Measuring Glide Reflection Symmetry in Human Movements

by

Chaitanya Prakash Potaraju

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved June 2017 by the
Graduate Supervisory Committee:

Pavan Turaga, Chair
Andreas Spanias
Narayanan Krishnamurthi

ARIZONA STATE UNIVERSITY

August 2017

ABSTRACT

Many studies on human walking pattern assume that adult gait is characterized by bilateral symmetrical behavior. It is well understood that maintaining symmetry in walking patterns increases energetic efficiency. We present a framework to provide a quantitative assessment of human walking patterns, especially assessments related to symmetric and asymmetric gait patterns purely based on glide reflection. A Gliding symmetry score is calculated from the data obtained from Motion Capture(MoCap) system. Six primary joints (Shoulder, Elbow, Palm, Hip, Knee, Foot) are considered for this study. Two different abnormalities were chosen and studied carefully. All the two gaits were mimicked in controlled environment. The framework proposed clearly showed that it could distinguish the abnormal gaits from the ordinary walking patterns. This framework can be widely used by the doctors and physical therapists for kinematics analysis, bio-mechanics, motion capture research, sports medicine and physical therapy, including human gait analysis and injury rehabilitation.

Dedication

I dedicate this thesis work to my parents and Madhuri, for without their love, support and encouragement, this would not have been possible.

Acknowledgment

There are several people who have contributed towards this thesis. First, I would like to thank my thesis advisor Dr. Pavan Turaga, for showing confidence in me and for giving me the opportunity to work in his lab. His knowledge, guidance and constant support have kept me focused during the duration of my thesis. I would also like to thank the other members of my thesis committee, Dr. Narayanan Krishnamurthi and Dr. Andreas Spanias, for their invaluable inputs and for taking time off their schedule to serve on my committee. I am also grateful to Qiao Wang, Ankita Shukla and Kowshik at ASU for introducing me to the area of research and helping me finish my thesis. I am thankful to my graduate advisor, Sno Kleespies from the department of Electrical Computer and Energy Engineering at ASU, for helping me make my stay at ASU, a comfortable one. This thesis would not have been possible without the love, guidance and unconditional support of my parents Chandrasekhar and Radha Sekhar.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Chapter 1

INTRODUCTION

Gait analysis is quite a long-studied topic with many tools and methodologies which are both mathematically advanced and practically useful for practitioners. In a very general sense there has always been a trade-off between required accuracy and algorithmic simplicity while using methods in gait analysis. Accuracy can be achieved through more sophisticated algorithms, but many such approaches limit the utility of systems to offline or post-hoc analysis. There is a significant need to develop methods that can provide real-time feedback on movement qualities, which can be used for the purposes of re-training. Examples of feedback approaches include visual, sonic, or haptic feedback. In this thesis, we explore the notion of space-time symmetries in human walking, and propose mathematically well-grounded approaches for quantification of the said symmetries, while also operating at or near real-time, such that this can be used in real-time feedback systems.

Gait is a way in which we move our whole body from one point to another. Most often this is done by Walking, running etc. Gait analysis is a method used to assess the way we walk to highlight biomechanical abnormalities. It is important for one to efficiently move around to avoid injuries that can affect kinetics and Kinematics of the walking system. If due to any injury or a disease, the range of motion in any joint is limited, the body then finds another optimal way of moving resulting in the biomechanical abnormalities in Humans. Gait analysis is performed by a professional, such as Physiotherapist for analyzing the abnormality present in the patient. Even today there are many clinical environment where gait abnormalities are found by either watching the patient walk across the pathway or video record the patient while

they walk on a treadmill and analyze based on freeze frames of the video. These days there is a widespread demand for such systems in the field of sports. They train to have a more symmetric pattern in their movements. Many sports shops are now using the equipment and staff trained in gait analysis. Most gait analysis usually involves walking or running on treadmill or over-ground followed by detailed analysis based on the equipment used.

The more energy efficient walking patterns is usually the symmetric. Let us consider the trajectories of a palm in healthy humans while they are waking. Clearly the trajectory of left palm is spatially symmetric with the trajectory of right palm over different time windows. The main idea of this thesis work is to present a framework that can quantify the asymmetry between the two gait cycles using the whole trajectories of corresponding joints and investigate if these features demonstrate adequate sensitivity for various abnormal gait analysis.

Let us consider the patterns of a palm in healthy humans while they are waking. When we try to understand it, our brain analyses the flow of your left palm to be symmetric with the right palm. Clearly the left palm is spatially symmetric over different time windows. It is also intuitive to understand that maintaining such symmetry during walking helps maintain the balance and increase energy efficiency. When we observe a person with injury that inhibits the movement of one side of the body, we are able to understand that he/she has abnormality in one side, since he/she will not be able to maintain symmetry in their walking style. The main idea of this thesis work is to present a framework that can quantify the asymmetry between the two gait cycles using the whole trajectories of corresponding joints and investigate if these features demonstrate adequate sensitivity for various abnormal gait analysis.

1.1 Background

1.1.1 Glide Reflection

In mathematics, we are aware of affine transformations like Rotation, Translation, Reflection, Scaling. However in real life scenarios, it is difficult to consider a few patterns as in Figure 1.1 to be symmetric in some sense. Our brain perceives such patterns to be symmetric and soothing and there lies a challenge to use the simple transformation available to mathematically define such symmetries. These patterns show what is usually referred to as Glide reflection symmetry.

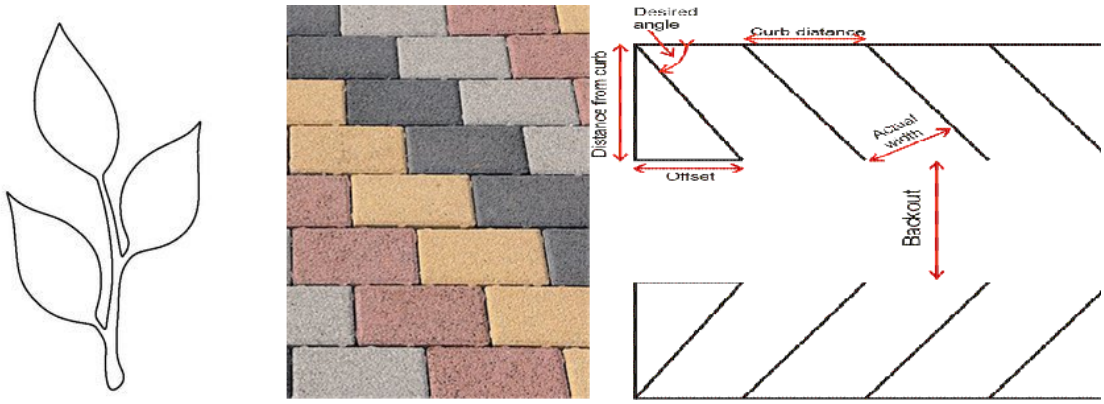


Figure 1.1: Examples of glide reflection symmetry.

For better understanding let us consider Figure 1.2, ΔABC in red is a triangle in $\mathbb{R}^2$. $\Delta A'B'C'$ can be constructed from ΔABC using a simple reflection transformation about the line shown. Similarly $\Delta A"B"C"$ in blue can be constructed from $\Delta A'B'C'$ using a simple translation transformation. When we combine both the two transformations we can traverse from ΔABC to $A"B"C"$ and such transformation is referred to as Glide reflection transformation. Glide reflection symmetry detection based on feature point matching was proposed in Lee and Liu (2012). Feature points are found in sub-images using canny edge detector followed by SIFT descriptor. Re-

flection axis ϕ_{axis} and translation T_{ij} are found separately and then combined to find the glide reflection. In our approach we use a similar technique to achieve the glide reflection for the whole trajectory.

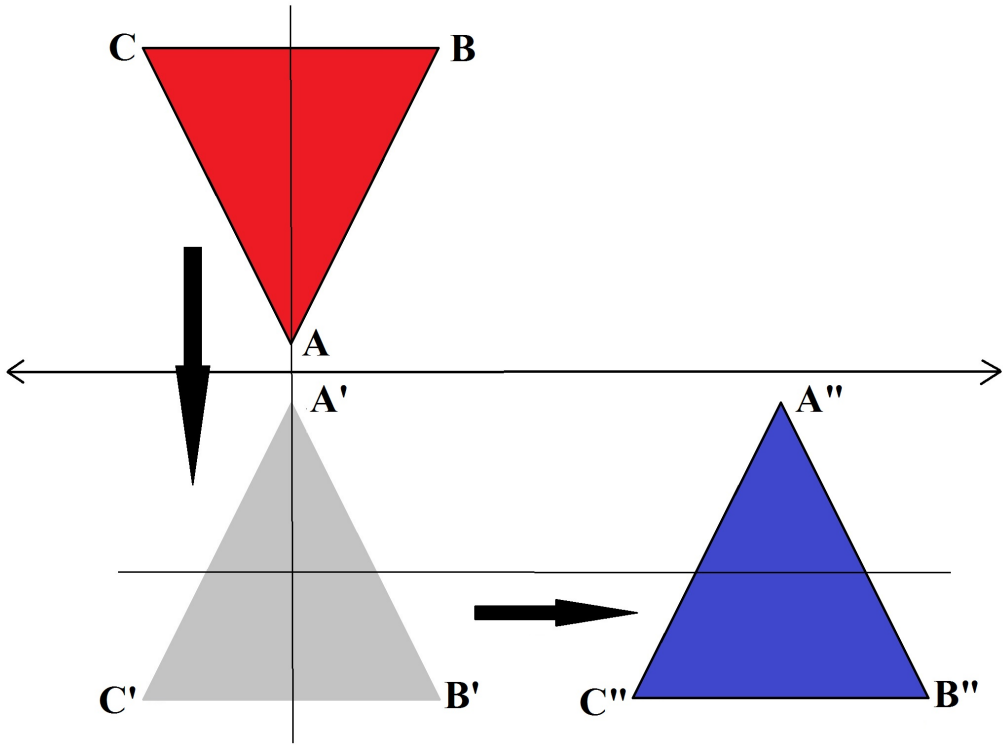


Figure 1.2: Demonstration of Glide reflection Symmetry.

1.1.2 Gait analysis

Now, We consider one left gait cycle starting from heel strike of left foot to the next heel strike of the same left foot. For better understanding let us consider Figure1.3. We can consider one complete left gait cycle starting from heel strike at t_1 to the next heel strike at t_3. We can also refer to one complete left gait cycle starting from toe-off at t_2 to the next toe-off at t_4. There are many other features of a gait cycle that can be considered such as upper body movement, twists in the torso and many more. However, we shall limit our work to the six main joints in our body: Shoulder,

Elbow, Palm, Hip, Knee, Foot. We define symmetry based on the relative positions of the feature points of every joint. Let us consider all the feature points of the foot we have as shown in Figure1.3. We can compare the left gait cycle starting from the heel-strike at t_1 to toe-off at t_2 with the right gait cycle starting from the heel-strike at t_2 to toe-off at t_3 by translating the feature set of left gait cycle in time by $(t_2 - t_1)$ and then reflecting it about the median plane of the body. Analysis can also be based on stance phase and swing phase of the gait cycle. However we will stick to one complete gait cycle as discussed. More about the temporal glide will be discussed in later section.

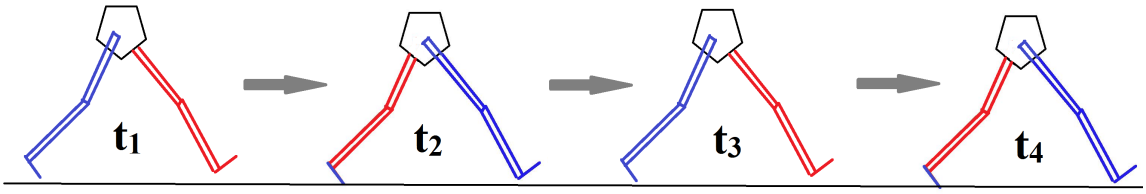


Figure 1.3: Demonstration of Glide reflection Symmetry.

1.1.3 Symmetry Index

Spatiotemporal parameters of a gait are usually compared using few ratios. Spatio temporal characteristics include step length, step duration, stance phase, load response, single support, pre-swing and swing phase. In Błażkiewicz *et al.* (2014) there are four important symmetry indices that are compared with a simple assumption made about the ratio of respective parameters.With X_r and X_l being the feature of right and left joints where features can be stride length, single limb support etc. It is assumed that $X_r \leq X_l$. This assumption is useful to force the ratio $X_r/X_l \leq 0$. Here three symmetry indices namely: Ratio Index (RI), Symmetry Index (SI) and Gait asymmetry (GA) are considered. Ratio Index (RI): This factor helps us identify

the variable that has higher value causing asymmetry. The RI value greater than 100% indicates the asymmetry. This is a very good index that can be used to identify the side in which asymmetry is caused in people with disability. The equation used for finding RI is given by

$$RatioIndex(RI) = \left(1 - \frac{X_R}{X_L}\right) \times 100\% \tag{1.1}$$

Symmetry Index (SI): This reflects the differences between kinetics and kinematic properties of any gait. This is the most commonly used and most cited index for gait assessment. SI is given by

$$SymmetryIndex(SI) = 2\left(\frac{X_L - X_R}{X_L + X_R}\right) \times 100\% \tag{1.2}$$

Gait Asymmetry is the logarithmic transform of RI and is commonly used for comparing time of different phases in the gait. $GA \geq 100\%$ indicates asymmetry. It is given by:

$$GaitAsymmetryIndex(GA) = ln\left(\frac{X_R}{X_L}\right) \times 100\% \tag{1.3}$$

GGA or global gait assymetry index Cabral *et al.* (2016) was computed using three dimensional joint angles of the hip, knee, ankle, trunk in relation to pelvis angles. In this approach the right and left gait cycles are normalised to have 101 equally distributed samples for every gait cycle. GGA is then computed as:

$$GGA = \sum_{v=v_1}^{v_{15}} \sqrt{\sum_{t=t_1}^{t_{101}} \Big(x_l(t) - x_r(t)\Big)^2} \tag{1.4}$$

Restricting the number of data samples is undesirable in most cases. All the above indices were compared in Błażkiewicz *et al.* (2014). However, when we see the parameters chosen to evaluate symmetry, it does not consider the irregularities happening inside the gait cycle. These indices can be used to find the side of the body that is

affected. A more robust and mathematical approach is in order for gait assessment and this thesis work tries to bring forth one such framework.

