MACHINE LEARNING APPROACHES TO CENTER-OF-MASS ESTIMATION FROM NOISY HUMAN MOTION DATA

By

AYESHA SIDDIQUA

Bachelor of Science in Computer Science & Engineering
Shahjalal University of Science and Technology
Sylhet, Bangladesh
2006

Submitted to the Faculty of the
Graduate College of
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
December, 2011

MACHINE LEARNING APPROACHES TO CENTER-OF-MASS ESTIMATION FROM NOISY HUMAN MOTION DATA

Thesis Approved:

Dr. Guoliang Fan

Thesis Advisor

Dr. Martin Hagan

Dr. Damon Chandler

Dr. Sheryl A. Tucker

Dean of the Graduate College

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

CHAPTER 1

Introduction

The **center of mass (COM)** is the balance point of an objects mass. If the object is supported at the point of Center of mass it would remain in place. To be precise, center of mass is the mean position of all the mass in an object. In the case of a rigid body, the position of the center of mass is fixed in relation to the body. Total mass in a body is considered to be the total quantity of 'matter' comprising the body. COM is a unique point in every rigid object around which the object's mass is equally distributed in all directions. Estimating the COM of an object is an important task in biomechanics since in many cases COM is the unique representative of postural stability and gait stability and therefore a true representative of human motion.

The purpose of this research is to compute and estimate COM. The research constitutes of two parts. In first part of this research, we have computed COM from motion capture data (MOCAP) and anthropometric data and have computed COM using anthropometric equations. The second part of the research involves machine learning approaches where we have used mapping based approach (RBF) and learning based approach (GPR) of machine learning to estimate COM. The main idea of this research is to accurately estimate COM in case of noisy or incomplete or inaccurately acquired motion data.

In this research, we compute and estimate COM from a very few representative training motions. We want to accurately estimate COM without using the anthropometric measurement which requires lots of manual input for different categories of people. In the first part of this research we have computed COMs from human motion data/(Motion Capture data) to acquire our training input (motion data) and training output (COM). The training

input consists of motion capture data for different individuals. Human motion data in our case consist of motion data of 31-joint (joints derived from markers) body model. The motion data basically consists of frames of 3D joint angles (relative to each other) or 3D joint positions. This body model is then compared and made compatible with an anthropometric body model to compute COM using anthropometric measurements. As a result the training output consists of the COMs computed from the body model. The second part of the research involves machine learning approaches where we have used manifold mapping (using RBF) based approach and Gaussian process regression based approach of machine learning to estimate COM.

In the field of biomechanics, human motion analysis is needed for understanding normal and pathological movement [1] to help mobility for paralyzed people [2,3], in the field of character animation real human motion analysis is need for character animation [4], in manufacturing industries manufacturing humanoid robots require human motion analysis feedback [5, 6] etc. Understanding and locating the COM greatly contributes to our understanding of human motion. COM acts as the key contributor to understand the human motion and analyze it.

Fall risk assessment is an aid for elderly people (65 years or older) who suffer or experience falls each year [7]. Two main reasons of falls in the elderly people are stepping over obstacles and gait imbalance [7]. Maintaining the whole body balance requires precise control of the motion of the whole bodys COM. The coordination of COM with COP, BOS or any other biomechanical unit is also important to measure the required support in the kinetic body joints while stepping over obstacles or to understand the gait balance [7,8].Therefore, the computation and estimation of COM is essential equipment in the biomechanical researches to measure the gait imbalance as an aid for fall risk assessment.

In this research we estimate COM readily (without using anthropometric equations) where we use limited number of training inputs to accurately estimate COM from an unknown (not present in the training motion data) motion. So it would be helpful to estimate

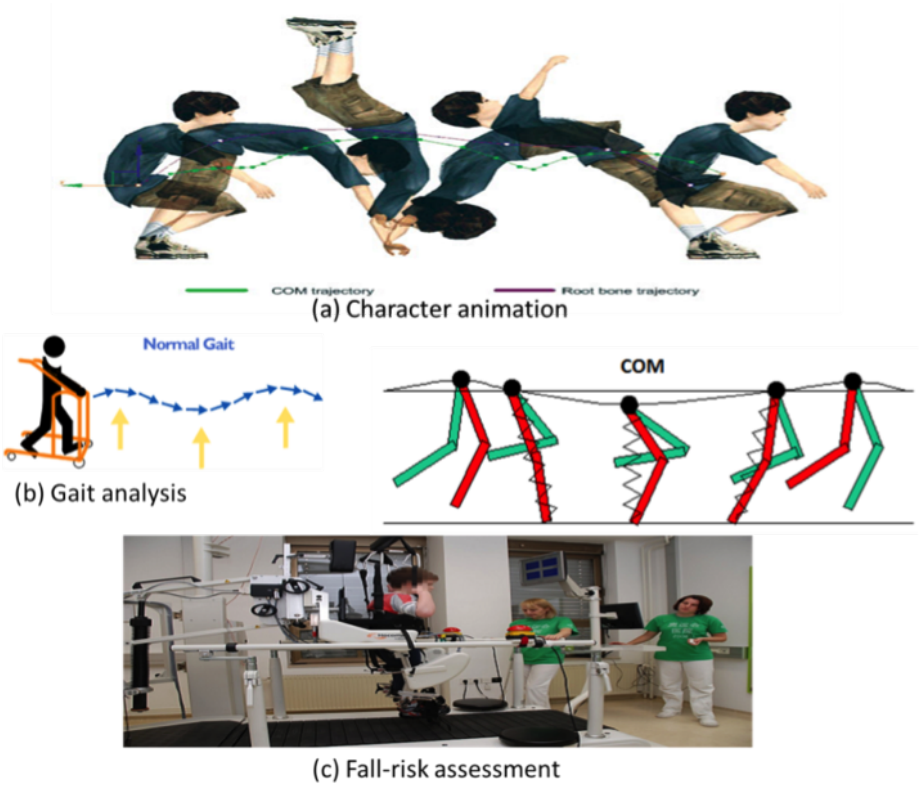

Figure 1.1: Applications of COM (a) Animation [9], (b) gait analysis [10] and (c) Fall-risk assessment [11]

COM readily where we use only the information of the three-dimensional positions of the body joints and body segment endpoints. The COM propagation is then estimated using machine learning approaches.

Motion analysis is one of the active research topics in the world currently. Human motion consists of high dimensional data. It often requires high amount of computation which is expensive and also inefficient to work with high dimensional motion data. COM is the point on a rigid body where the mass of the body can be considered to be concentrated for translational motion analyses. So COM decreases the amount of information to be recorded about the body. The bodys shape and structure can be ignored and only COM needs to be quantified only [12]. The trajectory of COM is completely different from walking to running. In the models of walking and running which can be considered as two different gaits, COM shows nearly opposite patterns of vertical movement [13]. So, from COM we can

characterize these two different gaits as walking and running respectively by dealing with COM rather than dealing with high dimensional full body motion. Determining the trajectory of the COM during walking, running and many other gaits plays an important role in the field of motion analysis. As a result the accurate estimation of COM which does not involve too much biomechanical measurement at each step of gait could play an efficient role.

In the fields of postural stability, gait stability, fall risk assessment, motion analysis, character animation and many others, COM computation and estimation is involved to reduce the amount of expense and time which we cannot avoid while dealing with high dimensional motion analysis.

1.1 Motivation

COM estimation from noisy/incomplete motion data: The most useful property of this shared latent space would be the filtering capacity in a particular sense. We have designed the latent space from noiseless training motion data and COMs, consequently when we extrapolate a noisy test motion data from the manifold it tries to extrapolate the noisy motion data from the manifold designed from noiseless training motions, and we get a reasonable noise free extrapolated motion data and the corresponding COM from the manifold.

All the biomechanics areas stated earlier, use optical or electro-magnetic or mechanical motion data acquisition system which is expensive. In our research, we are trying is to replace expensive motion data capture system such as optical, electro-magnetic and mechanical motion data with low cost motion sensors, inertial sensors etc. requiring less complex experimental set up.

Using inertial motion sensors our data acquisition would be less expensive, however, at the same time the disadvantage with these sensors is that often the data they provide would be less accurate, noisy or incomplete. Since the data could be noisy or incomplete, direct computation of COM from these noisy sensor data could not be helpful where high

accuracy of COM computation is needed in the areas such as diagnose postural and gait stability, detecting fall risk etc. As a result we would have to provide a filtering technique or estimation method for COM instead of direct computation when we are to acquire data from low cost inertial sensors.

Unknown motion data estimation: To estimate COM from new test motion data we have mapped/designed a shared latent space or manifold. This shared latent/space or manifold is shared by human motion data and the corresponding COM at the same time. This type of manifold/shared latent space has several different and exclusive advantages. One of the exciting characteristic of this shared latent space is that it provides continuous space to extrapolate new/unknown human motions (not present in the training motion data) and the corresponding COM.

Efficient tracking of High dimensional motion data onto Low dimensional manifold: Another important characteristic to be mentioned is the lower dimensionality of the manifold. The higher dimensional human motion data propagation corresponds to the lower dimensional point propagation onto the manifold. As a result we are able to track higher dimensional motion data on to the manifold efficiently since the manifold is a lower dimensional space and help us working with motion data by dimension reduction. And also we can extrapolate or interpolate intermediate motions from the manifold/latent space. **It is our assumption** that on the manifold each point maps to a human pose of a certain human gait as well as maps to the corresponding COM (for that pose/frame) at the same time. This is due to the fact that, the COM has a correlation with the pose/frame from which it has been calculated. The extrapolation of the test/unknown motion data as well as the COM is an added advantage of this shared latent space or manifold.

1.2 Research Objective

Estimate COM from noisy or incomplete motion data by

1. Shared latent space, shared by motion data and COM (motion data dominated latent space). Human motion is mapped onto the manifold by RBF. The manifold is torus shaped an ideal surface to reflect the kinematics of human gait.

2. Gaussian Process Regression, a machine learning technique, approximates the function (output) by taking into account the covariance structure of the high dimensional motion data.

We have compared the two different approaches at the end. Gaussian process regression performed well when there was no noise in the motion data. On the other hand for noisy motion data the manifold approach (Torus) performed better than GPR since the training motions mapped on to the torus is noiseless and the surface of the manifold is a guide to the noise motion data to find out what should be the noiseless motion data and corresponding COM.

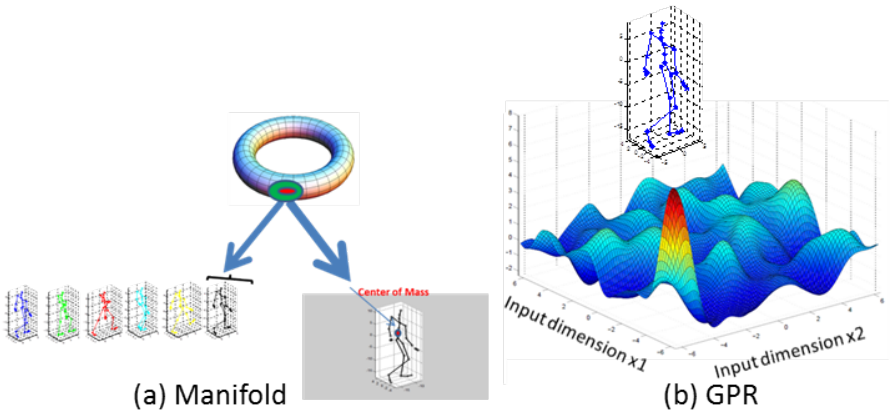

Figure 1.2: (a) Manifold based approach and (b) Gaussian Process Regression based approach to estimate COM.

1.3 Our Approaches

Dempster's Technique to compute training COM: In Biomechanics Dempster's created a table showing the segmental masses as proportion of the total body mass and lengths of the radii gyration as proportions of the segments' length for segment parameters [12]. Later, Barter (1957) worked with modified Dempster's data to compute segmental masses more accurately [12]. We have used our marker skeleton model and anthropometric model compatible to use anthropometric body segment parameters and formulas to calculate COM. We have used Dempester's Technique to compute COM. At first, we have computed COM for our training data to have our training input and training output. The training input consists of motion capture data for different individuals. From motion capture data we get motion data in terms of angles and positions for a 31-joint sensor body model. This body model is then compared and made compatible with an anthropometric body model to compute COM using anthropometric measurements. As a result the training output consists of the COM computed from the body model.

Manifold based mapping approach to estimate COM: Radial basis function (RBF) approximates multivariable functions by a single uni-variate function. We have used RBF between higher dimensional motion data and a lower dimensional manifold (Torus shaped). Again RBF was used in lower dimensional manifold and COM. This is how a manifold (Torus) has been designed or mapped. It allows two way mapping, where the first mapping works between motion data and motion data manifold and the second mapping works between COM and COM manifold. By acquiring extrapolated motion data COM has been extrapolated using the mapping relationship between Torus and COM where the lower dimensional manifold (Torus) is motion data dominated. We will discuss this issue in details later.

Gaussian Process based Regression approach to estimate COM: Gaussian Process regression provides inference in the function space directly. This regression process is non-parametric Bayesian approach. The high level information present in the training mo-

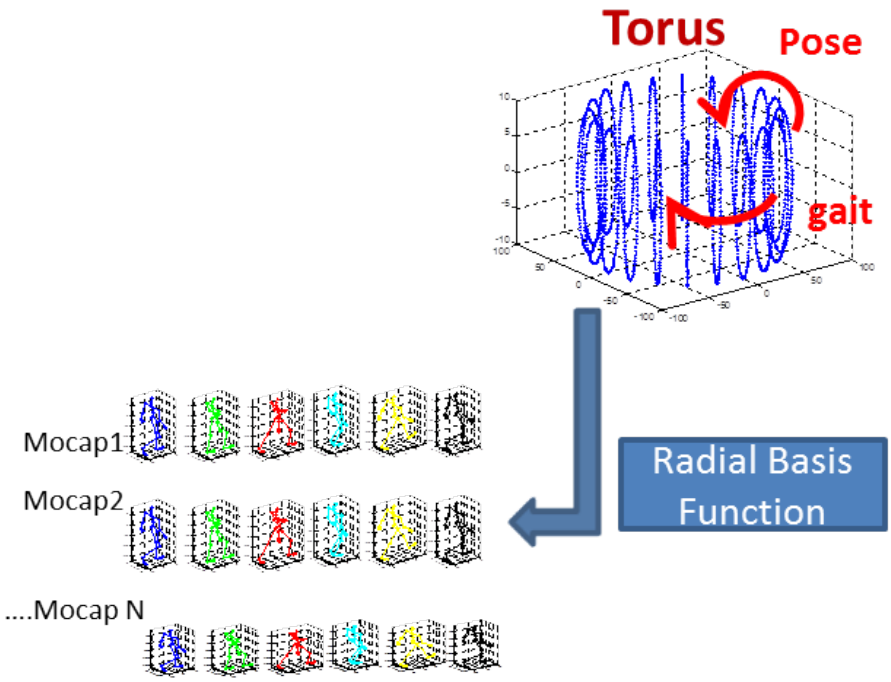

Figure 1.3: Application of Radial Basis Function.

tions that similar neighborhoods are strongly correlated, the output COMs are predicted from training examples where internal information of the test motion data is similar to the training motions. Gaussian process regression is essentially based on the assumption that similar inputs tend to give similar outputs [14].

1.4 Research Challenges

Noisy Motions tend to give estimation of COM with greater error: The main challenge of this research is working with noisy motion data acquired by sensors such as inertial sensors and get almost accurate estimation of the COM. The machine learning approaches such as RBF mapping and Gaussian process regression all work reasonably well when there is no noise or less noise in the testing motion data. But if the test data is noisy the estimation of COM is affected. The higher the variance of noise in testing motions, the higher amount of inaccuracy is added in the estimation of COM. So our challenge is to provide a way to handle noisy motion data and estimate COM accurately.

Representative combination of training motions: The main challenge of this research is to estimate COM for an unknown motion data. In this case, if the unknown motion data is completely different from the training motions given to the machine learning approach such as RBF (Radial Basis Function) or GPR (Gaussian Process Regression) model then there are chances of failing to estimate the appropriate COM. Gaussian process regression is basically based on the assumption that close inputs are likely to give almost same output [14]. As a result the choice of the combination of training motions should be an ideal representative of all the possible testing motions.

Working with less number of training data: In this research we are dealing with very few training data. For training we have used around twenty to twenty one training data. Our focus is to use small training motions because of the difficulty and complex data acquisition process by using markers and inertial sensors. So it is definitely a challenge to estimate test motion data accurately from such small number of training motions.

1.5 Contribution

Contribution 1: We have successfully computed COM from 16 joints from a 31 joint body model by comparing it with an anthropometric body model. We have made some changes in terms of trunks to make our sensor body model compatible with the anthropometric model. We have got the top views, side views and front views of the COMs have shown typical characteristics for normal walking.

Contribution 2: We have estimated COM using two different machine learning techniques. A manifold (Torus) has been designed or mapped to estimate COM. Also, we have implemented Gaussian process regression approach which predicts COM by encoding the high level information that similar neighborhoods training motions are strongly correlated. Gaussian process regression has been proved to be better working than torus-based approach when there is no noise or low noise in the testing motion data. The manifold based approach gives us a continuous space to extrapolate human motion data as well as COM.

Contribution 3: The shared latent space/manifold (Torus) approach where the manifold is shared by motion data and COM together, has been proved to be better working than the GPR to estimate COM from noisy test motion data. When the noise is higher, Torus-based approach has been proved to be better working than GPR since the torus structure guides the COM estimation not to be totally meaningless when the noise level is higher.

CHAPTER 2

Literature Review

The computation or estimation of COM is a biomechanics research topic. Postural stability, gait stability, fall risk assessment, pathological gait analysis etc. are important biometrics researches require the computation or almost errorless estimation of COM. In machine learning research approaches such as Gaussian process regression, radial basis function for interpolation, manifold learning to represent higher dimensional object etc. are generally used to predict robotic movement, robotic position, noisy sensor data etc. Biomechanics approaches need the computation of biomechanics units such as COM, base of support, center of pressure etc. with higher rate of accuracy. To meet the required accuracy, the acquisition of data in these fields are optical motion capture [7, 8, 15], electro-magnetic motion capture [13, 16], mechanical motion capture etc. Obviously optical, electromagnetic or mechanical data capturing approaches are expensive and require a lot of complex experimental setup.

Our focus in this research is to estimate COM from noisy, less accurate or incomplete human motion data. Estimation from noisy motion data provides an advantage over direct computation of COM using anthropometric equation. Anthropometric equations require various body segment parameters and ratio information as shown in table 3.1. As a result computation is time consuming and not flexible since a lot of manual input required.

It is natural that acquisition of human motion data from various sources might include some noise due to acquisition method and many other reasons. In our research what we are trying is to replace expensive motion capture system such as optical, electro-magnetic and mechanical motion data with low cost motion sensors, inertial sensors etc. requiring

less complex experimental set up. Using inertial motion sensors our data acquisition would be less expensive but at the same time the disadvantage with these sensors is that often the data they provide would be less accurate, noisy or incomplete. Since the data could be noisy or incomplete, direct computation of COM from these noisy sensor data could not be helpful where high accuracy of COM computation is needed in the areas such as diagnose postural and gait stability, detecting fall risk etc. As a result we would have to provide a filtering technique or estimation method for COM instead of direct computation when we are to acquire data from low cost inertial sensors.

In our research, to maintain the higher rate of accuracy in COM estimation, we acquire data by optical motion data to train up our COM machine learning estimation model (manifold mapping, Gaussian process regression). After that we acquire motion data to estimate COM from some low cost inertial motion sensors. Obviously our training phase requires COM computation where we apply biomechanical COM computation approaches and testing phase requires machine learning approaches to estimate COM, which is why the literature review we have been done here are from two different fields. As biomechanics and machine learning approaches are two different fields and there are almost very few researches have been done to combine these two fields, we will talk about the related works in these two fields separately.

