

## Marker set

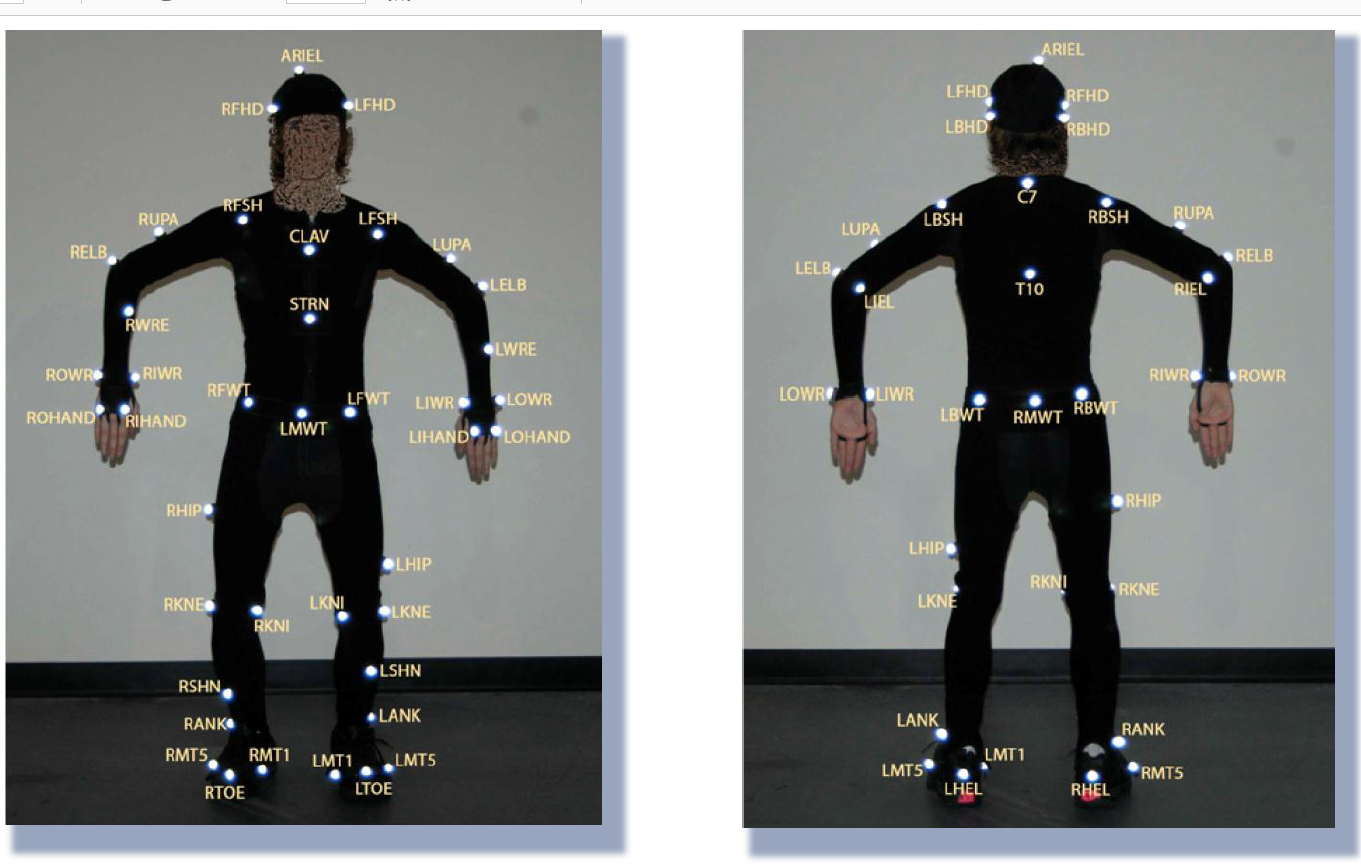
The marker set consist of 109 markers, placed over the torso, hands, fingers, head and face. The focus in this paper is in hands movement, not the movement of each marker. And trajectories of the four markers on top of hand can be interpreted as hand trajectory, so called solving process. Hand trajectory is calculated for each frame as follows:

Figure 12: Hand markers

## Segmentation

Segmentation is performed in the meaning of finding start and end frames of all lexical items in a dictionary file. A common approach for motion segmentation uses kinematics features of the movement.

