SURGE Program 2014
ProjectReport

RoadMap Based Robot Motion Planning

Submitted by:
Siddhant Manocha (12714)
Department of Computer Sci. and Eng.
IIT Kanpur.

Under the guidance of:
Dr. Amitabha Mukerjee
Department of Computer Sci. and Eng.
IIT Kanpur.

Indian Institute of Technology , Kanpur
Uttar Pradesh, India – 208016

Abstract

In this work , we attempt to learn robot trajectories that can be generalised to unknown environments.We use vision based methods to learn an effective path from source configuration to destination using roadmap based approach.We capture images of random robot configuration(Y) with corresponding joint parameters(θ).Both data are seperately processed and are coupled to learn an effective map Y $\rightarrow \theta$ using Gaussian Process Regression.Dimensionality Reduction Techniques including Random Projections and Gaussian Process Latent Variable Model are implemented to form a lower dimensional embedding of image feature space to improve the mapping.Statistical evaluation of generated joint parameters against ground truth data for path generation match favourably.

Contents

List of Figures

Chapter 1

Problem Definition

As a central element of robotics, motion planning has important applications in areas such as navigation,protein folding, animation,virtual prototyping,search-and-rescue,etc.The objective of the motion planning problem is to find a valid (e.g., collision-free) path that takes a movable object (e.g., a robot) from an initial configuration to a goal configuration. We define configuration as an assignment of values to all of an objects degrees of freedom, which are essentially the various ways (e.g., by translation in the X direction or rotation about a joint) that the object can move.This approximation generally takes the form of a visual roadmap of the environment, with nodes representing feasible configurations and edges representing the transitions between them. Finding a path from an initial node to a goal node on this roadmap corresponds to finding a feasible path for the robot itself in its environment.

However , in most motion planning approaches, the configuration space is assumed to be known, implying a complete knowledge of both robot kinematics and obstacles. Uncertainty of these characteristics, however, is prevalent which makes such motion planning techniques inadequate for practical purposes.However, a sensing mechanism, for example, which uses video cameras and computer vision techniques, can help in overcoming uncertainties for guiding the motion of a robot.Vision and Image Processing techniques can be used to extract a set of relevant image parameters that are adequate to explain the relation between images and the corresponding joint angles of the robot.A set of images with the joint angles of the robot are taken in random positions in the given configuration space of the robot.The datasets generated are manipulated using techniques to automate the motion of a robot and planning of path from source to destination position.

The map y $\rightarrow$ θ ,i.e mapping from image space(y) to the joint angle space (θ) can be coupled with the proximity graph based on the neighbourhood points in the configuration space to generate a visual roadmap.Once we obtain the map, θ = f (y) , then given a new image y* , we can estimate the corresponding joint angle (θ) that can induce robot motion along the generated trajectory or roadmap.However, the input image feature space is high dimensional which adds to time as well as space complexity for any supervised learning algorithm.Thus,image space has to preprocessed to learn a lower dimensional representation that can be used for effective regression.

Thus the task of robot motion planning using vision based algorithms can be reduced to systematic preprocessing of the input image feature space followed by regression on the reduced space to learn a mapping between the image feature space (y) and the joint angle space (θ).This mapping can be used to generate a roadmap in the configuration space.

Chapter 2

Introduction

2.1 Background

2.1.1 Dimensionality Reduction

Most computer vision problems suffer from one major constraint that the number of degrees of freedom of the representation of input data is much larger than its intrinsic representation. For eg,representation of a image captured by a camera as a matrix of pixel values.Each pixel of image matrix corresponds to a single light sensor which are allowed to vary independently of the other sensors on the lens. However,for any natural image captured by a camera ,the neighbouring pixels are strongly corelated.The corealtion between different pixel values will actually manifest themselves on a lower dimensional manifold.The intrinsic representation can be found as a mapping from the observed representation(Y) to its lower dimensional representation(X).This mapping is referred to as the generating mapping,

$$Y=f(X)$$

Dimensionality reduction serves the purpose of reducing the number of parameters needed by a specific representation.

Spectral dimensionality reduction

Spectral dimensionality reduction is based on the assumption that the generating mapping f between the actual representation(Y) and the intrinsic representation(X) is invertible. Some of the approaches are stated below:

1)MultiDimensionalScaling(MDS):

Given dissimilarity measure δ_{ij} between i and j the aim is to find a geometrical configuration of points X=$[x_1, x_2..., x_N]$ such that the Euclidean distance $d_{ij} = \|x_i - x_j\|$ approximates the dissimilarity δ_{ij} . Classical MDS is formulated as a minimization of the following energy,

$$argmin_X \sum_{ij}(\delta_{ij} - d_{ij})$$

.The soluion for X=$[x_1, x_2..., x_N]$ that minimizes the error function can be found by spectral decomposition of the dissimilarilty matrix.

2)Random Projections:

The method of random projections[13] is a powerful dimension reduction technique that uses random projection matrices to project the data into lower dimensional spaces.The original data $X \in \Re^p$ is transformed to the lower dimensional $S \in \Re^k$, with $k \ll p$, via S = RX,where the columns of R are realizations of independent and identically distributed (i.i.d.) zero-mean normal variables, scaled to have unit length.Random Projections is computationally cheap and can be easily implemented.It has been shown empirically that results with the random projection method are comparable with results obtained with PCA, and take a fraction of the time PCA requires . To reduce the computational burden of the random projection method, at a slight loss in accuracy, the random normal projection matrix R may be replaced by thresholding its values to -1 and +1, or by matrices whose rows have a fixed number of 1s (at random locations) and the rest 0s.