2.1 Biomechanics Approaches for COM Estimation

Biomechanical studies basically rely on estimation of COM, center of pressure, base of support etc. A mixture of those is also used to evaluate balance, postural stability in various movements important in biomedical studies such as upright stance comparison for healthy and aged individuals for fall detection and other treatments. In [8, 17] COM and COP have been simultaneously compared to understand the imbalance in movement while standing. In [15] horizontal COM displacement of total body COM has been estimated to evaluate balance and posture while standing. In [7] COM motion has been estimated to evaluate

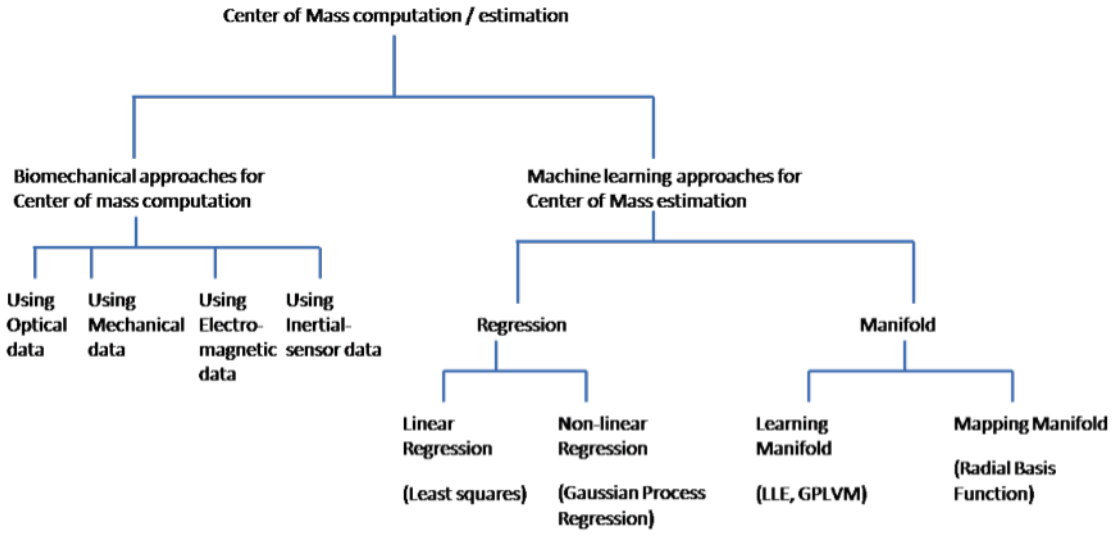

Figure 2.1: Literature review hierarchy to compute and estimate COM.

balance while stepping over obstacles. In [18] BOS has been measured with help of COM.

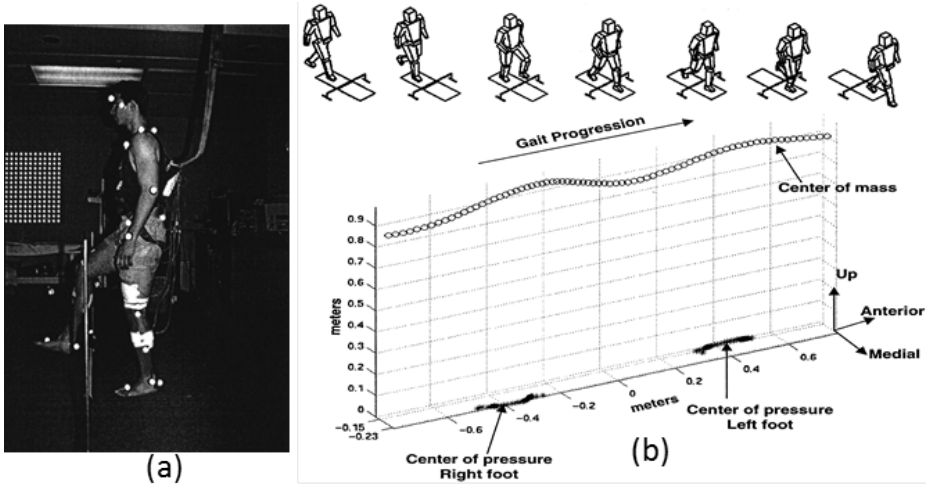

Figure 2.2: Biomechanical COM computation, (a) optical data collection with markers, (b) COM trajectory [7].

Biomechanics and biomedicine, sports, rehabilitation of impaired motion, joint force evaluation for human locomotion analysis, estimating joint forces and moments all these important research require the analysis of human motion indispensably [3]. Character animation, computer graphics, manufacturing humanoids, computer vision etc. are the

field where human motion analysis is the key. The pattern of propagation of COM, COP, BOS etc. in certain direction can detect various human motions and speed [19] as well as can define difference in human motions such as walking, running, jumping and many other motions [13], termination and initiation of gait [17]. Analyzing COM, COP, BOS of Biomechanics under different conditions help detecting activities like walking and running [13], fall risk assessment [7, 8, 17], age effect analysis and rehabilitation [20], clinical gait analysis [3, 16], gait and posture stabilities [8, 17, 21] in the areas of biomechanics and biomedicine.

2.2 Mapping-based Manifold

Human motion data is obviously high-dimensional which consist of all the joints motions or all the limbs motions together. Although human motion data has high dimension yet we can have human motion characteristics basically spanning a lower dimensional manifold because of the relative positions of the joints or body segments to each other. Another reason is cyclic nature of most of the movements or motions. In [22] 3D body configuration has been recovered from silhouettes by manifold where a strong prior is provided to relate the shape space with body configuration space. In [23] Gaussian Process Dynamic Model has been used to estimate a latent space or manifold simultaneously with nonlinear dynamic observation model. In [24] continuous 3D body configuration due to relative view variability has been tracked from a model which ties body kinematics manifold and visual manifold together.

In [25] object recognition has been done by visual manifold where visual appearance is changing under certain view and illumination. In [26] it has been shown that from a certain point of view the motion lies in 1D closed manifold and for a fixed posture the view varies along a circle and the changes of view lies in 1D closed manifold, having this observation [where view and posture both are 1D manifolds] in hand one can say that when view and posture will change together the observation will lie on a torus manifold [27]. In [26], the

tracking of the posture, 3 different low dimensional representation have designed where first one is view-shape invariant, second one is configuration-shape invariant and the third one is configuration-view invariant. In this work view and body configuration of human motion data have been tracked from a single monocular camera by a manifold where kinematic and visual data tied together with a parameterized generative mapping function where a person can change his/ her pose with respect to the camera but the view is limited to a one view circle [24].

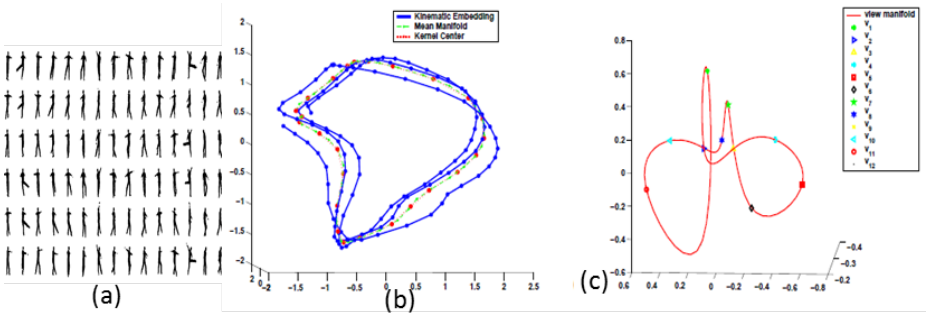

Figure 2.3: RBF mapped manifold (a) View variation, (b) Kinematic manifold, (c) View manifold [24].

Radial basis function (RBF) is commonly used in mapping based manifold to map HD data onto LD (manifold). RBF approximate multivariate function based on a single uni-variate function where multivariate function is defined as the linear combination of the uni-variate functions [28] and the value of RBF indicates distance from the origin or distance from some other points, this is how it has been radialised so that more than one dimension can be considered [28]. We can only efficiently approximate the multivariate functions having values in some discrete and finite number of points by RBFs [29, 30]. Frequent display of multivariate functions by simple functions is very common in the area of computer graphics [31], neural networks [28]. Scattered data interpolation by RBF is required in many areas: repairing mesh, reconstruction of surface, range scanning, geographic surveys, medical data, visualizing 2D or 3D field, Artificial Intelligence, image warping, morphing, registration, motion data extrapolation [31, 32].

In [27] a joint manifold (torus-shaped) was introduced to track 3D kinematic and pose (visual representation of the corresponding kinematic) together. Torus is a supervised structure for human motion data where human motion is periodic and closed 1D manifold. In [22] the view variability of human motion lies horizontally in the torus and body configuration lies vertically in a torus. In [32] a torus-based manifold has been designed to estimate human motion using RBF, this torus is purely mapped and does not include any data influence.

2.3 Learning-based Manifold

Learning based manifold involves data influence instead of mapping a manifold purely. The example of learning a manifold includes pose estimation using GPLVM (Gaussian Process latent variable model) [33], Eigenmaps [34] etc. The GPLVM is the generalization of the probabilistic PCA that estimates the joint density of the data sample and their latent coordinates. GPLVM and its variants Gaussian Process Dynamic Model [23], Scaled GPLVM [23], Back-constrained GPLVM [35], Balanced GPDM [36] and hierarchical GPLVM [37] were used for relatively small training data sets of particular style such as walking of a particular subject.

To estimate more than one factor such as pose along with gait, two non-linear Gaussian Process Kernel methods was proposed in [23], OD-GPLVM (introduced a controlling variable), Switch GPLVM (imposes a graphical model), LL-GPLVM (encouraging a desired topology) merge multiple motions manifolds into one latent space [32]. But none of these patterns can deal with new motion style or pattern estimation or extrapolation from training data.

In [32], three different torus have been introduced, torus-based (purely RBF mapped), torus-constrained (learnt manifold using LL-GPDM) and torus-like (learnt manifold using two step local-global GP learning algorithm). In this work, a torus is a joint gait-pose manifold (capable to estimate unknown gaits) where torus-constrained and torus-like manifolds

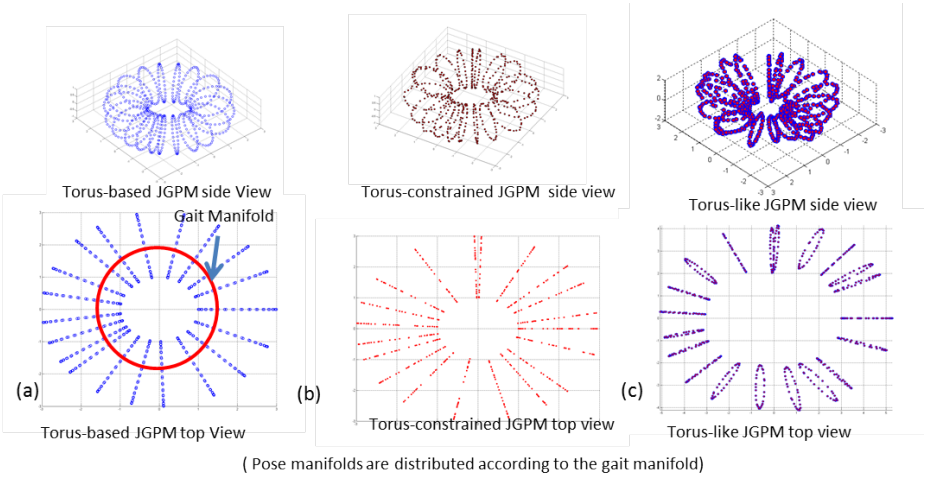

Figure 2.4: (a) Mapped manifold, (b) Learnt manifold, (c) Learnt manifold [32].

are not purely torus shaped rather they are encouraged to maintain the torus-shape while preserving the data influence.

2.4 Gaussian Process Regression

In statistics and machine learning areas Gaussian process regression is an important tool to interpret and analyze for complex datasets where Gaussian process provides a form of supervised learning of the data in the form of regression (needed for continuous outputs) and classification (needed for discrete outputs) [14].

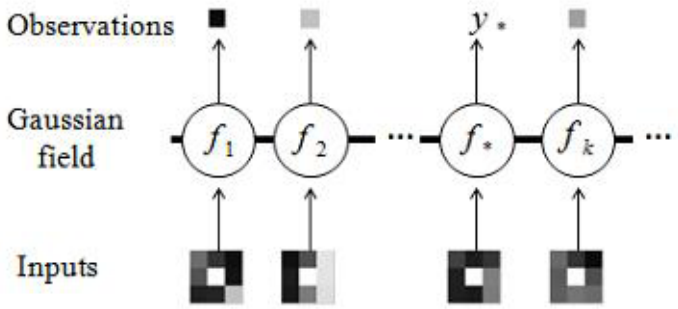

Figure 2.5: Gaussian Process Regression [14].

Gaussian Process is a powerful tool in the area of machine learning which can handle lots of real world problems by providing a representative probabilistic model for the

problem. Gaussian process regression is not completely non-parametric rather it is a less-parametric technique where we have to have some very simple and basic assumption about our function [38]. In Geostatistics Gaussian Process is used to analyze large amount of datasets, Gaussian process is an essential tool for supervised, unsupervised, reinforcement learning, in principal component analysis Gaussian process works as a prior about intuition, system identification and control [39], rendering music performance [40], optimization and many other areas are applying Gaussian process in practice.

CHAPTER 3

COM Computation

Computation of COM requires body segment parameters to quantify the body. There are several techniques to determine the body segment parameters, these techniques can be categorized mainly into four groups which are: Cadaver studies, Mathematical modeling, Scanning and imaging techniques and kinematic techniques [12]. Mathematical modeling models most of the segments as frusta of right circular cones which requires 242 direct anthropometric measurements to determine the inertial properties of the segments of a 42-DOF, 17-segment body model, scanning and imaging techniques require scanning the whole living body with various radiation techniques and kinematic techniques require a body part to be set into oscillation with an instrumented spring to quantify the body segment parameters [12].

In this research to compute 3D COM we have used Dempsters technique. Computing inertial properties such as body mass, COM, moment of inertia are difficult for living persons whereas it is much easier for rigid robot bodies since all the segments can be separated from the body which makes the measurements of body segment parameters easier [41]. Dempster collected data from eight cadavers; he segmented the cadavers according to his own method and then recorded the lengths, masses and volumes carefully. He computed the segmental COM using a balancing technique and segmental moment of inertia using a pendulum technique [12].

Dempster created a table where he showed the mass of a segment as a ratio of the mass of that particular segment to the total body mass. He presented the length of the segments as a proportion of length of the COM and the total segmental length. As a result, Dempster

could create a table of segmental mass, length and other inertial properties basically irrespective of individual persons segmental characteristics. This table was modified by Miller and Nelson in 1973, Plagenhoef in 1971 and D.A. winter in 1990 and in this research we have used this table to acquire the body segment parameters to apply and use Dempsters technique to compute COM [12].

Segment	**Endpoints (proximal to distal)**	**Segmental mass/total mass** (P)^b	**COM/segmental length** ($R_{proximal}$)^c	(R_{distal})^c
Hand	Wrist center to knuckle II of third finder	0.0060	0.506	0.494
Forearm	Elbow to wrist center	0.0160	0.430	0.570
Upper arm	Glenohumeral joint to elbow center	0.0280	0.436	0.564
Forearm and hand	Elbow to wrist center	0.0220	0.682	0.318
Upper extremity	Glenohumeral joint to wrist center	0.0500	0.530	0.470
Foot	Ankle to ball of foot	0.0145	0.500	0.500
Leg	Knee to ankle center	0.0465	0.433	0.567
Thigh	Hip to knee center	0.100	0.433	0.567
Lower Extremity	Hip to ankle center	0.1610	0.447	0.553
Head	C7-T1 to ear canal	0.0810	1.000	0.000
Shoulder	Sternoclavicular joint to glenohumeral joint center	0.0158	0.712	0.288
Thorax	C7-T1 to T12-L1	0.216	0.82	0.180
Abdomen	T12-L1 to L4-L5	0.1390	0.440	0.560
Pelvis	L4-L5 to trochanter	0.1420	0.105	0.895
Thorax and abdomen	C7-T1 to L4-L5	0.355	0.630	0.370
Abdomen and pelvis	T12-L1 to greater trochanter	0.281	0.270	0.730
Trunk	Greater trochanter to glenohumeral joint	0.497	0.495	0.505
Head, arms, and trunk	Greater trochanter to glenohumeral joint	0.678	0.626	0.374
Head, arms, and trunk	Greater trochanter to mid-rib	0.678	1.142	−0.142

Table 3.1: Dempster's body segment parameters [12].

3.1 Dempster's Technique to Compute COM

Computing total bodys COM requires computing the COM for each segment. In our research we have covered the whole human body as a unification of 16 segments. After computing the COM for all the segments, total bodys 3D COM position has been computed.

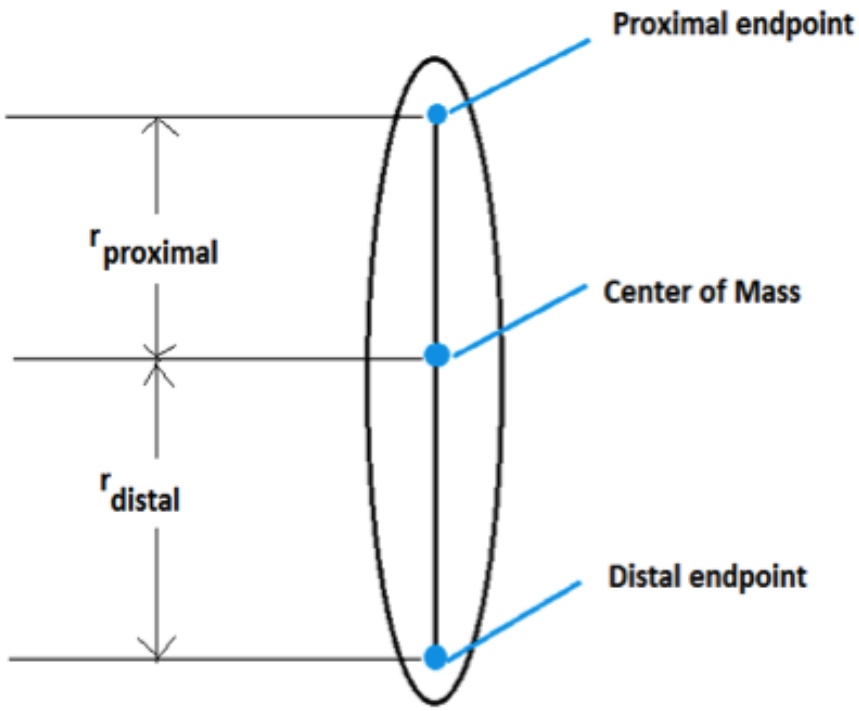

Figure 3.1: Proximal, distal end points for a certain segment [12].

3.1.1 COM computation for each segment of the body

Dempster(1955) simplified the process of computing a segments center of gravity by defining $R_{proximal}$ and R_{distal} which are the R-values of the segments length (l) where it is given that $r_{proximal}$ and r_{distal} are the distances from proximal end point to the COM and distal end point to the COM respectively [12]. Considering l = length of the segment,

$$R_{proximal} = \frac{r_{proximal}}{l} \tag{3.1}$$

$$R_{distal} = \frac{r_{distal}}{l} \tag{3.2}$$

R-values determined by equation 3.1 and 3.2 are used to compute segmental COM. In this respect one can choose either $R_{proximal}$ or R_{distal} but usually COM is usually computed from their proximal end [41].

If we consider $x_{proxmial}$, $y_{proxmial}$, $z_{proxmial}$ as the 3D coordinates ofthe proximal end of a segment and x_{distal}, y_{distal}, z_{distal} as the 3D coordinates of the distal end of a segment then by the following equations can determine the x, y, z coordinates of the COM of a particular segment [12].

$$X_{cg} : X_{proximal} + R_{proximal}(X_{distal} - X_{proximal}) \tag{3.3}$$

$$Y_{cg} : Y_{proximal} + R_{proximal}(Y_{distal} - Y_{proximal}) \tag{3.4}$$

$$Z_{cg} : Z_{proximal} + R_{proximal}(Z_{distal} - Z_{proximal}) \tag{3.5}$$

$$R_{proximal} + R_{distal} = 1 \tag{3.6}$$

Equation 3.6 holds because these two ratios constitute the whole length ratio of a segment. If we want to get back to the original distance we have to use equation 3.7 [12].

$$r_{proximal} = R_{proximal}\ l \tag{3.7}$$

3.1.2 COM computation for the whole body

The COMs of all the body segments constituting a body is used to compute total bodys COM. Total COM can be considered as the weighted average of all the segment COMs.

$$X_{total} : \sum (P_s \times X_{cg}) \tag{3.8}$$

$$Y_{total} : \sum (P_s \times Y_{cg}) \tag{3.9}$$

$$Z_{total} : \sum (P_s \times Z_{cg}) \tag{3.10}$$

$$P_s = mass\ proportion = \frac{Segmental\ mass}{Total\ mass} \tag{3.11}$$

Where X_{total}, Y_{total}, Z_{total} are the 3D coordintaes of the total bodys COM point and P_s is the segmental weight of the corresponding segment comprising the body and $\sum P_s = 1$ [41].

