Tuner - finding the best parameters for your algorithm

Torsten Möller
Graphics, Usability, and Visualization Lab
Simon Fraser University

Quiz - question 1

- "Each match must agree within 15 degrees orientation, $\sqrt{2}$ change in scale, and 0.2 times maximum model size in terms of location. If fewer than 3 points remain after discarding outliers, then the match is rejected."
- Lowe, 1999, Object recognition from local scale-invariant features (SIFT)

Quiz - question 2

- "In our numerical experiments, we generally choose the parameters as follows: $\lambda_1=\lambda_2=0$, $\nu=0$, $h=1$ (the step space), $\Delta t=0.1$ (the time step). We only use the approximations $H_{2,\varepsilon}$ and $\delta_{2,\varepsilon}$ of the Heaviside and Dirac delta functions ($\varepsilon=h=1$), in order to automatically detect interior contours, and to insure the computation of a global minimizer. Only the length parameter μ, which has a scaling role, is not the same in all experiments."
- Chan+Vese, 2001, Active contours without edges

Overview

- Problem setting
- Sampling multi-d spaces
- Exploring multi-d spaces
- Trading off multiple objectives
- Results

Problem Setting

Image segmentation

- many thresholds, e.g.
  - max vessel diameter
  - max vessel curvature, etc.
- variational / energy minimisation

$$E(\phi, I) = \alpha_1 E_1(\phi, I) + \alpha_2 E_2(\phi, I) + ... + \alpha_k E_k(\phi, I)$$

- abstracted (black-box) scheme:

$$\mathcal{R}^n \longrightarrow \boxed{f} \longrightarrow \mathcal{R}^m$$

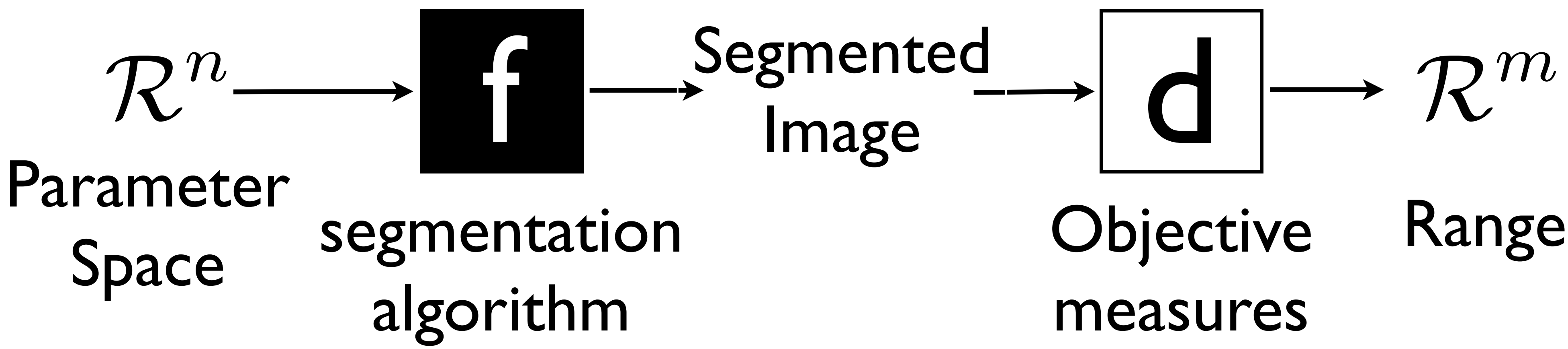
Parameter Space

segmentation algorithm

Range

- we can tell **Inputs** from **Outputs**
- we can query this box / algorithm at every "point"

- abstracted (black-box) scheme:



- we can tell **Inputs** from **Outputs**
- we can query this box / algorithm at every "point"

Objective measure

- one-dimensional ("goodness") rating:
  d(object) = quantitative grade
- two-dimensional comparison:
  $d(O_1, O_2)$ = quantitative similarity
- objective measures can be
  - exact (reliable)
  - approximate - about right, but not 100% precise
  - unknown (active learning)

in our case ...

