GENERALIZED NON-PARAMETRIC FUNCTION APPROXIMATION

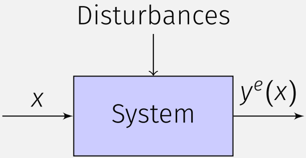
Suren Kumar
Research Fellow
Electrical Engineering and Computer Science

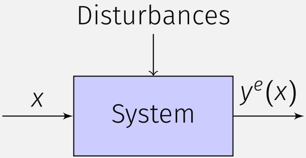
August 20, 2016

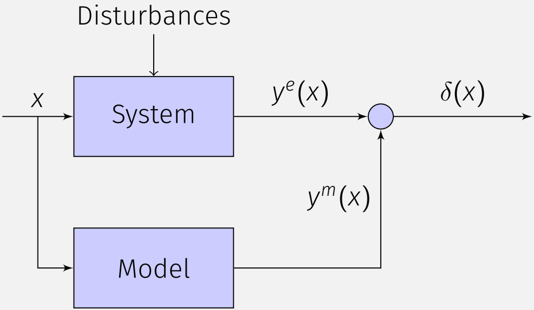
University of Michigan, Ann Arbor
https://github.com/surenkum/uq_gaussian_processes

INTRODUCTION

INTRODUCTION




Disturbances
x
System
$y^e(x)$


Disturbances
x
System
$y^e(x)$
$\delta(x)$
$y^m(x)$
Model

Bias Correction

$$y^e(x) = y^m(x) + \delta(x) + \epsilon$$

where ϵ is the experimental uncertainty.

Bias Correction

$$y^e(x) = y^m(x) + \delta(x) + \epsilon$$

where ϵ is the experimental uncertainty.

Parameter Calibration

$$y^e(x) = y^m(x, \theta^*) + \epsilon$$

Bias Correction

$$y^e(x) = y^m(x) + \delta(x) + \epsilon$$

where ϵ is the experimental uncertainty.

Parameter Calibration

$$y^e(x) = y^m(x, \theta^*) + \epsilon$$

Bias Correction and Parameter Calibration

$$y^e(x) = y^m(x, \theta^*) + \delta(x) + \epsilon$$

- Lack of simulation results $y^m(x, \theta)$
- Parameterizing discrepancy function $\delta(x)$

Why Bayesian

- Integration of prior knowledge

Bayesian calibration of computer models, Marc C. Kennedy, Anthony O'Hagan in Journal of the Royal Statistical Society, 2001

GAUSSIAN PROCESSES

Function Approximation

$\mathrm{f} : x \mapsto y$

Function Approximation

$f : x \mapsto y$

Parametric Models: Pros

Easy to interpret

Parametric Models: Cons

- Simpler models lack expressive power
- Complex models require lots of data
- Predictions are dependent on the model
- Representation of uncertainty in the input space

$$y = x + 0.005x^2 + \epsilon, x \in [1, 10] \cup [21, 25], \epsilon \sim \mathcal{N}(0, 10)$$

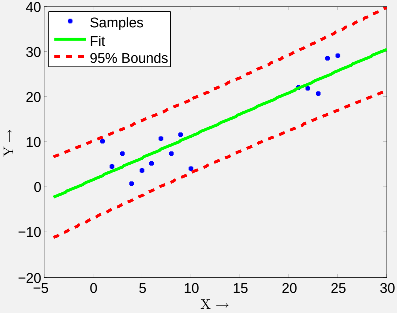


Figure 1: Linear Regression yields the model $y = mx + c$ with $m = 0.9644$ and $c = 1.5891$

$$y = x + 0.005x^2 + \epsilon, x \in [1, 10] \cup [21, 25], \epsilon \sim \mathcal{N}(0, 10)$$

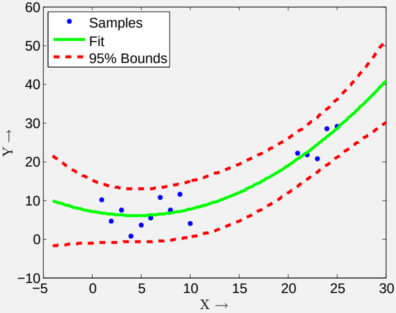


Figure 2: Quadratic model yields the model $y = ax^2 + bx + c$ with $a = 0.0534$, $b = -0.04781$ and $c = 7.1231$

$$y = x + 0.005x^2 + \epsilon, x \in [1, 10] \cup [21, 25], \epsilon \sim \mathcal{N}(0, 10)$$

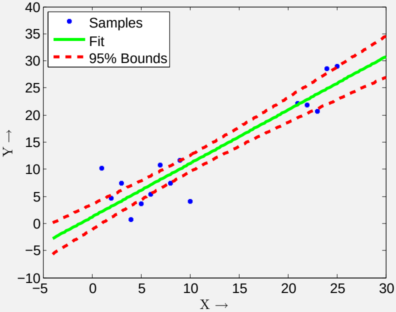


Figure 3: Bayesian Regression fit uses a zero mean Gaussian observation noise with variance 10 and a zero mean Gaussian with diagonal variance 5 prior on slope and intercept. The estimation process results in a mean parameter estimate of 0.9867 slope and 1.1797 intercept.