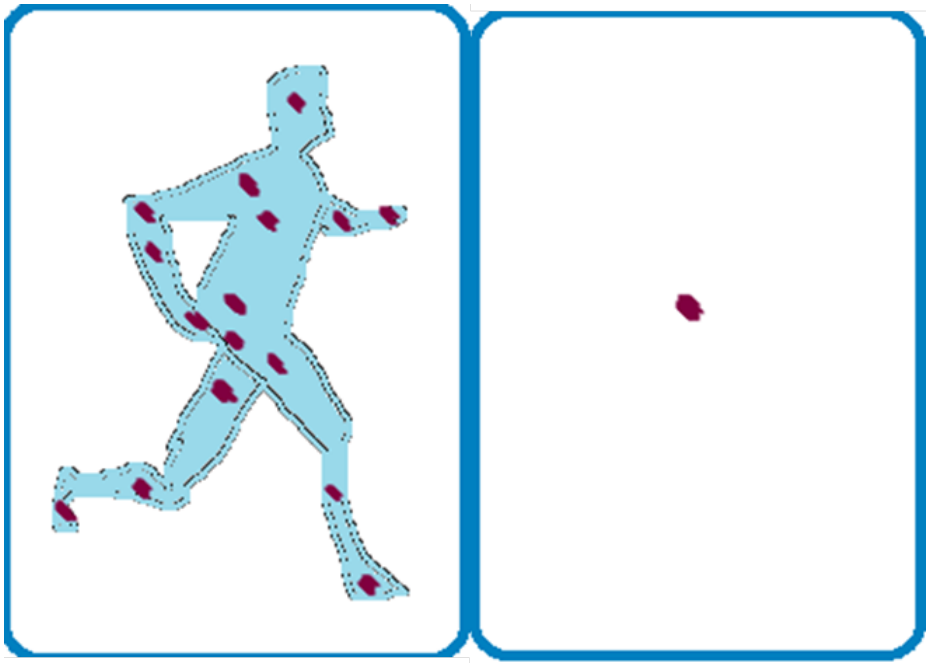

Figure 3.2: The total body COM is the weighted average of all the segmental COMs [12].

3.2 Motion Capture Data Info

In this research we have used Motion Capture Database created by Carnegie Mellon Graphics Lab to research on human motion data and compute COM during walking cycle. We have focused on motion capture data on normal walking. In this chapter we have described

how the motions of different subject are captured and what file formats we are using to deal the motions.

Motion Data Capture Environment: Motions are captured in a rectangular area (3m $\times$ 8m) placed in the center of the lab room around which 12 Vicon infrared MX-40 cameras were placed. The cameras are capable of 120 Hz video recording and 4 megapixel image resolution [42].

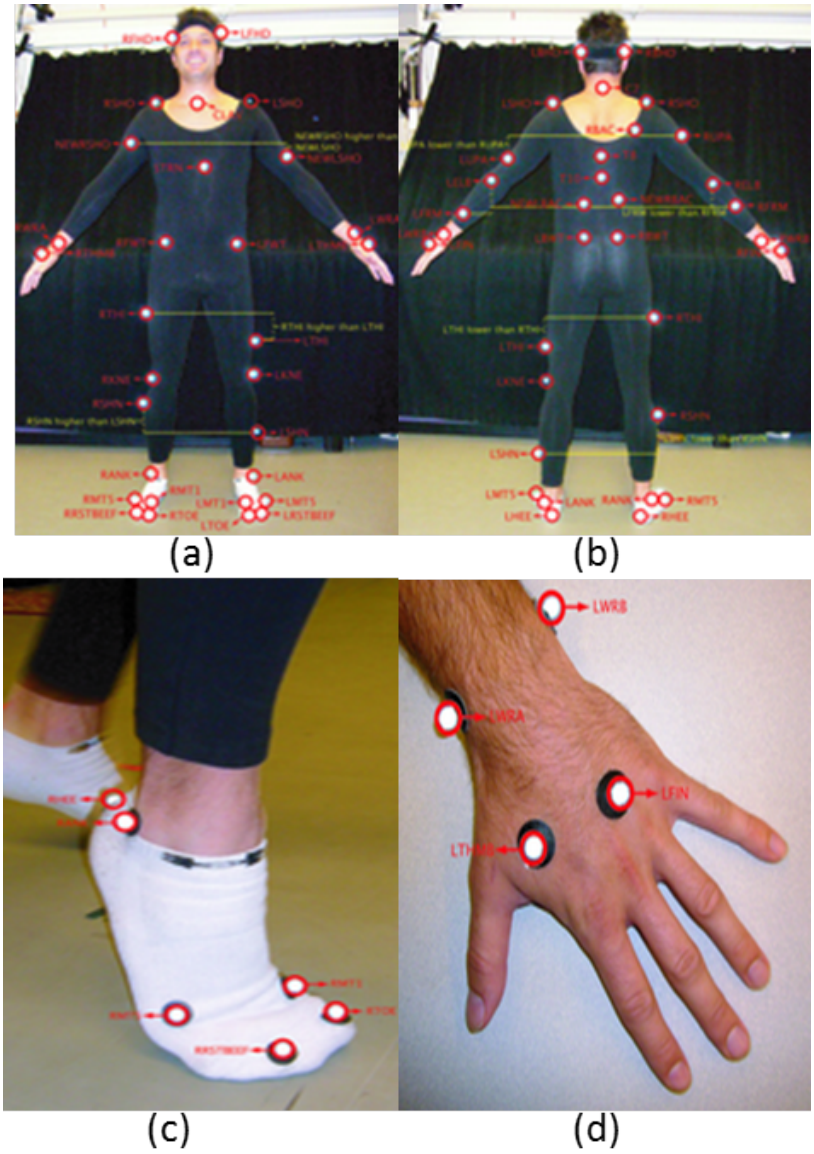

Figure 3.3: (a) Marker set from front view, (b) Marker set from back view, (c) Marker set on feet and (d) wrist marker set [42].

Marker positions, subject & skeleton: To capture a particular motion data from a particular subject, the subject (human) has to wear a black jumpsuit and 41 markers are taped on the suit over the entire body. The marker positions are recognized by the infrared rays from the markers. The images captured by all the cameras are triangulated to get 3D data [42].

3D data usage: One can use either the marker position (.c3d) or skeleton (.vsk/.v pair files)or movement (.asf/.amc pair files), among these formats we have used movement files (.asf/.amc) with the skeleton. If the subject is definite but is captured in different clips then different .amc formats are acquired. The camera data is processed by "ViconIQ" (Vicon software system) and stored as .vsk/.v. A .vsk of a skeleton is unique to each person because the segment lengths for each person are different. A .v depends on how many different clips of a certain person have been taken. By a software named Body Builder .vsk/.v pair is converted to .asf/.amc [42].

File format: The .asf/.amc format is ASCII and can be parsed. These files contain angles (Euler angles) to represent movement.

Data Unit: The unit of ASF files in CMU database is 0.45 and they are stored in inches. The data stored in ASF has to be multiplied with the following scale (1.0/0.45)(2.54/100.0).

3.3 Motion Capture Skeleton Model and Anthropometric Model

To compute COM we have used Dempsters technique. This technique requires a skeleton which has all the required segments for which segment parameters are provided in Dempsters body segment parameters table (Table 3.1). One problem we have faced in this part is the 31 joint motion capture skeleton model that we acquire from 41 marker model is not in the same format of the required anthropometric skeleton format.

Here in this section we have created a technique to match our motion capture skeleton model with the anthropometric body model.

It is easy to match up head, shoulder, upper arm, forearm, wrist, thigh, calf, foot all

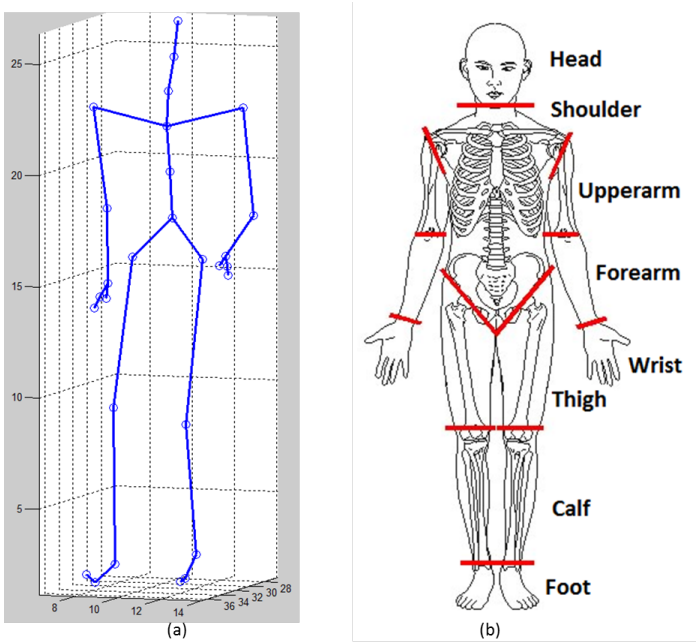

Figure 3.4: (a) Motion capture skeleton, (b) Anthropometric skeleton.

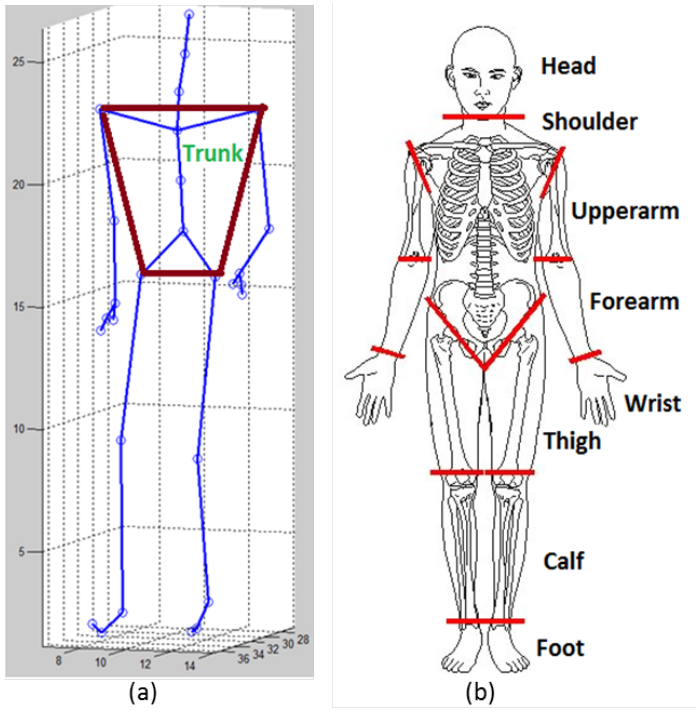

Figure 3.5: Trunk position in (a) Motion capture skeleton, (b) Anthropometric skeleton.

these anthropometric segments with our motion capture skeleton model segments. The segment trunk is not well defined in our motion capture skeleton model. Trunk is a complex

segment which roughly consists of thorax, abdomen and pelvis. In Dempsters table we have P_s for thorax, abdomen and pelvis individually as well as P_s for trunk as a whole. It is reasonable for us to use trunk as a full segment rather than using thorax, abdomen and pelvis individually because it is a complex task to define thorax, abdomen and pelvis individually in our motion capture skeleton model. We have taken the middle point of two shoulder joints which we have considered as the distal end point of trunk and the middle point of two hip joints (right and left) which we have considered as the proximal end point of trunk and then computed the COM for trunk.

3.4 COM Pattern from Global Motion Capture Data

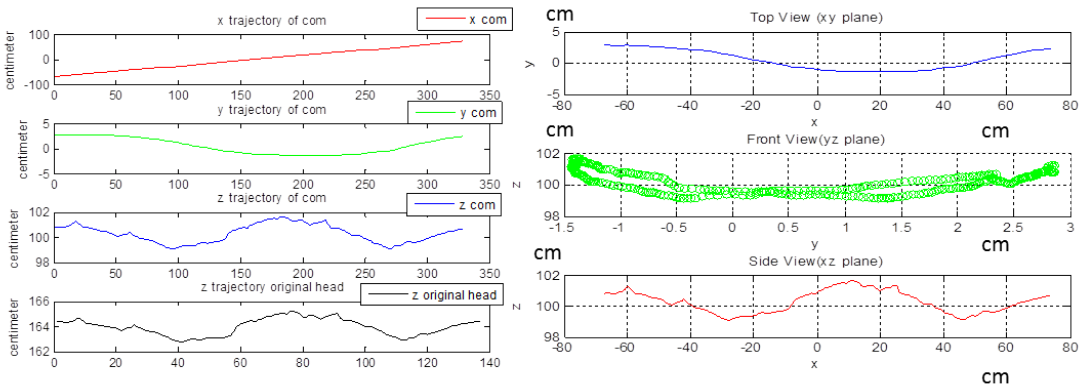

Figure 3.6: COM trajectory, Subject No: 35, Motion No: 03(one cycle period).

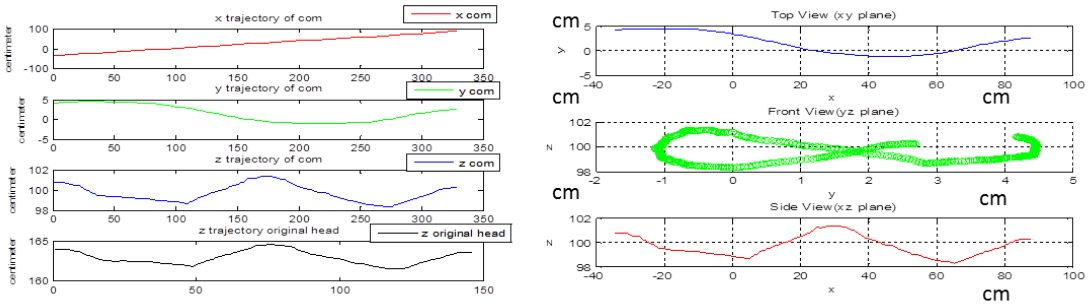

Figure 3.7: COM trajectory, Subject No: 16, Motion No: 16(one cycle period).

Motion capture data give us global motion data of the subject. Figure 3.6 and 3.7

portray some COM trajectories for x, y and z coordinates. X-trajectory of COM is just an increasing straight line, this is reasonable because the subject walks along X-coordinate. Z-coordinate represents the height of the subject. The Y-coordinate trajectory looks like a periodic cycle since the person is the walking straight along X-coordinate and there is no bias in the motion data while walking. For global motion data COM trajectory in the YZ plane is 8 shaped and the total width of the YZ is within 5 centimeter.

3.5 Human Gait and Pose

Human gait is the pattern by which human locomotion is achieved. In this research we are interested in gait during normal walking. Every Gait is different from person to person. Gait is characterized by speed, velocity, body segment movement pattern, kinematic energy cycle, surface contact speed and frequency; all these make one gait different from another gait [43].

Human pose or posture can be defined as a unit of gait. If we divide out gait into a number of frames then the combination of 3D body joint positions or relative body joint angles for a particular frame can be regarded as a pose or posture.

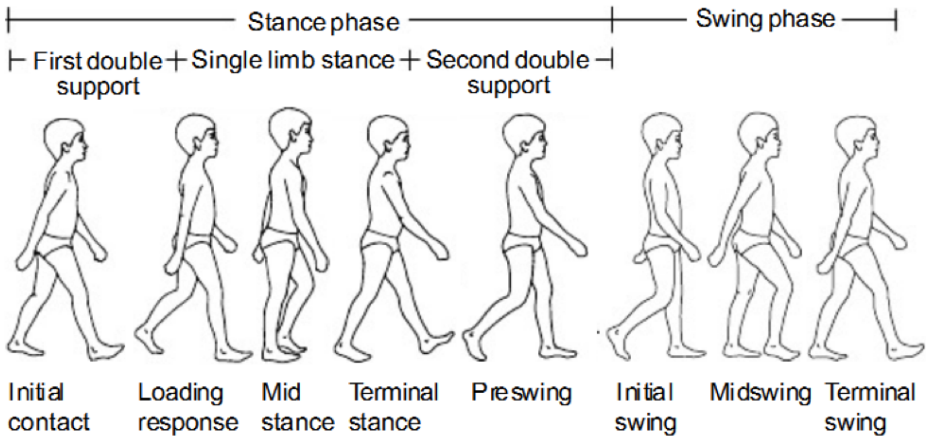

Figure 3.8: A cycle from a human gait. The gait is a combination of certain poses [43].

3.6 Periodicity of Human Gait

Human motion has cyclic nature because we see that after a while the same poses are getting repeated over and again. As a result, for any analysis purpose people take one cycle where poses are non-repetitive where we assume that in the successive cycles will have the combination of same poses during one cycle. Though this assumption may not be true yet it is reasonable for most people [43]. In biomechanics researches there are two phases in one gait cycle, they are stance phase and swing phase, these two phases continue to get repeated during the whole walking cycle. For this cyclic nature, most researches take one cycle of a gait into account for the analysis purpose.

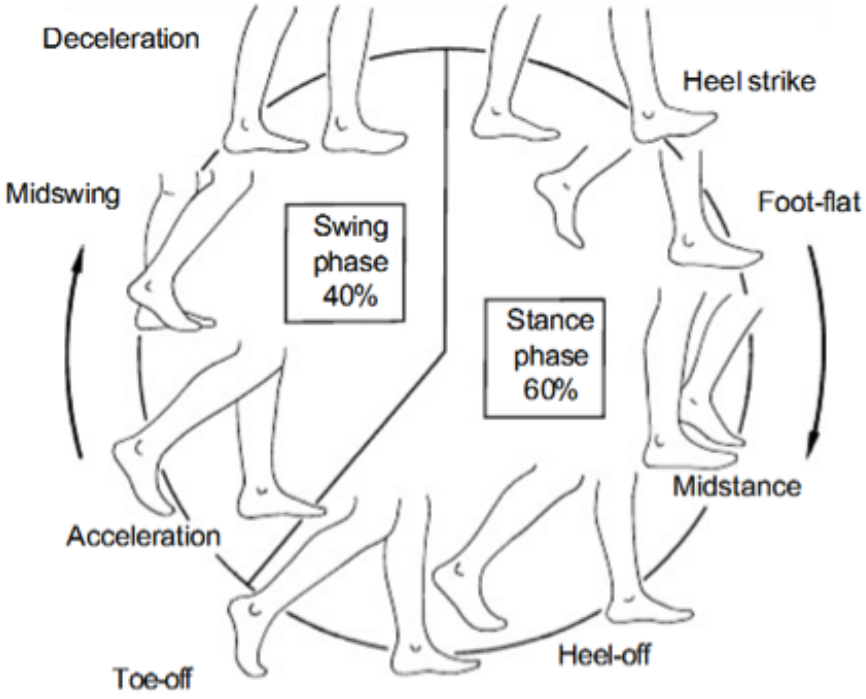

Figure 3.9: Eight events in one gait cycle [43].

There are two main phases: stance phase and swing phase. During stance phase, the foot is on the ground and in the swing phase the same foot is no longer in contact with the ground [43].

In biomechanics the gait cycle has been characterized into eight events or periods, five

events take place during stance phase and the other three events take place during swing phase. Heel strike (initiates the gait cycle, COM is at its lowest point in this event), foot flat (plantar surface of foot considered as the large area covering the arch of the foot is on the ground), midstance (COM is at its highest point), heel-off (the moment as the heel leaves the touch of the ground), toe-off (the stance phase comes to an end as the foot leaves the touch of the ground). Acceleration (moment when accelerates the leg forward), midswing (foot passes directly just under the body), deceleration (in preparation for the next heel strike slow the leg and stabilize the foot) [43].