- used (several) one-dimensional measures
  - dice
  - error
  - precision / recall
- need the user in the loop to find the best trade-off

We would like to ...

- sample the input space exhaustively
- explore / understand the space of all segmentations
- find **the** best segmentation (and its parameters)

Three challenges

- How to sample the input space exhaustively?
- How to explore / understand the space of all segmentations?
- How to find **the** best segmentation (and it's parameters)?

Sampling

Sampling

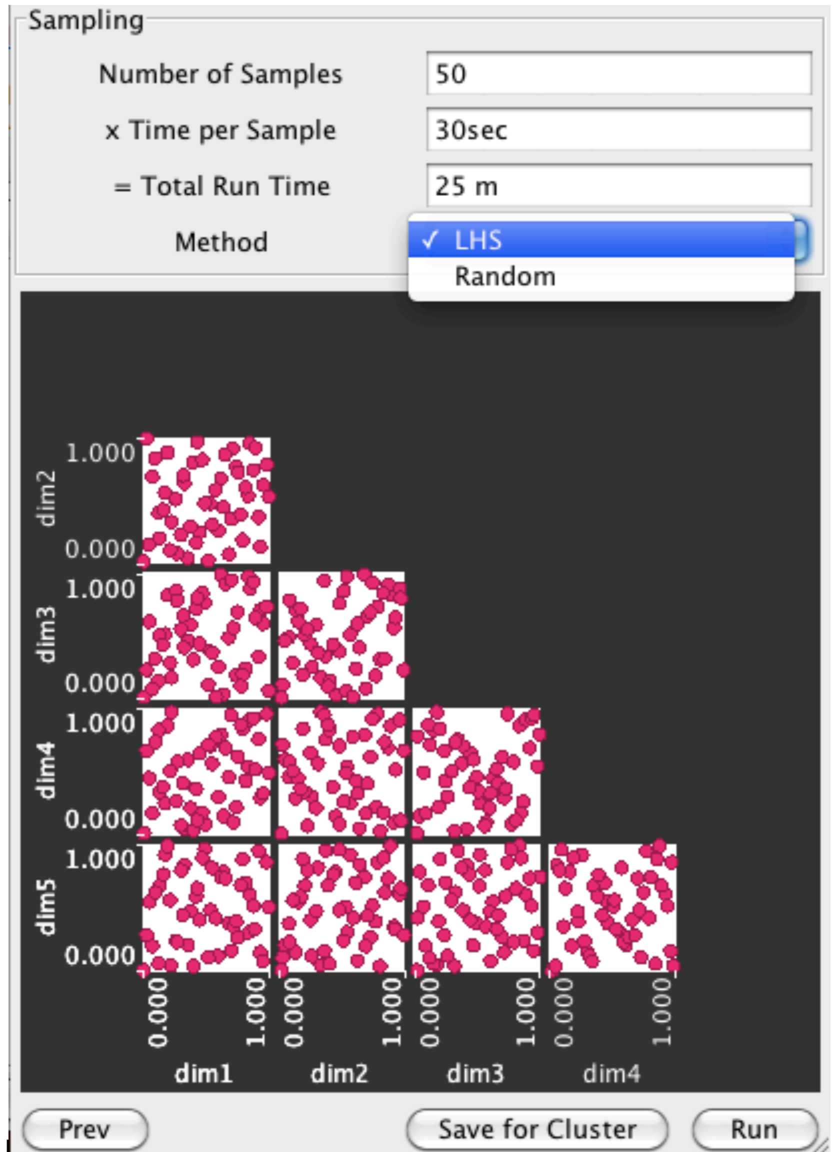
- trading-off time and accuracy
- time - would like to get an answer in less than a day (samples are expensive!)
- accuracy - would like to have as dense a sampling as possible
- typically reconstruct / infer values at non-sampled values from sampled neighbors

Common strategy

- user gives a sampling budget
- split into
  - uniform sampling at start
  - adaptively refine according to some refinement criteria

Our implementation

- 2 sampling strategies
  - Random
  - Latin Hypercube
- live preview
- estimate running time