3)Isomap:

Isomap was presented as a non-linear modification of MDS[9].Isomap is based on the proximity graph approach that represent a specific neighborhood relationships in the data.The fundamental idea behind proximity graph based algorithms for dimensionality reduction is that locally the data can be assumed to lie on a linear manifold. This means that locally the distance in the original representation of the data will be a good approximation to the manifold distance. Therefore the neighborhood relationship used for proximity graphs in dimensionality reduction is the inter-distance between points in the original representation. Usually the graphs are constructed either from an N nearest neighbor algorithm where the N closest points are connected. In Isomap it is suggested that the manifold distance be approximated by the shortest path through the proximity graph. By computing the shortest path

through the proximity graph a dissimilarity measure can be found between each data point onto which MDS can be applied.
4)Locally Linear Embedding:
Local Linear Embeddings (LLE)[7] is also based on the preservation of a proximity graph structure. LLE is based on the assumption that the manifold can be locally approximated using small linear patches.By rotating and translating each of these patches the full manifold structure can be modeled. LLE is a two step algorithm, in the first step each point in the data set is described by the nodes connected in the proximity graph as expansion,

$$W^{'} = argmin_W \sum_i^N \left\| y_i - \sum_{j \in N(i)} W_{ij} y_j \right\|$$

where y lies image feature space and N(i) is the index set of points that are connected to i in the proximity, subject to

$$\sum_j W_{ij} = 1$$

The optimal weights W can be solved in closed form from above equation.[7].Assuming that the manifold is locally linear, the reconstruction weights should summarize the local structure of the data and should therefore be equally valid in reconstructing the manifold representation of the data X. To find this manifold representation a second minimization is formulated,

$$X^{'} = argmin \sum_i \left\| x_i - \sum_j W_{ij} x_j \right\|$$

where $X^{'}$ is the intrinsic representation of the data.

Generative dimensionality reduction

Generative approaches to dimensionality reductions aim to model the observed data as a mapping from its intrinsic representation. The underlying representation is often referred to as the latent representation of the data and the models as latent variable models for dimensionality reduction[8][3].Description of one such technique (Gaussian Process Latent Variable Model) is given under Gaussian Process.

2.1.2 Gaussian Process

A D dimensional Gaussian distribution is defined by a D × 1 mean matrix and a D × D covariance matrix.Gaussian processes (GPs) extend multivariate Gaussian distributions to infinite dimensionality.It is chracterized by the mean and covariance where the mean and covariance is defined not by fixed size matrices but by a mean $\mu(x)$ and a covariance $k(x,x^{'})$ function, defined over infinite index sets, x.Now, the n observations in an arbitrary data set, $Y = [y_1, \ldots, y_n]$ can always be imagined as a single point sampled from some multivariate (n -variate) Gaussian distribution.Very often, its assumed that the mean of this GP is zero everywhere.What relates one observation to another in such cases is just the covariance function,$k(x,x^{'})$.A popular choice for the covariance function is the "squared exponential" often referred to as the Gaussian Kernel.

$$k(x,x^{'}) = \sigma^2 exp\left[-\frac{(x-x^{'})^2}{2l^2}\right]$$

Gaussian Process Regression [1],[6]

In regression , we are interested in modelling a relationship between the input domain $X \in \Re^D$ and target domain $Y \in \Re^k$ from a set of observations $x_i \in X$ and $y_i \in Y$ where i=1,2....n.Assuming the given functional relationship and that the observations have been corrupted by additive Gaussian noise , we are interested in modelling

$$y_i = f(x_i) + \varepsilon$$

where $\varepsilon \in N(0,\sigma^2)$
A GP can be used to specify a prior distribution over the functional relationship,i.e, f $\sim$ GP (μ, k) with squared exponential covariance function

$$y_i = f(x_i) + N(0,\sigma^2)$$

This is equivalent to the regression problem

$$y_i = f(x_i)$$

where

$$k(x,x^{'}) = \sigma_f^2 exp\left[-\frac{(x-x^{'})^2}{2l^2}\right] + \sigma_n^2\delta(x,x^{'})$$

where $\delta(x,x^{'})$ is the Kronecker delta function.So,given set of n observations ,our aim is to predict y^* for a new input value x^*.To prepare for Gaussian process regression, we have to calculate covariance function for all possible combinations of these points summarizing in three matrices,

$$K = \begin{pmatrix} k(x_1,x_1) & \cdots & k(x_1,x_n) \\ \vdots & \ddots & \vdots \\ k(x_n,x_1) & \cdots & k(x_n,x_n) \end{pmatrix}$$

$$K_* = [k(x_*, x_1)k(x_*, x_2)..........k(x_*, x_n)]$$

$$K_{**} = k(x_*, x_*)$$

Since the key assumption in GP modelling is multivariate Gaussian distribution, we have that

$$\begin{pmatrix} y \\ y_* \end{pmatrix} \sim N \left[0, \begin{pmatrix} K & K_*^T \\ K_* & K_{**} \end{pmatrix} \right]$$

where T indicates matrix transposition.We are interested in the conditional probability $p(y_*|y)$:given the data how likely is the prediction of y_*?It can be shown that

$$y_*|y \sim N(K_* K^{-1} y, K_{**} - K^{-1} K_*^T)$$

For the above distribution,our best estimate of y_* is the mean of the above distribution.

$$\mu(y_*) = K_* K^{-1} y$$

and the uncertainity in the distribution is captured in its variance

$$var(y_*) = K_{**} - K^{-1} K_*^T$$

However the reliability of our regression is dependent on how well we select the covariance function. Clearly if its parameters call them $\theta = (\sigma_f, \sigma_n, l)$ are not chosen sensibly, may lead to poor results. Our maximum a posteriori estimate of θ occurs when $p(\theta|x, y)$ is at its greatest.Bayes theorem tell us that assuming we have little prior knowledge what θ should be this corresponds to maximizing the likelihood $p(y|x, \theta)$,given by

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

We can simply run some multivariate optimization algorithm,for eg SCG(Scalable Conjugate Gradient) on this equation to find a reasonable estimate of the covariance function parameters.

