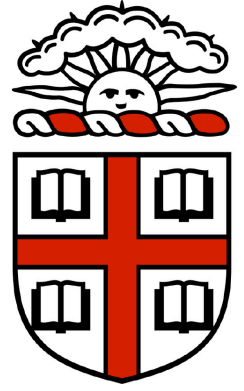


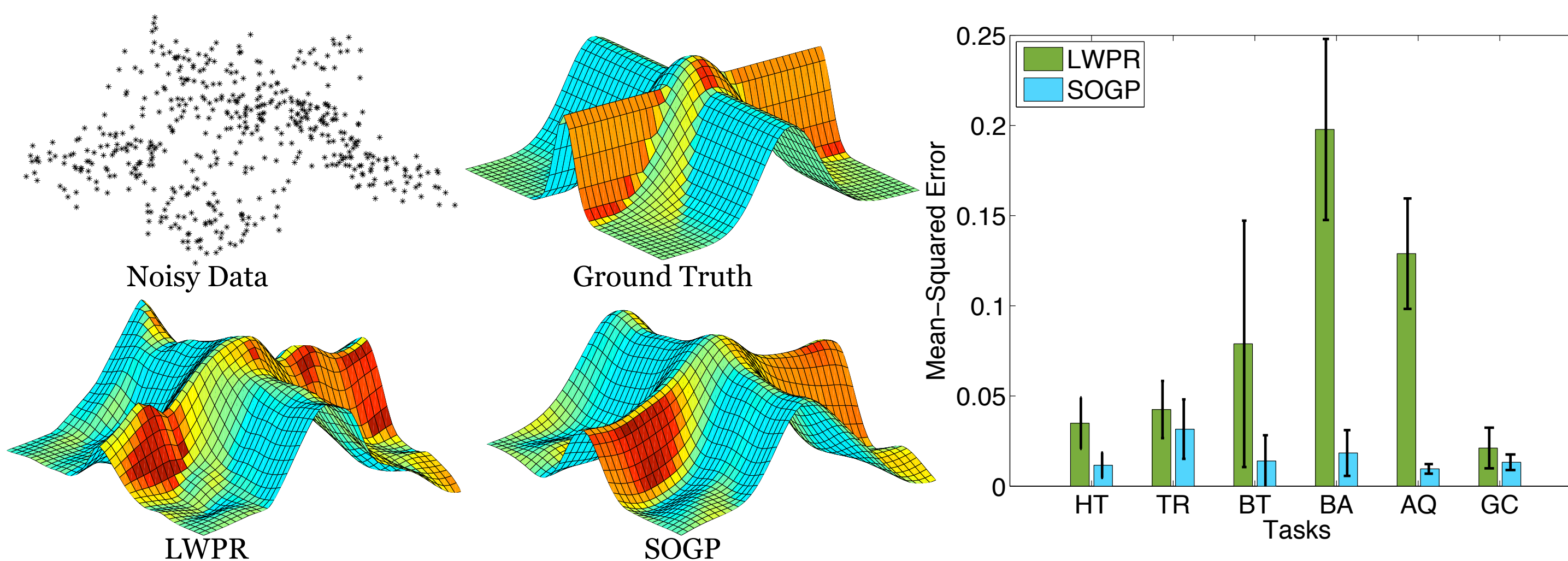
# (Machine) Learning Robot Control Policies



Daniel H Grollman, Odest Chadwicke Jenkins  
Robotics, Learning and Autonomy at Brown (RLAB)