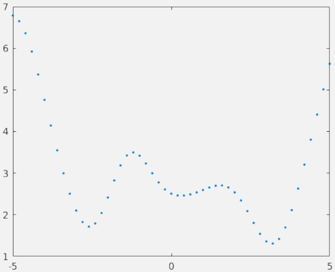
Definition

Gaussian Process is a collection of random variables, any finite number of which are joint Gaussian distribution.
$f(x) \sim \mathbb{GP}(m(x), k(x, x^1))$

- Generalization from mean vector and covariance matrix
- Indexed by x
- At every value of x, we have an associated random variable

VISUALIZING GAUSSIAN PROCESS

$f(x) \sim \mathbb{GP}(m(x), k(x, x^1))$
$m(x) = \frac{1}{4}x^2$, and $k(x, x^1) = \exp\left(-\frac{1}{2}(x - x^1)^2\right)$
Let's sample at n different locations
$\mu(x_i) = \frac{1}{4}x_i^2$, and $\Sigma_{ij}(x_i, x_j) = \exp\left(-\frac{1}{2}(x_i - x_j)^2\right)$, $i, j \in [1, ..., n]$



$f(x) : x \mapsto y$.

- Input: Ground truth data from $(\mathrm{x}, \mathrm{f}(\mathrm{x}))$ from n observations
- Output: Output f_* for a novel input x_*, $p(y_*|x_*)$.
- Measurement Process: $y = f(x) + \epsilon$, $\epsilon \sim \mathcal{N}(0, \sigma^2)$

Joint Distribution:

$$\begin{bmatrix} f \\ f_* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu \\ \mu_* \end{bmatrix}, \begin{bmatrix} \Sigma & \Sigma_* \\ \Sigma_*^T & \Sigma_{**} \end{bmatrix}\right) \tag{1}$$

Marginalization:

$$f_*|f \sim \mathcal{N}(u_* + \Sigma_*^T \Sigma^{-1}(f - \mu), \Sigma_{**} - \Sigma_*^T \Sigma^{-1} \Sigma_*) \tag{2}$$

Posterior Process:

$$\begin{aligned} \mathsf{f}|\mathcal{D} \sim & \mathbb{GP}(m_D, k_D) \\ & m_D(x) = m(x) + \Sigma(X,x)^T \Sigma^{-1}(f - m) \\ & k_D(x,x^1) = k(x,x^1) - \Sigma(X,x)^T \Sigma^{-1} \Sigma(X,x^1) \end{aligned} \tag{3}$$

Observation Noise:

$$y \sim \mathbb{GP}(m, k + \sigma_n^2 \delta_{ii^1}) \tag{4}$$

Parameterizing GP Priors
$m(x) = ax^2 + bx + c$, and $k(x, x^1) = \sigma_y^2 \exp\left(-\frac{(x-x^1)^2}{2l^2}\right) + \sigma_n^2 \delta_{ii^1}$
Hyperparameters: $\theta = \{a, b, c, \sigma_y, \sigma_n, l\}$
Fitting:

$$\begin{aligned} L &= \log p(y|x, \theta) \\ &= -\frac{1}{2}\log|\Sigma| - \frac{1}{2}(y-\mu)^T \Sigma^{-1}(y-\mu) - \frac{n}{2}\log 2\pi \end{aligned} \tag{5}$$

Estimation:

$$\begin{aligned} \frac{\partial L}{\partial \theta_m} &= -(y-\mu)^T \Sigma^{-1} \frac{\partial m}{\partial \theta_m} \\ \frac{\partial L}{\partial \theta_k} &= \frac{1}{2} trace(\Sigma^{-1} \frac{\partial \Sigma}{\partial \theta_k}) + \frac{1}{2}(y-\mu)^T \frac{\partial \Sigma}{\partial \theta_k} \Sigma^{-1} \frac{\partial \Sigma}{\partial \theta_k}(y-\mu) \end{aligned} \tag{6}$$

Non-parametric does not mean no parameters. Parametric models can through away data once parameters are tuned.

$$y = x + 0.005x^2 + \epsilon, x \in [1, 10] \cup [21, 25], \epsilon \sim \mathcal{N}(0, 10)$$