Reconstruction

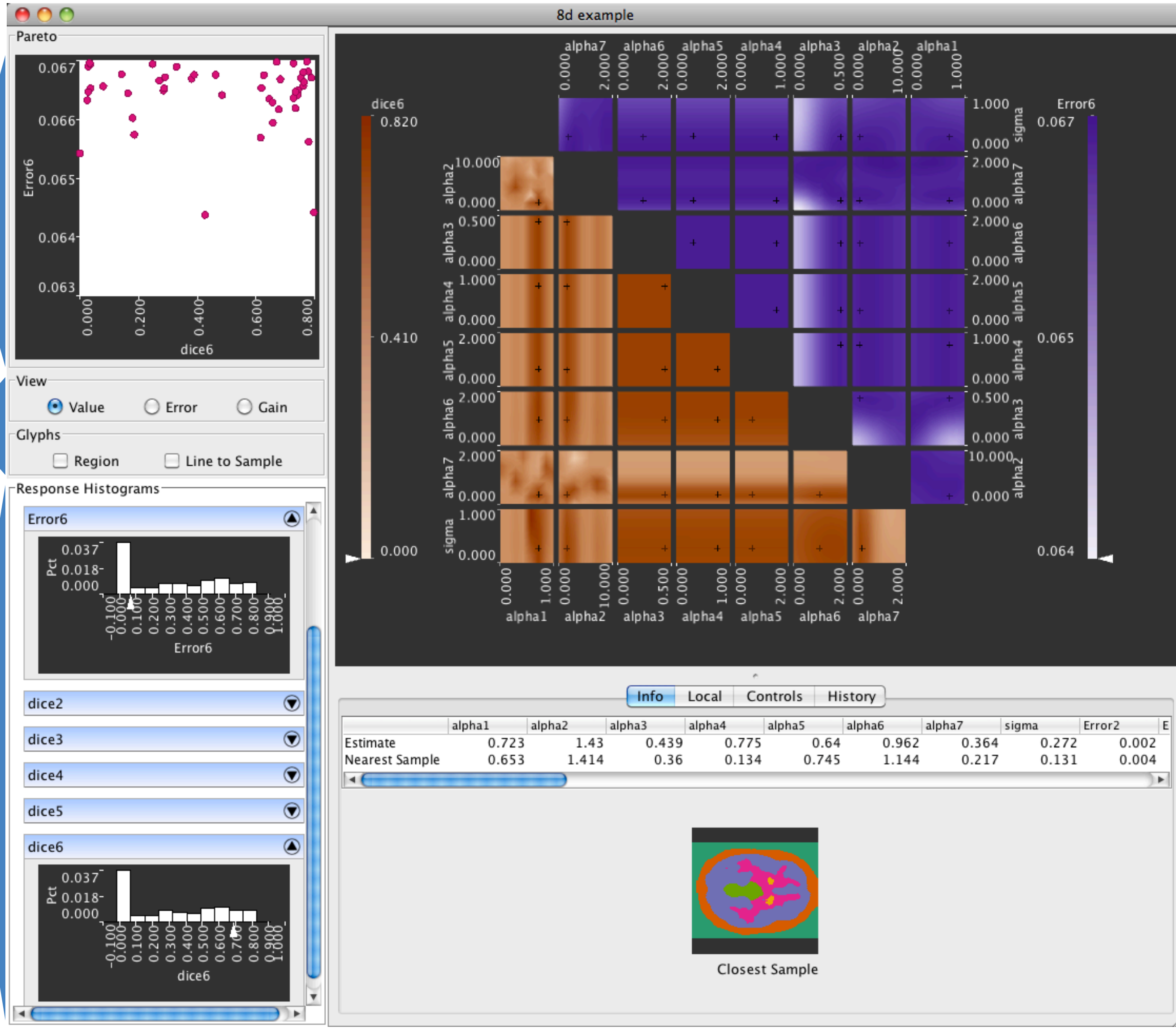
- Gaussian process model, i.e.
  - essentially a convolution with adapted kernel parameters
- refine where the user tells us to