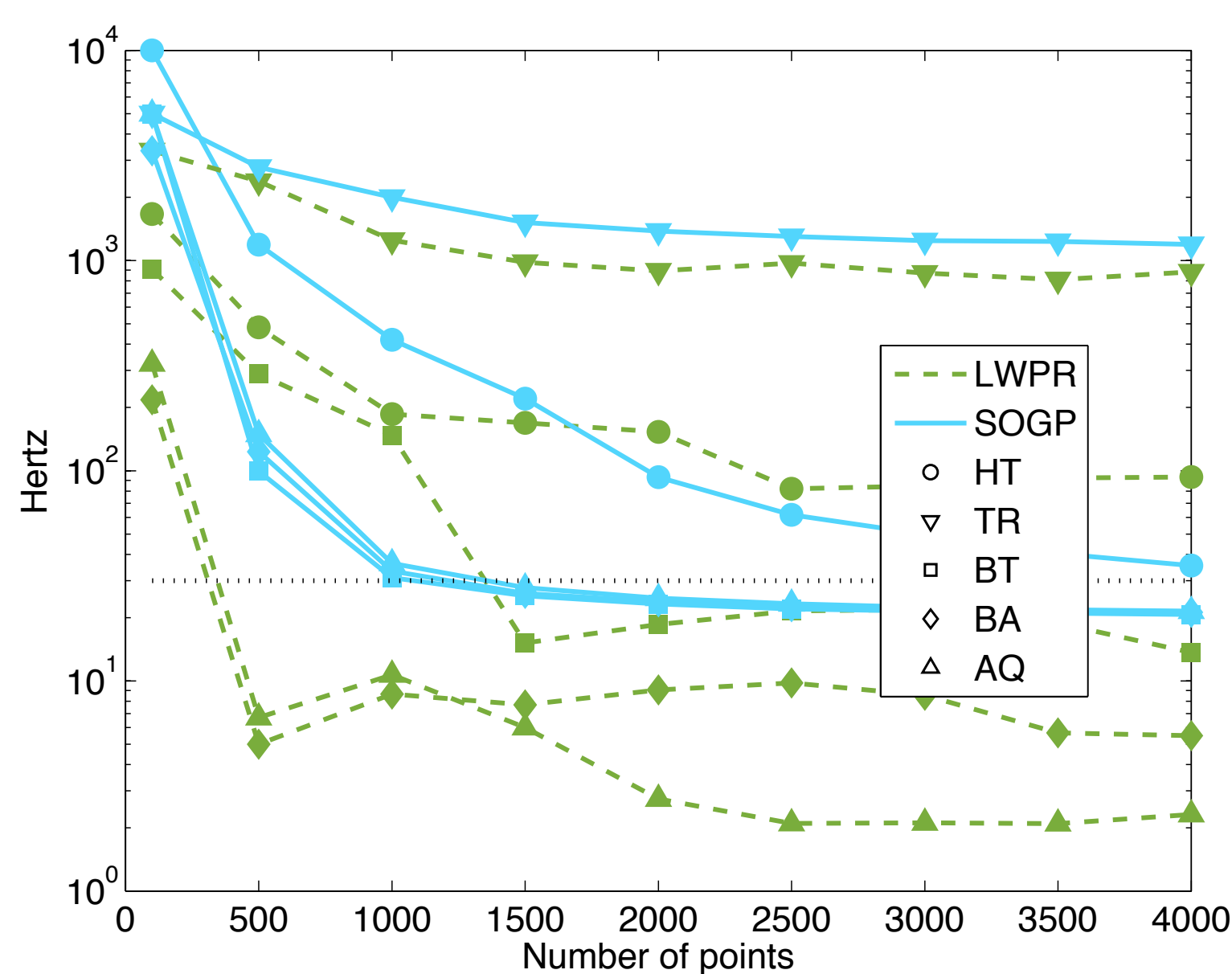
## Robot Policy Transfer from Demonstration:

Currently, most robots are given control policies by skilled users who explicitly program them to perform desired tasks. This paradigm restricts users without these skills to using robots with limited flexibility, that cannot go beyond what they are pre-programmed to do. As robots become more long-lived, the ability to learn new tasks, or modify old ones, from non-programmers may become more important. We look to machine learning to enable this policy transfer by allowing non-programmers to demonstrate tasks to a robot, which can then learn to perform the task.



## Comparing Learning Algorithms:

Because our system is not tied to any particular learning algorithm, it can be used to perform direct comparisons. We have so far compared two incremental regression techniques, Locally Weighted Projection Regression (LWPR) and Sparse Online Gaussian Processes (SOGP). Above, toy data is shown at left and on the right is the mean squared error when learning to perform a variety of soccer-skill policies. Below we examine how learning speed, in datapoints per second, relates to total data observed.



## SOGP:

Global function approximator that maintains a distribution over functions in terms of a set of basis functions. The size of the basis set can be limited to achieve sparsity. Functions are added and removed incrementally based on KL divergence.