* **Displacement -** defines the change in position that occurs over given period of time. It is a vector. Figure 14, shows an example of hand marker trajectory over time (measured in frames) for X, Y, Z axes.



Figure 13: Hand displacement

* **Velocity –** it is the change in displacement over time. It is calculated along the three axes. For the purpose of segmentation, I am more interested in the general movement of the hand rather than the movement along each axis, therefore a normalization is made by The Frobenius norm equation over each axis for each frame.

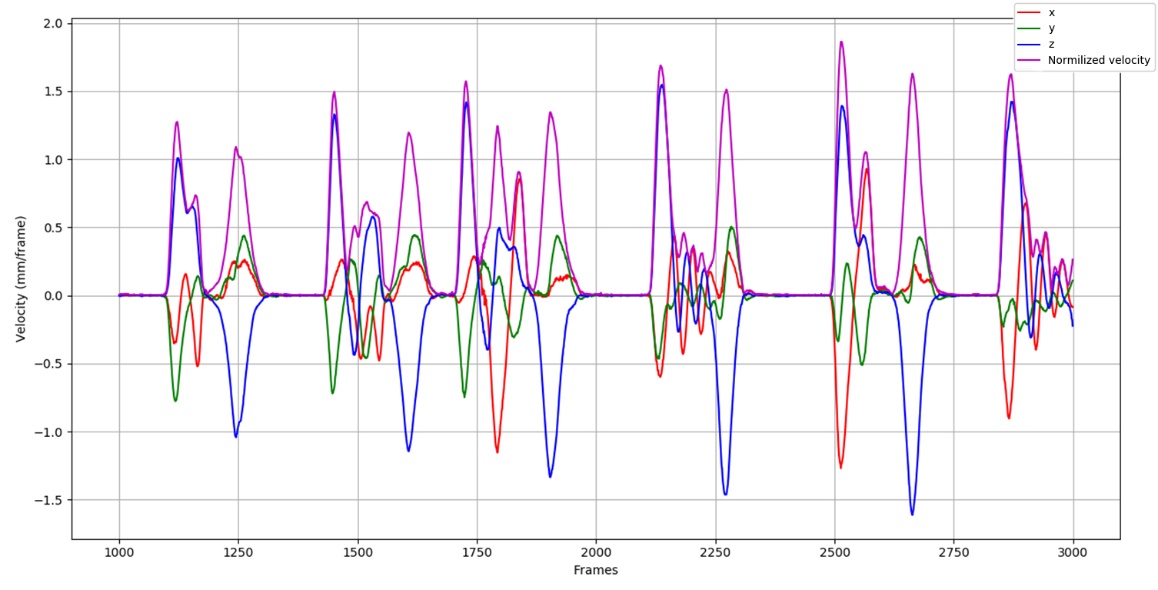


Figure 14: Hand velocity

Figure 14 shows a graph of hand velocity along the tree axes and the normalized velocity for a period of time.

* **Acceleration -** describes how the velocity changes over time.