Understanding the parameter space

Pareto Panel

View Controls

Histograms

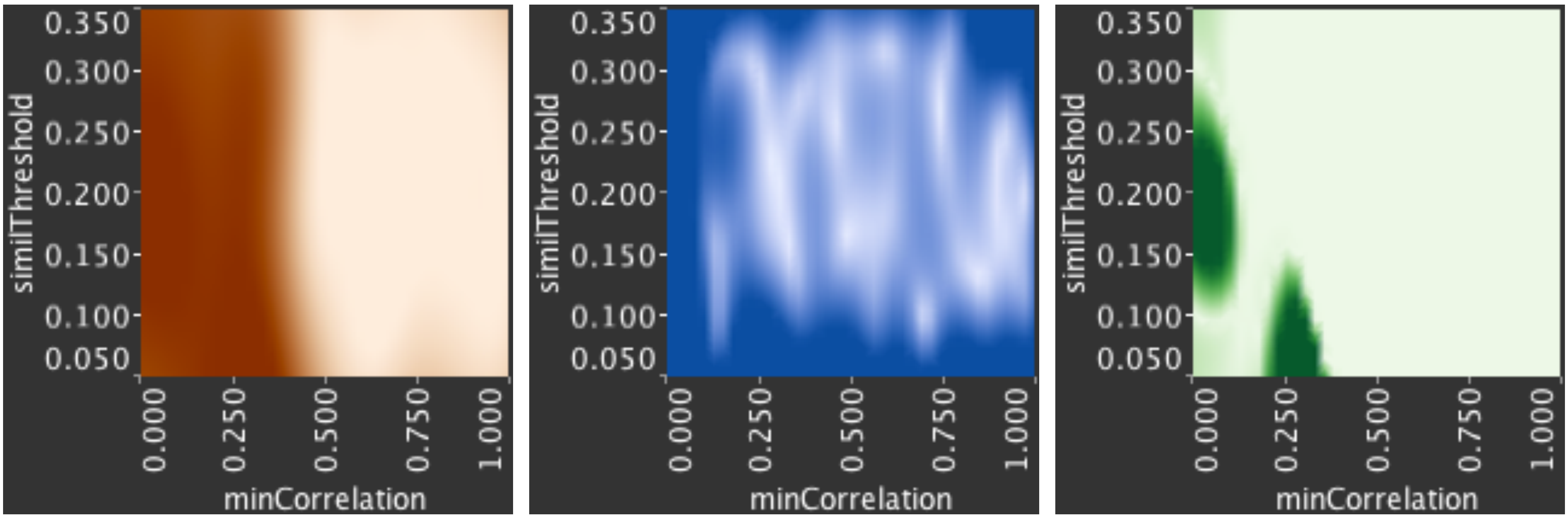


Response View

Plot Controls

Response Views

Predicted Value | Uncertainty | Expected Gain

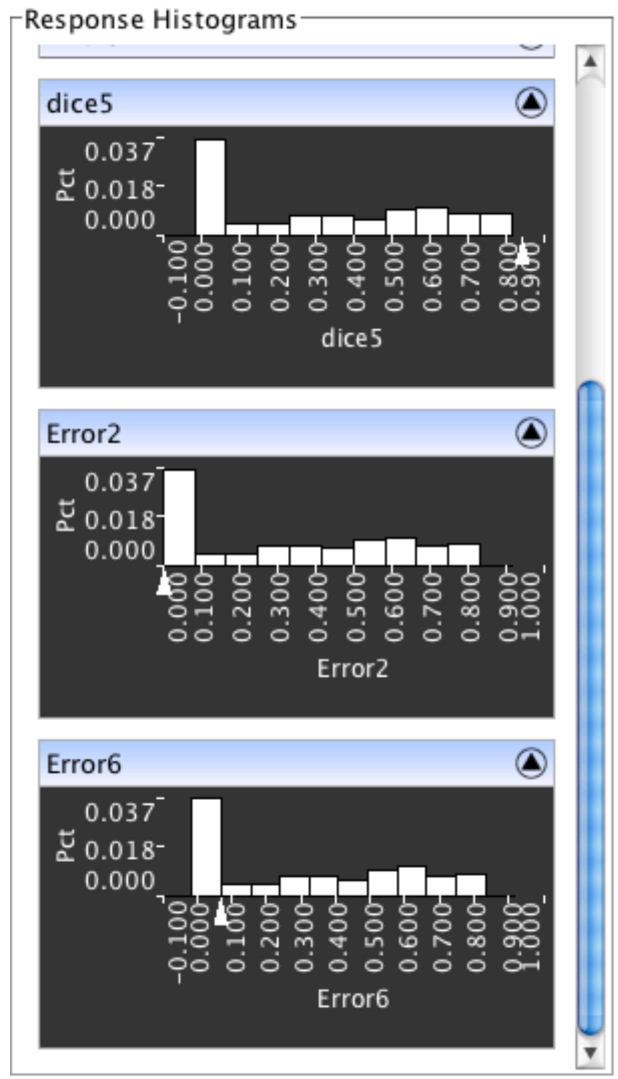


Improve global estimate

Likely optimum

Histograms

- One histogram per output
- Glyph shows where point of focus lies w.r.t. optimum
- People didn't really seem to use these