3.7 Motion Capture Data Normalization

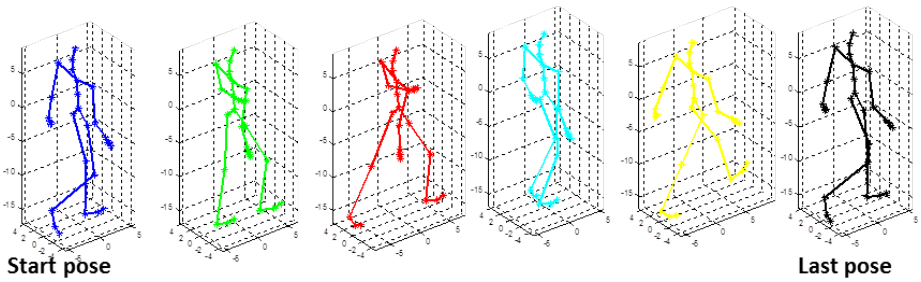

Figure 3.10: Sample poses from a full-cycle and normalized human motion (motion capture data).

Motion capture data requires normalization since we want to extrapolate new/unknown/test motion data from the training/known motions. We make all the motions taken under one skeleton so that all the joints motions vary in a certain range from gait to gait (person to person) to get a better extrapolation for a new or unknowngait/motion data. We take one cycle from a certain motion data and make all the cycles/gaits equal length (all the gaits have same number of poses/frames) so that the time information as well as speed information is taken out from the gait of all training individuals and the pattern of the cycle/gait is only brought into light. What we do is make the trajectory of hip zero. If we make hip positions of all the training gaits zero then we lose the global information about the motion

data and the motions of all the joints are just relative motion to each other. And also making the hip positions of all the gaits to be zero makes the X-coordinate-trajectory of any motion changing in a cycle or period which can compared with the motion on a tread mill which is responsible for producing a rotary motion. Figure 3.10 demonstrates a full-cycle normalized human motion where the cycle ends when the motion is about to repeat itself.

We can summarize our normalization process by the following steps:

1. Take one full cycle from a motion data of a particular subject and a particular motion data.

2. Take all the motions of different subject/skeleton under a certain skeleton to have all the joints motion vary in a certain range to get better extrapolation for new/unknown motion data.

3. Make the hip position of all the motions zero so that x, y, z coordinates of the motions vary in a cyclic way which also ensures better extrapolation because it can better represent the cyclic nature of a gait.

4. Make all the training and testing gaits to be equal length (same number of poses/frames) so that the time and speed information is gone, only the pattern of the COM trajectory is taken into account.

CHAPTER 4

Joint Gait and Pose Manifolds for COM Estimation

To estimate COM from human motion data we have mapped/designed a shared latent space or manifold. This shared latent/space or manifold is shared by human motion data and the corresponding COM at the same time, provides continuous space to extrapolate new/unknown human motions and the corresponding COM.

We have designed the latent space from noiseless training motions and COMs. Consequently when we extrapolate a noisy motion data from the manifold it tries to extrapolate the noisy motion data from the manifold designed from noiseless training motions. As a result we get a reasonable noise free extrapolated motion data and the corresponding COM from the manifold. In case of noisy or incomplete motion data, direct computation of COM could not be helpful where high accuracy of COM computation is needed in the areas such as diagnose postural and gait stability, detecting fall risk etc. COM estimation may have some advantage over direct computation of COM from noisy motion data.

4.1 Joint Gait and Pose Manifolds (JGPMs) for Motion Modeling

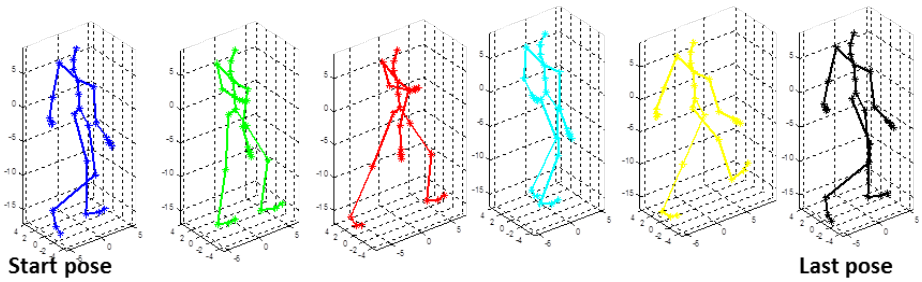

Figure 4.1: Our gait cycle consists of one midswing position of right leg to other midswing.

Our purpose is to achieve a manifold which will accommodate both gait and pose variables along with COM and will estimate unknown gait. Both Gait and pose have their own manifolds. Our new joint manifold has to be a structure which will allow both the manifold topologies of gait and pose simultaneously. Pose manifold is well defined closed 1D manifold because of its cyclic nature during a full cycle of gait. In our research we are intended to know unknown subjects gait which will be extrapolated from the manifold. For gait manifold we have chosen 1D closed manifold by using shortest path.

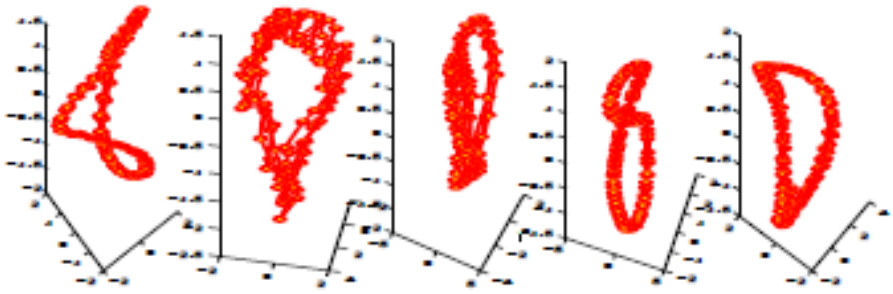

Figure 4.2: 1D closed pose manifold [22].

Shortest path problem is a problem where we have to find the shortest path among all the possible paths which connects two specific nodes/vertices in a graph. In the graph every edge/connection between two particular nodes/vertices must have a weight associated with it. The graph could be directed or undirected. The best example of shortest path problem could be the travelling salesman problem where the salesman travels the cities where the total distance he travels has to be the shortest.

In this research our problem is more like the travelling salesman problem because we do not have two specific gaits, rather we want to find a path which is the shortest of all the paths and will give us a smooth transition from gait to gait (Fig 4.3). In our case the graph is undirected. Given an undirected weighted graph where the set of vertices V and the set of edges is E, then the sum of the weights of the edges constructing the shortest path is the shortest/smallest [32].

The intuition working behind 1D closed manifold for gait is the characteristic of an ideal

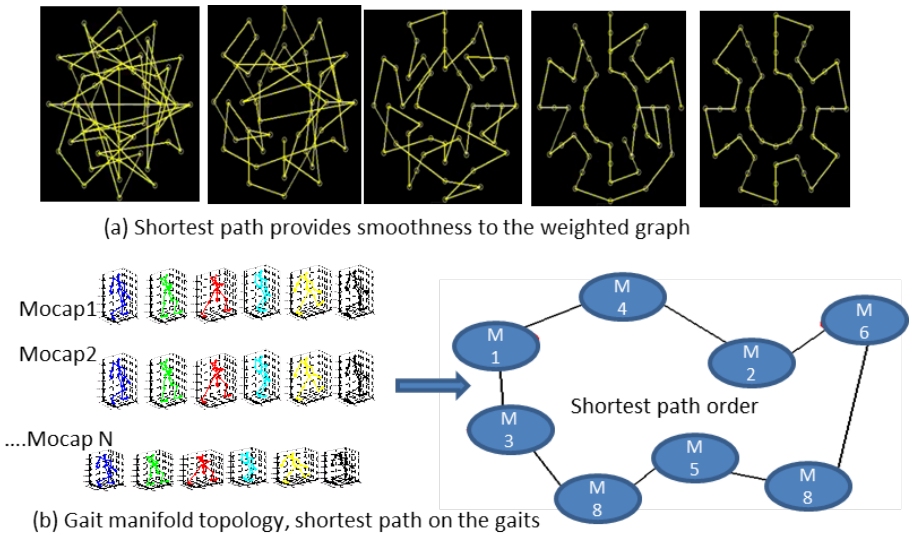

Figure 4.3: Gait manifold topology by shortest path [44].

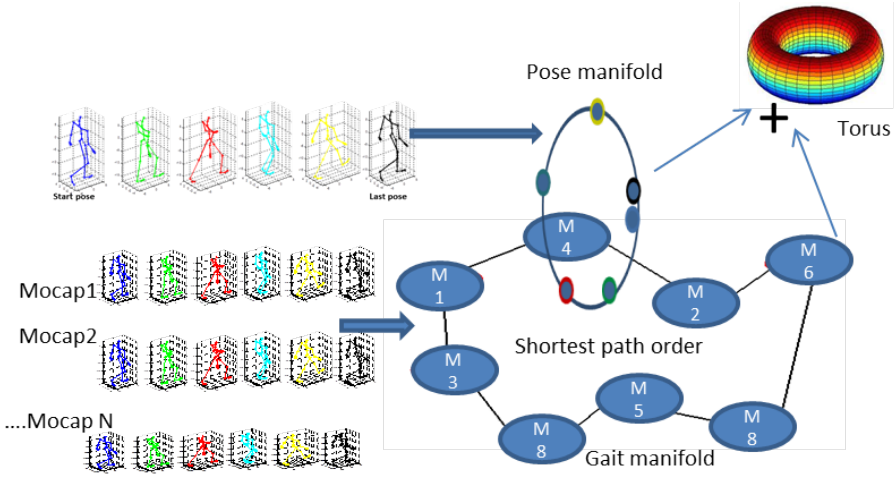

Figure 4.4: Gait and pose manifold topology followed by Torus.

manifold where similar data points tend to be placed with a short distance. The distance from one gait to another gait in our case is the normal eucledian distance where each gait is a bunch of 3D positions of 31 joints. Before taking the distance we normalize all the training gaits by taking the hip positions of all the gaits to be the same. Then we take the difference of two gaits to measure the kinematic dissimilarity. Shortest path algorithm has been applied on these gait distances which gives us a smooth ordering of the gaits. It is our assumption that the shortest path finds a path through the gaits with minimum travel

distance, as a result we get a smooth path which can be an efficient manifold topology for gaits from different persons where tow gaits having small distance are placed closed to each other. As a result to combine two 1D closed manifold Torus is the best structure to unify those two gait and pose variables (Fig. 4.4). This tours structure is the joint gait pose manifold (JGPM) which will be used later to estimate COM.

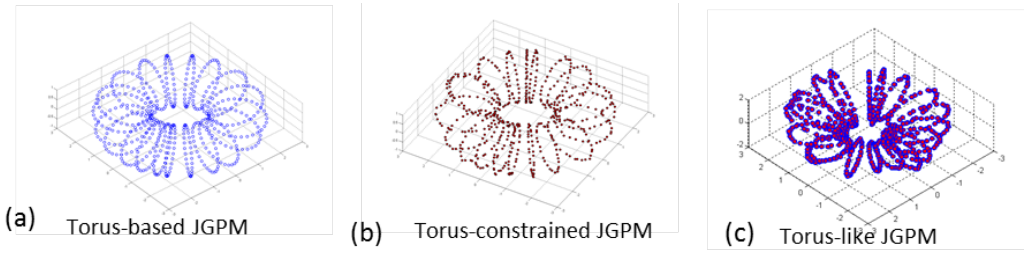

Figure 4.5: Torus-based, Torus-constrained and Torus-like JGPMs [26].

In [26] three different JGPMs (Joint gait pose manifolds) have been implemented to estimate human motion data. In the first JGPM no data influence has been learnt, it is just a pure mapping from human motion data to manifold. The first JGPM is learnt via non-linear mapping by RBF as shown in equation 4.1.

$$y_{(i,j)} = \mathbf{B} \cdot \Phi(x_{(i,j)}), \tag{4.1}$$

Where $\Phi(.)$ is a non-linear kernel function, $y_{(i,j)}$ is the gait having a pose id i and gait id j and B is the mapping matrix. The second JGPM is torus-constrained JGPM. LL-GPDM has been used to learn the latent space where pose and gait are the controlling latent variables. The latent variables are optimized by minimizing $\sum_{i=1}^{N} ||\tau(i,j) - \mathbf{W}\tau(m,n)||^2$, where $\tau(i,j)$ is the corresponding latent position of $y_{(i,j)}$ to get the weight matrix $\mathbf{W}$ and this minimization is done by equation 4.2 where $\mathscr{L}_d$, $\mathscr{L}_l$, $\mathscr{L}_s^j$, $\mathscr{L}_p$, $\mathscr{L}_w$ are negative log likelihoods and $\mathscr{L}_s^j$ is learnt for every gait respectively [26]. Torus-constrained JGPM is also an ideal torus figure 4.5(b).

$$\mathscr{L}_d = \mathscr{L}_l + \sum_{j=1}^{j=N_g} \mathscr{L}_s^j + \mathscr{L}_p + \mathscr{L}_w \tag{4.2}$$

In the third torus, two-step local global GP learning algorithm has been used to learn a JGPM (torus-like) [26]. This JGPM is not a strict torus figure 4.5(c), rather it is a torus-like manifold where the torus shape has been encouraged but not forced. This independence has given the most meaningful LD gait-pose manifold. In the first step of this algorithm pose manifold for each gait has been learnt by equation 4.3 and then gait variability is characterized by equation 4.4 where pose manifold structure is preserved locally.

$$\{G_i, \Phi_i\} = \arg\max_{G_i, \Phi_i} \; p(Y_i|G, \beta_i)\; p(G_i|\alpha_i)\; p(\alpha_i)\; p(\beta_i) \tag{4.3}$$

$$x_{ij} = R_i(\gamma_x, \gamma_y, \gamma_z) T_i(\rho) g_{ij}, \tag{4.4}$$

Where G_i is the latent space for gait y_i, Φ_i is the hyper parameter set including β_i and α_i, $P(G_i|\alpha_i)$ is the dynamic model, R_i and T_i are rotation and translation matrix for G_i, and $\gamma_x, \gamma_y, \gamma_z$ are the rotations along x, y, and z respectively.

4.2 Shared Torus-based JGPM for Joint Motion and COM Modeling

We can extrapolate or interpolate intermediate motions from the manifold/latent space. It is our assumption that on the manifold each point maps to a human pose of a certain human gait as well as maps to the corresponding COM (for that pose/frame) at the same time since the COM has a correlation with the pose/frame from which it has been calculated. The extrapolation of the test/unknown motion data as well as the COM is an added advantage of this shared latent space or manifold.

Radial Basis Function: Radial basis functions approximate multivariate function based on a single uni-variate function $\Phi(||x - c||)$. Multivariate function is defined as the linearly combination of the uni-variate functions [28].The uni-variable of radial basis function indicates distance from the origin or distance from some other points, this is how it has been

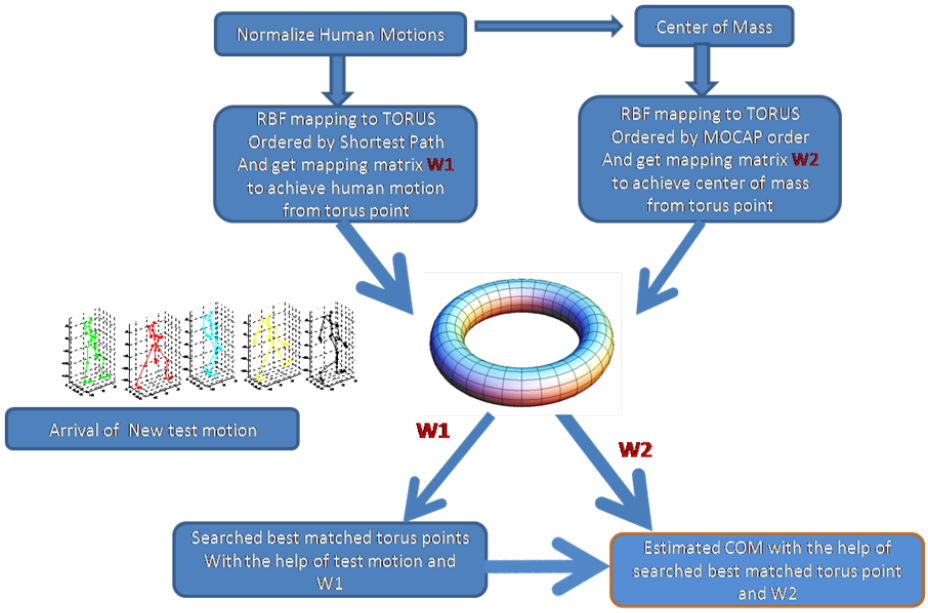

Figure 4.6: Flow-chart for estimating COM with shared latent space shared by jointly gait-pose and COM.

radialized so that more than one dimension can be considered [28]. We can only efficiently approximate the multivariate functions having values in some discrete and finite number of points by radial basis functions [29, 30].

Standard radial basis function types are given below:

$$\text{Gassian: } \phi(r) = e^{-\sigma^2}$$
$$\text{Reciprocal multi quadric: } \phi(r) = \frac{1}{\sqrt{r^2 + c^2}}$$
$$\text{Thin plate spline: } \phi(r) = r^2 \log(r),$$

Where $r = ||x - rbf\_center||$ is the distance from rbf center point to the training data point. For reciprocal multi quadric c is a scale parameter which could be adjusted to get better function approximation. In case Gaussian kernel hyper parameter $c \neq 0$ also has to be adjusted for better approximation.

Among lots of popular choices of radial basis functions such as thin-plate spline is good for achieving smooth function constituting of two variables. Gaussian kernel is ideal for neural networks and multi quadric kernel is good for topological data [31]. We can use any

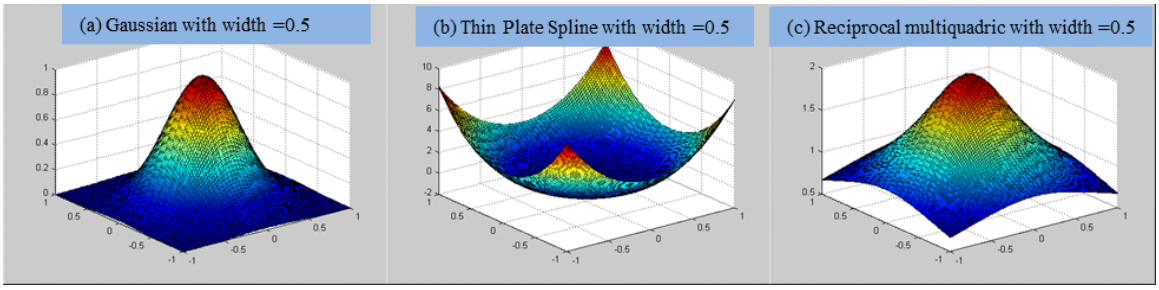

Figure 4.7: Some standard radial basis function types [45].

form of radial basis function. It is difficult to define the properties of the RBF suited for predicting a continuous surface for an arbitrary basic function. Gaussian and multi quadric kernels have an advantage over thin plate spline kernel since thin plate spline require not co-planar dataset whereas Gaussian and multi-quadric do not put any restriction on the location of data points. In other words, there is no such restriction that the data lie on any sort of regular grid required by RBFs [31]. In this research we have chosen Gaussian radial basis function to have a continuous representation of the torus surface to extrapolate new/unknown human motion data and the corresponding estimation of COM.

In general RBF is a function of the following form

$$s(x) = \sum_{i=1}^{N} \lambda_i \phi \left(||x - x_i|| \right), \tag{4.5}$$

Where

- x_i are the centers of the manifold surface which represents the manifold
- x are points at which we want to evaluate our approximation of the higher dimensional function.
- Φ is the RBF, Φ is uni-variate which takes the distances from data point to the radial basis centers as its unit variable to approximate he higher dimensional function.
- $||.||$ is the norm of the distance, usually the distance is eucledian but any form of useful or meaningful distance could be used.

- λ is the scalar parameter which has to be adapted for better approximation.