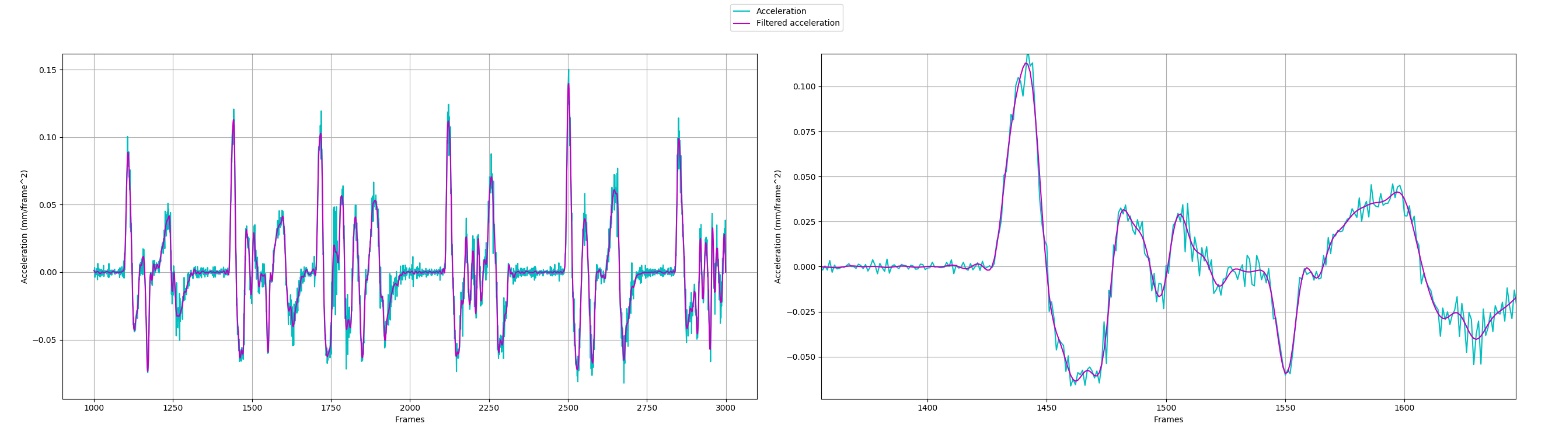


Figure 15: Hand acceleration

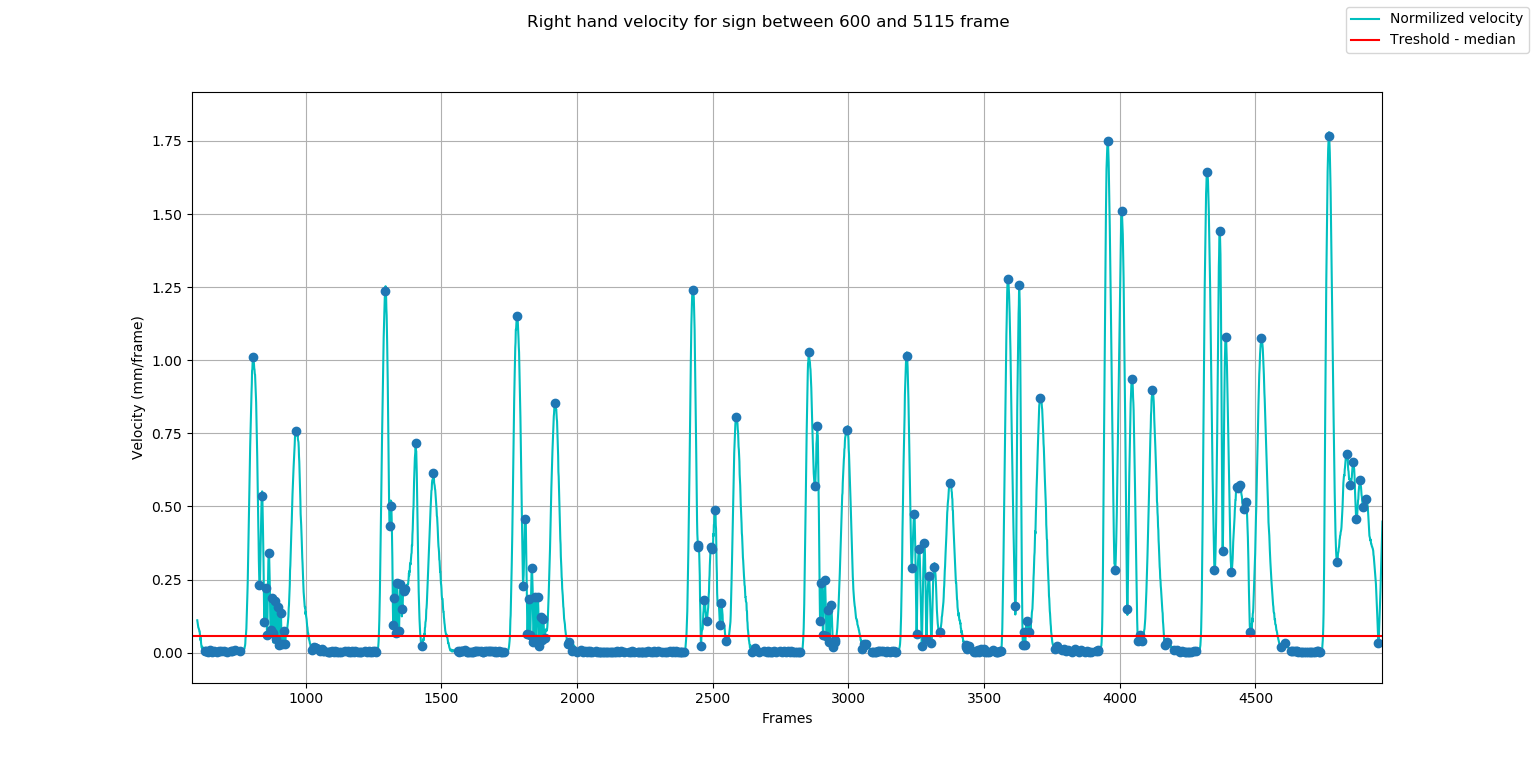
There is another aspect the has to be considered here – noise. There are many reasons for noisy data and different source of noise. Although the data presented in the time graph in Figure 13 seems quite smooth, calculating the acceleration greatly increases even the slight variations in it, as it can be seen on Figure 15 (the blue line). Butterworth filter is applied. The issue of data noise and filtering is described in more details in section 3.10.