Gait symmetry is usually assessed based on stride length, velocity profiles, step length, single limb support, trunk movement patterns, accelerometers based balance evaluation and ground reaction forces. The use of triaxial accelerometers with other sensors has become more common since it is easily adoptable both in clinical environments and homes or elsewhere of patients interest. iGAIT is a software package available in MATLAB and uses accelerometer data. Symmetry can then be derived into a simple text file. Yang *et al.* (2012) gives a detailed description of the software package. A smartphone in which tri-axial accelerometers was attached to the back of the participants and the data was collected to analyze their gaits. In Yang *et al.* (2010) 21 gait features were extracted from the data collected from Complex Regional Pain Syndrome(CRPS) and based on SNR(Signal to Noise ratio) ranking, few features were given as an input to a multi-layer neural network and were able to achieve around 97 accuracy in discriminating the CRPS patients. Electronic pathways have also gained considerable interest with the availability of transportable walkways and descent costs. Initially the variables like the step length were assessed manually. The patient was asked to cross a grid patterned walkway. The observer or physician walked behind the patient calling out heel strike location from the grid into a tape recorder. This methodology was then improved by having a soft heel counters with dye fixed at the feet. And readable marks on paper walkways were used for analysis. These time consuming methods were then replaced with electronic pathways that can record the position and time of foot strike and take off. Companies like GAITRite systems provide these devices at affordable prices. Ground reaction forces acting on foot during standing or walking are measured using force plates. They give us the force vector: vertical and two other loads acting on the surface of the plate

surface. Vaverka *et al.* (2015) proposes a kinetic analysis based on the assessment of the ground reaction forces (GRF) recorded on two force plates during the stand phase of both left and right gait cycles. Initial contact is considered when the vertical GRF exceeds 5% of the body weight and toe-off is considered as the instant when the vertical component of GRF goes below 5% of the body weight. A symmetry index is then computed using force-time variables and other symmetry index available. Motion analysis based on 3D camera system like opti-track system are also gaining interest for research work. A set of 9 or more cameras are calibrated to understand their relative positions among each other and with respect to ground. Reflective markers are placed on the subject and then the body is tracked as the subject moves around on the stage. Zhang *et al.* (2011) uses one such system for walking pattern analysis. They propose a fourteen-linkage model for optimal analysis. Our thesis work uses a similar model but with small changes. In the proposed methodology, they use motion capture system in conjunction with tri-axial accelerometer unit along with all other markers. Various features are extracted from the data and effect of aging on cycle stability is studied. This paper used Dynamic Time Warping to calculate the similarity, which is a method for flexible pattern-matching algorithm.

The main contributions in this thesis work is to compare three optimal methods and finally conclude on the framework which can be used to optimally distinguish abnormal gaits from normal walking patterns. We also try to minimize the use of various mechanical devices on the subject and use simple reflective markers on subjects to track the joints of interest.

Chapter 2

OVERVIEW OF PROPOSED MATHEMATICAL FRAMEWORK

This chapter details the steps involved in building the framework. There are many multivariate algorithms that combine non-optical data such as electromyographic, kinematics data with optical data for achieving high accuracy. We intend to use only the 3D position data of six main joints: Shoulder, Elbow, Palm, Hip, Knee, Foot. Finding asymmetry in the joints involves four stages: (1) capturing and normalizing the data collected from the Motion Capture system like the opti-track systems, (2)segmenting each left and right gaits, (3)finding the sagittal or the median plane to divide the body into left half and right half of the body, (4)reflecting the left gait about the plane of symmetry and finding the symmetry scores over time. In the subsequent sections we will be discussing each stage in detail

2.1 Fourteen Linkage model and Pre-processing of data

A good human model for a gait analysis should be simple, but must also capture the most important features of most of the walkers if not all. We have used the Fourteen Linkage model Zhang *et al.* (2011) shown in the figure 2.1 where p_i is the marker placed on the body. We do not consider the markers p_2, p_4, p_1, p_3 as they give details about the whole structure of the foot. We consider the whole foot using a single marker. We can use multiple markers to define a joint for better analysis. However since this work mainly aims at building a prototype, we stick to a more basic model as explained earlier/ Figure 2.2 shows us the trajectory of palm and foot over time as the subject moves around the stage. Our framework tries to use only the 3D positions of the joints.We export the data into a csv file and at this stage we have a

continuous data stream that needs to be normalized and segmented into gait cycles. At this stage we have raw data that needs some preprocessing before we can evaluate the symmetry. The next stage speaks about data preprocessing.

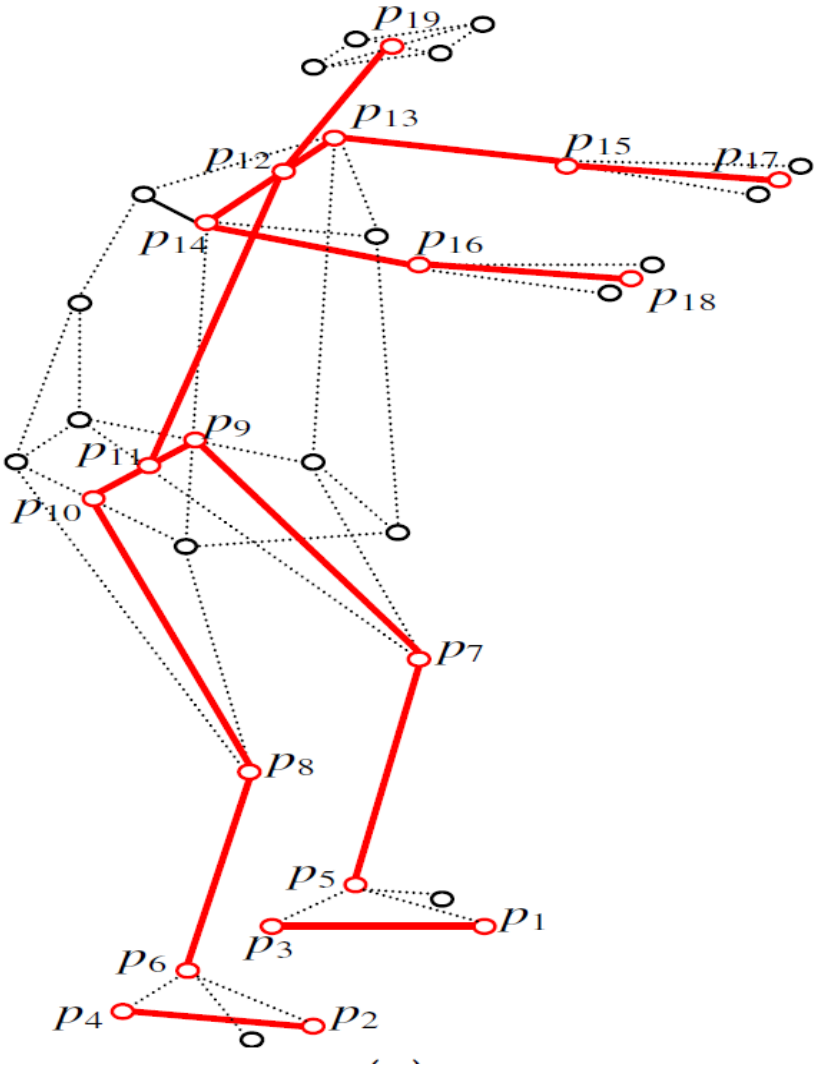


Figure 2.1: Fourteen Linkage Model taken fromm Zhang *et al.* (2011)

2.2 Data Preprocessing

Data preprocessing is a critical issue for data analysis, as real-world data tend to be incomplete, noisy, and inconsistent. Motion capture system might not be able

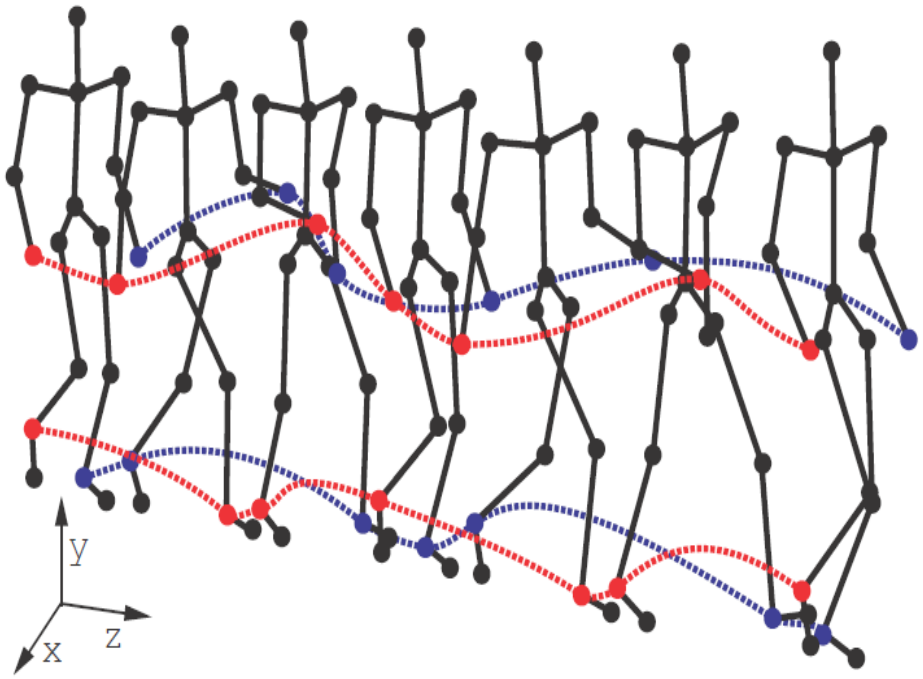


Figure 2.2: Motion capture data. Seven selected video frames of a walk sequence contain 3D coordinates of each joint in time. The red and blue lines track trajectories of hands and feet. Sinha *et al.* (2013)

to capture the 3D positions of any marker always which leads us to have a data preprocessing stage which involves interpolation. Although numerous methods of data preprocessing have been developed, data preprocessing remains an active area of research, due to the huge amount inconsistent or dirty data and the complexity of the problem. Data preprocessing includes data cleaning, data integration, data transformation, and data reduction. Figure 2.3 shows us the trajectory of left foot along Y-axis and green '+' signs indicate the interpolated data for the missing data points along the trajectory of the raw data indicated by red lines. Many interpolation methods could be used, such as nearest neighbor interpolation, linear interpolation, cubic spline interpolation, piecewise cubic Hermite interpolation and N-th degree

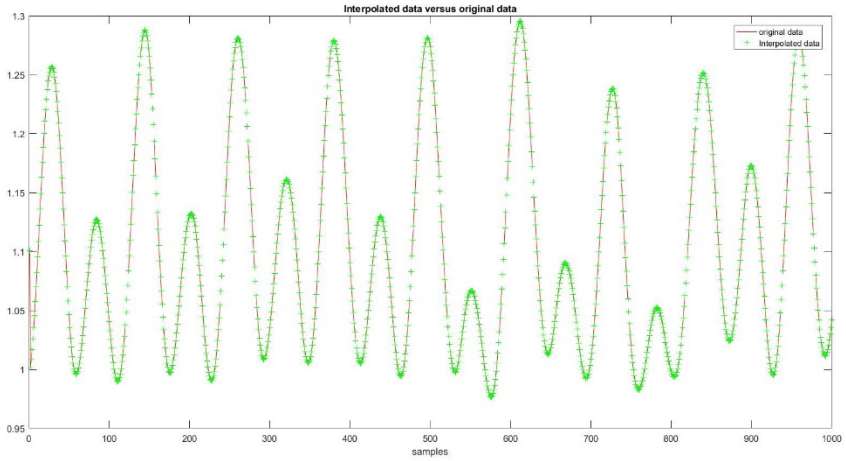


Figure 2.3: Interpolated data vs raw data

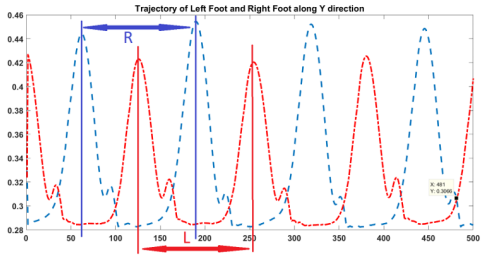
polynomial interpolation. We choose the spline method to interpolate the missing data because the cubic spline interpolation is a piecewise continuous curve, passing through each of the values in the source data. Secondly, we need to have a comparable data in all time and so we must take care of scaling and translation problem. Translation issue is dealt by finding the relative positions of every other joint with a particular joint. We then have to scale the joint suitably. Many scaling techniques have been used in various research studies. Here we scale the joints data in between 0 and 1.

2.3 Segmentation of the Data

Once the data has been collected and normalized, we need to segment the data into corresponding gait cycles so that the consecutive left and right cycles can be compared to generate the score. One way of dealing this issue is to have a slight elevation in the direction of walking. We had a 5cm block placed at the front end of the treadmill as shown in figure 2.4a. This way the foot always tends to achieve a peak in its trajectory along Y axis. Figure 2.4b shows the trajectory of left and right

foot along Y axis. We can use the peaks in the trajectory to define the start and end of every gait cycle. We can also achieve this task by integrating force plates or electronic pathways. However, our main goal is to minimize the equipment interaction with the subjects. In figure 2.2, we can see the trajectory of right and left foot in red and blue respectively. This method can induce some extra error into assessment but takes away the need of having another device on the subject and help evaluate the gaits with minimal feature set.