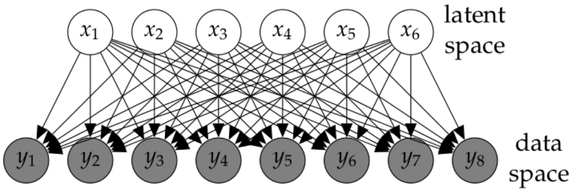


Figure 2.1: Graphical rep. of GPLVM

Gaussian Process Latent Variable Model

The GPLVM is a probabilistic dimensionality reduction technique that uses Gaussian Processes (GPs) to find a non-linear manifold of some data that seeks to preserve the variance of the data in latent space[2],[3],[4]. The latent space $X \in \Re^{n\times q}$ is assumed to be related to the mean centered data set, $Y \in \Re^{n\times p}$ through a projection matrix $W \in \Re^{p\times q}$ that is corrupted by noise.This linear relation is non-linearised using the kernel trick.Moreover the noise is assumed to possess a normal gaussian distribution.

$$y_i = Wx_i + \varepsilon_i$$

where

$$\varepsilon_i \sim N(0, \sigma^2 I)$$

Using standard latent variable approach for generative dimensionality reduction , we can obtain the given distribution $p(Y|X, W)$ as

$$p(Y|X, W) = \prod_{i=1}^{n} N(y_{i,:}|Wx_{i,:}, \sigma^2 I)$$

To solve for unknown weight matrix , we assume some prior knowledge over the weight parameters which serves as the trick for Gaussian Process Latent Variable Model. We define Gaussian prior over parameters W with unit variance and integrate them to obtain a marginal likelihood .

$$p(W) = \prod_{i=1}^{p} N(w_{i,:}|0, I)$$

Using the equations,

$$p(Y|X, W) = \prod_{i=1}^{n} N(y_{i,:}|Wx_{i,:}, \sigma^2 I)$$

$$p(W) = \prod_{i=1}^{p} N(w_{i,:}|0, I)$$

and integrating out the parameters W, we obtain the marginal likelihood,

$$p(Y|X) = \prod_{j=1}^{p} N(y_{:,j}|0, XX^T + \sigma^2 I)$$

which is equivalent to

$$p(Y|X) = \prod_{j=1}^{p} N(y_{:,j}|0, K)$$

$$K = XX^T + \sigma^2 I$$

The covariance matrix here is actually a covariance function.The covariance matrix $K = XX^T + \sigma^2 I$ is a product of Gausian processes with linear kernels.We replace this linear kernel with a non linear kernel to obtain a non-linear model(kernel trick).
This linear kernel can be replaced with squared exponential kernel which has the form

$$k(x_{i,:}, x_{j,:}) = \alpha exp(\frac{\|x_{i,:} - x_{j,:}\|^2}{2l^2})$$

However we now possess X,$\theta = (\alpha, l, \sigma^2)$ as unknowns which can be obtained by maximizing the mariginal likelihood function $p(Y|X, \theta)$ which is given as

$$\log p(Y|X, \theta) = -\frac{p}{2} \log |K| - \frac{1}{2} tr(K^{-1}YY^T) + const.$$

For non-linear mappings, a closed-form solution is not available and the likelihood function is optimised with respect to the latent values X using conjugate gradient optimisation.Maximising the marginal likelihood with respect to the latent points and the hyperparameters θ results in the latent space representation of the GPLVM.

$$(X^{'}, \theta^{'}) = argmax_{X,\theta} P(Y|X, \theta)$$

Shared Gaussian Process Latent Variable Model

Gaussian Process Latent Variable Model has been extended to construct a shared latent space model between two observation spaces.The two sets of variables, Y and Z possess shared latent space X,and may also possess private independent space .The likelihood function is taken to the the product of each individual likelihood function, conditioned on a common latent space[10].This leads to the optimisation of two different sets of hyperparameters for the two kernel functions.The joint likelihood of two observation spaces is given by

$$P(Y, Z|X, \theta_S) = P(Y|X, \theta_Y)P(Z|X, \theta_Z)$$

where $\theta_S = (\theta_Y, \theta_Z)$ is a concatenation of two sets of hyperparameters.

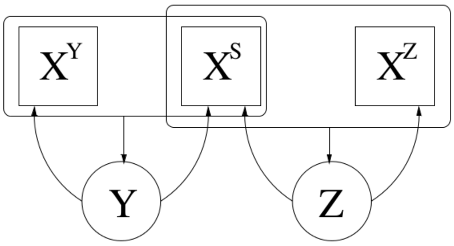


Figure 2.2: Graphical rep. of Shared GPLVM

The SGPLVM can be viewed as a non-linear extension of Canonical Correlation Analysis (CCA). CCA learns a correspondence between two datasets by maximising their joint correlation. The Shared GPLVM (SGPLVM) has been used to learn a mapping between pose and silhouette data by Ek et al[10].In this technique , NCCA(Non Consolidating Component Analysis) has been used to initialise the private spaces while CCA(Canonical Corelation Analysis) has been used for initialisation of shared spaces.