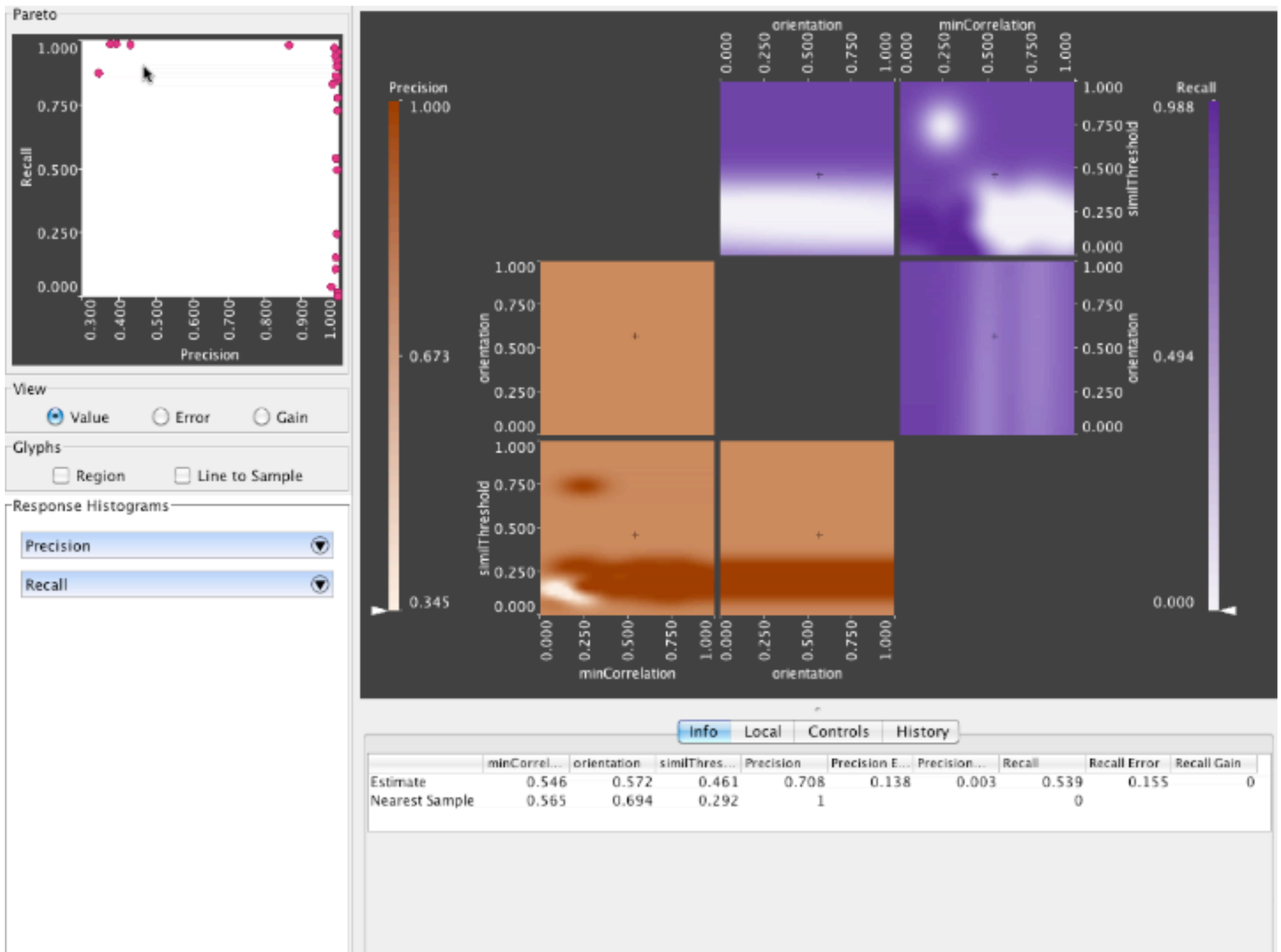


The best segmentation

Optimization: subtasks

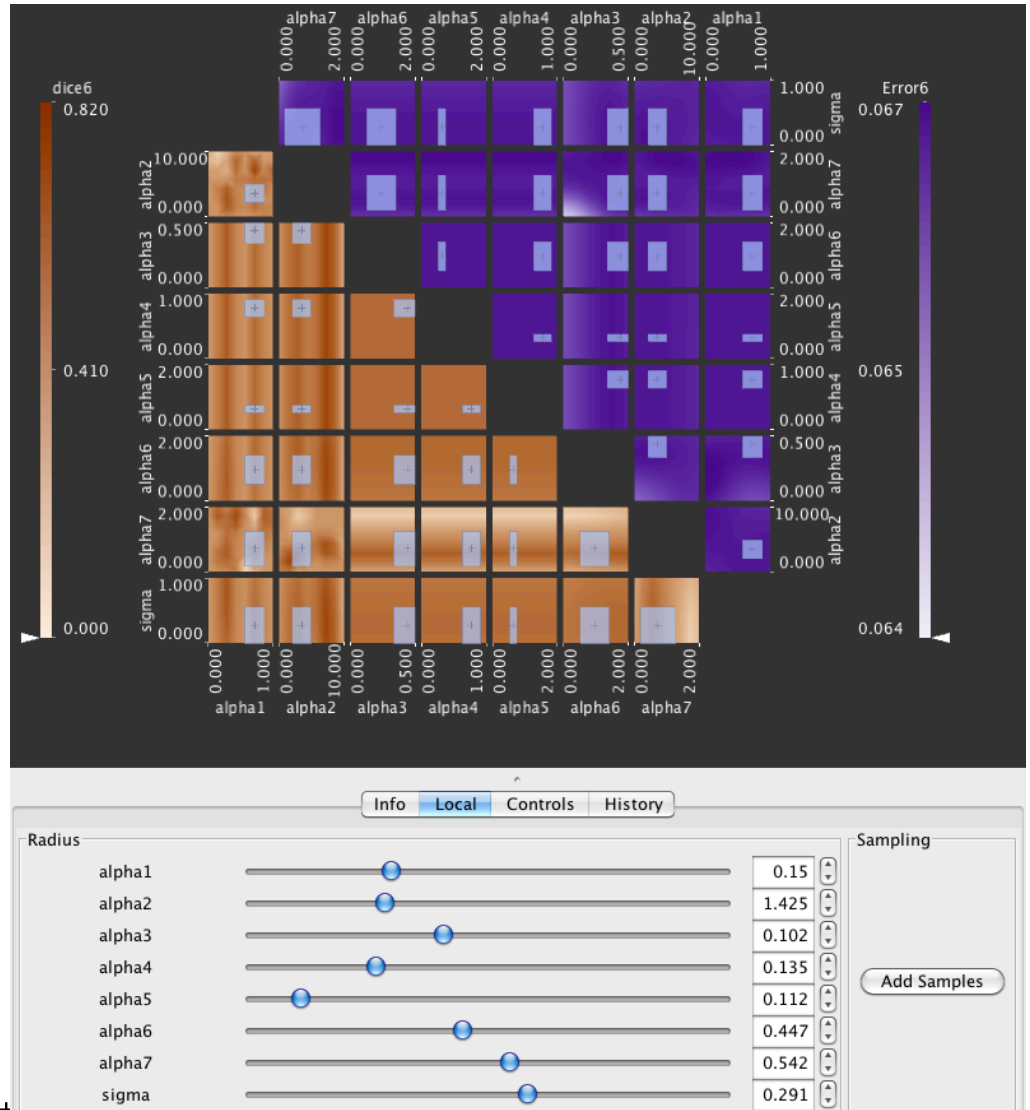
- facilitate understanding of trade-offs
- applying constraints on (output) parameters
- refine sampling at potential optima