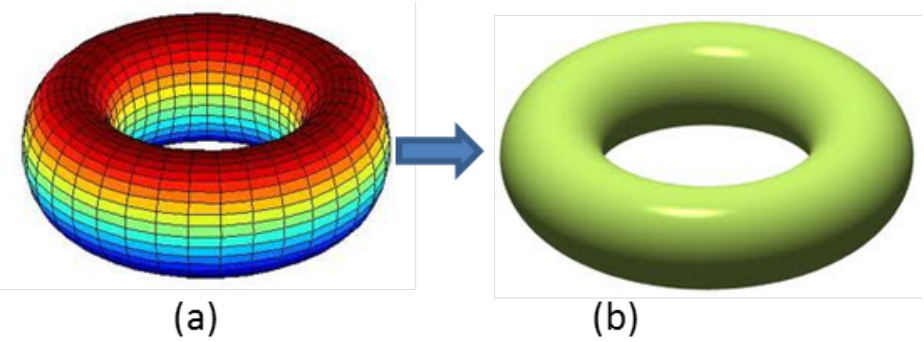

Figure 4.8: Radial basis function provides us with a continuous approximation (showed in figure b) of the higher dimensional function, where we know the higher dimensional function for only few discrete points (showed in figure a).

Gait and pose placement on Tours: We have designed an ideal torus. The torus is basically is a continuous representation of two variables, gait and pose. The manifold therefore is a 2D torus surface which is regulated by pose variable vertically and gait variable horizontally or vice versa. Let on the pose variable is represented by p and the gait variable is represented by g respectively. For a particular pose id p and gait id g, a latent point $T^i_{(p,g)}$ is a unique point on the torus manifold surface defined as

$$T_{(p,g)} = [(r_h + r_v \cos(g)) \cos(p), (r_h + r_v \cos(g)) \sin(p), r_v \sin(p)]^T, \tag{4.6}$$

Where $(p, g)\epsilon[0, 2\pi]$ and r_h and r_v are the horizontal and vertical radius of the torus respectively.

Let we have total G number of training gaits and each gait has P number of poses. Radial basis function maps our training gaits on torus. Each point on the torus (lower dimensional space) corresponds to a pose of a certain gait (higher dimensional space). Consequently each higher dimensional function which is a pose in our case has a gait id (which gait it belongs to) and pose id (which particular pose it is).

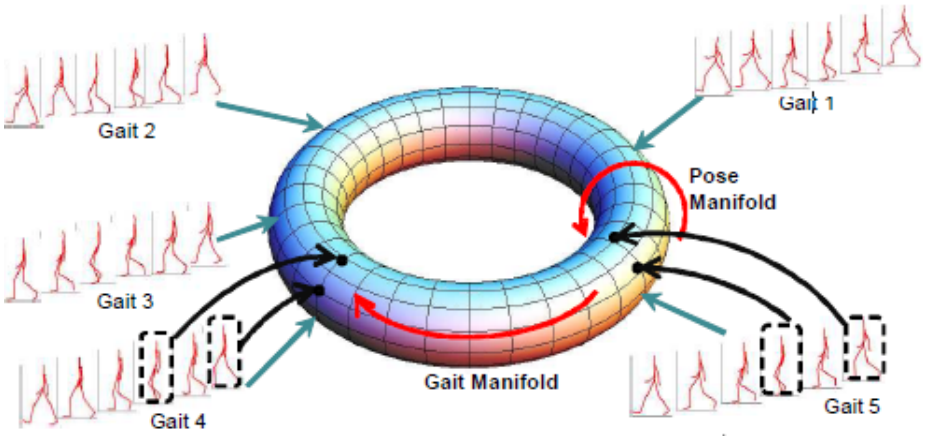

Figure 4.9: Gait and Pose placement on torus [32]

Let Y is the set of all training gaits, $Y = \{y_{i,j}\}$, where each element $y \epsilon Y$ has corresponding gait id i and pose id j. If t_{ij} is the point on tours surface representing gait $y_{i,j}$ then we can get the higher dimensional function y from the radial basis function equation 4.7

$$y_{ij} = B.\Phi(t_{ij}), \tag{4.7}$$

Where **B** is the weight matrix in terms of the weight decided by each of the radial basis centers representing the manifold surface and Φ is the uni-variate radial basis function where the uni-variable is the distance of a training point from each radial basis center defined by

$$\Phi(t_{ij}) = [\phi(t_{ij}, c_1), ..., \phi(t_{ij}, c_m)], \tag{4.8}$$

Where $C = c_1, \cdots, c_m$ is the set of m radial basis centers on the torus surface.

The RBF centers correspond to the mean location of Gaussian density function. If the centers are too close to each other there would be too much overlap in the functions. If the centers are very far from each other, there would be no overlap in the functions and the approximation of the function in the non-overlapped area would be zero. As a result, to get a better approximation of the function there should be a well distribution of the centers to have a better approximation of the function where the centers are neither far apart nor

too close. The variance of the RBF function determines how wide the Gaussian function provides its influence.

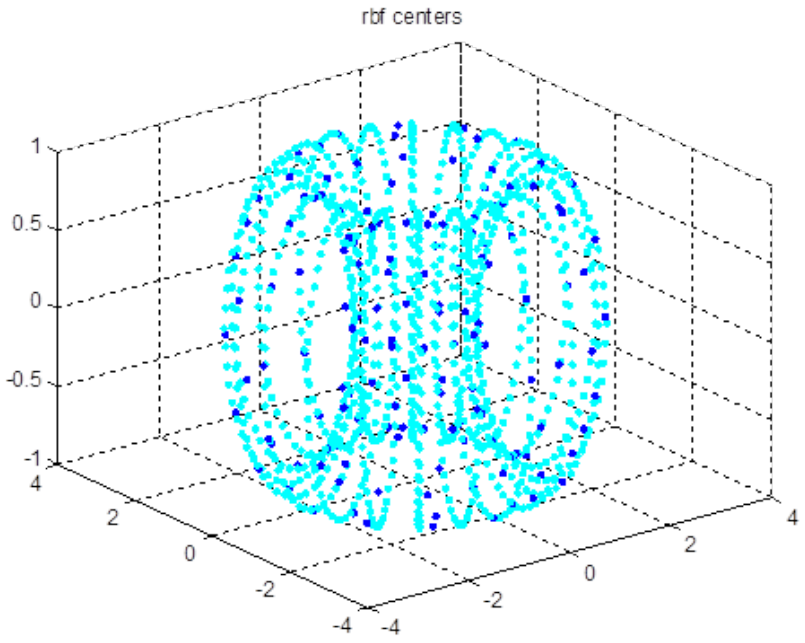

Figure 4.10: Radial Basis Centers representing torus surface

RBF mapping from COM to Torus: Let we have total G number of training gaits, each gait has P number of poses and G number of COM which are training COMs computed from training gaits. Radial Basis function maps our training gaits on torus as well as maps training COMs on to tours. Each point on the torus (lower dimensional space) corresponds to a pose of a certain gait (higher dimensional space) and the corresponding COM for that frame. Consequently each higher dimensional function (which is a pose in our case) and the corresponding COM both have a gait id (which gait it belongs to) and a pose id (which particular pose it is).

Let C is the set of all training COMs, $C = \{c_{i,j}\}$, where each element $c \, \epsilon \, C$ has corresponding gait id i and pose id j. If t_{ij} is the point on tours surface representing COM $c_{i,j}$ then we can get the higher dimensional function COM cfrom the radial basis function equation defined by

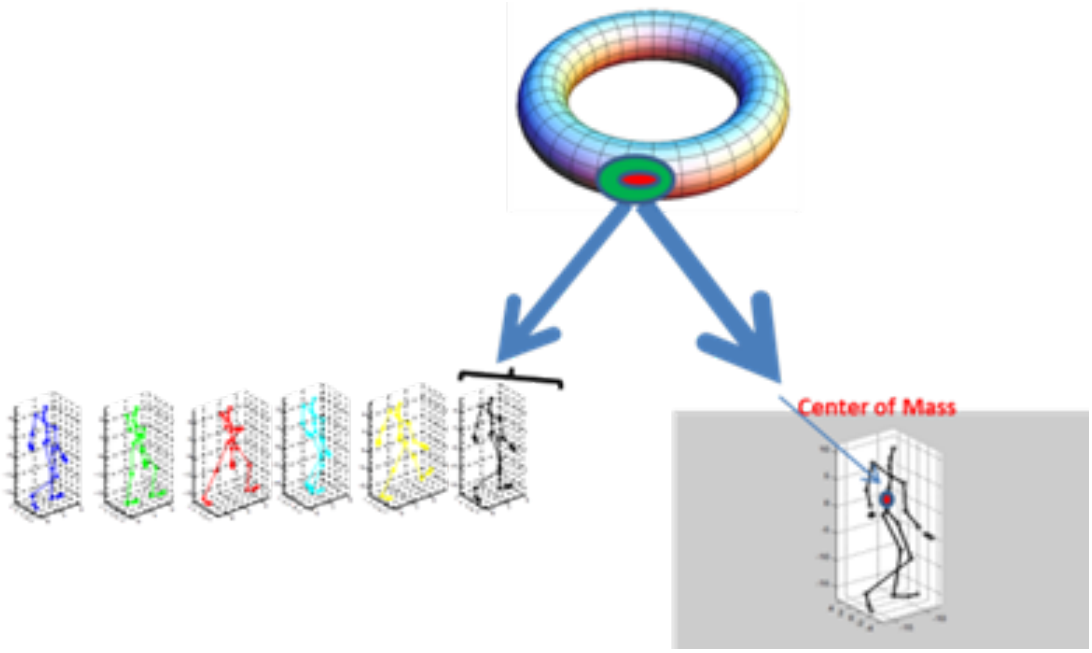

Figure 4.11: motion data and COM extrapolation from torus.

$$c_{ij} = B.\Phi(t_{ij}), \tag{4.9}$$

Where **B** is the weight matrix in terms of the weight decided by each of the radial basis centers representing the manifold surface and Φ is the uni-variate radial basis function where the uni-variable is the distance of a training point from each radial basis center defined by

$$\Phi(t_{ij}) = [\phi(t_{ij}, c_1), ..., \phi(t_{ij}, c_m)], \tag{4.10}$$

Where $C = c_1, \cdots, c_m$ is the same set of m radial basis centers we picked for mapping motion data on the torus surface. .

4.3 Torus-based COM Estimation

Best matched torus point is the point by which the computed pose has the lowest error. Using window based approach we search the torus for a motion data to estimate COM. The position of the window for the next pose is decided by the current poses best matched torus

point. To search the next best matched torus point we take the current best matched torus point as the center (figure 4.13).

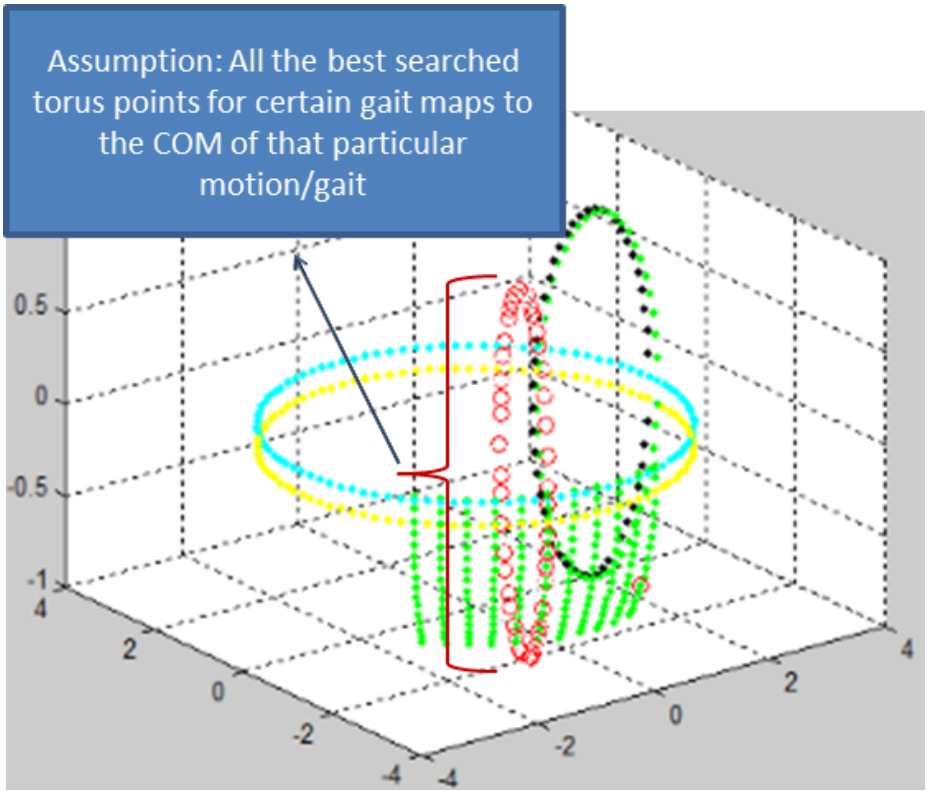

Figure 4.12: Assumption for shared latent space: a particular motion and its COM share the same points on shared torus surface.

We can extrapolate or interpolate intermediate motions from the manifold/latent space. It is our assumption that on the manifold each point maps to a human pose of a certain human gait as well as maps to the corresponding COM (for that pose/frame) at the same time since the COM has a correlation with the pose/frame from which it has been calculated. The extrapolation of the test/unknown motion data as well as the COM is an added advantage of this shared latent space or manifold.

COM computation: We directly compute COM using Dempsters technique (anthropometric equation) from the best matched torus points.

COM estimation: We use torus to COM mapping matrix (w2) to get the corresponding COM for the test motion data.

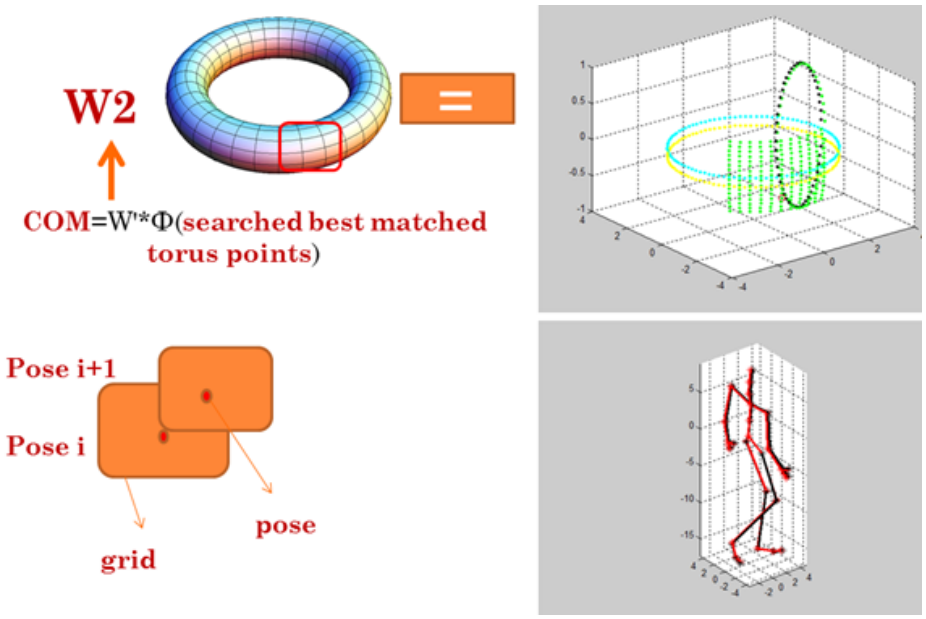

Figure 4.13: Searching best matched torus point.

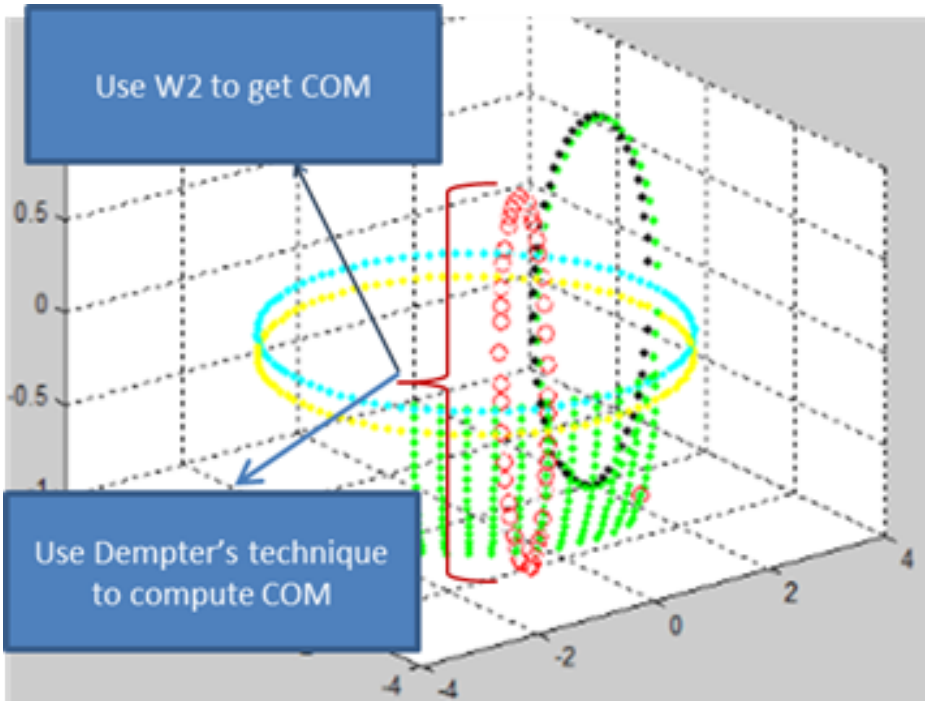

Figure 4.14: COM computation and estimation from best matched Torus points.

Algorithm for estimating COM by JGPM:

Step 1: Normalize training motion data (human motions) based on hip joints of all the training motions. Make the hip position of all the motions zero so that x, y, z coordinates

of the motions vary in a cyclic way. Make all the training and testing gaits having one gait cycleperiod to be equal length (same number of poses/frames) so that the time and speed information is gone, only the pattern of the COM trajectory is taken into account.

Step 2: Acquire a smooth order/path by applying shortest path on the normalized training motion data where each gait is a group of 3D positions from 31 joints.

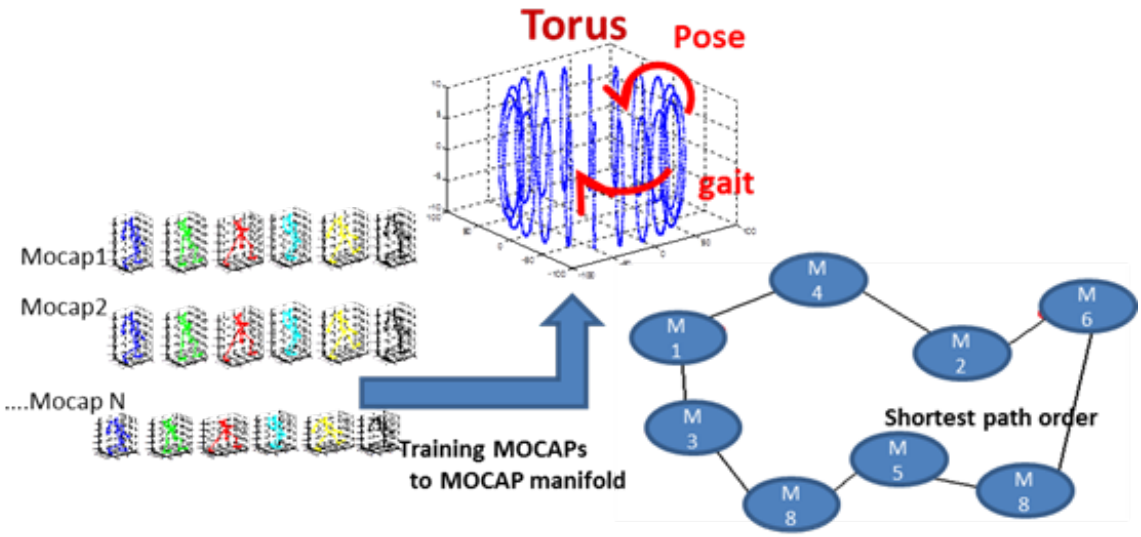

Figure 4.15: Placement of training gaits onto torus based on shortest path order.