2.2 Motivation

Mapping between the image space (Y) and joint parameters space(θ) is generally non linear.Most of the regression techniques do piece wise linear approximations and are therefore not able to capture non linearity in the data.Gaussian Process Regression is characterized by a non linear squared exponential covariance function that is capable for finding an effective map between two feature spaces.

However,the input image feature space is high dimensional(640*480 dimensional) and any type of supervised learning motion planning algorithm require a very high time and space complexity .It has been shown by Awasthi, Mukerjee et.al [12] that ISOMAP, a nonlinear dimensionality reduction technique,if used to reduce the dimensionality of the images of a simulated planar arm with 2 joints from 30000 to a 3-dimensional embedding, the performance of a neural network in regressing the images with the angles improved significantly. Moreover, since the images are smooth functions of the angles,its intrinsic dimension is the same as the number of degrees of freedom of motion.Thus,image feature space can be preprocessed to find a lower dimensional manifold which is capable of capturing maximum variance in the data.Moreover,due to their cheap computation and distance preserving nature, Random Projection can be used in conjunction with other Non Linear Dimension Reduction techniques such as GPLVM, to speed up dimensionality reduction process.

Gaussian Process Latent Variable Model has been used for dimensionality reduction since it can be extended to model shared latent space between two different feature spaces and learn a direct mapping between two feature spaces.Ek et al suggested SGPLVM model that has been used to model a relation between human pose and corresponding silhouettes.

Chapter 3

Work Done and Results

3.1 Dataset Collection

We used two perpendicular cameras to collect videos of different robot configurations for 6 degrees of freedom CRS Robotic arm using only three degrees of freedom.We restricted the joint angles to specific range due to certain robot constraints.We collected 32 videos chracterized by 135 different configurations given as input to the robot.In order to form a labelled dataset , we exploited the time lag that existed between two moving frames.We extracted 4320 images corresponding to the joint angles.Each image was 640*480;ie 307200 dimensional.We used Vibe algorithm[11] to perform background subtraction and form silhouettes from these images.We rescaled these images to 220*165 which is around 36,300 dimensional to reduce image dimension. We projected this 36300 dimensional image feature space to 7175 dimensions using random projections.We now had 7175 dimensional images feature space corresponding to the images from two different cameras,say Y_1 and Y_2. We concatenated them to form 14350 dimensional image feature space i.e

(a) Original Image

(b) Vibe Processed Image

Figure 3.1: Background Subtracted Image

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix}$$

where $Y_1 \in \Re^{7175}$, $Y_2 \in \Re^{7175}$,thus $Y \in \Re^{14350}$.

We once again applied random projection on Y to reduce it to Y' where $Y' \in \Re^{7175}$.

As we varied only 3 degrees of freedom so our joint angle vector is 3 dimensional.Out of these 4320 collected datapoints, we used 3645 images for training and remaining 675 images for validation with their corresponding joint angles.

3.2 Regression

3.2.1 Approach1:SGPLVM Model

We implemented Shared Gaussian Process Latent Variable Model as suggested by Ek et.al[10] as an extension of GPLVM .Shared GPLVM assumes that the two feature spaces, namely the image space(Y) and joint parameter space(θ) possess a shared lower dimensional manifold.We used 7175 dimensional images and 3 dimensional joint angles as our input feature space for testing the SGPLVM model.We used 15 top eigen vectors for images and 3 eigen vectors for angles from PCA to represent the overall variance present in the data.We further assumed two latent spaces being shared between the observation spaces and 1 independent latent space for both feature spaces.However , we cannot obtain satisfactory results from this technique and average value of absolute error was as high as 12 degrees per joint angle.Thus we switched to the second approach for our regression problem.Refer figure 3.2 and 3.3

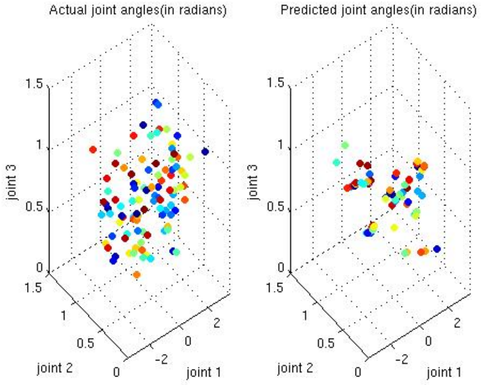


Figure 3.2: Comparison:SGPLVM result with ground truth(100 test points)

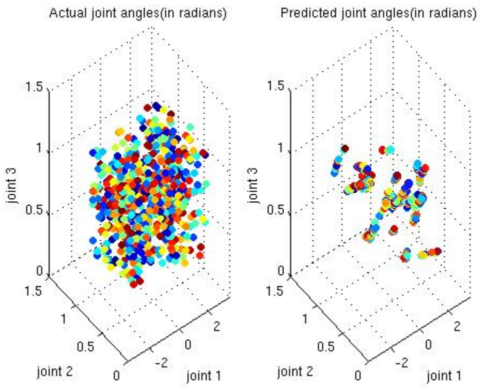
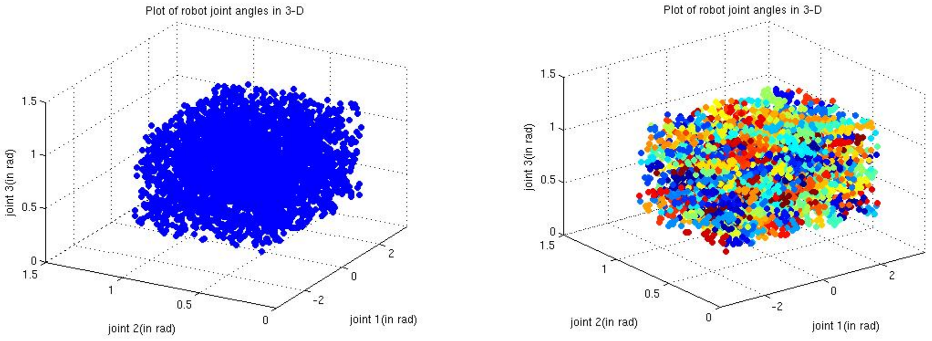