(a) Elevation in the treadmill to help in segmentation of data.



(b) Trajectory of the Left and right foot along Y axis. (Left foot trajectory is shown in red and Right foot trajectory is shown in blue.)

Figure 2.4: Examples of possible warping paths that can violate the conditions mentiond above

When we look into a continuous scores then having the peaks will help us determine the glide. Let us consider the 1st peak in left gait cycle to be L_{p1} and the 1st peak in the right gait cycle be R_{P1}. The temporal glide between the two cycles or the number of frames by which we need to temporally glide the left gait cycle to make it comparable to the right gait cycle is calculated as

$$Glide(G) = \left| L_{P1} R_{P1} \right| \tag{2.1}$$

At this stage we have the normalized, interpolated and segmented data. More details about how we can segment the data for a continuous score in subsequent sections. As discussed before, our similarity score or symmetry score should compare a segment of trajectory on one side with the spatially reflected and temporally glided segment on the other side. Using the above Glide we can move the trajectory temporally. However, the next task in hand is to determine the plane about which the trajectory needs to be spatially reflected. Next section speaks about the different planes of motion and the most significant plane to be considered.

2.4 Plane of symmetry

There are three different planes of motion: Coronal, Transverse and Sagittal. For more refer to Source. The coronal plane divides the body into front and back. When we move along this plane, we are moving toward or away from the mid-line. Adduction and abduction are movements along this plane. Many of our daily movements and exercises involve very little abduction. The transverse (or horizontal) plane divides the body into top and bottom, but it is a little less straightforward. Any time we rotate a joint we are moving along the transverse plane. In daily life, this is the action we do least frequently, particularly with the large joints in the hips, shoulders, and spine. The median plane is that sagital plane bisecting the body vertically through

the navel, dividing the body exactly in left and right side. When we move along this plane, we are using the strength of our muscles to move parts of the body forward or backward. Extension and flexion happen along the sagittal plane. This means most walking, running, biking, rowing, and lifting movements make use of this plane. Figure 2.5 shows the possible planes of motion.

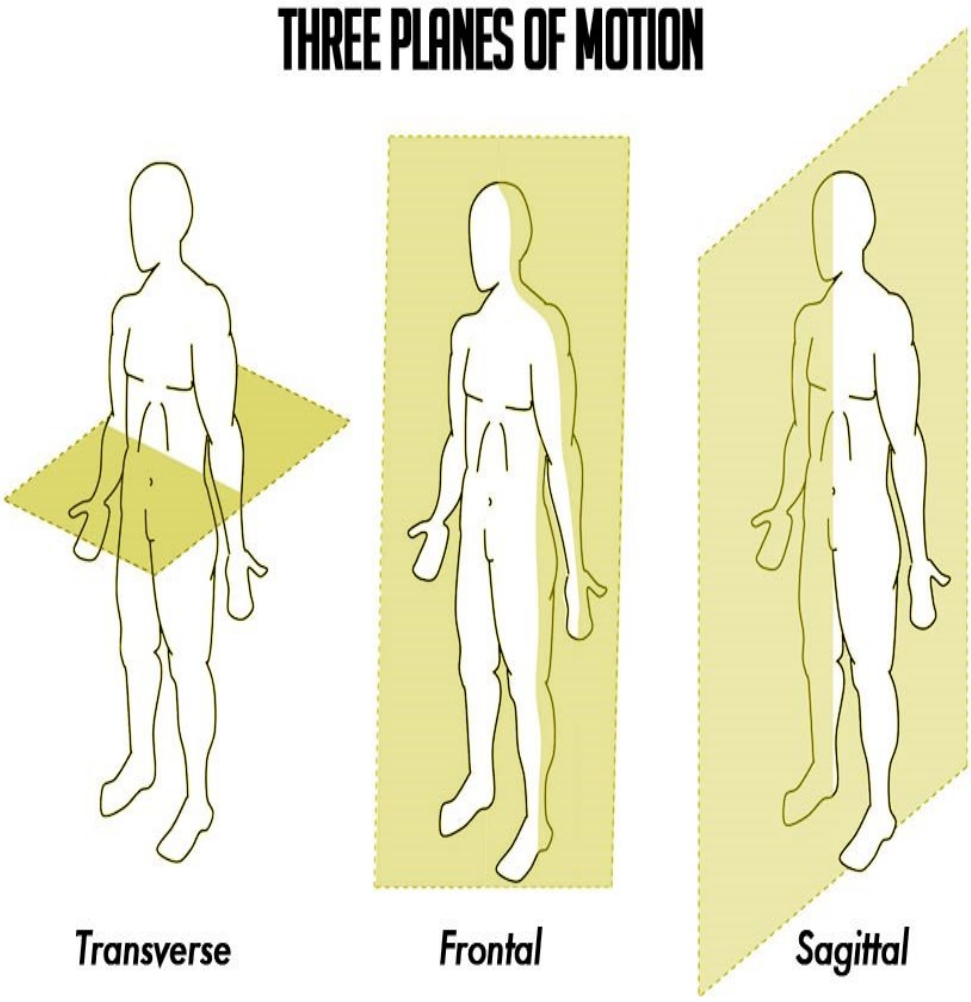


Figure 2.5: Three different planes of motion

This is a very important part of the framework. We need to find the plane about which the trajectory of any joint need to be reflected to its corresponding joint in

the other side of the body. One way to approach this problem is by considering it as a two-class or binary classification problem. We try to come up with a hyper-plane that can divide the body into exatly two halves: Left body and Right body, with some simple assumptions. One of the most important assumption made is that the plane is always perpendicular to the ground. The second assumption made is that the plane needs to pass through the center of the hip. In other words, med-sagittal plane or the median plane is what we are formulating. A simple gradient descent is used to find the parameters of sagittal or the median plane.

2.5 Gradient Descent

Gradient descent Ruder (2016) is one of the classical algorithms available to minimize any given function with certain constraints levied on the parameters. Given a function defined by a set of parameters, gradient descent starts with an initial set of parameter values and iteratively moves toward a set of parameter values that minimize the function. This iterative minimization is achieved using calculus, taking steps in the negative direction of function gradient.

Let us consider an objective function $J(\theta)$ parameterized by a models parameters $\theta \in \mathbb{R}^d$, let $\nabla_\theta J(\theta)$ be the gradient of the objective function, and let us have a learning rate η. We shall look at a few flavors of gradient descent before we can implement it in our algorithm. Based on how much data we use to compute the gradient of the function, there are three main variants of gradient descent, namely: Batch gradient descent, Stochastic gradient descent, Mini-batch gradient descent. Batch gradient descent, computes the gradient of the cost function w.r.t. to the parameters θ for the entire training dataset:

$$\theta = \theta - \eta \cdot \nabla_\theta J(\theta) \tag{2.2}$$

We update our parameters in the direction of the gradients with the learning rate

determining how big of an update we perform. As we need to calculate the gradients for the whole dataset to perform just one update, batch gradient descent can be very slow and consumes more memory. Batch gradient descent also doesn't allow us to update our model online, i.e. with new examples available. Batch gradient descent is guaranteed to converge to the global minimum for convex error surfaces and to a local minimum for non-convex surfaces.

Stochastic gradient descent (SGD) performs a parameter update for each training example $x^{(i)}$ and label $y^{(i)}$. The parameters are updated in a similar fashion as below:

$$\theta = \theta - \eta \cdot \nabla_\theta J(\theta; x^{(i)}; y^{(i)}) \tag{2.3}$$

Batch gradient descent performs more number of computations for large datasets, as it computes gradients for similar examples before each parameter is updated. Unlike Batch gradient descent, SGD reduces these redundant computations by performing one update at a time. It is therefore usually much faster and can also be used to for real time applications. SGD performs frequent updates with a high variance that cause the objective function to fluctuate heavily as in Figure 2.7.

While batch gradient descent converges to the nearest minimum of the convex surface the parameters are placed in, SGD's fluctuation, on the one hand, enables it to jump to new and potentially better local minima. However, this action of SGD might not be desired as it can keep overshooting and resulting in unstable values. This behavior can be handled when we slowly decrease the learning rate making SGD show the same convergence behavior as batch gradient descent, almost certainly converging to a local or the global minimum for non-convex and convex optimization respectively.

Mini-batch gradient descent finally takes the best of the above two methods and

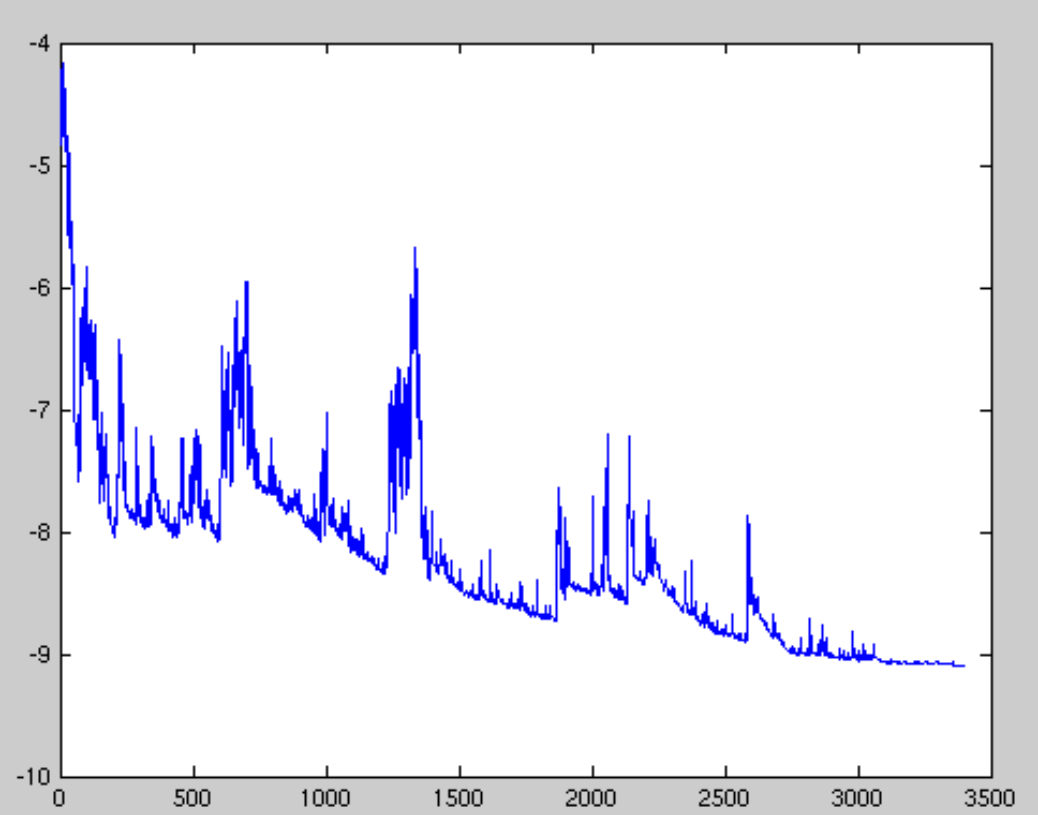


Figure 2.6: SGD Fluctuations. (source: Wikipedia)

performs an update for every mini-batch of n training examples:

$$\theta = \theta - \eta \cdot \nabla_\theta J(\theta; x^{(i:i+n)}; y^{(i:i+n)}) \tag{2.4}$$

This way, it a) reduces the variance of the parameter updates, which can lead to more stable convergence; and b) can make use of highly optimized matrix optimizations that make computing the gradient w.r.t. a mini-batch very efficient. Common mini-batch sizes range between 50 and 256, but can vary for different applications. We use this kind of gradient descent approach to find optimal values of the median plane.

Challenges while implementing gradient descent can be: Choosing a proper learning rate can be difficult. A learning rate that is too small leads to painfully slow

convergence, while a learning rate that is too large can hinder convergence and cause the loss function to fluctuate around the minimum or even to diverge. Additionally, the same learning rate applies to all parameter updates. If our data is sparse and our features have very different frequencies, we might not want to update all of them to the same extent, but perform a larger update for rarely occurring features. Another key challenge of minimizing highly non-convex error functions common for neural networks is avoiding getting trapped in their numerous suboptimal local minima and saddle points, i.e. points where one dimension slopes up and another slope down. Saddle points are usually surrounded by a plateau of the same error, which makes it notoriously hard for SGD to escape, as the gradient is close to zero in all dimensions.

The next step is to define the objective function $J(\theta)$. Since our main target is to find the median plain about which one side of the body can be spatially reflected to the other side of the body, our objective function should be a plane that can separate the two sides of the body. We can consider any binary classifier to find the optimal plane of symmetry. Intuitively, we must have the labels $y \in 0, 1$ where the data from the left side of body is labelled 0 and right side of the body is labelled 1. Thus our $J(\theta)$ takes the sigmoid function as shown below.

$$J(\theta) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}, where g(x) = \frac{1}{1 + e^{-x}} \tag{2.5}$$

$g(x)$ is called the logistic function or the sigmoid function. Notice that g(x) tends towards 1 as $x \to \infty$ and g(x) tends towards 0 as $x \to -\infty$. This bounds our objective function $J(\theta)$ between 0 and 1. More about sigmoid function and its useful properties can be found in Ng (2000).