The idea behind kinematic approach is finding the zero-crossings points in acceleration to identify significant changes in velocity. It is possible to find points of interest based on velocity extremums. Through experiments I found it more useful to base them on zero-crossing in acceleration.

The task of finding this points can be narrowed down to finding the points where the function changes its sign, meaning at which frame there are extremums in hand velocity

Next step is to decide which points are really releated with the beginning and end of the lexical item. First they are filtered by threshold function. The threshold value is chosen as median value of velocities. It gives an safe value, not to low to miss some point, neither too hight to have too many points. Figure 16 shows an example of hand velocity during movement in a whole dictionary

file, marked points of interest and the chosen theshold.

Figure 16:Hand velocity and points of interest

At this graph the correlation between hand kinematics and points of interested is easeally obesrvable.

* **Labeling -** Frames at which velocity value is unther the threshold are labeled with 0, others with 1. Then this labels are examin as vector of zeros and ones. The idea is that if the start of a sign is at certan frame it will be precceeded by at least one zero and followed by at least one 1 (ex. 0011), respectively if it is end, it would be precceded by one and fowolled by zero (1100). Ofcourse there might be some false detections, becouse of the bad threshold value or sudden halts and abrupt changes. The aim at this step is to determaine frames where at least one of the hands leaves rest-pose or where enters it. So a check of hands positoin can be performed in order to precise the labeling and reduce the errors.

Final result is pairs of frames pointing to beginning and ending of each lexical item, or frames where RP ends and next one begins. Knowing frames in which hands might be in RP, means that their beaviour during a well defined and almost static position, can be analyzed and the threshold can be refined.

## Co-ordinate system

Before recording anything with motion capture system, the systems needs to be calibrated. At this process a co-ordinate system is set. Usually the absolute co-ordinate system is on the floor in the middle of the captured volume (as in the case of recordings for SL database). This world origin may be at differ, but it has to be set before the recording.

Before any processing of the data by this tool, a relative co-ordinate system is introduced for several reasons.

SL project is based on data-driven method, so it needs a rich and reliable set of data. It should consist performances of different signers. Meaning that further analysis should not depend of person’s physique, like height weight, etc.

Also, most of the sign’s features are reviewed according to signer’s body plane, not according its global position.

For these reasons a method for changing the origin point was implemented. Considering all things said the new origin point is fixed in the middle of pelvis.

## Noise and methods for filtering

Although motion capture systems are improving over the time and tends to provide precise and accurate data, there is still the issue with noisy data. It is one of the most significant challenges that researches have been studied. In general, under the term “noise“ is understood the unwanted modifications of the motion data. There might be several sources of noise.