Step 3: Apply radial basis function to map training motion data onto torus and get mapping matrix w1 to go from torus to human motion data.

Step 4: Compute COMs from training motion data using Dempters technique.

Step 5: Apply radial basis function to map training COMs onto torus and get mapping matrix w2 to go from torus to COM.

Step 6: Estimate COM from test motion data by searching best torus points using W1. Using best torus points and W2 estimate new COM.

The lower dimensionality of the manifold provides efficient tracking of human motion. The higher dimensional human motion propagation corresponds to the lower dimensional point propagation onto the manifold. As a result we are be able to track higher dimensional motion on to the manifold efficiently since the manifold is a lower dimensional space and

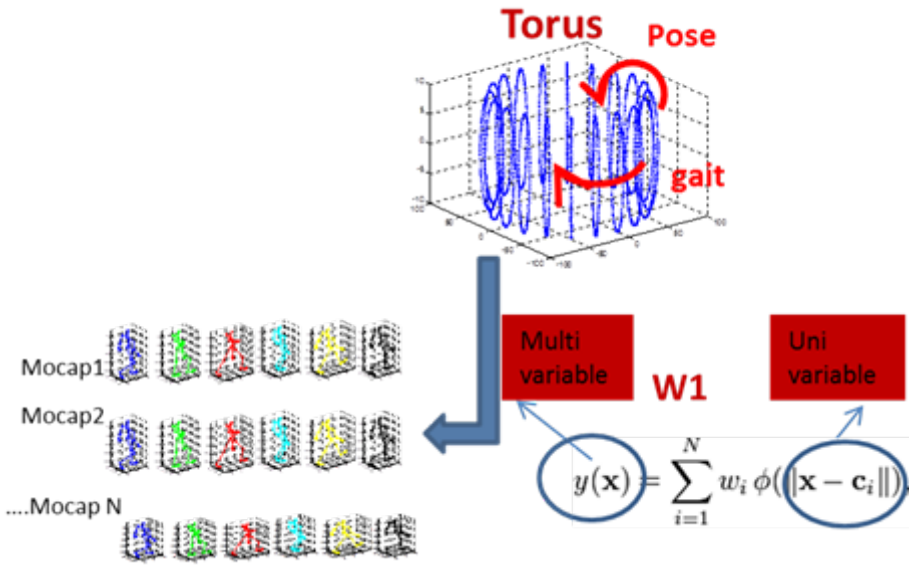

Figure 4.16: RBF mapping of training gaits onto torus.

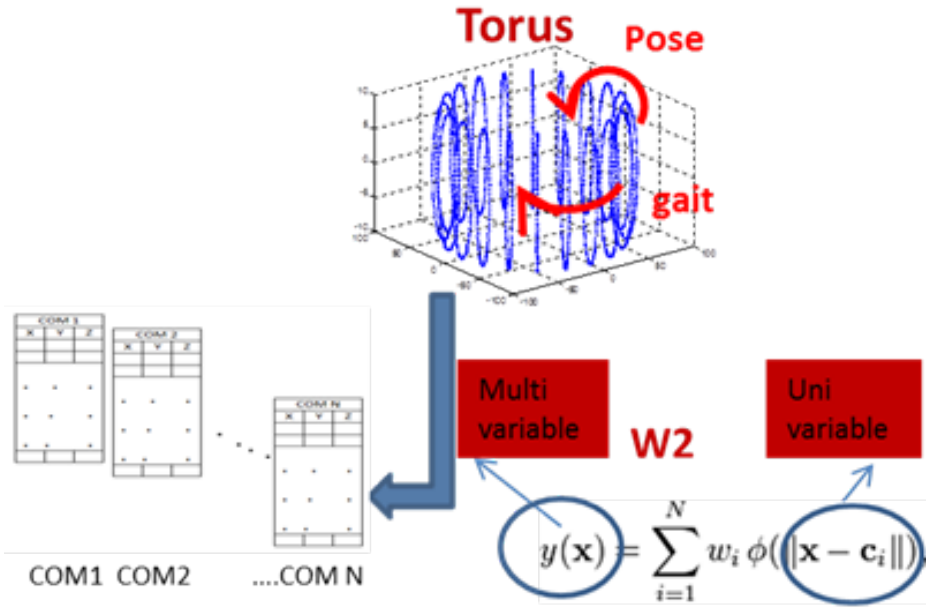

Figure 4.17: RBF mapping training COMs onto torus based on training motion data order (decided by shortest path in Figure 4.6).

help us working with motion data by dimension reduction. And also we can extrapolate or interpolate intermediate motions from the manifold/latent space.

The most useful property of this shared latent space would be the filtering capacity

in a particular sense. We have designed the latent space from noiseless training motions and COMs, consequently when we extrapolate a noisy test motion data from the manifold it tries to extrapolate the noisy motion data from the manifold designed from noiseless training motions, and we get a reasonable noise free extrapolated motion data and the corresponding COM from the manifold.

CHAPTER 5

Gaussian Process Regression for COM Estimation

In statistics and machine learning areas Gaussian Process Regression is an important tool to interpret and analyze for complex datasets. Gaussian Process provides a form of supervised learning of the data in the form of regression needed for continuous outputs [14]. Gaussian Process is a powerful tool in the area of machine learning which can handle lots of real world problems by providing a representative probabilistic model for the problem.

5.1 Gaussian Process (GP)

Gaussian process is any set of function variables (random variables) having joint Gaussian distribution [14]. Gaussian process imposes a distribution over a function f which maps input space χ to output space $\mathscr{R}$ [46].

Let χ is an input space. Any subset of χ has been mapped to another space $\mathscr{R}$ by a function f. Gaussian process here tells that the joint distribution of f($\mathbf{x}_1$), f($\mathbf{x}_2$),...f($\mathbf{x}_n$) has a joint distribution which is a Gaussian distribution. Gaussian Process is fully defined by its mean function and covariance function. Where m(**x**) is the mean function and $k(x_i, x_j)$ is the kernel function.

$$\mathbf{f}|\mathbf{X} \sim \mathscr{N}(m(\mathbf{x}), K(\mathbf{X}, \mathbf{X})) \tag{5.1}$$

A normally distributed multivariate random variable is a linear combination of univariate random variables where each of its uni-variable has normal distribution. Any set of correlated real-valued random variables clustered around a mean value is basically represented by multivariate random variable.

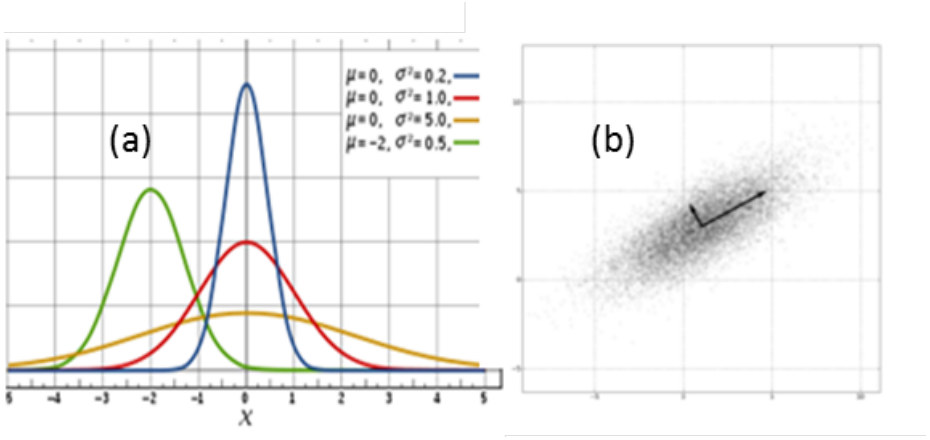

Figure 5.1: Uni-variate random variable (a) and multivariate random variable (b) [14].

The multivariate Gaussian distribution of a variable has the following distribution (equation (5.2)) where **x** is multivariate Gaussian and $\sum$ is the covariance matrix. The covariance matrix $\sum$ of the multivariate normal distribution is symmetric and positive definite [14]. The Gaussian distribution function is given below

$$f_x(\mathbf{x}) = \frac{1}{\sqrt{(2\prod)^n \det(\sum)}} \exp(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}))^T \sum^{-1}(\mathbf{x}-\boldsymbol{\mu}) \tag{5.2}$$

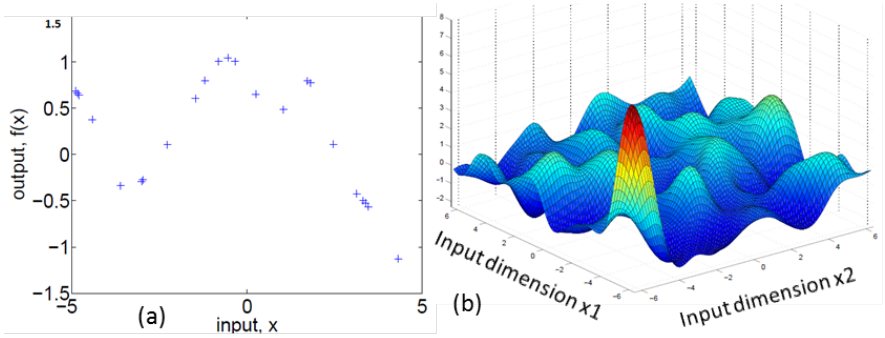

Figure 5.2: Uni-variate random variable function (a) and multivariate random variable function (b) [14].

The uni-variate random variable function has one dimensional input (Fig. 5.2(a)). The input has 20 data points. So the covariance of these data points is a 20×20 matrix, or the joint distribution of these data points has 20×20 covariance matrix. The function drawn

here is just one sample from its distribution drawn randomly [14].

The right one is one sample drawn from the function distribution of a function of 2-dimesional input. From these examples it could be imagined how the sample from multi-variate function distribution look like since it is hard to draw. We want to estimate COM from a 31 joint body model where each joint has 3D position information. As a result the function we want to estimate (computing COM) is a function of multi-variable input where the input has 31×3 dimensions.

One of the most wonderful properties of Gaussian is that the family of Gaussian is preserved under many different conditions. Marginal, conditional and joint distributions of Gaussians are also Gaussians.

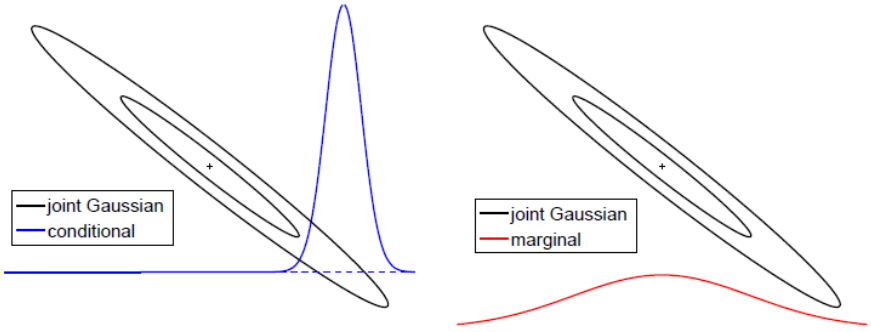

Figure 5.3: Conditional and Marginal distribution of Gaussian [14].

Let $\boldsymbol{X}$ be a multidimensional random variable with normal distribution. If we decompose $\boldsymbol{X}$ into two parts $\mathbf{X}$ and $\mathbf{Y}$ where index of $\mathbf{X}= 1, 2, ..., k$ and index of $\mathbf{Y}= k+1, ..., n$ where $1 \leq k \leq n$, $\bar{\boldsymbol{\mu}}$ and K are the mean and covariance for $\boldsymbol{X}$, $\boldsymbol{\mu}$ and $\boldsymbol{\mu}_*$ are the means for $\mathbf{X}$ and $\mathbf{Y}$ respectively, then we can prove that $\mathbf{X}$ and $\mathbf{Y}$ have normal distributions.

$$\bar{\boldsymbol{\mu}} = \begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\mu}_* \end{bmatrix} \tag{5.3}$$

$$K = \begin{bmatrix} K(\mathbf{X},\mathbf{X}) & K(\mathbf{X},\mathbf{Y}) \\ K(\mathbf{Y},\mathbf{X}) & K(\mathbf{Y},\mathbf{Y}) \end{bmatrix} \tag{5.4}$$

$$\mathbf{X} = \begin{bmatrix} X_1 \\ \vdots \\ X_k \end{bmatrix} \tag{5.5}$$

$$\mathbf{Y} = \begin{bmatrix} X_{k+1} \\ \vdots \\ X_n \end{bmatrix} \tag{5.6}$$

If A is an affine matrix of k by n dimensional which is an identity matrix in its $k \times k$ dimension and other elements are zero and $\boldsymbol{X}$ is multidimensional random variable with mean $\bar{\boldsymbol{\mu}}$ and covariance K, then by definition $A * X \sim \mathcal{N}(A\bar{\boldsymbol{\mu}}, AKA^T)$. Here, $A * X = \mathbf{X}$ and $A\bar{\boldsymbol{\mu}} = \boldsymbol{\mu}$ and $AKA^T = K(\mathbf{X},\mathbf{X})$, As a result $\mathbf{X}$ has normal distribution where $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, K(\mathbf{X},\mathbf{X}))$ and the same way we can prove that $\mathbf{Y} \sim \mathcal{N}(\boldsymbol{\mu}_*, K(\mathbf{Y},\mathbf{Y}))$ and the marginal distribution of a Gaussian is also Gaussian.

Then the zero mean conditional distribution $\mathbf{Y}|\mathbf{X}$ also has Gaussian distribution with mean $\mathbf{M}$ and covariance $\mathbf{C}$.

$$\mathbf{M} = K(\mathbf{Y},\mathbf{X})(K(\mathbf{X},\mathbf{X}))^{-1}\mathbf{X} \tag{5.7}$$

$$\mathbf{C} = K(\mathbf{Y},\mathbf{Y}) - K(\mathbf{Y},\mathbf{X})(K(\mathbf{X},\mathbf{X}))^{-1}K(\mathbf{X},\mathbf{Y}) \tag{5.8}$$

5.2 Gaussian Process Regression (GPR)

Gaussian process works as a prior in building the Gaussian Process Regression model [46]. We want to impose the Gaussian Process prior that the function distribution we will get is

also Gaussian [14]. Let Y be the observed function with Gaussian noise ϵ for a multidimensional random variable ***X*** where ***X*** = $[X_1, \cdots, X_k, X_{k+1}, \cdots, X_n] \; \epsilon \; S$, $\mathbf{Y} = [Y_{x1}, \cdots, Y_{xk}, Y_{xk+1}, \cdots, Y_{xn}] \; \epsilon \; R$, Gaussian noise $\boldsymbol{\epsilon} = [\epsilon_1, \cdots, \epsilon_n]$. The noise $\boldsymbol{\epsilon}$ is independent of the observation $\mathbf{Y}$. The Gaussian Process regression model we have

$$\mathbf{Y} = f(\mathbf{X}) + \boldsymbol{\epsilon} \tag{5.9}$$

In order to inference the posterior distribution gives us the opportunity to predict unobserved function distribution from given observed function. Let $\mathbf{y}$ be the known function values and Y_b the unknown or test function values observed for input values $\mathbf{X}$ and $\mathbf{X}_*$ respectively where $\mathbf{X} = [X_1, \cdots, X_k]$, $\mathbf{X}_* = [X_{k+1}, \cdots, X_n]$, $X = [\mathbf{X}, \mathbf{X}_*]^T \; \epsilon \; S$, $\mathbf{y} = [Y_{x1}, \cdots, Y_{xk}]$, $\mathbf{f}_* = [Y_{xk+1}, \cdots, Y_n]$ and $Y = [\mathbf{y}, \mathbf{f}_*]^T \; \epsilon \; R$.

The joint distribution of $\mathbf{y}$ and $\mathbf{f}_*$ with a zero mean function is

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim N \left(0, \begin{bmatrix} K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 I & K(\mathbf{X}, \mathbf{X}_*) \\ K(\mathbf{X}_*, \mathbf{X}) & K(\mathbf{X}_*, \mathbf{X}_*) \end{bmatrix} \right) \tag{5.10}$$

The posterior/conditional/predictive distribution would be

$$\mathbf{f}_* | \mathbf{y}, \mathbf{X}, \mathbf{X}_* \sim \mathcal{N}(\mathbf{m}, \mathbf{D}) \tag{5.11}$$

To get the mean **m** and covariance **D** we would follow equation 5.7 and equation 5.8.

$$\mathbf{m} = K(\mathbf{X}, \mathbf{X}_*)(K(\mathbf{X}_*, \mathbf{X}_*) + \sigma_n^2 I)^{-1} \mathbf{f}_* \tag{5.12}$$

$$\mathbf{D} = K(\mathbf{X}_*, \mathbf{X}_*) - K(\mathbf{X}_*, \mathbf{X})[K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 I]^{-1} K(\mathbf{X}, \mathbf{X}_*) \tag{5.13}$$

Choice of kernel function: In Gaussian Process covariance function is the key based on which it approximates function. Covariance function encodes the prediction about the function we want our input to be mapped to. Gaussian Process regression provides inference in the function space directly. This regression process is non-parametric Bayesian

approach. The high level information present in the training motions that similar neighborhoods are strongly correlated, the output COMs are predicted from training examples where internal information of the test motion data are similar to the training motions. Gaussian Process Regression is essentially based on the assumption that similar inputs tend to give similar outputs [14]. In our research we have used the following kernel function to assume the function:

$$k(x, x^{'}) = \nu^2 \exp\left(-\frac{(x - x^{'})^2}{2l^2}\right) + \sigma_n^2 \delta_{xx} \tag{5.14}$$

The characteristic of this covariance function can be well described by figure 5.4,

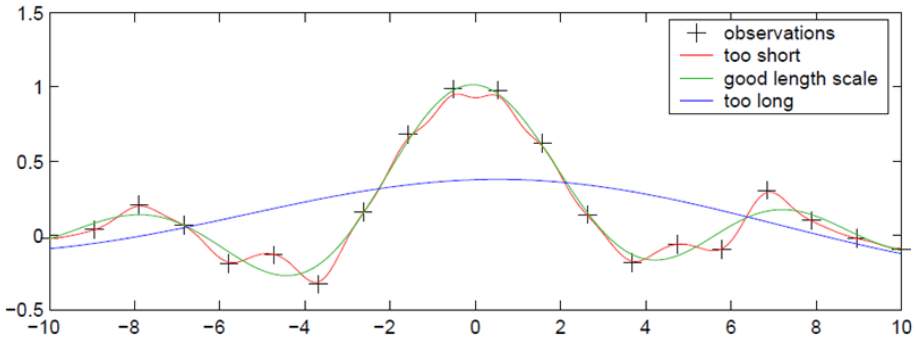

Figure 5.4: Fitting length scale parameter [14].

where ν and l are the two parameters of this covariance function. ν is the signal variance and l is the length scale parameter. The effect of l defines the meaning of 'closeness' between two data points. Depending on this parameter remotely placed or closely placed two data points can have high correlation with each other. So setting this parameter depends on the nature of the data structure to assume a function.

We can get marginal likelihood from Bayesian Inference in parametric model where we multiply prior with the likelihood and compute the integral. Then we take the log of marginal likelihood and get equation 5.15. The higher the marginal likelihood the better is the approximation of function or the higher the assumption about the function, and optimize it to get parameters of the covariance function.

$$\log p(y|x, M_i) = -\frac{1}{2} y^T K^{-1} y - \frac{1}{2} \log |K| - \frac{n}{2} \log(2\pi) \tag{5.15}$$

The marginal likelihood has three terms. The first term is the data fit term because this is the only term that involves the test data and in this term K involves test and training both inputs. The second term is complexity penalty which regulates over-fitting the training data and is taken care of by itself. There is no need to take extra care to optimize this term. The marginal likelihood has a trade-off between the data-fit term and the complexity term.

$$\frac{\partial \log p(y|x, \theta, M_i)}{\partial \theta_j} = \frac{1}{2} y^T K^{-1} \frac{\partial K}{\partial \theta_j} K^{-1} y - \frac{1}{2} trace \left(K^{-1} \frac{\partial K}{\partial \theta_j} \right) \tag{5.16}$$

If we want to learn the Gaussian process model for a data set we need to find the appropriate covariance function and optimize its unknown hper-parameters. In our research we have optimized log-marginal likelihood to optimize the parameters. We have optimized the covariance parameters by line-search method with the equation 5.16.