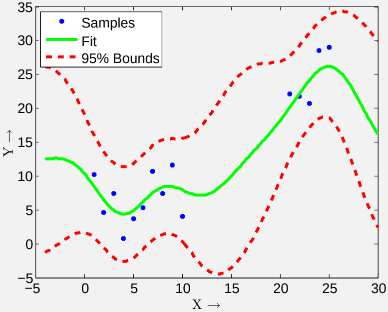
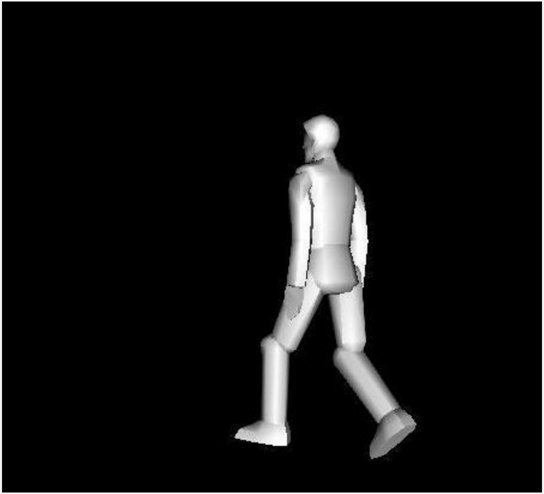


Figure 4: GPR with constant mean function, Gaussian likelihood and Squared Exponential covariance function

POSE ESTIMATION

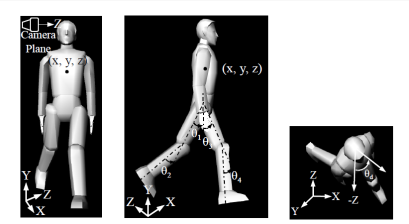


PARTS OF POSE ESTIMATOR

- Motion Model : Second Order motion continuity
- Observation Model : Mapping features to pose using Gaussian Processes
- Estimation : Kalman Filter

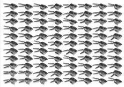
PREVIOUS WORK

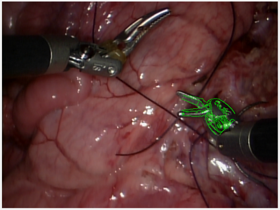
Model Based Optimization [Agarwal et. al, RSS 2012]:



- Computationally expensive
- Model Required: Articulation, CAD
- Often requires background subtraction

Template Based Methods [Reiter et. al, CARS 2012]



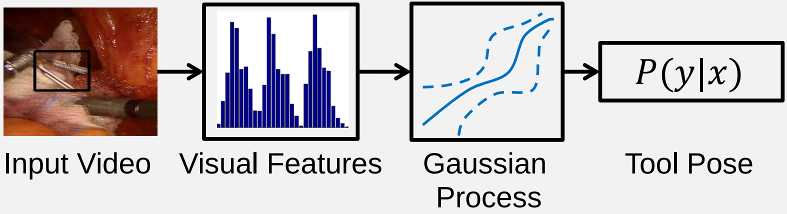


- Curse of Dimensionality
- Model Required: Articulation, CAD, Visual

MODEL-LESS POSE ESTIMATION

Guiding Principles

- Real-time performance
- Measure of confidence in estimates



- Model Required: Articulation
- Map generic visual features x to tool end-effector pose estimate y
- Tool Pose y represents orientation, end-effector opening angle

Problem with Regression Models

- Parametric Models
- Prediction is dependent on the chosen model

Gaussian Process Regression

$f(x) \sim \mathbb{GP}(m(x), k(x, x^1))$

- Inference takes place in space of functions
- Makes minimal assumptions on the underlying data distribution
- High variance in regions with sparse ground truth data

- Regression process predicts pose solely using one image
- Smoothness in surgical actions

Motion Continuity

Consider a single k^{th} element of the pose state $y(k)$,

$$\begin{bmatrix} y(k)_t \\ \dot{y}(k)_t \end{bmatrix} = \begin{bmatrix} 1 & \delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y(k)_{t-1} \\ \dot{y}(k)_{t-1} \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\delta t^2 \\ \delta t \end{bmatrix} \ddot{y}(k)_{t-1} \tag{7}$$

Acceleration is modeled as zero mean Gaussian white noise $\ddot{y}(k)_t \sim \mathcal{N}(0, \sigma_a^2) \forall t$.