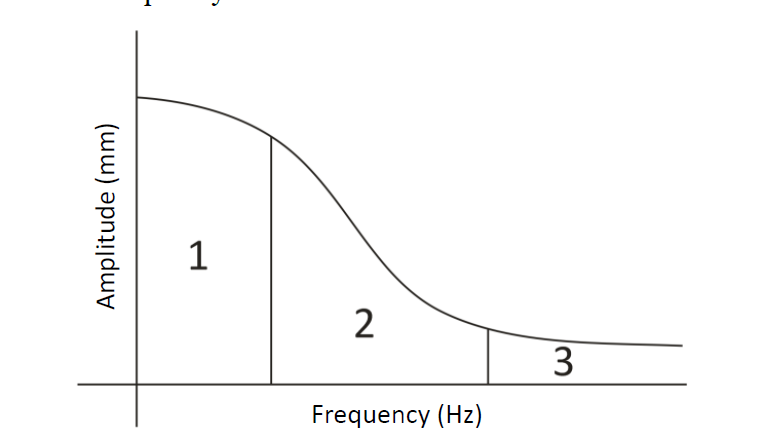
The nature of the noise and its linguistic relevance is reviewed by McDonald et al in their study for a prosodic analysis of a motion [[4](#_J._McDonald,_R.)].They analyze trajectories of different markers and use Fourier transformation to get signal’s spectrum in frequency domain. The frequency spectrum is divided into three sections which have different impact over sign analysis. According to their study based on medicals researches oscillations with frequencies higher than 10-12Hz (section 3 in figure 17) cannot be produced by human muscles thus these frequencies can be considered as noise caused by recording technology or other external factors and can be safely eliminated with low-pass filter. Other frequencies may be produced by human motion, but not all of them have meaning for sign language. The cutoff frequency between section “1” and “2” may vary according to the type of motion. Human body may produce different types of motion. For example, fingers, because of their lower mass and smaller range of movements are capable of higher frequencies than hips. They also present a table of body parts and their cutoff frequencies. Another important finding in his research is that although frequencies in the section marked with ‘2” have no linguistic meaning, they are produced by human motion and therefore its important from realistic point of view. Further analysis of these frequencies can contribute to achieving more human-like animation.

Figure 17: Division of frequency domain

A crucial step in the process of segmentation is filtering the signal. For this purpose, a Python implementation of Butterworth filter was used. It is a type of signal processing filter, based on Fourier transform. Through experimentation and based on results from McDonald’s research, the cutoff frequency parameter of 12Hz was chosen.

The following figures shows acceleration in time domain and the resulting graph after applying the filter with different cutoff frequencies.