Using the above algorithm, we can find the parameters for the plane of symmetry. Spatial reflection can be derived using the Householder transformation Householder (1958) given by $H(u) = I - 2uu^T$ where u is a unit vector in $\mathbb{R}^3$, we can flip the

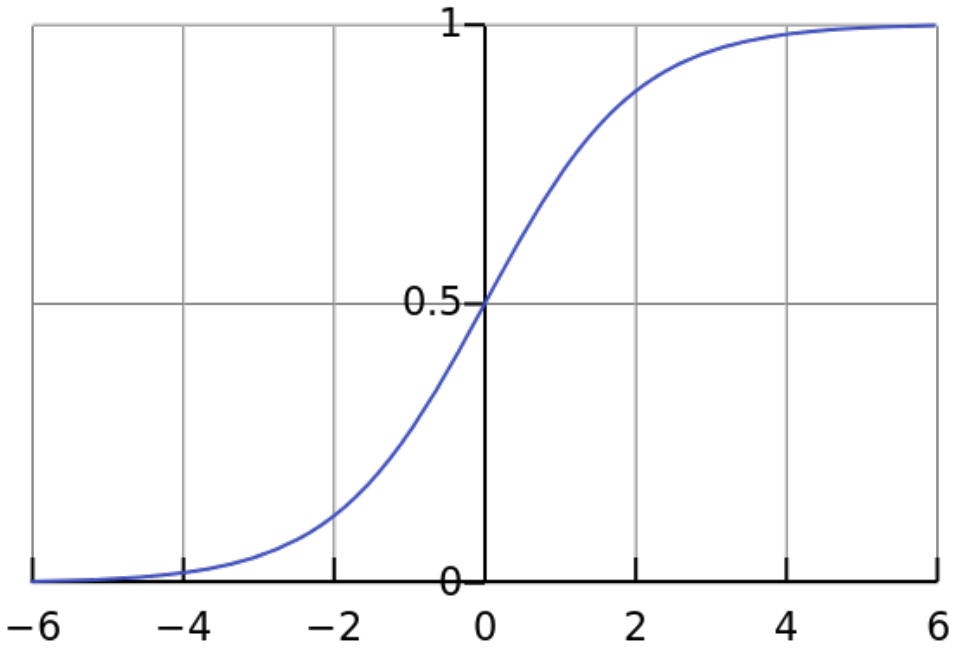


Figure 2.7: Plot of sigmoid function. (source: Wikipedia)

trajectories of left joints to the corresponding trajectories of right joints and then compare the two sequences: right gait cycle and flipped left gait cycle in some sense. For better understanding of formulation of the median plane and understanding how householder transformation can be used to bring the two sequences as close as possible. In Figure 2.8 we have the trajectories of all the six joints. we also have a black hyper plane formulatted using the above technique. The set of all points below the median plane belongs to the joints of the left side of the body and the set of all points above the median plane belongs to the right side of the body. Figure 2.9 gives us the spatially reflected trajectories of the left joints towards the trajectories of the right joints. The ones in red belong to th spatially reflected joints from the left side of the body and the ones in blue belong to the joints from the right side of the body.

To begin with we can have a fixed number of samples in the trajectory by interpolating each gait cycle to have 160 frames or sample. We then take Euclidean distance

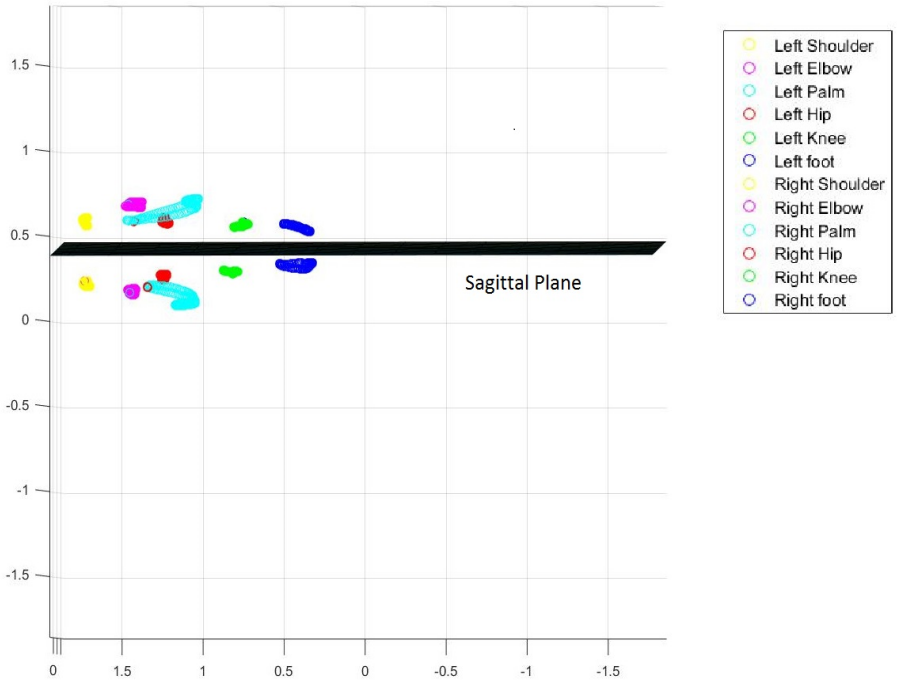


Figure 2.8: Trajectory of Left and right joints along with the sagital plane.

between the two as shown in figure 2.10. We refer to this score as the A_E which is computed as:

$$\sqrt{\sum_{t=t_1}^{t_1 60} \left(x_l(t) - x_r(t)\right)^2} \tag{2.6}$$

It is intuitive that such a score cannot be used as it is vulnerable to errors while comparing the symmetry.

Keeping the above as a benchmark, we trying to use two mapping techniques. If we consider the data, time period of left gait cycle and right gait cycle might not be same at all times and interpolating the data to a fixed length might not seem a good idea. So we need a mapping function that can map every frame in left gait cycle to corresponding frame in right gait cycle and then find the Euclidean distance to reflect the symmetry. Dynamic time warping can help us map the two sequences of unequal

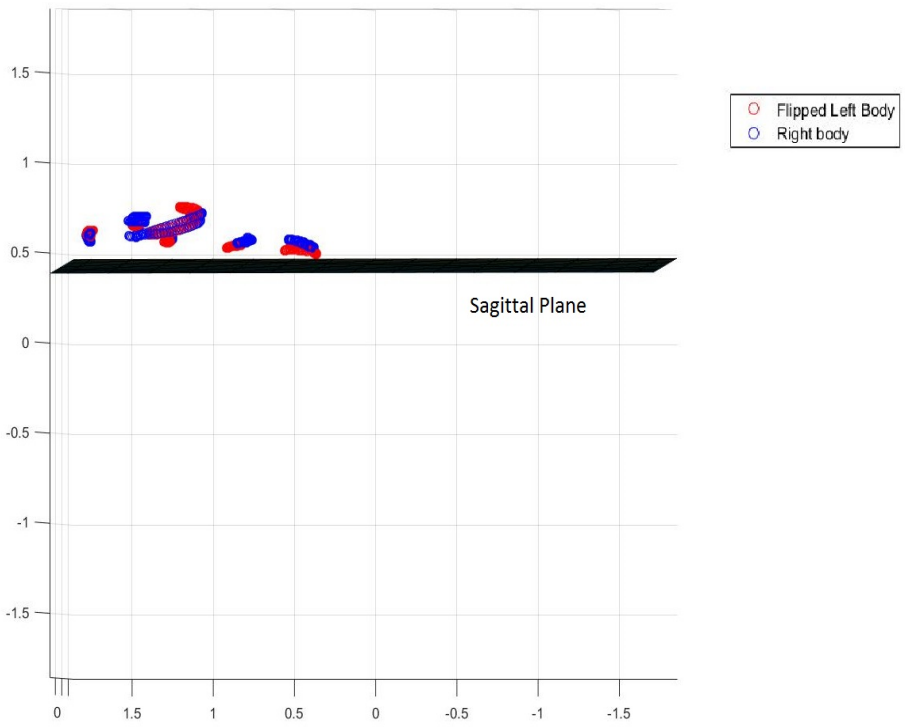


Figure 2.9: Trajectory of flipped Left(in red) and right joints(in blue) along with the sagital plane.

length optimally.

2.6 Dynamic Time Warping

Dynamic time warping (DTW) is a well-known technique to find an optimal alignment between two given (time-dependent) sequences under certain restrictions (Figure 2.14). Intuitively, the sequences are warped in a nonlinear fashion to match each other. Originally, DTW has been used to compare different speech patterns in automatic speech recognition. In fields such as data mining and information retrieval, DTW has been successfully applied to automatically cope with time deformations and different speeds associated with time-dependent data. In this section we shall discuss the main ideas of classical DTW.

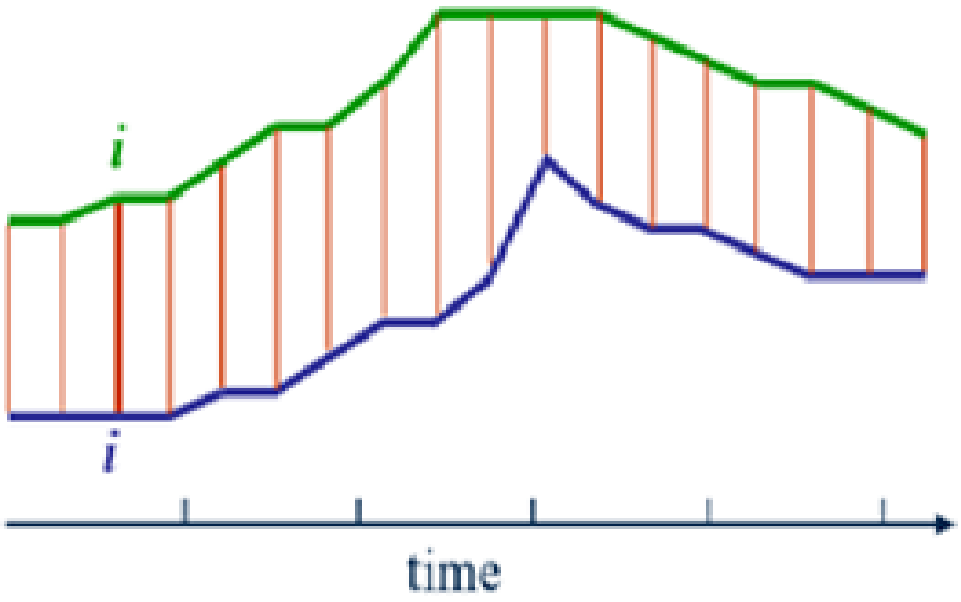


Figure 2.10: Baseline method

The objective of DTW is to compare two time-dependent sequences $X := (x_1, x_2.....x_N)$ of length $N \in \mathbb{N}$ and $Y := (y_1, y_2.....y_M)$ of length $M \in \mathbb{N}$. These sequences are feature sequences that are sampled at a equidistant points in time. In the following we fix a feature space denoted by F. then $x_n, y_m \in F for n \in [1 : N]$ and $m \in [1 : M]$. To compute the distance between X and Y, we need a distance measure which is given by a function $c : F \times F \to \mathbb{R}_{\geq}$ Typically, $c(x, y)$ is small if x and y are similar to each other, and otherwise large if they are different. One can obtain the cost matrix $C \in \mathbb{R}^{N \times M}$ defined by $C(n, m) := c(x_n, y_m)$.

Then the goal lies in finding the optimal path using the cost Matric C. This is the warping phase of DTW. An (N, M)- warping path is a sequence $p = (p_1, ..., p_L)$ with $p_l = (n_l, m_l) \in [1 : N] \times [1 : M]$ for $l\ in [1 : L]$ fsatisfying the three conditions.

- Boundary condition: $p_1 = (1, 1)$ and $p_L = (N, M)$.

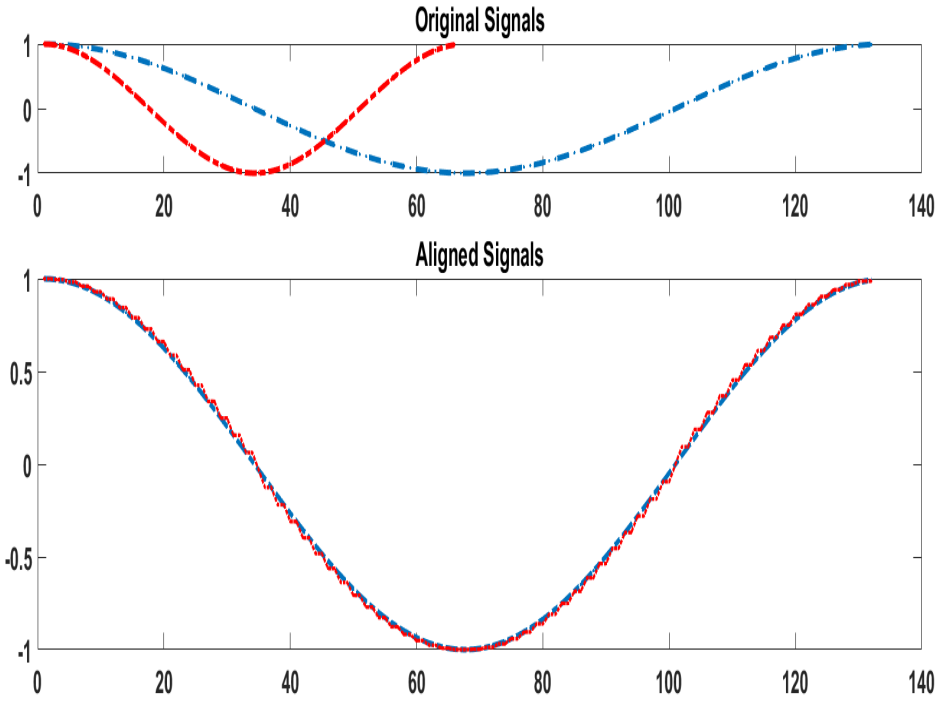


Figure 2.11: Time alignment of two time-dependent sequences.