Figure 3.3: Comparison:SGPLVM result with ground truth(675 test points)

3.2.2 Approach2:Gaussian Process Regression

The dimensionality of the input image feature space is still too high which adds to the computational complexity of our regression problem and it cannot be easily used to learn an effective mapping with the joint angles.Since the intrinsic dimensionality of the image space will be a lot less than the given reduced dimension, we implemented non linear dimensionality reduction techniques to project the image space to lower dimensional space.Ref figure 3.4 and 3.5 for plot of image space and joint angle space

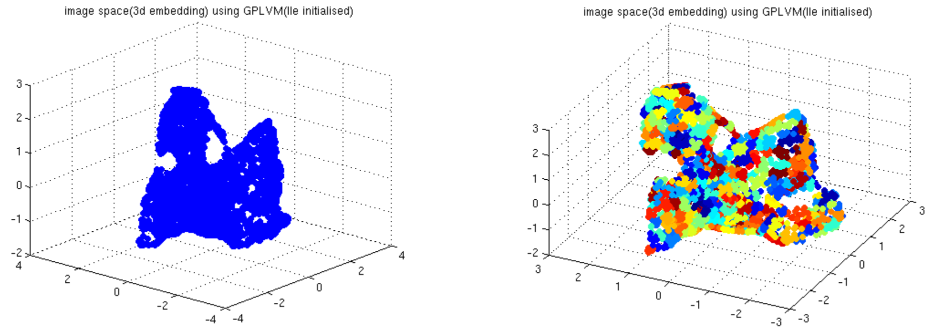


(a)

(b)

Figure 3.4: plot for joint angles in joint angle space
(colors are derived from proximity on basis of K means clustering)

We used different techniques as Isomap, Locally Linear Embedding and Gaussian Process Latent Variable Model(GPLVM) and compared their performance.Since GPLVM requires initialisation using other spectral dimensionality reduction method, we considered GPLVM initialised with LLE and



(a) (b)

Figure 3.5: 3d embedding of image space using GPLVM(lle init.)
(colors are derived from proximity on basis of K means clustering)

initialised with Isomap seperately.The 7175 dimensional feature space was reduced to as low as 30 dimensional which served as the input feature space for Gaussian Process Regression.Refer figure 3.6,3.7,3.8

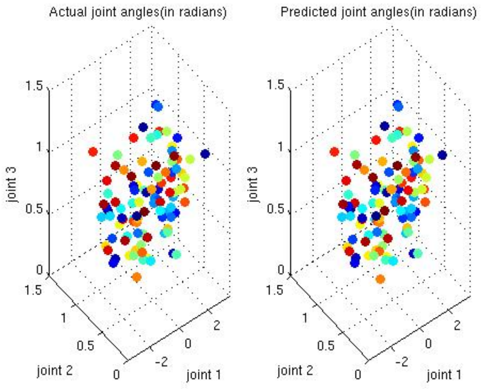


Figure 3.6: Comparison:Gplvm(lle init.) followed by GP Regression results with ground truth(100 test points)

It was observed that GPLVM gave better results under GPRegression where GPLVM initialised with LLE gave superior results than that initialised with Isomap.Moreover on statistical analysis of the output,it was observed that the results were close to ground truth.Refer figure 3.9

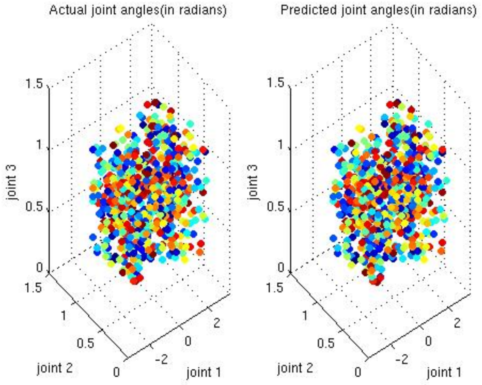


Figure 3.7: Comparison:Gplvm(lle init.) followed by GP Regression results with ground truth(675 test points)

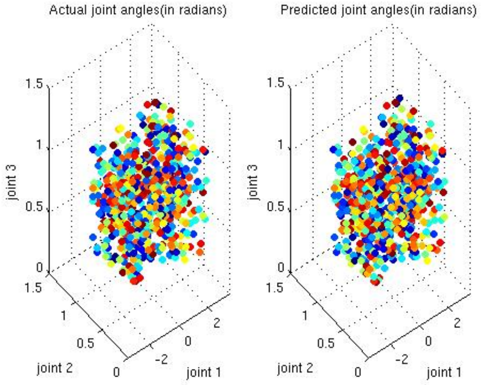


Figure 3.8: Comparison:LLE followed by GP Regression results with ground truth(675 test points)