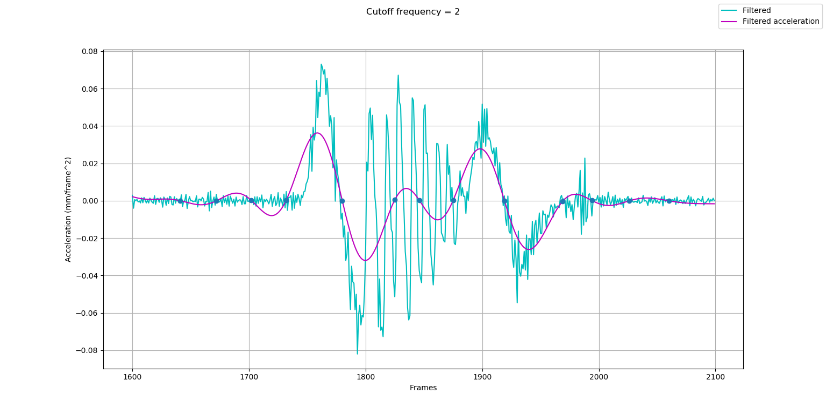


Figure 18: Cutoff frequency = 2

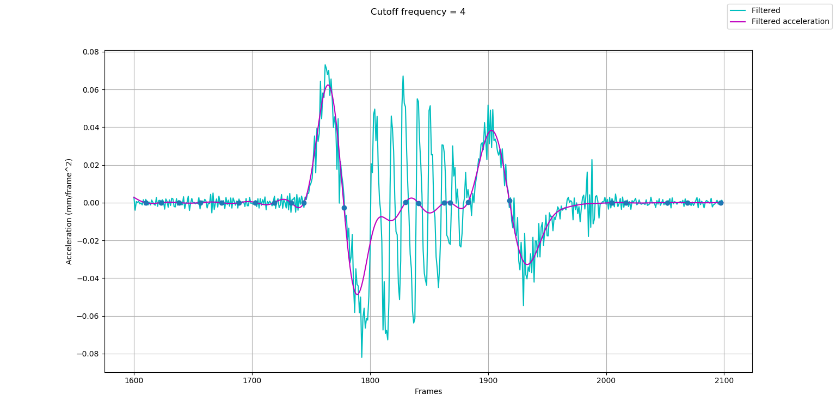


Figure 19: : Cutoff frequency = 4

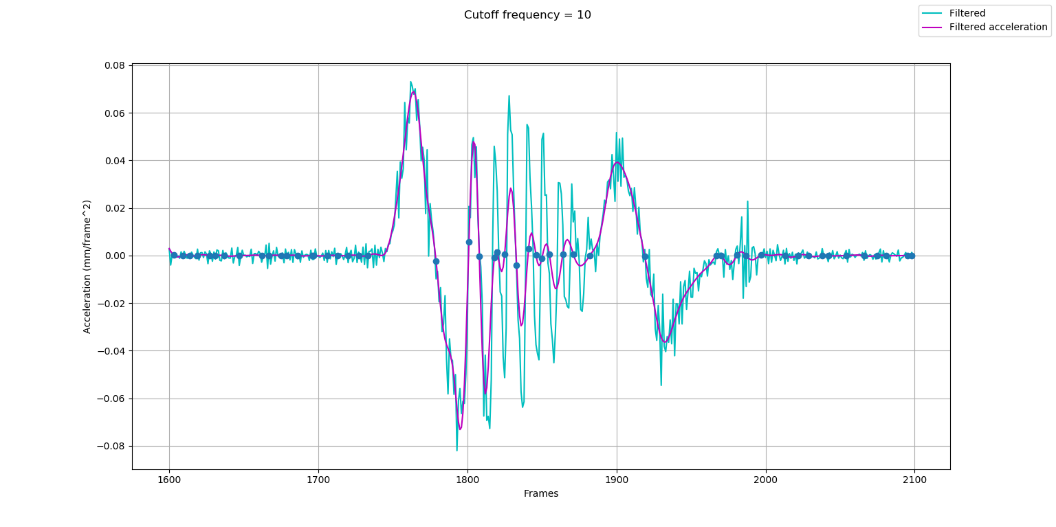
As discussed earlier according to McDonald’s research wrist limit frequency is 2Hz and fingers limit is never more than 4Hz. Compared to the results after filtering with cutoff frequency equal to 10Hz or 12Hz, I found the first two cases unsatisfying for the raw segmentation. Cutting the frequencies higher than this in section “1” cause too much loss of data and therefore may result in false segmentation.

Figure 20: Cutoff frequency = 10



Figure 21: Cutoff frequency = 12

## Sign properties

Each lexical item is investigated in the meaning of HamNoSys, which was discussed earlier in section 2.2.1. The aim of this module is to extract properties of a sign, which describe it from motion point of view. The idea is these properties later to be used for sign classification.

The dominant hand can be easily determined by computing the total length of trajectory for both hands and compare the two measurements. The idea is that no matter if the sign is performed by one hand or both, the dominant hand’s length of the trajectory will be longer. Total distance of each hand can also be used to determine if the sign is one or two-handed, by simply comparing both measurements to threshold. The threshold is dynamically computed over the general movement. If ????

### Hand Location

Hand location is determined for each frame relatively to the signer’s body plane. The space in front of the body is divided in regions according to HamNoSys (Figure 22 shows an example of body plane sectioning). The exact boundaries of each region are defined dynamically for each frame, by the position of body markers. During the sign, hands can have different handshapes, so in order to be more accurate, each region has margin, which is also calculated for each frame. The method also keeps statistical information about the number of changes in hand location and time spent in each region.

Figure 22: Example of hand location regions

### Hand Orientation

The task of defining Wrist and palm orientation can be narrowed down to minor tasks in linear algebra. The idea is to representing body parts as 3-dimensional vectors in a relative co-ordinate system.

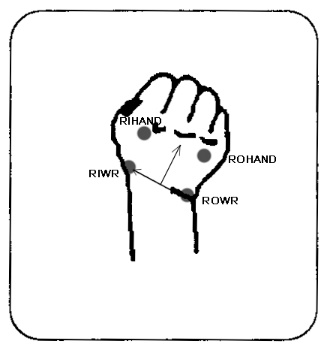
According to HamNoSys wrist orientation is notated if the wrist is bend toward the pulse or back of the hand, toward the thumb or the little finger. The task of defining the wrist bending can be solved by finding the angle between two vectors. One vector is defined by wrist markers (e.g. RIWR and ROWR), the other is defined by the middle point of segment defined of the two wrist markers and the middle point of the segment defined by the two hand markers (e.g. LIHAND and LOHAND).

Figure 23: Vectors for wrist orientation

The second mentioned vector is important, because this vector also describes the Extended Finger orientation. As it was explained earlier (see section 2.2.1.2) Palm orientation is two-step process. First has to be determined in which of the two Basic Position is the hand. This can be achieved by determining the Extended Finger orientation in the space according to the body plane.