- Monotonicity condition: $n_1 \leq n_2 \leq ... \leq n_L$ and $m_1 \leq m_2 \leq ... \leq m_L$
- Step size condition: $p_{l+1}p_l \in (1,0),(0,1),(1,1)$ for $l \in [1:L1]$

An (N, M) -warping path $p = (p_1, ..., p_L)$ defines an alignment between two sequences $X := (x_1, x_2.....x_N)$ and $Y := (y_1, y_2.....y_N)$ by assigning the element x_{n_l} of X to the element y_{m_l} of Y. The boundary condition enforces that the first elements of X and Y as well as the last elements of X and Y are aligned to each other. In other words, the alignment refers to the entire sequences X and Y . The monotonicity condition enforces that when an element in $x_n \in X$ is mapped to an element $y_m \in Y$ then we can map $x_{n+1} \in X$ to any element in Y that occurs after y_n and cannot be mapped to elements before y_n. One element in X can also be mapped to multiple elements in Y or in other words we can have many to one mapping between the

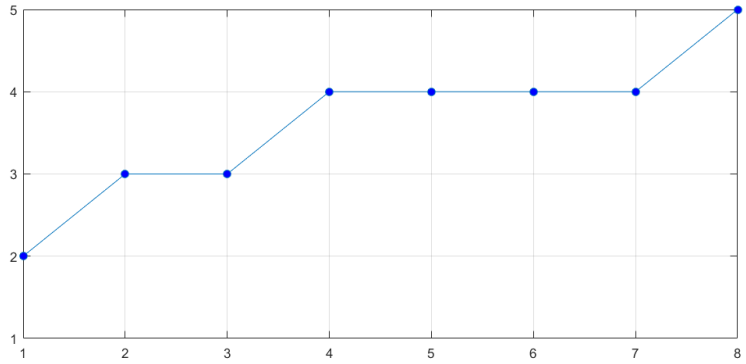
two sequences. The step size condition makes sure every element in X has a unique mapping to elements in Y. It also means that the mapping is pairwise distinct. Step condition makes sure that there is no element in X and Y that are not mapped at all. For illustration, let us consider the mapping and warping paths shown in figure 2.12. Consider the figure 2.12a, we see that the first element in Y is not mapped to first element in X violating the boundary condition. In figure2.12b, we see that the slope of the mapping function is going negative when $x_6 \in X$ is mapped to $y_2 \in Y$ violating the monotonicity condition. In figure 2.12c, the second element in Y is not mapped at all violating the step size condition. Looking into figure 2.13, we can see that the warping path shown has not violated the above conditions, and so this is an optimal warping path for the two given sequences. Let us consider the cost matrix built between the two sequences X and Y as in figure 2.14, the red boxes with black dashed lines indicates the optimal warping path that starts at the first element of both the sequences and ends at the first element of both sequences.

The total cost $c_P(X, Y)$ of a warping path p between X and Y with respect to the local cost measure matrix c is defined as,

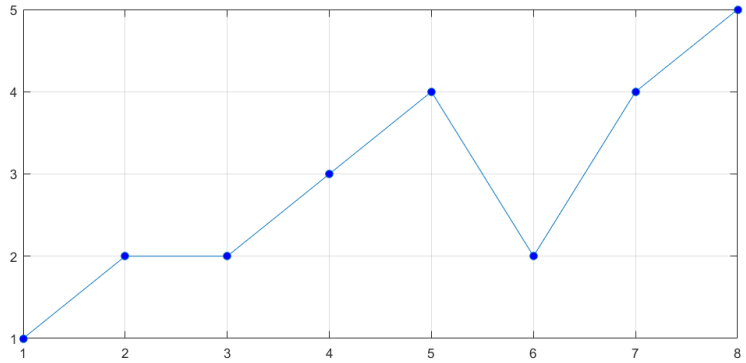
$$c_p(X, Y) := \sum_{n=1}^{L} c(x_{n_l},\ y_{m_l}).$$

Furthermore, an optimal warping path between X and Y is a warping path p^* having minimal total cost among all possible warping paths. The DTW distance DTW(X, Y) between X and Y is then defined as the total cost of p^*:

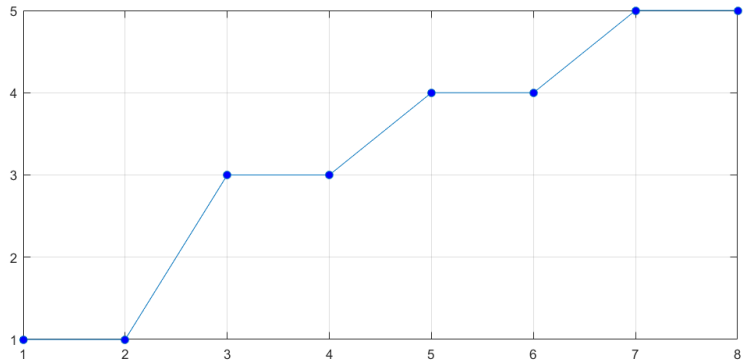
$$DTW(X,\ Y) = \begin{cases} \mathrm{c}_{p^*}(X,\ Y) \\ \min\ \ \mathrm{c}_p(X,\ Y) \text{ where p is an (N, M)-warping path} \end{cases}$$



(a) Warping path violating the boundary condition



(b) Warping path violating the monotonicity condition



(c) Warping path violating the step-size condition

Figure 2.12: Examples of possible warping paths that can violate the conditions mentiond above

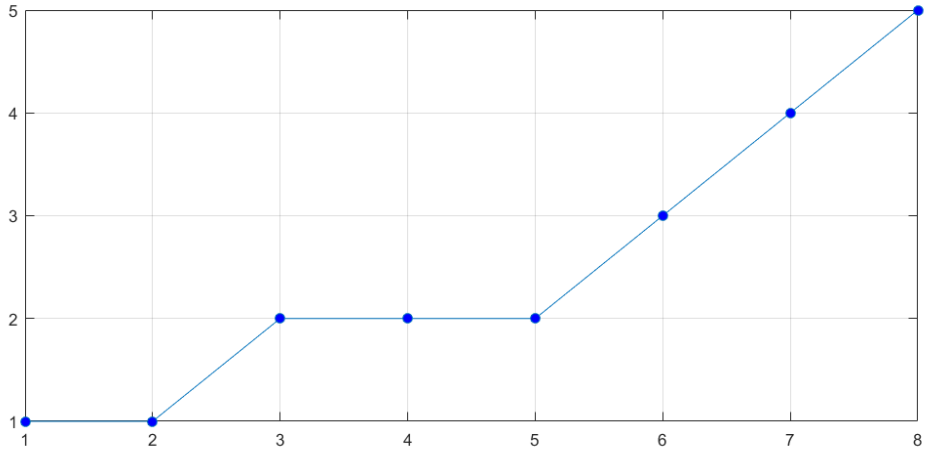


Figure 2.13: Possible warping path given by DTW that satisfies all th three conditions

11	7	3	1	1	2	6	3	0
8	4	0	2	4	5	3	0	3
6	2	2	4	6	7	1	2	5
11	7	3	1	1	2	6	3	0
3	1	5	7	9	10	2	5	8
	4	8	10	12	13	5	8	11

Figure 2.14: Simple constructionn of Cost matrix. Black dashed lines indicate the path from bottom to the top

A few remarks need to be made before we move further. First, we need to note that DTW distance is well-defined even though there may be several warping paths of minimal cost. Second, DTW is symmetric only if the local cost measure matrix c is symmetric. DTW distance does not satisfy triangle inequality even if c satisfies it. To determine an optimal path p^*, one could test every possible warping path between X and Y. Such a procedure is computationally very expensive. So we prefer dynamic programming to find an optimal path and thus we can have a new $N \times M$ cost accumulated matrix D. The accumulated cost matrix D is build using the following identities:

- $D(n,1) = \sum_{k=1}^{n} c(x_k, y_1)\ for\ n \in [1:N]$,
- $D(1,m) = \sum_{k=1}^{m} c(x_1, y_k)\ for\ m \in [1:M]$,
- $D(n,m) = min\{D(n-1,m-1), D(n-1,m), D(n,m-1)\} + c(x_n, y_m)\ for\ n \in (1:N)\ \&\ m \in (1:M)$

In this manner we can accumulate the matrix D both in row wise and column wise. The optimal path $p^* = (p_1, ..., p_L)$ is computed in reverse order of the indices starting with $p_L = (N,\ M)$. then the path p^* is computed as:

$$p_{l-} = \left\{ \begin{array}{ll} (1,\ m-1), & \text{if } n = 1 \\ (n-1,\ 1), & \text{if } m = 1 \\ argmin\big\{D(n-1,m-1), D(n-1,m), D(n,m-1)\big\}, otherwise, & \end{array} \right\} \tag{2.7}$$

However, there are a few limitations with respect to classical DTW. The continuity condition may map a single element of one sequence to many consecutive elements in other sequence, leading to a vertical and horizontal segments in the warping path, this drawback can be avoided by replacing the step size conditions with $p_{l+1}p_l \in \big\{(2,1),(1,2),(1,1)\big\}$ for $l \in [1:L]$. This results in accumulated cost matrix D given by:

$$D(n,m) = \begin{cases} min\big\{D(n-1,m-1), D(n-2,m-1), D(n-1,m-2)\big\} + c(x_n, y_m), \\ \text{for } n \in [2:N] \text{and } m \in [2:M]. \end{cases} \tag{2.8}$$

As initial values we set $D(0,0) := 0, D(1,1) := c(x_1, y_1)$. First and second row of accumulated matrix can be made ∞. First and second column can be made ∞ as well. This will help in better coding. We can impose constraints on the slope of the warping path. To favor the vertical and horizontal or diagonal directions in the

alignment we can have a weight vector (w_d, w_h, w_v) in $\mathbb{R}^3$, yielding the recursion.

$$\text{D(n,m)=min} \begin{cases} D(n-1, m-1) + w_d.c(x_n, y_m), \\ D(n-1, m) + w_h.c(x_n, y_m), \\ D(n, m-1) + w_v.c(x_n, y_m) \end{cases}$$

The classic DTW discussed above can align a pair of 1D sequence at a time. When we try to implement it on the data from the MoCap system we might have to consider each dimension separately. With few changes in the algorithmwe can implement it for a multi-dimensional data. For instance we can have the average of the scores along each dimention. Multidimensional Dynamic time warping as proposed in tries to modify the cost metric $c(x_n, y_n)$ by using the vector norm (usually L2) to calculate the distance between the pair of points. So now let us consider two sequences X and Y such that $X \in \mathbb{R}^{K \times L_x}$ and $Y \in \mathbb{R}^{K \times L_y}$, where K is the number of attributes or the number of dimensions in the feature space, L_x and L_y are the length of sequences X and Y respectively. Then the cost matrix c is given by,

$$c(n, m) = \sum_{k=1}^{K} \Big(X(k, n) \;\; Y(k, m) \Big)^2$$

Once we have this cost matrix we can generate the accumulated matrix D from C. and then find an optimal path p^* using the above mentioned method. The final score A_{dtw} is then given by:

$$A_{dtw} = D(N, M)/length(p^*). \tag{2.9}$$

There are many other ways to handle it. DTW is a very old and well known technique that has been studied by many researchers. There are many variants of

DTW and for more reading refer to Albrecht and Muller (2009). The above score is not symmetric and we need to have a weighted sum of the scores to have a symmetric score, which in turn increases computation and time. DTW is a very good algorithm to map two time dependent sequences and has also been used in fields like speech recognition. However the main disadvantage we have is that DTW gives a very low score when we compare two sequences that look alike. In other words it does not consider the length of the two sequences. To illustrate this let us consider figure 2.15. All the three signals in red, green and blue give very similar and low scores. However when the signal in blue is compared to the signal in red, it is desirable to have a higher score. Looking at this problem in Euclidean space might not be the right choice. The next section gives us a different approach to look at the same problem.

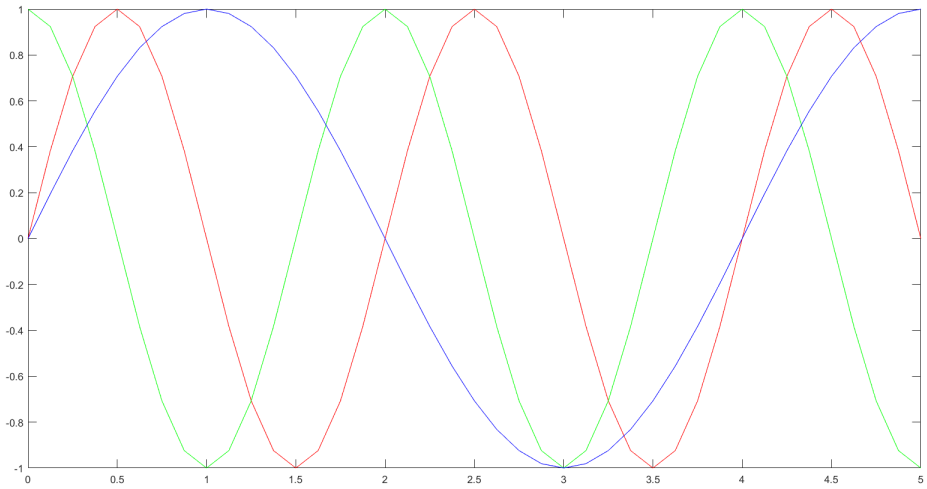


Figure 2.15: Illustrating when DTW fails

2.7 A geometric analysis of warp functions

Here we adopt the methodlogy of Su *et al.* (2014), which address the aforementioned limitations of the classical dynamic time-warping approach to compare se-

quences. Let ${A_m}^1 : \mathcal{M} \to \mathbb{R}_+$ be a function that maps a trajectory to a real number to reflect the asymmetry score. Let $\gamma \in \tilde{\Gamma} = \gamma : [0,1] \to [0,1] | \gamma(0) = 0, \gamma(1) = 1$, be a non-decreasing warping function such that $\alpha \circ \gamma$ is a trajectory that follows a similar path of α but with a different evolution rate. This warping function must not affect the quantification of the score A_m. In other words,

$$A_m(\alpha) = A_m(\alpha \circ \gamma) \tag{2.10}$$

If the warping function γ yields in the exact same function to which it is compared, then $A_m = 0$. Mathematically,

$$A_m(\alpha) = 0, \qquad if \quad \alpha = \alpha \circ \gamma \tag{2.11}$$

The score must satisfy the above two conditions. Using the Square-root velocity function(SRVF) representation of a trajectory α proposed in [2], we have

$$h_\alpha(t) = \frac{\dot{\alpha}(t)}{\sqrt{|\dot{\alpha}(t)|}} \tag{2.12}$$

and the distance between two SRVFs is given by,

$$d_h(h_{\alpha_1}, h_{\alpha_2}) = \Big(\int_0^1 \big| h_{\alpha_1}(t) - h_{\alpha_2}(t) \big|^2 \mathrm{d}t \Big)^{1/2} \tag{2.13}$$