3.3 Visual RoadMap

Constructing a roadmap and obtaining a valid path require us to create a proximity graph of the input image space based on its neighbourhood. To make a visual roadmap, we exploited two assumptions, firstly, since the image initially was very high dimensional and computing Euclidean distances between all possible pairs was expensive , we used 7175 dimensional image vector for the purpose , secondly, if two images are close together then their corresponding joint angles will also be close together.Under above stated assumptions,we implemented K-means clustering on these 3645 points and split them into 729 clusters , roughly 5 points in each cluster.All of these 3645 points in the configuration space are assumed to be connected where

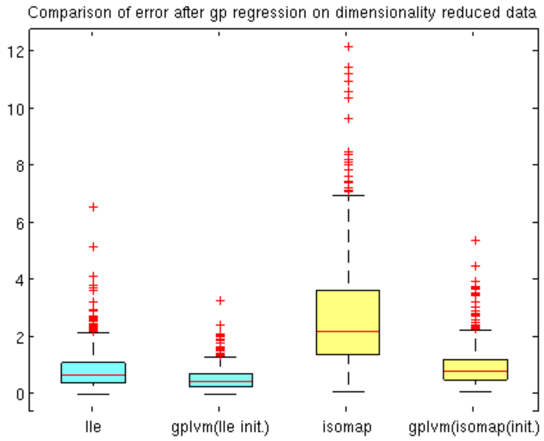


Figure 3.9: Comparative error analysis using different dimensionality reduction methods followed by GP Regression

Table 3.1: Comparison of error among various dimensionality reduction techniques

Technique	Avg. error	Min error	Max error	Standard Deviation
LLE	0.853	0.030	6.507	0.66
GPLVM(lle init.)	0.558	0.013	3.247	0.38
Isomap	2.713	0.081	12.10	1.85
GPLVM(isomap init.)	0.968	0.084	5.378	0.66

edge weights were derived from distance between centroids of the clusters.We represent each cluster by its centroid and find pairwise distance matrix between centroids of different clusters, let say DIST . Now for any given point Z_i(joint angle) we first find the cluster to which it belongs let say j and we then sort the array DIST(j,:) in increasing order. Now suppose we want to update the entry W(i,k) and if k point belongs to mth cluster then W(i,k) is updated as the rank of the mth cluser in the sorted array DIST(j,:)

For any two unknown configurations say A,B we first project corresponding images to 7175 dimensions using random projections followed by reduction to 30 dimensions using GPLVM. We then use GP regression to obtain its corresponding joint angles.Nearest neighbour for A and B is found in the above graph among 3645 points.Say A has closest neighbour A_1 while B has closest neighbour B_1.Path between A_1 and B_1 is found by applying Dijakstra's Algorithm on above distance matrix and shortest path is returned.

Thus $A - -A_1 - - - -$(shortest path from proximity graph)$- - - - -B_1 - -B$ is reported as a path between the unknown configurations.Refer figure 3.10,3.11,3.12,3.13,3.14,3.15