5.3 Learning of GPR

Two modes to train the GPR have been used in this research. Training mode 1 takes training motions pose by pose to train GPR to estimate COM. Training mode 2 takes all the poses of all the training motions at a time to train the GPR.

GPR Training Mode 1: We have used 20 training motions and 10 test motions. Each motion data (input) has 31 dimensions and each COM (output) has 3 dimensions. As we are estimating COM pose by pose, for each pose the covariance $K(\mathbf{X}_*, \mathbf{X})$ between training and testing motions is a 10×20 matrix, since each dimension has 20 data points for training and 10 data points for testing. Again for each pose the covariance $K(\mathbf{X}, \mathbf{X})$ among training motions is a 20×20 matrix. The mean prediction for function approximation m is a 10×1 matrix, the variance D is also a 10×1 matrix since we have to approximate 1 dimension for 10 training motions each time. The total dimension for output function is 3. As a summary, dimension of $\mathbf{X}$: 20×31, dimension of $\mathbf{X}_*$: 10×31, dimension of $\mathbf{y}$: 20×1 since

estimating one dimension each time, dimension of $\mathbf{f}_*$: 10×1, dimension of $K(\mathbf{X}, \mathbf{X})$: 20×20, dimension of $K(\mathbf{X}_*, \mathbf{X})$: 10×20 and dimension of $K(\mathbf{X}_*, \mathbf{X}_*)$: 10×10.

GPR Training Mode 2: In training mode 2 the correlation among all the poses are considered together to estimate COM. We have used a set of 20 training motions and 10 test motions. Each motion data has 50 poses. As a result, in each dimension we have 1000 data points for training and 500 data points for testing. The covariance $K(\mathbf{X}_*, \mathbf{X})$ between training and testing motions is a 500×1000 matrix, since each dimension has 1000 data points for training and 500 data points for testing. Again the covariance $K(\mathbf{X}, \mathbf{X})$ among training motions is a 1000×1000 matrix. The mean prediction for function approximation m is a 500×1 matrix, the variance D is also a 500×1 matrix since we have to approximate 1 dimension for 10 training motions each time. The total dimension for output function is 3. As a summary, dimension of $\mathbf{X}$: 1000×31, dimension of $\mathbf{X}_*$: 500×31, dimension of $\mathbf{y}$: 1000×1 since estimating one dimension each time, dimension of $\mathbf{f}_*$: 500×1, dimension of $K(\mathbf{X}, \mathbf{X})$: 1000×1000, dimension of $K(\mathbf{X}_*, \mathbf{X})$: 500×1000 and dimension of $K(\mathbf{X}_*, \mathbf{X}_*)$: 500×500.

5.4 GPR-based COM Estimation

In Gaussian process an assumption over the output mapping function is: the function has Gaussian distribution. It is covariance structure of the input based on which it approximates the function. Covariance function encodes the prediction about the function we want our input to be mapped to. Gaussian process regression provides inference in the function space directly. This regression process is non-parametric and has been summarized by the following steps.

Step 1: Our Input is high dimensional human motion. Output is COM and we want to predict the function of computing COM

Step 2: Split $\mathbf{Y}$ (COM) into $\mathbf{y}$, $\mathbf{f}_*$ and take zero mean joint distribution of $\mathbf{y}$ and $\mathbf{f}_*$

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim N\left(0, \begin{bmatrix} K(\mathbf{X},\mathbf{X}) + \sigma_n^2 I & K(\mathbf{X},\mathbf{X}_*) \\ K(\mathbf{X}_*,\mathbf{X}) & K(\mathbf{X}_*,\mathbf{X}_*) \end{bmatrix}\right)$$

Step 3: Take conditional distribution from Joint Gaussian distribution.

$$\begin{aligned} &\mathbf{f}_*|\mathbf{y},\mathbf{X},\mathbf{X}_* \sim \mathcal{N}(\mathbf{m},\mathbf{D}) \\ &\mathbf{m} = K(\mathbf{X},\mathbf{X}_*)(K(\mathbf{X}_*,\mathbf{X}_*) + \sigma_n^2 I)^{-1}\mathbf{f}_* \\ &\mathbf{D} = K(\mathbf{X}_*,\mathbf{X}_*) - K(\mathbf{X}_*,\mathbf{X})[K(\mathbf{X},\mathbf{X}) + \sigma_n^2 I]^{-1}K(\mathbf{X},\mathbf{X}_*) \end{aligned}$$

Step 4: Choose the kernel function, squared exponential has been chosen here to estimate COM (equation 5.14)

$$k(x,x') = \nu^2 \exp\left(-\frac{(x-x')^2}{2l^2}\right) + \sigma_n^2 \delta_{xx}$$

Step 5: Optimize the parameters ν and l of the Covariance function by optimizing Log marginal likelihood (equation 5.15)

$$\log p(y|x,M_i) = -\frac{1}{2}y^T K^{-1} y - \frac{1}{2}\log|K| - \frac{n}{2}\log(2\pi)$$

Step 6: Put optimized parameters into Covariance function to get f^* (COM In this case) (equation 5.16).

$$\frac{\partial \log p(y|x,\theta,M_i)}{\partial \theta_j} = \frac{1}{2}y^T K^{-1}\frac{\partial K}{\partial \theta_j}K^{-1}y - \frac{1}{2}trace\left(K^{-1}\frac{\partial K}{\partial \theta_j}\right),$$

where m is the mean of the estimation or estimation for COM and d is the variance of the estimation indicating the confidence about the approximation

The high level information present in the training motions that similar neighborhoods are strongly correlated, the output COMs are predicted from training examples where internal information of the test motion data are similar to the training motions. Gaussian

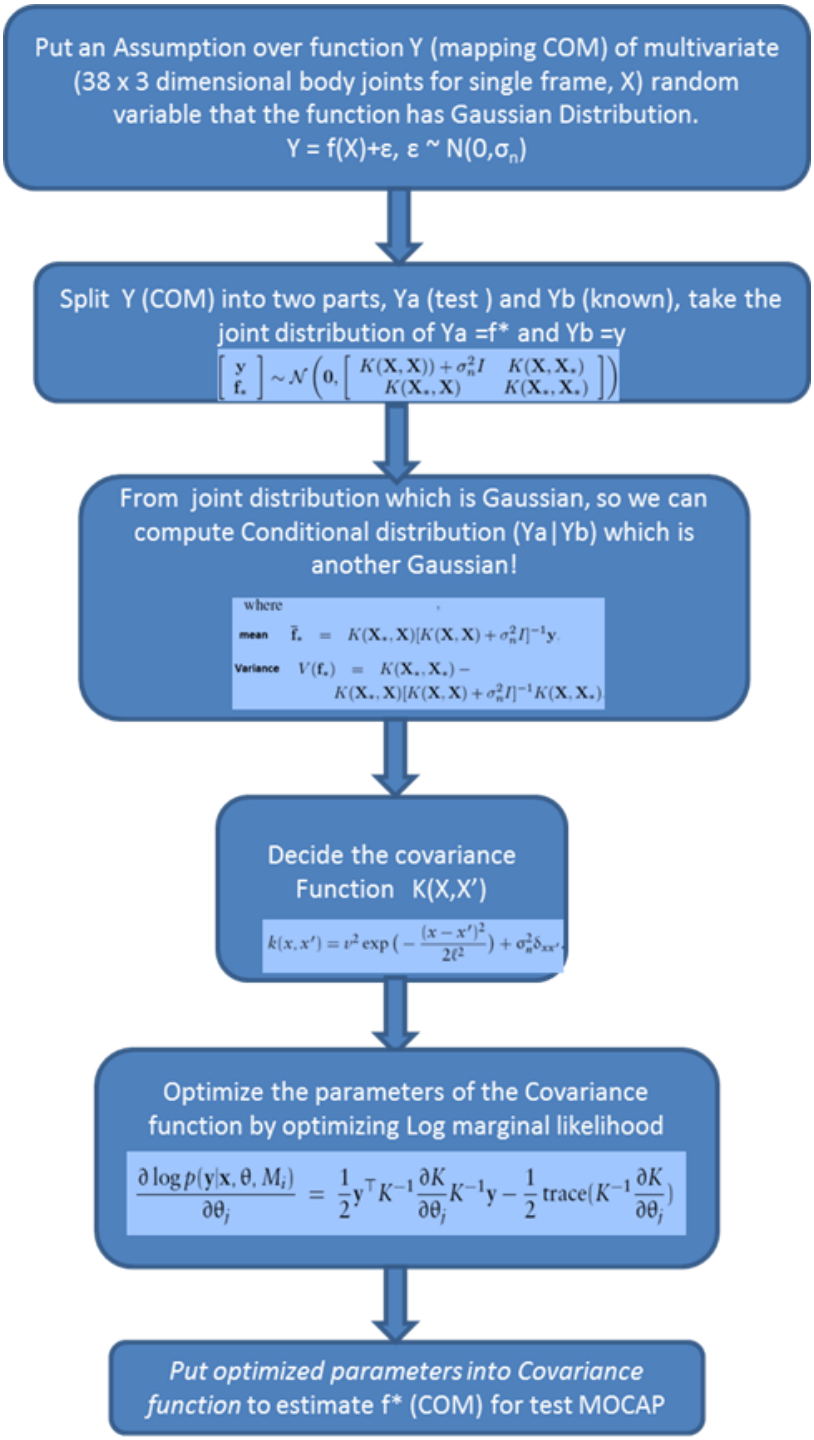

Figure 5.5: Gaussian Process Regression for COM estimation flow chart.

Process Regression is essentially based on the assumption that similar inputs tend to give similar outputs [14].

CHAPTER 6

Experimental Result

In this research we have estimated COM by two different approaches of machine learning, regression based approach (GPR), designed manifold based approach (torus). Gaussian Process Regression is a technique which takes into account the correlation among the inputs and the outputs. It basically tries to find the correlation of the mapping function which maps inputs to the outputs. This is a straight forward strategy which does not research over the structure of the data, motion or kinematics.

We have estimated COM from noise-free motion data and noisy motion data. We have got the best estimation of COM from noise-free test motion data by Gaussian Process Regression comparing to the torus manifold. The manifold based estimation tries to impose a topology over the data. The manifold surface/latent space is an ideal reflection of the data structure and kinematics. To estimate COM from noisy motion data we have adapted two approaches: first one is direct estimation of COM by the manifold. In second approach there is an intermediate stage where we have estimated test motion data from the manifold/torus and then using the de-noised motion data we have computed COM. In case of de-noised motion data, torus performs better job than GPR and torus +GPR is the best to estimate COM from noisy test motion data.

Figure 6.1 provides us with an overview of the experimental result organization in this chapter.

In table 6.1 we have shown the comparison of motion data estimation and COM estimation. In table 6.2 we have shown the comparison of mapped COM and computed COM. In table 6.3 we have compared the performance of torus and GPR to estimate COM from

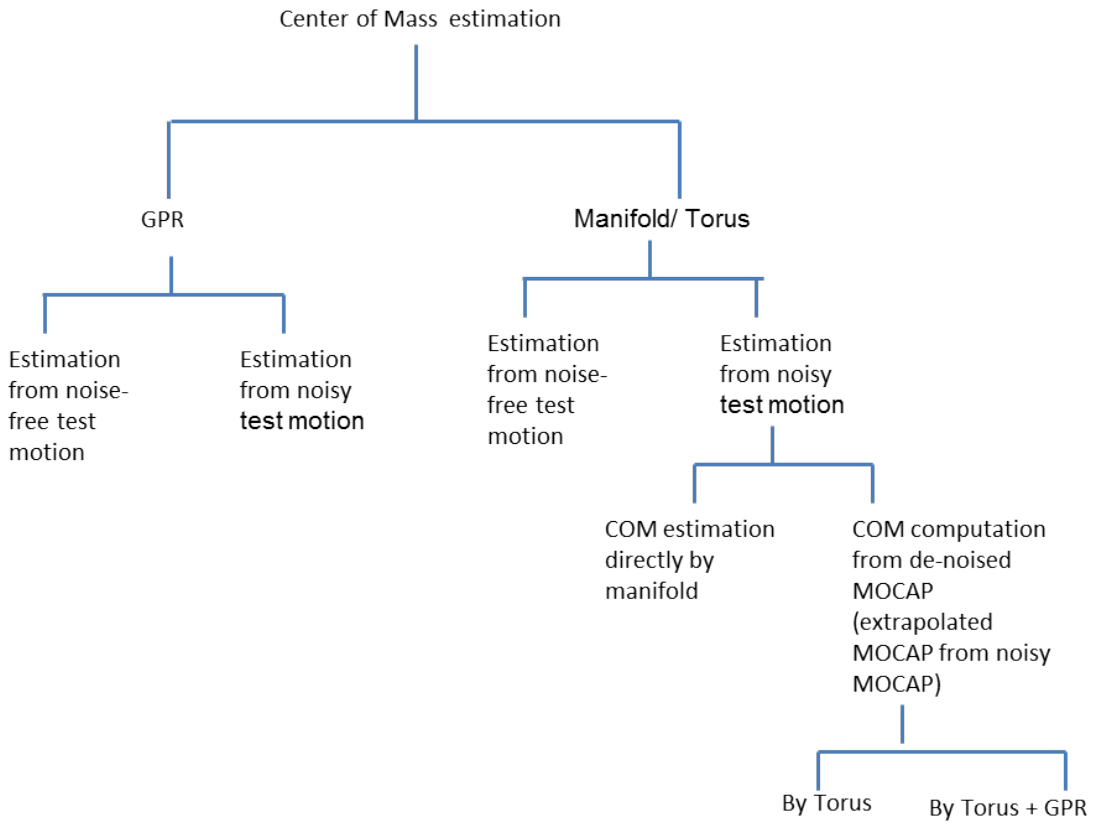

Figure 6.1: Experiment result hierarchy.

noiseless test data whereas tables 6.4 and 6.5 show the comparison among direct computation of COM from noisy motion data, torus and torus with GPR in computing COM from denoised motion data by torus.

6.1 Torus based COM computation and estimation

Unlike GPR, torus manifold gives us two extrapolations, first extrapolation is for 16 joints human body model and the second extrapolation is for COM.

motion data error: Per frame/pose, per joint error (an average error of 16 joints and 50 poses/frames).

COM error: Per frame error (an average of 50 poses/frames).

COM error is almost half comparing to motion data error since estimation of COM is just one joint estimation per frame whereas motion data error is 16 joints estimation per

Interpolation Error	X-error	Y-error	Z-error	Total sqrt(x + y + z) (cm)
motion data	0.934552	0.396132	0.320216	1.2849
COM	0.555039	0.216323	0.055882	0.9095

Table 6.1: Motion data versus COM error.

frame. The estimation of each joint induces error to the total estimation error of motion data, as a result it is reasonable to have motion data error greater than the COM error.

With a test motion data we search the torus to get the best matched points (the points for which the distance between the reconstructed motion data and the test motion data is the minimum). Using those best matched torus points we can get COM in two ways as shown in figure 4.14.

Error	X-error	Y-error	Z-error	Total (x + y + z) error	Total sqrt(x + y + z) error (cm)
COM ESTIMATION	0.555039	0.216323	0.055882	0.827244	0.9095
COM COMPUTATION	0.620487	0.248963	0.052986	0.922435	0.9604

Table 6.2: COM estimation versus COM computation.

The COM estimation appears slight better than the direct COM computation from best matched torus points. Here mapping has advantage over direct computation since motion data extrapolation from torus induces a bit error. This error is reflected in COM computation. On the other hand, COM mapped by 'torus to COM mapping matrix' does not have this error, as a result it performs better.

6.2 Torus-based COM Estimation vs. GPR-based COM Estimation (Noiseless Motion Data)

In case of COM estimation from noise free test motion data GPR2 does the best job and GPR1 does a better job than torus-manifold approach. In Gaussian process covariance function is the key based on which it approximates function. Covariance function encodes the prediction about the function we want our input to be mapped to. GPR2 performs better than GPR1 since GPR2 encodes training information from all the poses of all the training motions. As a result, GPR2 has more training information (all the poses of all the training motions) to predict COM for a certain pose. GPR provides inference in the function space directly. Similar neighborhoods are strongly correlated, this is the high level information that is present in the training motions. The output COMs are predicted from those training examples with which the test motion data has the greatest similarity. Gaussian process regression is essentially based on the assumption that similar inputs tend to give similar outputs [14].

COM EXTRAPOLATION Error	**X-error**	**Y-error**	**Z-error**	**Total (x + y + z) error**	**Total sqrt(x + y + z) error (cm)**
Torus	0.555039	0.216323	0.055882	0.827244	0.9095
Gaussian Process Regression1	0.073213	0.01429975	0.009815118	0.096007	0.3098
Gaussian Process Regression2	0.001508	0.000265	0.002037	0.00381	0.061726

Table 6.3: COM extrapolation by GPR, Torus computation, Torus estimation.

6.3 Torus-based COM Estimation vs. GPR-based COM Estimation (Noisy Motion Data)

Noise standard deviation	Direct COM computation error from noisy test motion (cm)	Torus COM estimation error from noisy test motion (cm)	GPR1 COM estimation error from noisy test motion (cm)	GPR2 COM estimation error from noisy test motion (cm)
0	0	0.9095	0.3098	0.061726
1	0.447634	0.904251	0.483482	0.416941
3	1.275239	1.020497	1.048368	1.192343
5	2.054045	1.096497	1.531762	1.776459
6	2.47407	1.1647037	1.676961	1.983968
7	2.886464	1.175772	1.862467	2.111276

Table 6.4: COM estimation by Torus and GPR from noisy motion data.

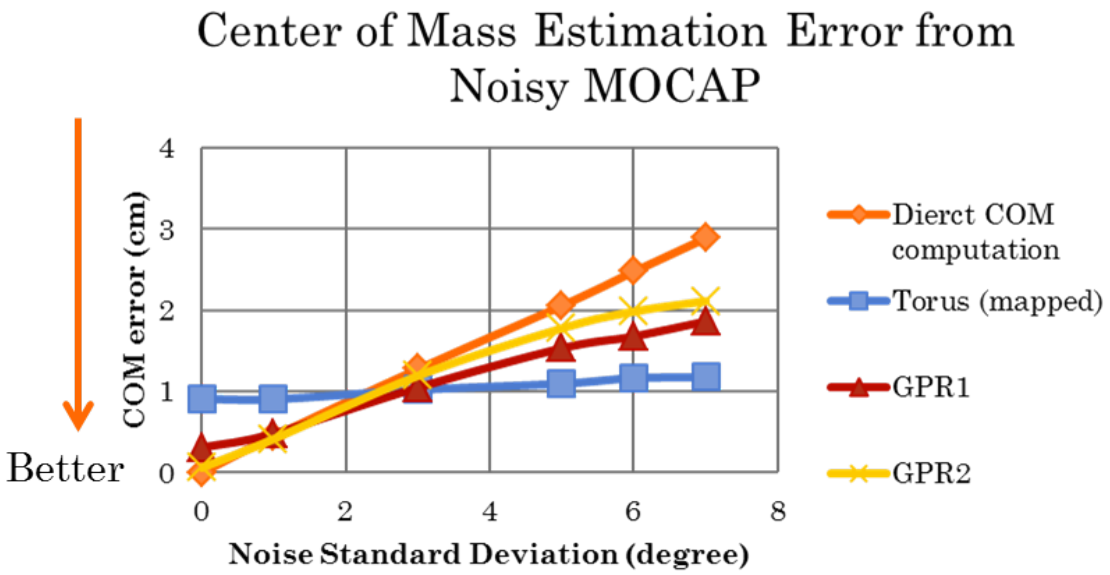

Figure 6.2: COM estimation by Torus, GPR from noisy motion data.