The equivalent class of h_α is defined as:

$$[h_\alpha] = \{ h_{\alpha \circ \gamma} | \gamma \in \tilde{\Gamma} \} \tag{2.14}$$

Therefore, the score A_m is the shortest distance between the equivalent classes of h_α and $h_{\alpha \circ \gamma}$ and is given by, $A_m(\alpha) = d_s([h_{\alpha_1}], [h_{\alpha_2}]) =$

$$\inf_{\gamma_1, \gamma_2 \in \tilde{\Gamma}} \Big(\int_0^1 \Big| h_\alpha\big(\gamma_1(t)\big) \sqrt{\dot{\gamma}_1(t)} - h_\alpha\big(\gamma_2(t)\big) \sqrt{\dot{\gamma}_2(t)} \Big|^2 \mathrm{d}t \Big)^{1/2} \tag{2.15}$$

[1] Asymmetry score in Riemannian manifold

where $\alpha_1 = \alpha \circ \gamma_1$, $\alpha_2 = \alpha \circ \gamma_2$ Su *et al.* (2014) As we can see, if $\alpha \circ \gamma_1$ is same as $\alpha \circ \gamma_2$ then $A_m = 0$. While implementing the algorithm, γ_1 and γ_2 can be replaced with a simgle $\gamma^* \in \tilde{\Gamma} = \{\gamma : [0,1] \to [0,1] | \gamma(0) = 0, \gamma(1) = 1, \gamma \text{ is a diffeomorphism}\}$ that aligns $h_{\alpha \circ \gamma^*}$ with h_α, a warping function that minimizes $d_h(h_\alpha, h_{\alpha \circ \gamma^*})$ over $\tilde{\Gamma}$.

2.8 Summary of the framework

We have discussed all the necessary components so far. This section combines all the necessary elements to give us an overview of the framework necessary to analyze human walking patterns based on symmetry.

- Collecting raw data from devices like RGBD sensors, Motion-Capture system: This stage involves collecting a stream of data that can give us the precise location of joints of interest. This data holds the 3D position with respect to the system calibration.

- Use the trajectory of left and right feet to find the peaks along Y axis (Axis perpendicular to the ground). Figure2.16 shows us the trajectory of left and right foot along Y axis.

- Use the location of peaks to mark the start and stop of each gait cycle as shown in the figure2.16 (both left and right gaits separately) and index the gait cycles.

- We now will have the left gait cycles and right gait cycles indexed in order. We need to compare the ith*gaitcycleinleftto*ith gait cycle in right (Note: For every joint, the length of the data samples constituting one gait cycle need not be same for left and right joints).

  - Take i^{th} gait cycle in left and i^{th} gait cycle in right and scale the trajectory of every joint wrt., left hip.

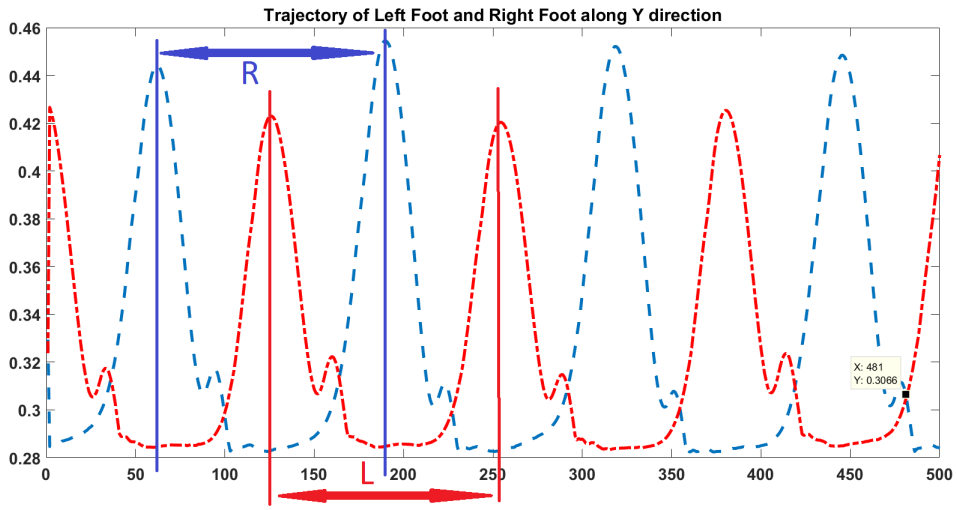


Figure 2.16: Trajectory of the left foot in red and right foot in blue along the Y axis

- Normalize the data samples to lie between 0 and 1.
- Label the joints in the left side of the body as 0 and right side of the body as 1.
- Using gradient descent find an optimal hyper plane that passes through the center of the hip dividing the body into exactly two equal halves and which is perpendicular to the ground. Figure2.17 gives us the hyper plane and the data samples of various joints for one gait cycle in assorted colors.

- Using the parameter of the hyper plane computed in the previous step, flip the data samples of the left joints towards the right side of the body (Householder transformation) and the new data samples after spatial reflection is referred to as flipped left joints. (At this point we have spatially reflected and temporally glided data samples of left joints of i^{th} gait cycle.) Figure2.18 gives us the flipped left joints in red and the right joints in blue for one gait cycle.

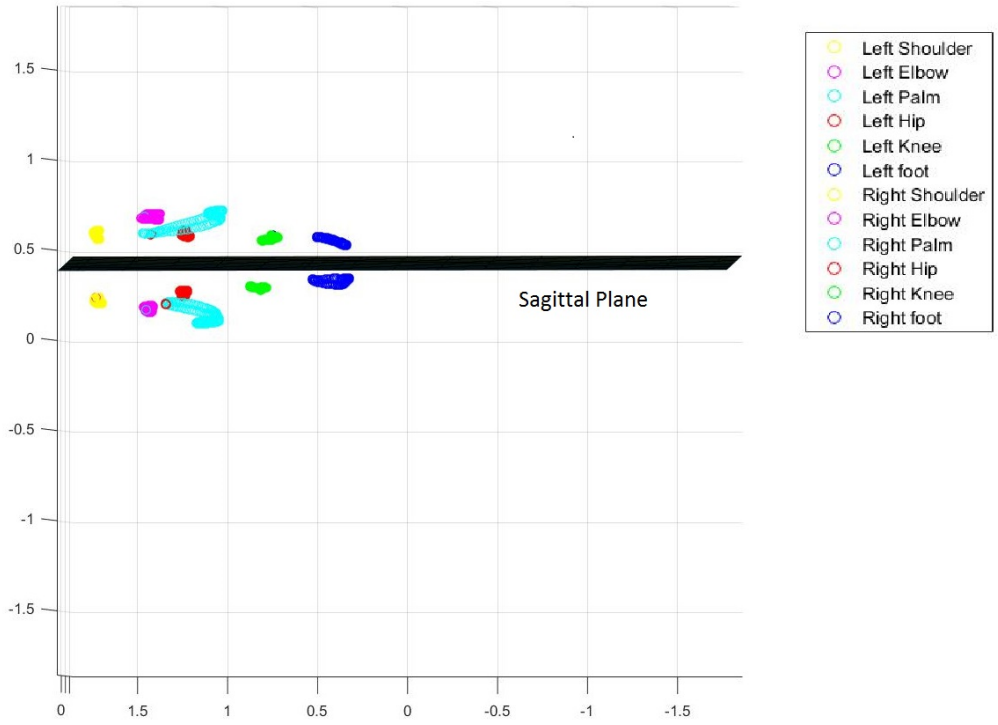


Figure 2.17: Data samples of all the joints for one particular gait cycle with the constructed sagittal plane

- Repeat all the below methods for every gait cycle
- Method 1:
  - Resample the flipped left joint and the right joint to have 160 equally spaced samples.
  - Find the Euclidean distance between the two sequences of 160 data samples each.
  - This Euclidean distance can be used as a score to measure the similarity between the two sequences.
- Method 2: (DTW Score)

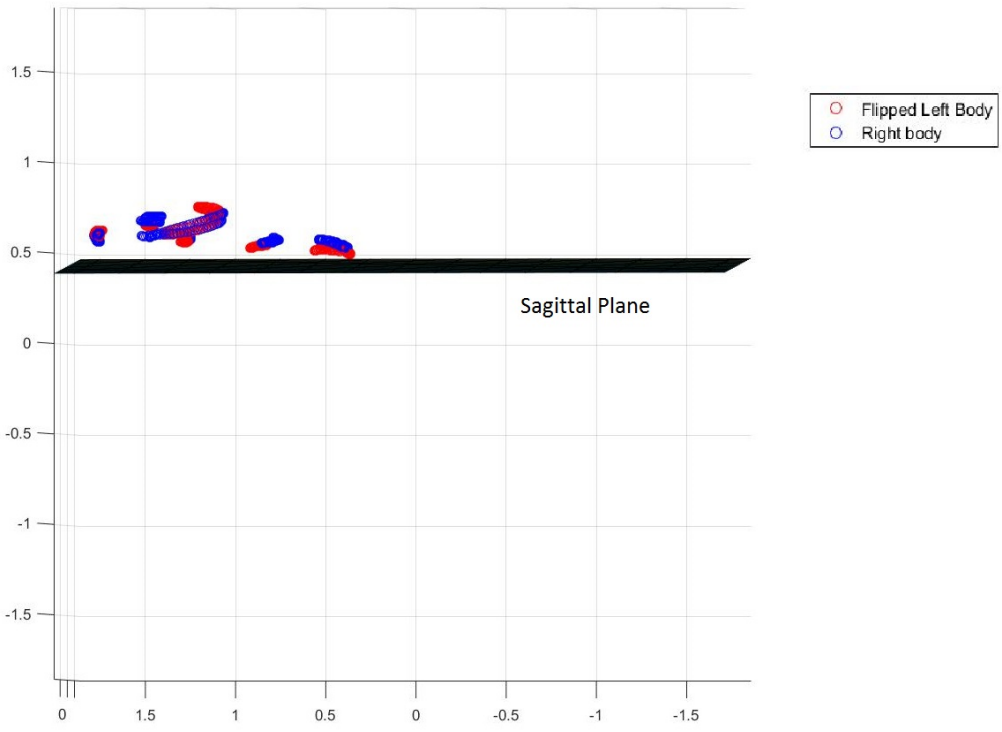


Figure 2.18: Data samples of flipped left joints in red and the right joints in blue for one gait cycle against the sagittal plane constructed.

* Create a $N \times M$ cost matrix C between the two sequences as explained in previous section.
* Construct an accumulated cost matrix D from C.
* Using the accumulated cost matrix to find an optimal warping path p starting at (N,M) all the way down till (1,1).
* Sum of all the elements along the warping path divided by the number of elements in the warping path is the score of similarity or the distance between the two sequences.

– Method 3: (SRVF representation of open curves)

* Use the SRV representation of the open curves in the quotient space to

find the distance between equivalent classes as explained in previous section.

- Plot the scores obtained from all three methods to check an optimal method we could use for our application.

2.9 Real time application

To extend the framework for real time approach, we need to find the appropriate temporal glide necessary in the segmentation stage. A symmetry score for glide-reflection symmetry should compare a segment of trajectory on one side with the spatially reflected version of a temporally glided segment on the other side. Since we want to obtain continuous and potentially real-time symmetry scores, at time $t \in [T_1, T_2]$, we will have access to current and history data $[T_1, t)$ but not future data $(t, T_2]$. For the current symmetry score A at any time t, the segment of trajectory on the one side should be the most current version (i.e, from $[t - T, t]$). Since we can choose both left and right side as the one side, there are two symmetry scores here, one comparing the latest segment from the left side with a past segment from the right side, and the other vice versa. We denote these two scores $A_{(l)}$ and $A_{(r)}$ respectively, and intuitively these two scores should be close to each other over the course of the whole movement. The choice for proper values of T needs to be considered. In theory the entirety of history data from T_1 to t are available for use. However, using too much of them may not be optimal, because it can be both computationally costly and undesirable in the context of real-time applications to have scores influenced by signals from the remote past. Conversely, T values that are too small may yield noisy scores. Since movements with glide-reflection symmetry are often also periodic, a natural choice is to use the fundamental period as T_l and T_r where T_l and T_r are the fundamental period of left and right gait cycle. This value may change over the

course of a movement (e.g. a person can adjust the speed of their walking), in which case T_l and T_r can be treated as a function of t and updated frequently as mentioned in the segmentation stage discussed in previous section. The temporal glide G is then computed as the difference between the two successive peaks in the trajectories of left and right gait cycle in some sense and this can be update in similar fashion as that of fundamental period T_l and T_r of left and right gait cycle respectively. We can thus compare the trajectories of left gait cycle between $[t - T_l, t)$ and the trajectories of the right gait cycle between $[(t - G - T_r), t - G)$.