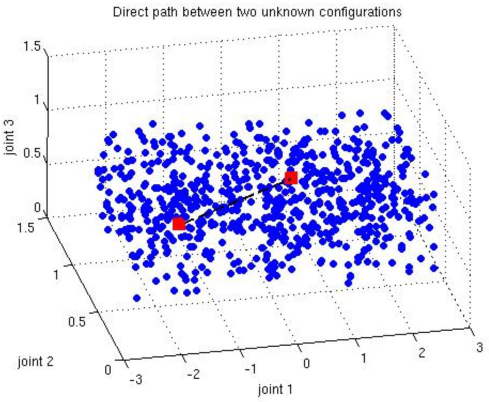


Figure 3.10: Direct Path between test points 1 and 100

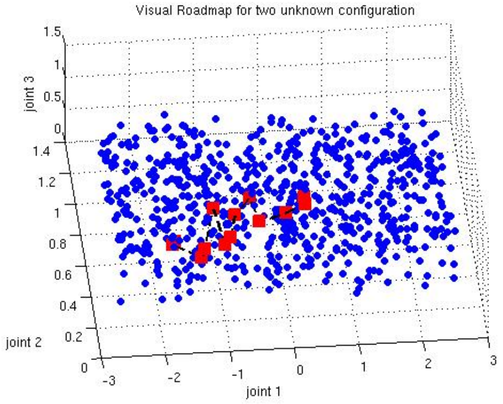


Figure 3.11: Roadmap between test points 1 and 100

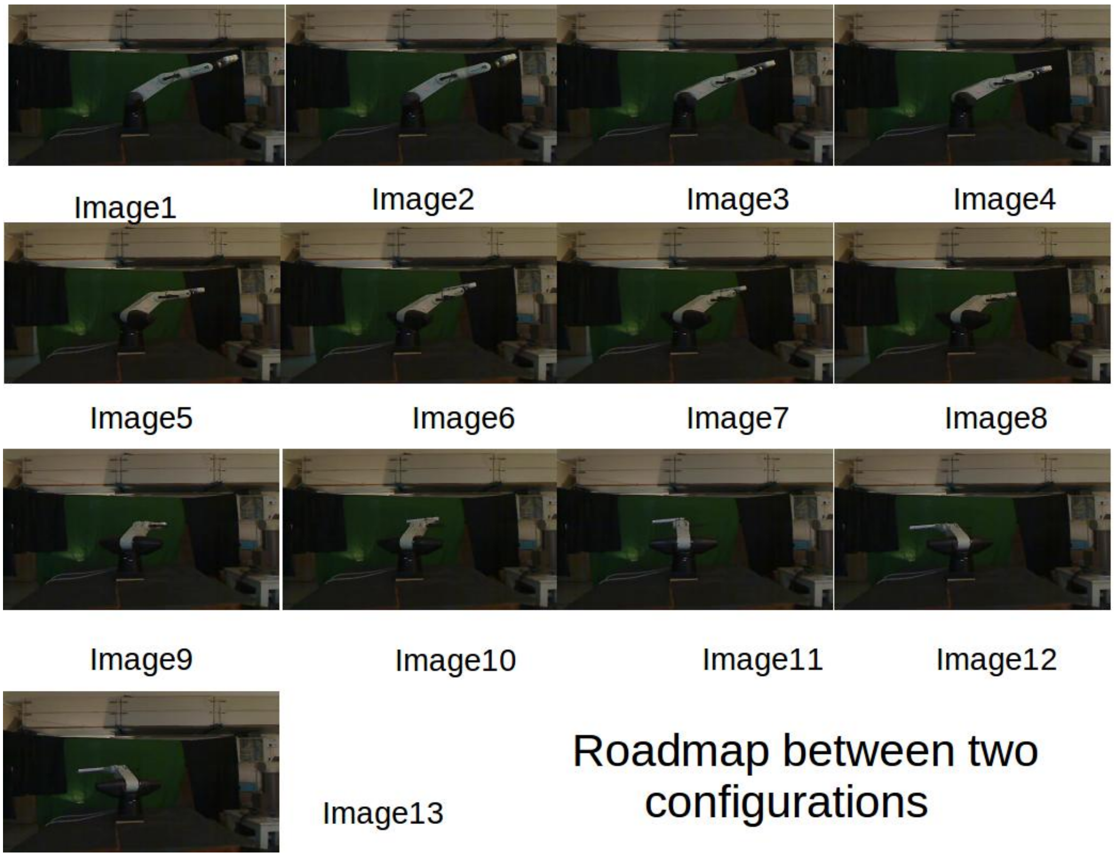


Figure 3.12: Transition showing various configurations on road map (test points 1,100)

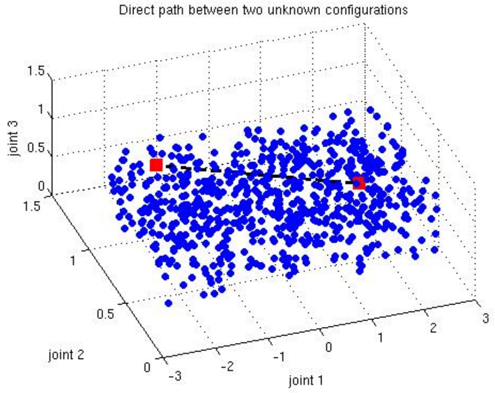
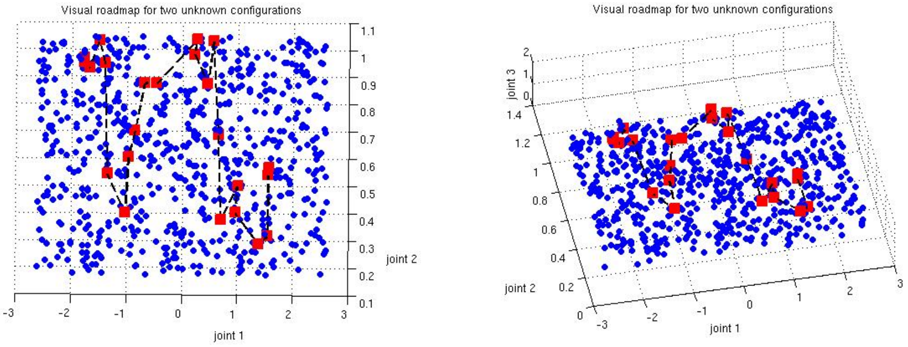


Figure 3.13: Direct Path between test points 17 and 289



(a) path between test points 17 and 289(2D view)

(b) path between test points 17 and 289(3D view)