As we know from linear algebra, a plane can be described by three non-collinear points. For example, body plane can be defined by LFSH, RFSH (markers on left and right shoulders) and one of the front waist markers (e.g. RFWT or LFWT). Therefore, two wrist markers and two hand markers are enough to define the palm plane in the space. This is useful when identifying Palm orientation.

### Handshape

Different hand shapes were discussed in section 2.2.1.1. Handshape aims to describe the complex configuration of the hand and all fingers.

At this point Handshape and Hand orientation modules are not included in the system output. They still need refinements. The description of hand orientation is tightly related to the specific handshape. A minor problem that has to be investigated more deeply is

Furthermore, during the sign handshape and orientation may change, thus leading to the question at which exact moment the handshape and orientation should be investigated.

## SVM

Support vector machines (SVMs) are powerful and flexible class of supervised algorithms for classification. Supervised means that it needs data set which has been labeled in order to train a model. The model hen can be used to cluster signs for further processing. This module is still not implemented in the tool, but it is included in the future work

# Testing

# Conclusion

The next step is to extended the number of sign features and implement the module for SVMs to cluster signs for further processing.

# Bibliography

#### T. Hanke, “Hamnosys-representing sign language data in language resources and language processing contexts," in LREC, vol. 4, pp. 1-6, 2004.

#### Stokoe, William C. (1960, 1978), “Sign language structure: An outline of the visual communication systems of the American deaf," Studies in linguistics, Occasional papers, No. 8, Dept. of Anthropology and Linguistics, University at Buffalo. 2d ed., Silver Spring: Md: Linstok Press.

#### Mihoko Kato, “A Study of Notation and Sign Writing Systems for the Deaf ", Intercultural Communication Studies XVII: 4 2008, Toyohashi University of Technology

#### J. McDonald, R. Wolfe, R. B. Wilbur, R. Moncrief, E. Malaia, S. Fujimoto, S. Baowidan, and J. Stec, “A new tool to facilitate prosodic analysis of motion capture data and a data- driven technique for the improvement of avatar motion," 2016.

#### E. Muybridge, “Muybridge's Complete Human and Animal Locomotion”. Collections of ne art in Dover books, Dover Publications, 1979

#### “Go Further with Vicon MX T-Series” Vicon Motion System Limited, rev. 1.3 August, 2010

#### N. Wheatland, Y. Wang, H. Song, M. Neff, V. Zordan, and S. Jörg, “State of the art in hand and finger modeling and animation” Computer Graphics Forum, vol. 34, no. 2, pp. 735–760, 2015

#### M. Dilsizian, Z. Tang, D. Metaxas, M. Huenerfauth, and C. Neidle, “The importance of 3D motion trajectories for computer-based sign recognition" 7th Workshop on the Representation and Processing of Sign Languages: Corpus Mining, The 10th International Conference on Language Resources and Evaluation (LREC 2016), 2016.

#### T. Hanke, S. Matthes, A. Regen, and S. Worseck, “Where does a sign start and end? segmentation of continuous signing" 5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon, Language Resources and Evaluation (LREC 2012), 05 2012.

#### Perlmutter, David M., "What is Sign Language?"LSA, 12 April 2014. Retrieved 4 November 2013.

#### Sandler, Wendy; & Lillo-Martin, Diane., “Sign Language and Linguistic Universals” , 2006. Cambridge: Cambridge University Press.

#### P. Jedli\_cka, Z. Kr\_noul, and M. \_Zelezn\_y, “Methods for recognizing interesting events within sign language motion capture data” in Universal Access in Human-Computer Interaction. Designing Novel Interactions (M. Antona and C. Stephanidis, eds.), (Cham), pp. 55-75, Springer International Publishing, 2017.

#### L. Naert, C. Larboulette, and S. Gibet, “Coarticulation analysis for sign language synthesis” in 7th Workshop on the Representation and Processing of Sign Languages, LREC, 2016.

#### [Python documentation](https://docs.python.org/3/library/index.html) - https://docs.python.org/3/library/index.html

#### “The C3D file format”, MotionLab Systems, Inc., 2002

#### F. Bevilacqua, J. Ridenour, D. J. Cuccia, “3D motion capture data: motion analysis and mapping to music”