It is also important to have the scores run smoothly and so we use a Half-normal distribution as a final kernal to convolve it with the running score. The probability density function of a half-normal distribution is given by:

$$f(x;\sigma) = \frac{\sqrt{2}}{\sigma\sqrt{\pi}}\ exp\big(\frac{x^2}{2\sigma^2}\big)|for x \leq 0 \tag{2.16}$$

Figure 2.19 gives us the distribution of a half-normal distribution under consideration. We have chosen $\sigma = 1$. The kernel parameters are given by:

0.00633	0.07715	0.34601	0.57049

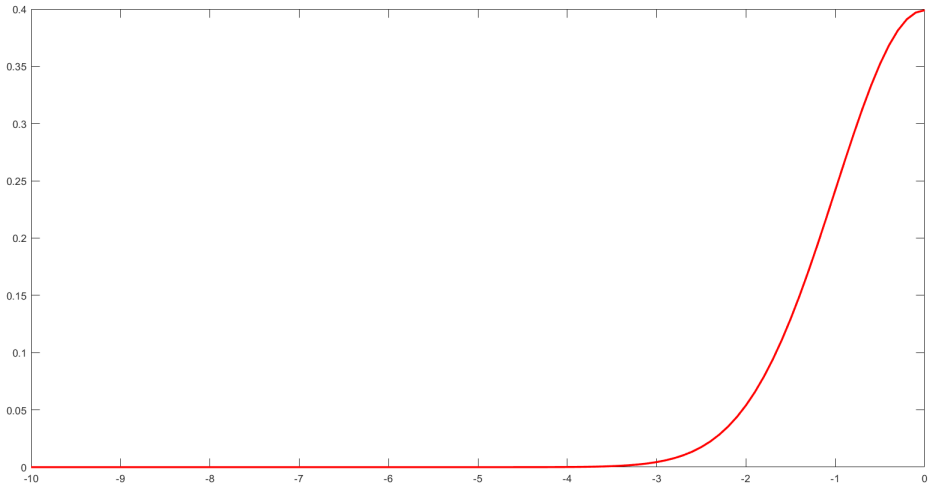


Figure 2.19: Probability density function of a Half normal distribution

2.9.1 Schematics for implementing Real time application

This section gives us the visualization of the extra measures that needs to be taken to implement real time applications as discussed above.

- Use the trajectory of left and right feet to find the peaks along Y axis (Axis perpendicular to the ground) as in the original framework and use the location of peaks to mark the start and stop of each gait cycle (both left and right gaits separately) and index the gait cycles.

- Define the left and right gait period as T_l and T_r using the peaks that are found in previous step as shown in figure2.20.

- Find the temporal glide G or the time required to find a right peak after the left peak has been found or vice-versa as shown in figure2.20.

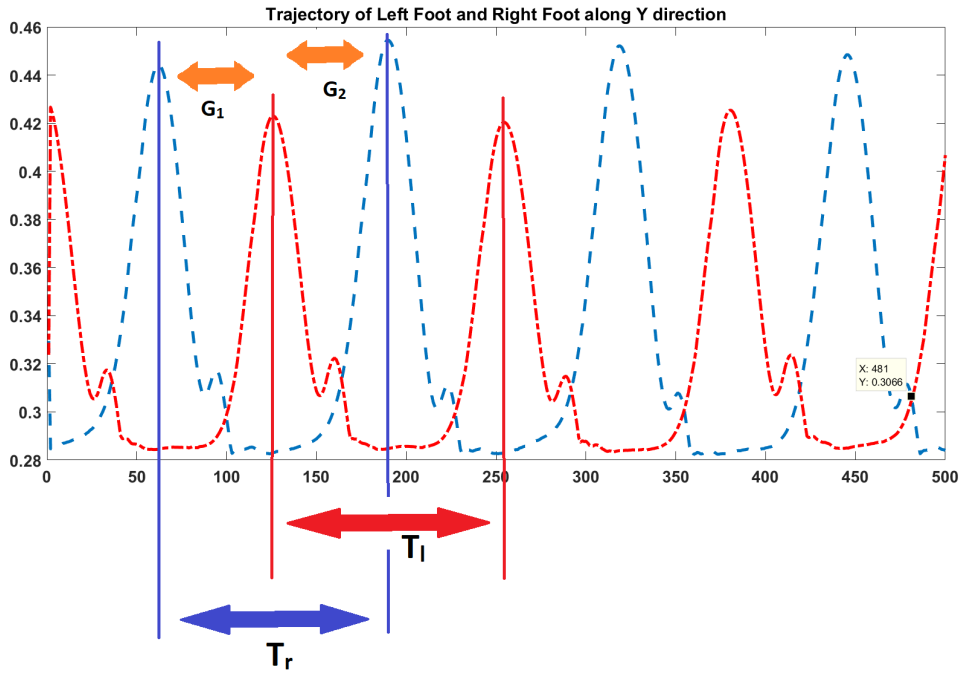


Figure 2.20: Illustrating gait period $T_l in red and T_r in blue$ and glide $G-1 and G_2$.

- Keep updating the gait periods and temporal glide for every new peak that appears on the time line.
- We can thus compare the trajectory of the left gait cycle between $/[t-T_r-G, t-G/)$ in red and the trajectories of the right gait cycle between $/[t-T_r-G, t-G/)$ in blue with the framework mentioned in the previous section.
- For every 20 new samples in the left gait cycle, we compute the score in a similar fashion as computed for every gait cycle as explained in previous section

Chapter 3

RESULTS AND EVALUATION

We have seen all the necessary components for human gait analysis. Since it is also necessary to show that the framework proposed can truly differentiate between the abnormal gaits and normal gaits, this section discusses more about the kind of gait abnormalities for which we are looking. To begin with we shall discus the characteristics of the three abnormal gaits we have chosen and then proceed to see if the symmetry score reflects the same. We have considered three abnormal gaits: Hemiplegic gait, Diplegic gait, Parkinson gait.

In Hemiplegic gait, also reffered to as stroke, the patient will hold his or her arm to one side and drags his or her affected leg in a semicircle (circumduction) fashion due to weakness of distal muscles (foot drop) and extensor hypertonia in lower limb. Figure 3.1 gives the foot prints and path of Hemiplegic gait for better understanding.

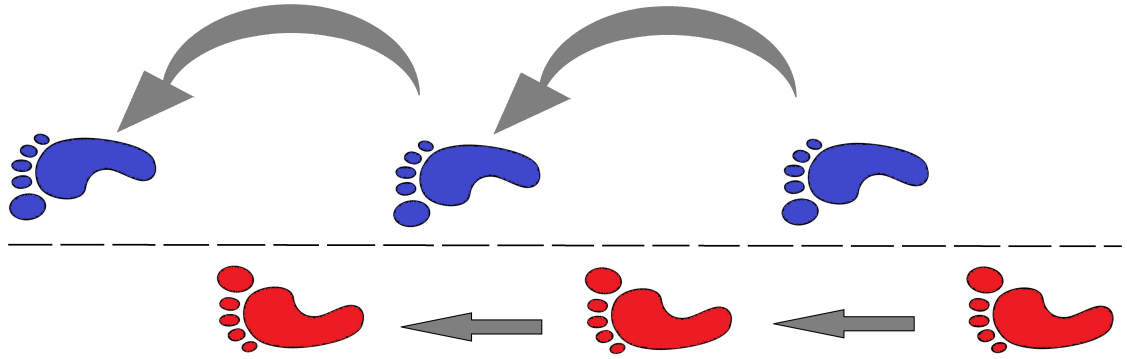


Figure 3.1: Foot prints of Hemiplegic Gait, where the arrow indicates the direction of motion.

Parkinson Gait is a special gait in consideration. Our first attempt to evaluate the gaits based on symmetry in Euclidean space to find the abnormal gait failed here. Figure 3.2 gives us the foot patterns observed in Parkinson gait. In this gait, the patient will have rigidity and bradykinesia. He or she will be stooped with the head and neck forward, with flexion at the knees. The whole upper extremity is also in flexion with the fingers usually extended. The patient walks with slow little steps known at marche a petits pas (walk of little steps). The patient may show an involuntary inclination to take accelerating steps, known as festination. We have also illustrated a normal gait foot patterns in figure 3.3.

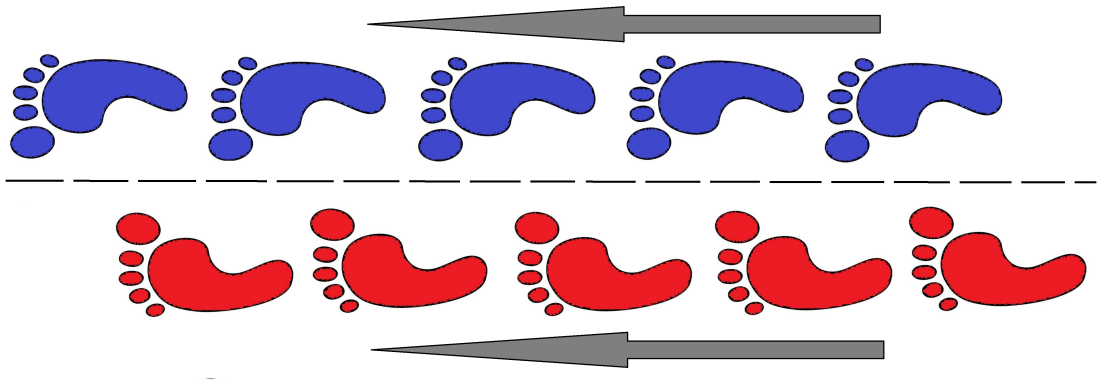


Figure 3.2: Foot prints of Parkinson Gait where the arrow indicates the direction of motion.

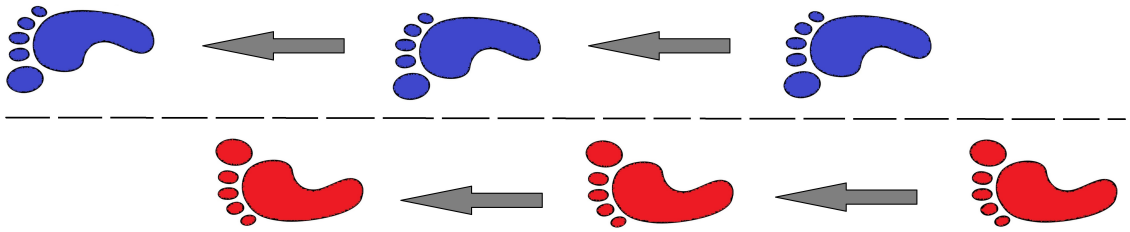


Figure 3.3: Foot prints of Normal Gait where arow indicates the direction of motion.

For better understanding, the human body is divided into upper body and lower body. The scores for upper body and lower body are given as a weighted sum of the individual socre of joints considered in upper body and lower body, and is given by:

$$A_{upper-body} = \frac{L_{ua}}{2L_{ua} + L_{la}} A_{elbow} + \frac{L_{ua} + L_{la}}{2L_{ua} + L_{la}} A_{hand} \tag{3.1}$$

$$A_{lower-body} = \frac{L_{ul}}{2L_{ul} + L_{ll}} A_{knee} + \frac{L_{ul} + L_{ll}}{2L_{ul} + L_{ll}} A_{foot} \tag{3.2}$$

Where A_{elbow} is the symmetry score of the elbow, A_{palm} is the symmetry score of the palm, A_{knee} is the symmetry score of the knee, A_{foot} is the symmetry score of the foot,

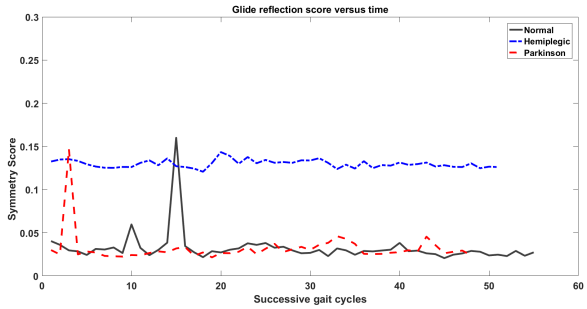
L_{ua}, L_{la}, L_{ul}, L_{ll} are the average lengths of upper arm, lower arm, upper leg, lower leg, respectively, measured from motion capture data. Adding the above coefficients gave hands and feet more weight proportional to their ranges of motion.

To evaluate our proposed method, we collected walking data from a motion capture system. The actor walked on a treadmill, with 12 sets of markers on his shoulders, elbows, hands, hips, knees, and feet. Each session has around 20 gait cycles. The actor followed instructions on gait abnormalities from Stanford Medicines websiteMedicine (2017) to simulate hemiplegic (stroke) and Parkinson gaits. For each gait type, 3 sessions of data were collected, with each session 15 to 30 seconds of duration, and symmetry scores calculated after the 4th second in each session. This is done to avoid any errors in the beginning of the session. The data were capture at 120 FPS. Having these three abnormal gaits in mind, we would like to see if the score we generate using our framework can help us distinguish them better. We have used three methods to find the symmetry score:

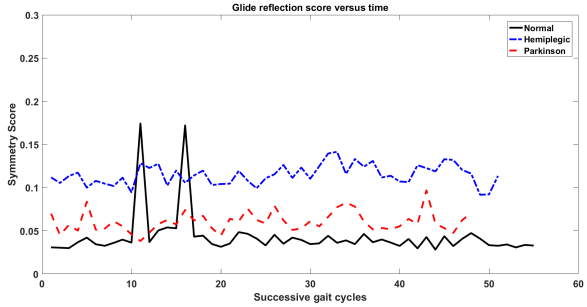
- Method 1: This is the baseline method, and the proposed framework will be compared to this approach. Let us consider two time dependent sequences L and R representing left and right gait cycles. Let $L := (l_1, l_2, \dots l_N)$ of length $N \in \mathbb{R}^3$ and $R := (r_1, r_2, \dots r_M)$ of length $M \in \mathbb{R}^3$. As a baseline for comparison, independent of the sampling rate (samples per gait cycle) every cycle was resampled to have 140 to 160 data sampleCabral *et al.* (2016) depending upon the velocity profiles of the gait in consideration. The score $A_{resample}$ is the $\mathbb{L}^2$ norm between the two vectors and is given by,

$$A_{resample} = \sqrt{\sum_{n=1}^{n=140} \Big(L(n) - R(n) \Big)^2} \tag{3.3}$$

Figure 4.1 gives us the scores of the discussed method. Plot in black refers to normal gait, plot in green refers to parkinson gait and plot in blue refers to hemiplegic gait. This method is not a good choice as there is improper matching between the samples of right and left gait cycle. Since it is also important to compare two appropriate parts of the gait cycle, we need a warping function.