Figure 6.2 draws the data shown in table 6.4. The orange curve is the COM computation

directly using anthropometric equations from noisy test motion data. This is very much evident from the graph that when noise level is higher Torus-A (designed manifold using RBF) does the best job in estimating COM from noisy test motion data. GPR can only perform well when noise level of test motion data is low. Unlike noiseless COM estimation GPR1 performs better than GPR2 for noisy test motion data. GPR2 performs worse than GPR1 since it takes all the poses of all the training motions as its training information. When a pose is noisy it might look like another pose, as a result GPR2 tries to predict that pose using the pose information close to it (which might be totally a different pose present in the training poses). The most useful property of torus/shared latent space would be the filtering capacity in a particular sense. We have designed the latent space from noiseless training motions and COMs, consequently when we extrapolate a noisy test motion data from the manifold it tries to extrapolate the noisy motion data from the manifold designed from noiseless training motions, and we get a reasonable noise-free extrapolated motion data and the corresponding COM from the manifold.

6.4 Torus-GPR-based COM Estimation

Noise standard deviation	**Direct COM computation error from noisy test motion (cm)**	**Torus COM estimation error from noisy test motion (cm)**	**Torus and GPR1 COM estimation error from noisy test motion (cm)**	**Torus and GPR2 COM estimation error from noisy test motion (cm)**
1	0.447634	0.923259	0.863435	0.88139
3	1.275239	0.960172	0.916728	0.944372
5	2.054045	1.008086	0.93534	0.977287
6	2.47407	1.02568	0.950237	0.997221
7	2.886464	1.138829	1.080728	1.119613

Table 6.5: COM estimation by Torus, Torus-GPR from de-noised motion data.

Figure 6.3 portrays the scenario of table 6.5 where table 6.5 shows the result of COM

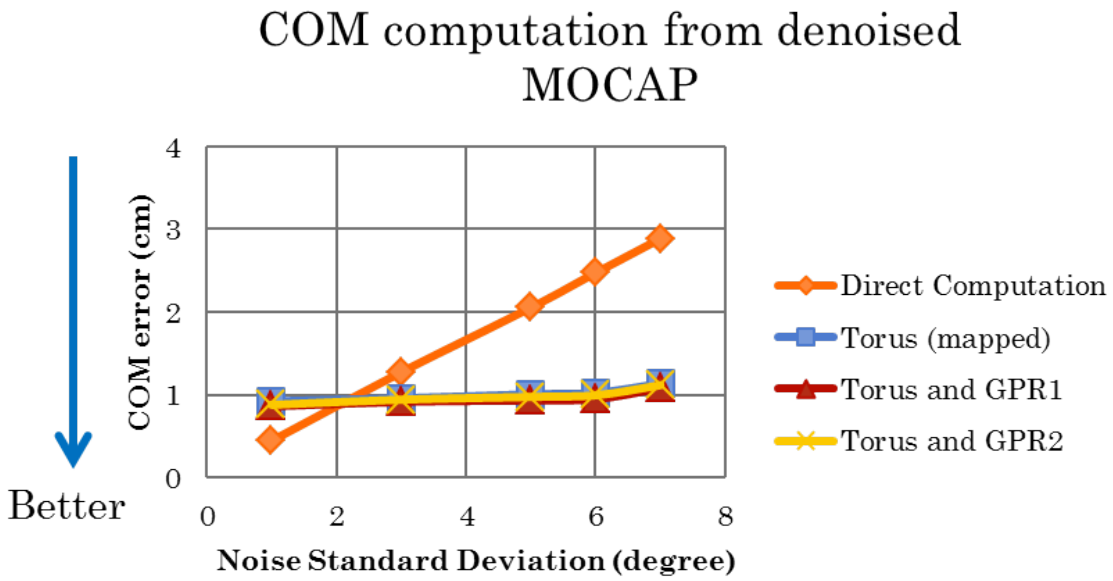

Figure 6.3: COM computation by Torus, Torus-GPR from denoised motion data.

computation directly by Dempsters technique (anthropometric equations) after denoising. Denoising means the extrapolation of noisy test motion data from the manifold. This denoising cannot be applied for GPR since GPR does not give an extrapolation for the test motion data, GPR directly approximates the function to compute COM.

From figure 6.3 it is evident that applying GPR approach after denoising by the torus outperforms direct COM computation from noisy motion data and direct computation of COM from denoised motion data by torus. It is also evident torus and torus-GPR are very close in performance to estimate COM. We can make a conclusion that the performance of torus alone is well enough since torus-GPR is a two-step method whereas applying torus once requires less time and expense. Figure 6.3 shows the promise of manifold approach with a tours structure (an ideal surface to represent gait and pose together) which denoises noisy human motion data and estimates COM than direct regression approach (which loses precision in the presence of noisy motion data).

6.5 Comparative analysis

Comparison title	GPR	Torus manifold
Characteristics	Regression based non-parametric approach, better for the approximation of noise-less test motion data	Manifold based parametric approach, performance is not better than GPR for the approximation of noise-less test motion data
Methodology	Approximation of function is based on the co-variance structure of HD dataset	Approximation of function is based on the structure and kinematics of HD data, the manifold surface works as an information for approximation of function
Robustness to noise	GPR is not better for function approximation from noisy test motion data, it considers noise in training data and approximates the function based on noisy training data	Best for extrapolation from noisy test motion data because the training motion data is noise free, as a result it can work as a filter for extrapolating new noisy test motion data and the corresponding function (COM)
COM estimation accuracy	Does not serve our research focus to filter out noise from noisy, incomplete or inaccurate motion data and estimate noise-free COM	Serves our research focus to filter out noise from noisy or incomplete motion data, therefore can work as an alternative to replace expensive optical, mechanical and electro-magnetic data acquisition and computation with COM estimation

Table 6.6: Comparison between GPR and manifold based approach.

The most useful property of our shared latent space (torus) is the filtering capacity

in a particular sense. We have designed the latent space from noiseless training motions and COMs, consequently when we extrapolate a noisy test motion data from the manifold it tries to extrapolate the noisy motion data from the manifold designed from noiseless training motions, and we get a reasonable noise free extrapolated motion data and the corresponding COM from the manifold.

In case of noiseless test motion data, GPR works better than torus which is okay but it does not sever our research focus since GPR does not show any advantage over direct computation from noiseless test data. Besides noise handling, we get another exciting characteristic of this shared latent space is that it provides continuous space to extrapolate new/unknown human motions and the corresponding COMs whereas from GPR we get the estimation of COM only. Estimation of COM can be an alternative for expensive motion capture system such as optical, electro-magnetic and mechanical motion data and filter out noise from noisy or incomplete motion data.

In case of noisy or incomplete motion data, direct computation of COM from noisy motion data could not be helpful where high accuracy of COM computation is needed in the areas such as diagnose postural and gait stability, detecting fall risk etc. As a result we would have to provide a filtering technique or estimation method for COM instead of direct computation when we are to acquire data from various noisy sources. Torus manifold serves our research focus and therefore can work as an alternative to replace expensive data acquisition systems by providing reasonable COM estimation from noisy sensor data.

CHAPTER 7

Conclusions and Future work

In this research we have computed COM using Dempsters technique and have estimated COM using two different machine learning techniques. A manifold (Torus) has been designed or mapped to estimate COM. Also, we have implemented Gaussian Process Regression approach which predicts COM by encoding the high level information that similar neighborhoods training motions are strongly correlated. Gaussian process regression has been proved to be better working than Torus-based approach when there is no noise or low noise in the testing motion data. The manifold based approach gives us a continuous space to extrapolate human motion data as well as COM. The most useful property of this shared latent space would be the filtering capacity in a particular sense. We have designed the latent space from noiseless training motions and COMs, consequently when we extrapolate a noisy test motion data from the manifold it tries to extrapolate the noisy motion data from the manifold designed from noiseless training motions, and we get a reasonable noise free extrapolated motion data and the corresponding COM from the manifold.

The shared latent space/manifold (Torus) approach where the manifold is shared by motion data and COM together has been proved to be better working than the GPR to estimate COM from noisy test motion data. Manifold based approach is better since the torus structure guides the COM estimation not to be totally meaningless when the noise level is higher.

In our future work we are interested to find a shared latent space for human motion data and COM. The space will share some common structure of human motion data and COM. Structural commonalities is an excellent space since we can interpolate new data in

one space where we are given observation of another space [47]. Some novel domains of this shared latent space include 3D object appearance estimation while we are given poses of another object by learning parameters common to both objects, "learning by watching" [48–50] where a robot learns to perform a task by observing another agent for example by observing a human instructor.

Also, we intend to work on incomplete (not all the joints of a human body are given) human motion data to estimate COM and involve GPR to deal noisy human motion.

BIBLIOGRAPHY

[1] D. Winter and P. Eng, "Kinetics - our window into the goals and strategies of the central-nervous-system," *Behavioural Brain Research*, vol. 67, pp. 111–120, Mar 1995.

[2] G. Yamaguchi and F. Zajac, "Restoring unassisted natural gait to paraplegics via functional neuromuscular stimulation - a computer-simulation study," *IEEE Transactions on Biomedical Engineering*, vol. 37, pp. 886–902, Sep 1990.

[3] B. Dariush, "Human motion analysis for biomechanics and biomedicine," *Machine and Applications*, vol. 14, pp. 202–205, Sep 2003.

[4] A. Menache, *Understanding motion capture for computer animation and video games*. Morgan Kaufmann, 2000.

[5] F. Amorouche, M. Xie, and A. Patwardhan, "Optimization fo the contact damping and stiffness coefficients to minimize human-body vibration," *Journal of Biomechanical Engineering-Transactions of the ASME*, vol. 116, pp. 413–420, Nov 1994.

[6] H. Hemami and Y. Zheng, "Dynamics and control of motion on the ground and in the air with application to biped robots," *Journal of Robotic Systems*, vol. 1, no. 1, pp. 101–116, 1984.

[7] L. Chou, K. Kaufman, R. Brey, and L. Draganich, "Motion of the whole body's center of mass when stepping over obstacles of different heights," *Gait & Posture*, vol. 13, pp. 17–26, Feb 2001.

[8] S. S. Hasan, D. W. Robin, D. C. Szurkus, D. H. Ashmead, S. W. Peterson, and R. G. Shiavi, "Simultaneous measurement of body center of pressure and center of gravity during upright stance. part i: Methods," *Gait & Posture*, vol. 4, no. 1, pp. 1 – 10, 1996.

[9] E. Oba, "Center of mass: Tools and techniques for animating natural human movement@http://www.gamasutra.com/view/feature/4594/center_of_mass_tools_and_.php?print=1.," Dec. 2011.

[10] "Restoring the ability to walk is a key step toward independence@http://www.easy-walking.com/up-n-go/what-is-it/how-it-works/," Dec. 2011.

[11] "University rehabilitation institute@http://www.ir-rs.si/en/Research_equipment," Dec. 2011.

[12] D. Robertson, *Research methods in biomechanics*. Human Kinetics, 2004.

[13] C. Lee and C. Farley, "Determinants of the center of mass trajectory in human walking and running," *Journal of Experimental Biology*, vol. 201, pp. 2935–2944, Nov 1998.

[14] C. Rasmussen and C. Williams, *Gaussian process for machine learning*. MIT Press, 2006.

[15] J. Eng and D. Winter, "Estimations of the horizontal displacement of the total body centre of mass: considerations during standing activities," *Gait & Posture*, vol. 1, no. 3, pp. 141 – 144, 1993.

[16] H. David and Sutherland, "The evolution of clinical gait analysis part l: kinesiological emg," *Gait & Posture*, vol. 14, no. 1, pp. 61 – 70, 2001.

[17] Y. Jian, D. Winter, M. Ishac, and L. Gilchrist, "Trajectory of the body cog and cop during initiation and termination of gait," *Gait & Posture*, vol. 1, no. 1, pp. 9 – 22, 1993.

[18] V. Lugade, V. Lin, and L.-S. Chou, "Center of mass and base of support interaction during gait," *Gait & Posture*, vol. 33, pp. 406–411, Mar 2011.

[19] M. Orendurff, A. Segal, G. Klute, J. Berge, E. Rohr, and N. Kadel, "The effect of walking speed on center of mass displacement," *Journal of Rehabilitation Research and Development*, vol. 41, pp. 829–834, Nov-Dec 2004.

[20] A. Hernandez, A. Silder, B. C. Heiderscheit, and D. G. Thelen, "Effect of age on center of mass motion during human walking," *Gait & Posture*, vol. 30, pp. 217–222, Aug 2009.

[21] S. Hasan, D. Robin, D. Szurkus, D. Ashmead, S. Peterson, and R. Shiavi, "Simultaneous measurement of body center of pressure and center of gravity during upright stance. part ii: Amplitude and frequency data," *Gait & Posture*, vol. 4, no. 1, pp. 11 – 20, 1996.

[22] M. Buhmann, *Radial Basis Functions: Theory and Implementations*. Cambridge University Press, 2009.

[23] E. Cheney, *Introduction to approximation theory*. AMS Chelsea Publisher, 1982.

[24] P. Davis, *Interpolation and approximation*. Dover Publisher, 1975.

[25] J. Carr, R. Beatson, J. Cherrie, T. Mitchell, W. Fright, B. McCallum, and T. Evans, "Reconstruction and representation of 3d objects with radial basis functions," (Los Angeles, CA, United states), pp. 67 – 76, 2001.

[26] X. Zhang and G. Fan, "Joint gait-pose manifold for video-based human motion estimation," *Proceedings of the Computer Vision and Pattern Recognition*, 2011.

[27] A. Elgammal and C.-S. Lee, "Tracking People on a Torus," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, pp. 520–538, Mar 2009.

[28] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models for human motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 283–298, Feb 2008.

[29] C.-S. Lee and A. Elgammal, "Modeling view and posture manifolds for tracking," (Rio de Janeiro, Brazil), 2007.

[30] H. Murase and S. K. Nayar, "Visual learning and recognition of 3-d objects from appearance," *International Journal of Computer Vision*, vol. 14, pp. 5–24, 1995.

[31] A. Elgammal and C.-S. Lee, "Inferring 3d body pose from silhouettes using activity manifold learning," vol. 2, (Washington, DC, United states), pp. II681 – II688, 2004.

[32] C.-S. Lee and A. Elgammal, "Simultaneous inference of view and body pose using torus manifolds," vol. 3, (Hong Kong, China), pp. 489 – 494, 2006.

[33] N. Lawrence, "Gaussian process latent variable models for visualisation of high dimensional data," in *In NIPS*, p. 2004, 2003.

[34] Z. Lu, M. A, and C. Sminchisescu, "People tracking with the laplacian eigenmaps latent variable model," in *NIPS*, 2007.

[35] N. D. Lawrence, "Local distance preservation in the gp-lvm through back constraints," in *In ICML*, pp. 513–520, ACM Press, 2006.

[36] K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popovic, "Style-based inverse kinematics," vol. 23, pp. 522 – 531, 2004.

[37] N. D. Lawrence and A. J. Moore, "Hierarchical gaussian process latent variable models," vol. 227, (Corvalis, OR, United states), pp. 481 – 488, 2007.

[38] M. Ebden, "Gaussian processes for regression: A quick introduction," tech. rep., 2008.

[39] D. Nguyen-Tuong and J. Peters, "Learning robot dynamics for computed torque control using local gaussian processes regression," (Edinburgh, Scotland, United kingdom), pp. 59 – 64, 2008.

[40] K. Teramura, H. Okuma, Y. Taniguchi, S. Makimoto, and S.-I. Maeda, "Gaussian process regression for rendering music performance," *Music Perception*, no. Icmpc 10, pp. 167–172, 2008.

[41] S. A. Gard, S. C. Miff, and A. D. Kuo, "Comparison of kinematic and kinetic methods for computing the vertical motion of the body center of mass during walking.," *Human Movement Science*, vol. 22, no. 6, pp. 597–610, 2004.

[42] "Cmu graphics lab motion capture database@http://mocap.cs.cmu.edu/info.php," Dec. 2011.

[43] L. Vaughan, B. Davis, and J. O'Connor, *Dynamics of Human Gait*. Human Kinetics Publisher, 1999.

[44] P. Meyer, "The travelling salesman problem@ONLINE," Dec. 2011.

[45] "Types of radial basis functions@http://www.physnet.uni-hamburg.de/physnet/matlab/help/toolbox/mbc/model/rbf3.html," Dec. 2011.

[46] H. He and W.-C. Siu, "Single image super-resolution using gaussian process regression," (Colorado Springs, CO, United states), pp. 449 – 456, 2011.

[47] A. P. Shon, K. Grochow, A. Hertzmann, and R. P. N. Rao, "Learning shared latent structure for image synthesis and robotic imitation," in *In Proc. NIPS*, pp. 1233–1240, MIT Press, 2006.

[48] S. Schaal, A. Ijspeert, and A. Billard, "Computational approaches to motor learning by imitation," *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, vol. 358, pp. 537–547, mar 2003.

[49] A. S. David, D. B. Grimes, C. L. Baker, and R. P. N. Rao, "A probabilistic framework for model-based imitation learning," in *In Proceedings of CogSci*, 2004.

[50] A. Meltzoff and W. Prinz, *The imitative mind: development, evolution, and brain bases*. University Press, Cambridge, 2003.

VITA

Ayesha Siddiqua

Candidate for the Degree of

Master of Science

Thesis: MACHINE LEARNING APPROACHES TO CENTER-OF-MASS ESTIMATION FROM NOISY HUMAN MOTION DATA

Major Field: Electrical and Computer Engineering

Biographical:

Personal Data: Born in Chittagong, Bangladesh on July 8th, 1982.

Education:

- Received the B.S. degree from Shahjalal University of Science and Technology, Sylhet, Bangladesh, 2006, in Computer Science & Engineering
- Completed the requirements for the degree of Master of Science with a major in Electrical and Computer Engineering Oklahoma State University in December, 2011.

Experience:

- Worked as a teacher's assistant for two undergraduate courses (2010–2011)
- Worked as a research assistant at Visual Computing and Image Processing Lab with machine learning for 2 years

Name: Ayesha Siddiqua Date of Degree: December, 2011

Institution: Oklahoma State University Location: Stillwater, Oklahoma

Title of Study: MACHINE LEARNING APPROACHES TO CENTER-OF-MASS ESTIMATION FROM NOISY HUMAN MOTION DATA

Pages in Study: 74 Candidate for the Degree of Master of Science

Major Field: Electrical and Computer Engineering

The focus of this research is to estimate Center Of Mass (COM) from noisy motion data. COM is a 3D point in the human body around which the mass of the whole body is equally distributed in each direction, and it plays an important role in many biomechanical studies of human motion, such as gait stability assessment. Traditionally, COM is computed using the Dempster's technique where the total COM is the sum of the weighted segmental COMs. Computation of COM normally requires expensive optical, mechanical and electromagnetic motion capture systems (MOCAP). Instead of high precision MOCAP systems, we can use low-cost sensors such as inertial motion sensors for efficient motion acquisition to compute COM. However, sensor-based motion acquisition could be noisy due to various ambient interference conditions and may be incomplete due to a limited number of sensors used. As a result, direct computation of COM from noisy motion data could be unreliable and even unusable in practice. In this research we have proposed two machine approaches to address this problem, i.e., manifold mapping and Gaussian Process Regression (GPR). First, we have designed a torus manifold which is a low-dimensional space to represent complex motion kinematics via two variables, i.e., the gait and the pose, representing different walking styles and different stages in a walking cycle, respectively. This torus manifold is shared by motion data (MOCAP) and the corresponding COM trajectories and provides with continuous space to extrapolate unknown motion along its COM trajectory. Moreover, given a noisy motion sequence, the torus manifold can be used to play a filtering role to denoise the motion data as well as a bridge to map the filtered motion data to the corresponding COM sequence. On the other hand, GPR does not account motion kinematics explicitly, and it directly approximates a non-linear mapping function between the input space (motion data) to the output space (COM data) where the covariance structure learned from noiseless motion data plays an important role for COM prediction. Our experiment has shown that GPR works better than the torus manifold for COM estimation from noiseless motion data. However, the performance of GPR degrades as the noise level increases in the motion data, largely due to the fact that its dependence on the covariance structure learned from the noiseless training data does not match that of the noisy motion data. In other words, unlike the torus manifold-based method, there is no filtering effect from GRP which makes it less accurate to estimate COM under noisy motion data. Still, both machine learning techniques have shown significant advantage over the method of direct computation of COM from noisy motion data.

ADVISOR'S APPROVAL: _____