The observation model is the Gaussian process regression framework which can be represented by

$$z_t = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} y(k)_t \\ \dot{y}(k)_t \end{bmatrix} + v_t \quad (8)$$

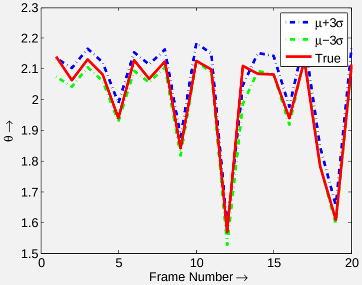


Figure 5: Gaussian Process regression with 3 sigma bounds plotted with true value of tool opening angle

Ideal Features

Unique, Invariant, Computationally Efficient

Histograms of Oriented Gradients (HOG) and Local Binary Patterns (LBP)

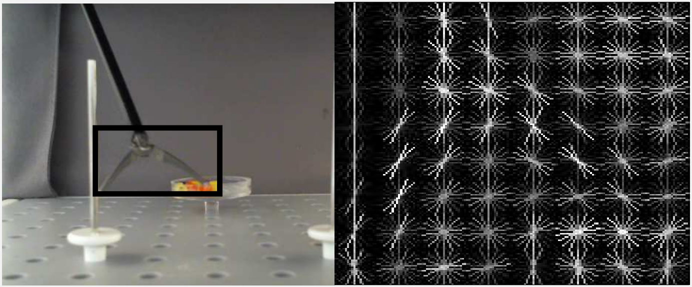


Figure 6: An example image with tool and corresponding HOG feature

EXPERIMENTAL DATA

- Both moving and fixed part of a tool are tracked at 50 fps
- Endoscopic camera : 640×480 pixels at 15 fps
- Entire dataset has 4346 different tool poses
- Sensing Noise: Motion blur, partial occlusions, lighting variation

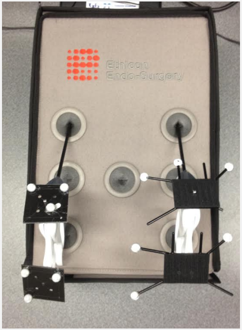
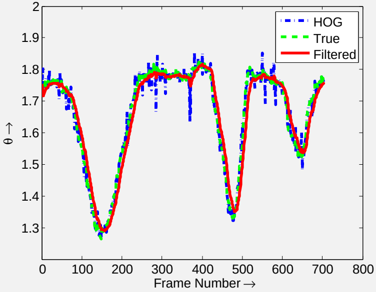


Figure 7: Customized Box Trainer Setup Retrofitted with Optical Reflective Markers

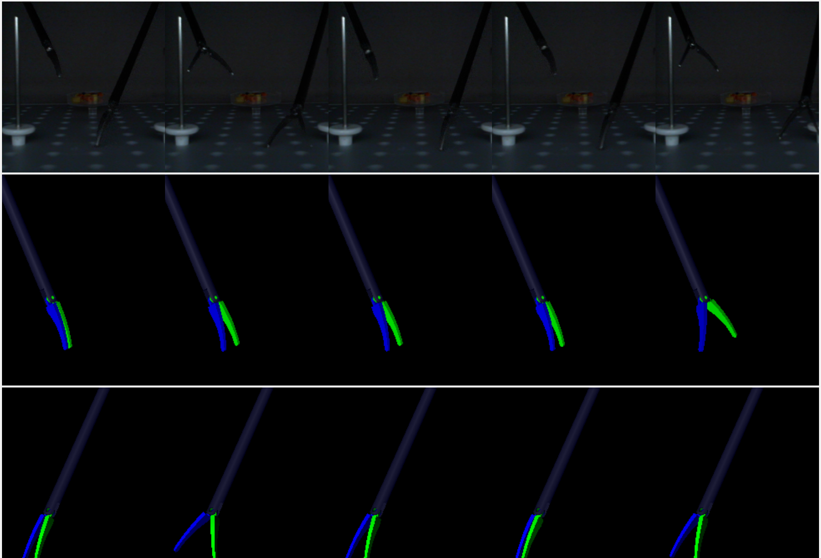
RESULTS

Features	Orientation Error	Opening Angle
HOG	2.42	2.49
LBP	1.92	2.48

Table 1: Tool pose estimate angular accuracy in degrees using different visual features



VISUAL RESULTS



CONCLUSION

- Real-time method to predict tool pose using generic visual features
- Robust variance estimates along with mean predictions
- Variance estimate is demonstrated to be useful for filtering
- Experimental results using a customized box trainer demonstrate good tool pose prediction

Journal

1. **S. Kumar**, J. Sovizi, V. Krovi, “Error Propagation on SE(3) for Surgical Tool Pose Filtering” (In Preparation)
2. P. Agarwal, **S. Kumar**, J. Ryde, J. Corso, and V. Krovi, “Estimating Dynamics On-the-fly Using Monocular Video For Vision-Based Robotics”, IEEE/ASME Transactions on Mechatronics, 2013.

Conference

1. **S. Kumar**, J. Sovizi, M. S. Narayanan, V. Krovi, “Surgical Tool Pose Estimation from Monocular Endoscopic Videos”, IEEE International Conference on Robotics and Automation (ICRA), 2015
2. P. Agarwal, **S. Kumar**, J. Ryde, J. Corso, and V. Krovi, “Estimating Human Dynamics On-the-fly Using Monocular Video for Pose Estimation”, Robotics: Science and Systems VIII, 2013

Questions?