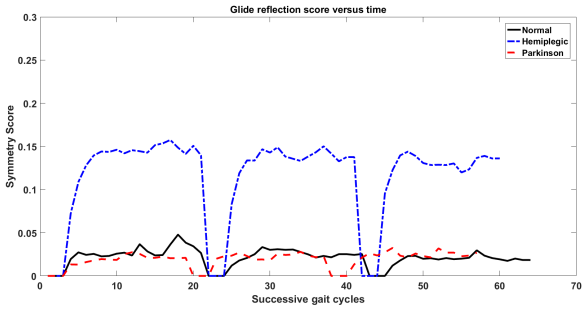
(a) Upper body score



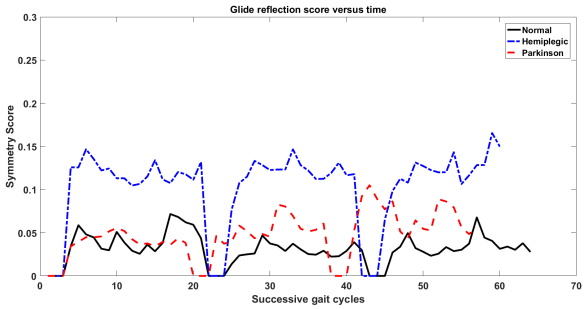
(b) Lower body score

Figure 3.4: Symmetry scores using method 1.

- Method 2: This is the implementation in Euclidean space using DTW to map the two sequences. The similarity score is got using the framework discussed above. Figure 3.5 gives the symmetry score computed using DTW over individual gait cycle. We have three sessions of the data and so there is a dip to 0 before the start of the next session. Plot in black refers to normal gait, plot in green refers to parkinson gait and plot in blue refers to hemiplegic gait.



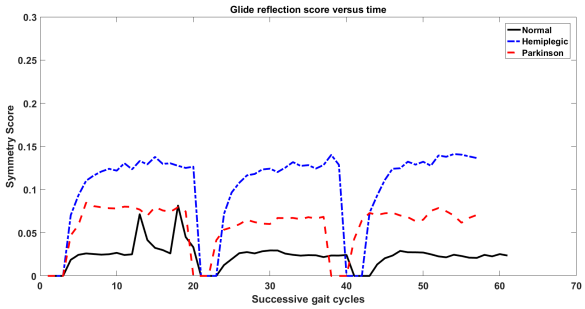
(a) Upper body score



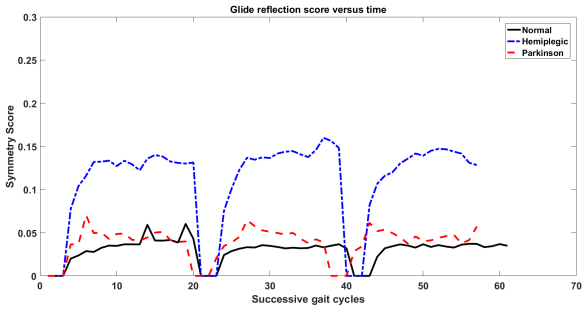
(b) Lower body score

Figure 3.5: Symmetry scores using DTW for every gait cycle.

- Method 3: This is the implementation in Riemannian manifold using SRVF representation. This framework has also been discussed in previous section. Figure 3.6 gives the symmetry score computed using SRVF representation over individual gait cycle. We have three sessions of the data and so there is a dip to 0 before the start of the next session. Plot in black refers to normal gait, plot in green refers to parkinson gait and plot in blue refers to hemiplegic gait. Figure 3.7 gives us the symmetry score when we need it in the real time for every 5 new successive frames with the glide G evaluated as discussed in Chapter 3.

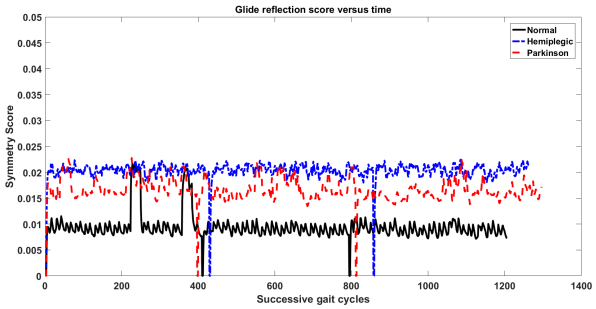


(a) Upper body score

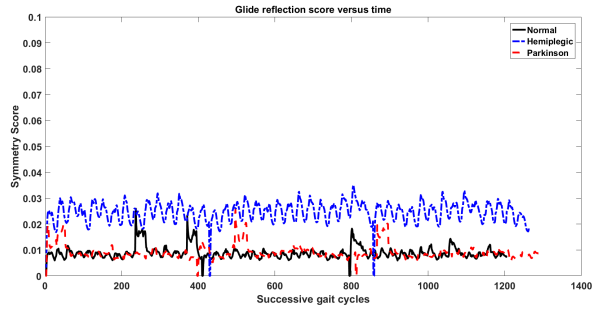


(b) Lower body score

Figure 3.6: Symmetry scores using SRVF representation for every gait cycle.



(a) Upper body score



(b) Lower body score

Figure 3.7: Symmetry scores using SRVF representation for real time applications.

From visual examination of the graphs, the scores from DTW in Euclidean space can be used to identify certain abnormal gaits better than the simple resampling technique mentioned in method 1. However it is seen that it is not able to distinguish between Normal and Parkinson gait in most of the scenarios. This is mainly because, Parkinson gait has a similar walking patterns as Normal walking patterns. To be more precise, Parkinson has symmetric walking patterns. So a simple approach in Euclidean space fails to identify such abnormal gait. SRVF representation of all the trajectories onto the quotient space helps us identify such abnormal gaits which is evident from 3.6. We have tabulated the results in table 3.1 for the symmetry indices discussed in section 1.1.3 with the gait cycle period $T_l and T_r$ as the chosen parameter for X_l and X_r.

Method	Normal	Hemiplegic	Parkinson
RI	-0.0081 / 0.0087	-0.0170 / 0.0108	-0.0155 / 0.0158
SI	-0.0080 / 0.0085	-0.0168 / 0.0106	-0.0153 / 0.0154
GI	0.0080 / 0.0085	0.0168 / 0.0106	0.0153 / 0.0154

Table 3.1: Mean/Variance of the symmetry scores obtained from the three indices

Table3.2 gives us the upper body score of all the three methods discussed in the thesis work. From the table below, we can also clearly say that method 3 is more robust approach to distinguishing abnormal gaits from normal gaits.

Method	Normal	Hemiplegic	Parkinson
Method 1	0.0322 / 0.0188	0.1298 / 0.0044	0.0318 / 0.0179
Method 2	0.0211 / 0.0104	0.1146 / 0.0508	0.0190 / 0.0092
Method 3	0.0232 / 0.0142	0.1025 / 0.0475	0.0577 / 0.0267

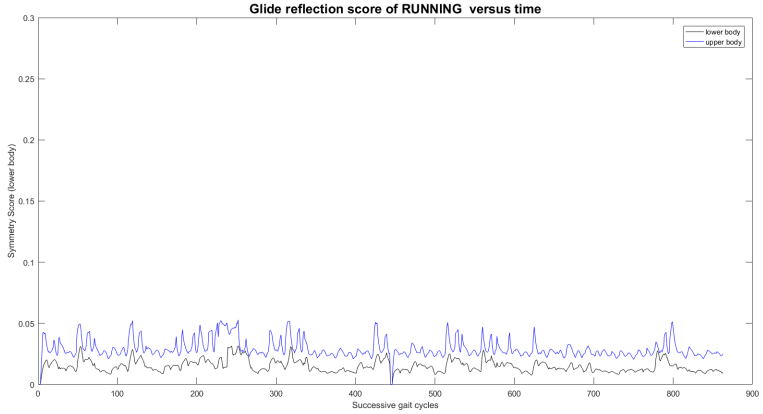
Table 3.2: Mean/Variance of upper body scores obtained from all the three methods discussed above

Chapter 4

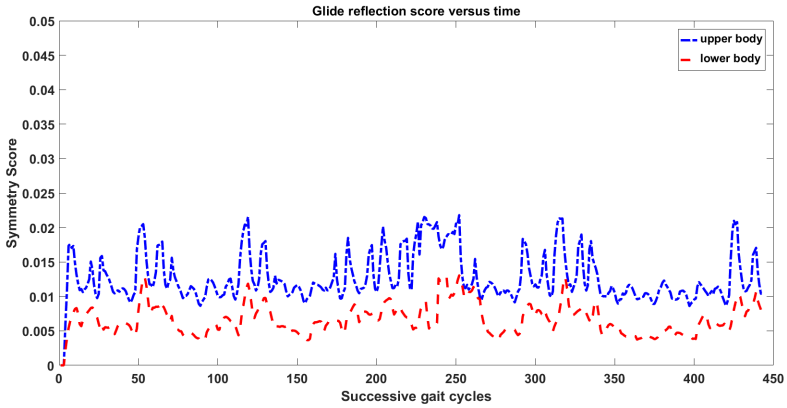
CONCLUSION AND FUTURE WORK

In my thesis work, we proposed two optimal frameworks to continuously measure the degree of glide-reflection symmetry in human movements, and saw different techniques like differential geometric tools from elastic shape analysis to quantify the symmetry and help identify gait abnormalities. Our preliminary experiment showed that our proposed method in Euclidean space worked well to distinguish between normal gait and simulated pathological gaits (under controlled environment), and can be used to generate stable outputs to drive feedback systems which is something similar to the system developed in Wang *et al.* (2016). In the later stage of thesis work, we also proposed a more mathematical approach that could be used to build a more robust system to help identify few other abnormalities in gait patterns that might have symmetry in them. We could also use this framework for evaluating few other human activities that show glide reflection symmetry. To start with, the same framework was used to quantify running patterns of a subject running at 4 miles per hour. Two sessions 20 seconds each were recorded and figure 4.1a gives the real time score. Similarly, we simulated front crawl swimming action by making the subject stand on the floor and imitate the motion of arms for 10 seconds. Figure 4.1b gives us the score of swimming action. So this framewowrk could be used for training athletes to have a more symmetric patterns. This framework can be used by physiotherapists for monitoring patients during rehabilitation after any injury that can hinder the normal human movements. It can also be used to study the complexity of locomotions in the developing children and elderly people. It is also important to know the amount of asymmetry that can be allowed in human motions before it can get fatal and

so can also be used in general clinics. However, the current implementation of the



(a) Symmetry score for RUNNING



(b) Symmetry score for SWIMMING

Figure 4.1: Symmetry scores of other actions

method was online but not real-time. Future work can also be focused on applying fast approximate methods to increase the speed of the method and aim for building real-time systems with it. On the other hand, having only the trajectory of joints might not be enough to analyze the symmetry discussed. We can also look into how the angle profiles of Torso, knee, Elbow can be used.

REFERENCES

Albrecht, T. and M. Muller, "Dynamic time warping (dtw)", Information Retreival for Music and Motion pp. 70–83 (2009).

Błażkiewicz, M., I. Wiszomirska and A. Wit, "Comparison of four methods of calculating the symmetry of spatial-temporal parameters of gait", Acta of Bioengineering and Biomechanics **16**, 1 (2014).

Cabral, S., R. A. Resende, A. C. Clansey, K. J. Deluzio, W. S. Selbie and A. P. Veloso, "A global gait asymmetry index", Journal of applied biomechanics **32**, 2, 171–177 (2016).

Householder, A. S., "Unitary triangularization of a nonsymmetric matrix", Journal of the ACM (JACM) **5**, 4, 339–342 (1958).

Lee, S. and Y. Liu, "Curved glide-reflection symmetry detection", IEEE transactions on pattern analysis and machine intelligence **34**, 2, 266–278 (2012).

Medicine, S. M. . S., "Gait abnormalities", Http://stanfordmedicine25.stanford.edu/the25/gait.html (2017).

Ng, A., "Cs229 lecture notes", CS229 Lecture notes **1**, 1, 1–3 (2000).

Ruder, S., "An overview of gradient descent optimization algorithms", arXiv preprint arXiv:1609.04747 (2016).

Sinha, A., K. Chakravarty and B. Bhowmick, "Person identification using skeleton information from kinect", in "Proc. Intl. Conf. on Advances in Computer-Human Interactions", pp. 101–108 (2013).

Su, J., S. Kurtek, E. Klassen, A. Srivastava *et al.*, "Statistical analysis of trajectories on riemannian manifolds: bird migration, hurricane tracking and video surveillance", The Annals of Applied Statistics **8**, 1, 530–552 (2014).

Vaverka, F., M. Elfmark, Z. Svoboda, M. Janura *et al.*, "System of gait analysis based on ground reaction force assessment", Acta Gymnica **45**, 4, 187–193 (2015).

Wang, Q., R. Anirudh and P. Turaga, "Temporal reflection symmetry of human actions: A riemannian analysis", (2016).

Yang, M., H. Zheng, H. Wang, S. McClean, J. Hall and N. Harris, "Assessing accelerometer based gait features to support gait analysis for people with complex regional pain syndrome", in "Proceedings of the 3rd International Conference on PErvasive Technologies Related to Assistive Environments", p. 48 (ACM, 2010).

Yang, M., H. Zheng, H. Wang, S. McClean and D. Newell, "igait: an interactive accelerometer based gait analysis system", Computer methods and programs in biomedicine **108**, 2, 715–723 (2012).

Zhang, B., D. Wei, K. Yan and S. Jiang, *Human Walking Analysis, Evaluation and Classification Based on Motion Capture System* (INTECH Open Access Publisher Winchester, 2011).