Figure 3.14: Roadmap between test points 17 and 289

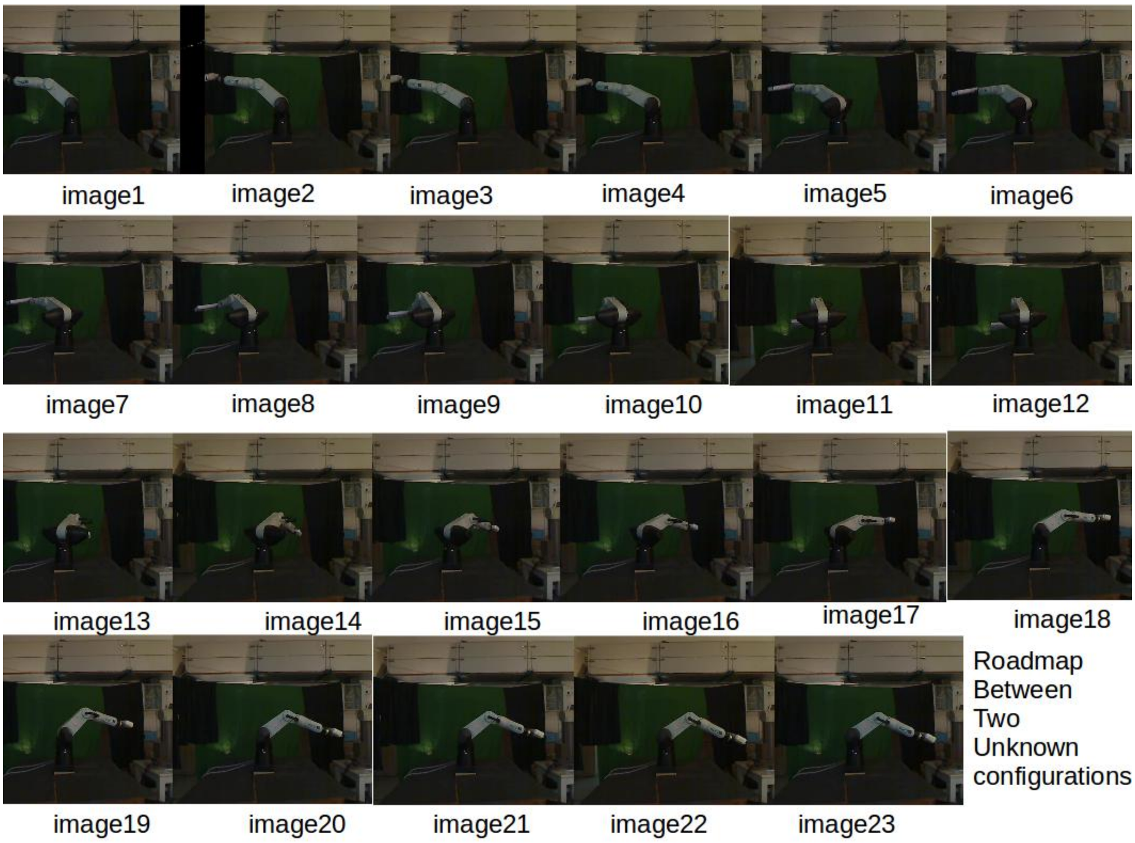


Figure 3.15: Transition showing various configurations on road map (test points 17,289)

Chapter 4

Conclusion and Future Work

We have shown how Gaussian process regression can be coupled with Gaussian Process Latent Variable Model to find an effective inverse mapping between the image space and the joint angle space.Statistical analysis of the results show that the obtained results are favourably close to the ground truth.The path generated between two configuration spaces also passes through a number of close observed configurations in the input configuration space.

In future work,we would like to update the lower dimensional manifold structure and reconstruct the distance matrix and the proximity graph to take into account any obstacles in the surrounding environment.Shared manifold model with non linear initialization can be implemented to model the relationship between the image space and the joint parameter space.Dynamical models can be included whereby a continuous sequence of robot motion images can be utilised to improve upon the manifold structure.This can help to solve multimodalities in the solution in case of multivalued regression.

Generating a labelled data (images and the corresponding joint angles) is a tedious and time taking task.However,a lot of unlabelled data is readily available from the videos.As proposed by Navratnam et al.[5] this unlabelled data can be combined with labelled samples to improve upon the manifold structure which can give better inferences even in case of sparse dataset.

Acknowledgments

I take this opportunity to express my profound gratitude and deep regards to Dr.Amitabha Mukerjee for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him time to time shall carry me a long way in the journey of life on which I am about to embark.

I also take this opportunity to express a deep sense of gratitude to my partner Satvik Gupta, for his cordial support and valuable information, which helped me in completing this task through various stages.

My thanks and appreciations also go to my friends who helped me during the course the project and people who have willingly helped me out with their abilities.

I am highly indebted to Neil Lawrence for his publicly available Gaussian Process Software.

Siddhant Manocha
July,2014
Indian Institute of Technology,Kanpur

References

[1] Q. Candela and C. Rasmussen. A unifying view of sparse approximate gaussian process regression. Journal of Machine Learning Reserch Volume, 6:19351959, 2005.,

[2] N. D. Lawrence. Gaussian Process Models for Visualisation of High Dimensional Data. In S. Thrun, L. Saul, and B. Sch olkopf, editors, Gaussian Process Models for Visualisation of High Dimensional Data, volume 16, pages 329336, Cambridge, MA, 2004., `<urlhere>`

[3] N. D. Lawrence. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. The Journal of Machine Learning Research, 6:17831816, November 2005.

[4] N. D. Lawrence and Qui nonero-Candela. Local distance preservation in the GP-LVM through back constraints. In R. Greiner and D. Schurmans, edi- tors, Local distance preservation in the GP-LVM through back constraints, volume 21, pages 513520, New York, NY, USA, 2006. ACM.

[5] R. Navaratnam, A. Fitzgibbon, and R. Cipolla. The joint manifold model. In IEEE International Conference on Computer Vision (ICCV), 2007.

[6] C. E. Rasmussen and C. K. I. Williams. Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning). The MIT Press, 2005.

[7] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. Science, 290(5500):23232326, 2000.

[8] M. Tipping and C. Bishop. Probabilistic principal component analysis. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 61(3):611622, 1999.

[9] H. Choi and S. Choi. Kernel isomap. Electronics letters, 40(25):16121613, 2004.

[10] C. H. Ek. Shared Gaussian Process Latent Variable Models. PhD thesis, Oxford Brookes University, 2009.

[11] M. Van Droogenbroeck and O. Barnich. ViBe: A Disruptive Method for Background Subtraction. In T. Bouwmans, F. Porikli, B. Hoferlin, and A. Vacavant, editors, Background Modeling and Foreground Detection for Video Surveillance, chapter 7. Chapman and Hall/CRC, June 2014

[12] Ankit Awasthi, Sadbodh Sharma, and Amitabha Mukerjee. Learning of motor maps from perception: a dimensionality reduction approach.

[13] Chinmay Hegde, Michael Wakin, and Richard Baraniuk. Random projec- tions for manifold learning. In Advances in neural information processing systems, pages 641648, 2007.