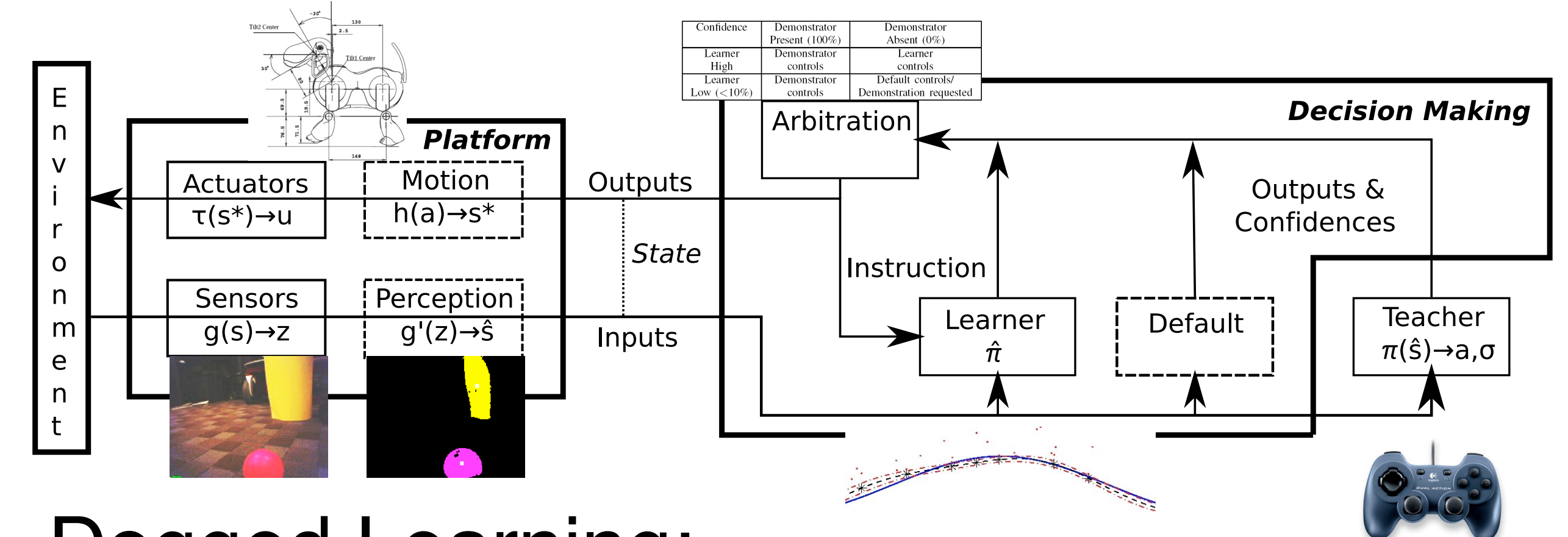
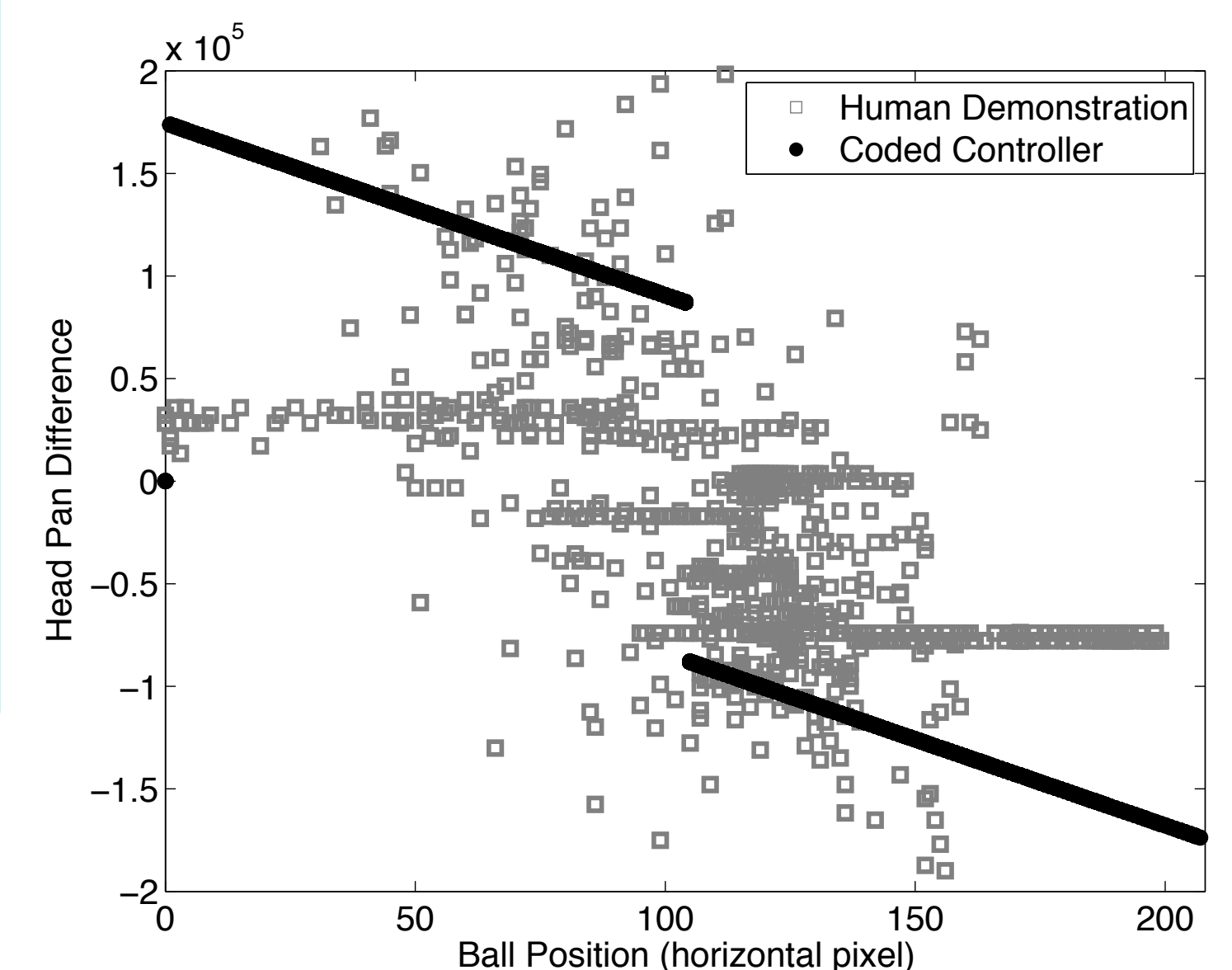
$$\text{Kernel distance (Radial Basis Function): } k(\mathbf{x}, \mathbf{x}') = \exp\left(\frac{e^{-\|\mathbf{x} - \mathbf{x}'\|^2}}{2d\sigma^2}\right)$$

For a basis set  $BV = \{\mathbf{x}_i\}, i = 1 : P$  of size  $P$   
Inverted Gram Matrix:  $\mathbf{Q} = \mathbf{K}^{-1}, k_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$

New point  $\mathbf{x}', k^* = k(\mathbf{x}', \mathbf{x}'), k_i = k(\mathbf{x}_i, \mathbf{x}')$   
Residual projection error:  $\gamma = k^* - \mathbf{k}^T \mathbf{Q} \mathbf{k}$

## Learning from Humans:

Human-generated data is much noisier and inconsistent than that from a handcoded controller. Here we show data for the ball-tracking task, where the robot's head must move to keep an orange ball in the center of its field of view. We show only one axis of movement, the horizontal axis. An autonomous controller was successfully learned from the human data using SOGP.



## Dogged Learning:

A learning from demonstration framework designed to be agnostic to robotic platform, demonstration interface, arbitrator and learning algorithm. Abstractly, a platform (robot) extracts information from the world via its sensors and uses perception to generate an estimate of the world's state,  $\hat{s}$ . A demonstrator uses a latent control policy,  $\pi$ , to generate a desired action,  $a$ . The learning algorithm's job is to discover an approximation of the demonstrator's control policy,  $\hat{\pi}(\hat{s}) \rightarrow a$ .  $\sigma$  are confidence values used to arbitrate which controller has access to the physical robot.

## LWPR:

Performs local function approximation by learning a set of receptive fields. Each field maintains a collection of univariate regressions chosen with partial least squares regression, resulting in a sparse, incremental algorithm. Prediction output from each field is weighted and combined to produce system predictions. New fields are added based on a threshold activation value.

$K$  Receptive Fields with centers,  $\mathbf{c}_k$  and Gaussian areas of influence,  $\mathbf{D}_k$

For a data point  $\mathbf{x}$ :

$$\text{RF activation: } w_k = \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{c}_k)^T \mathbf{D}_k (\mathbf{x} - \mathbf{c}_k)\right)$$

$$\text{System Prediction: } \hat{y} = \frac{\sum_{k=1}^K w_k * y_k}{\sum_{k=1}^K w_k}$$

Acknowledgements:

NSF: IIS-0534858, Brown Salomon Grant, Brown #, Ugur Cetintemel