Pareto Panel



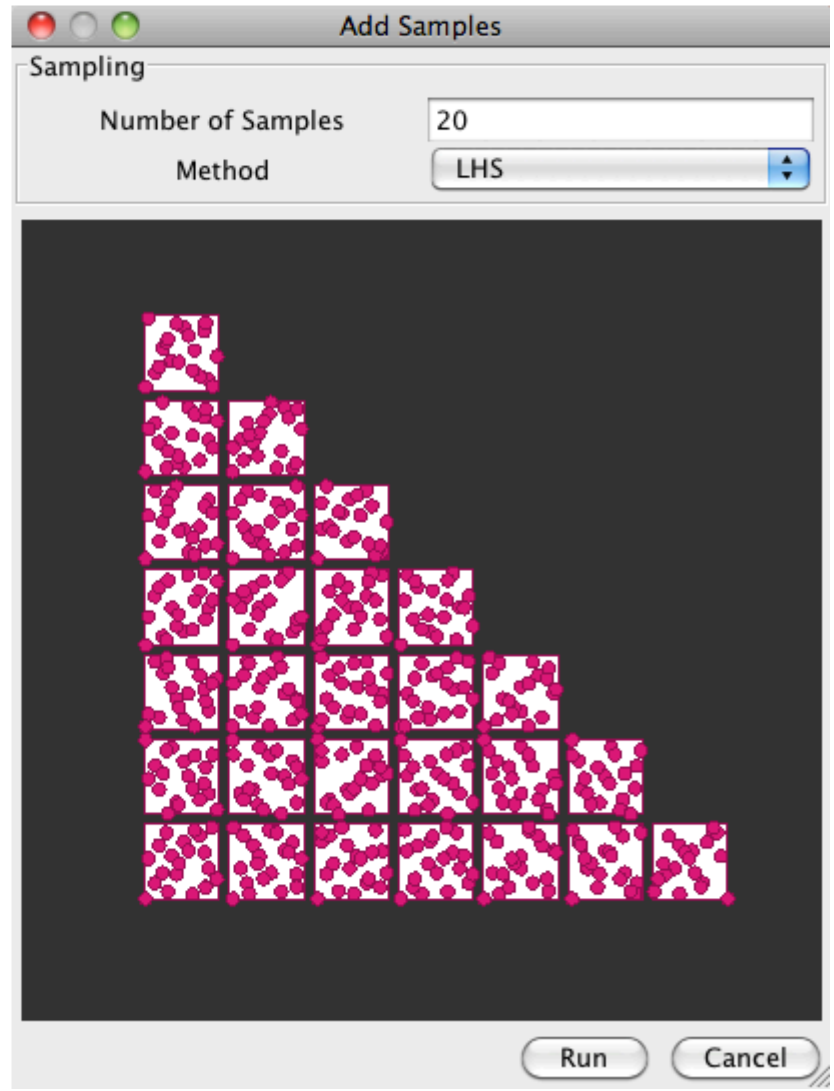
Refinement

- Mark the region of interest



Refinement

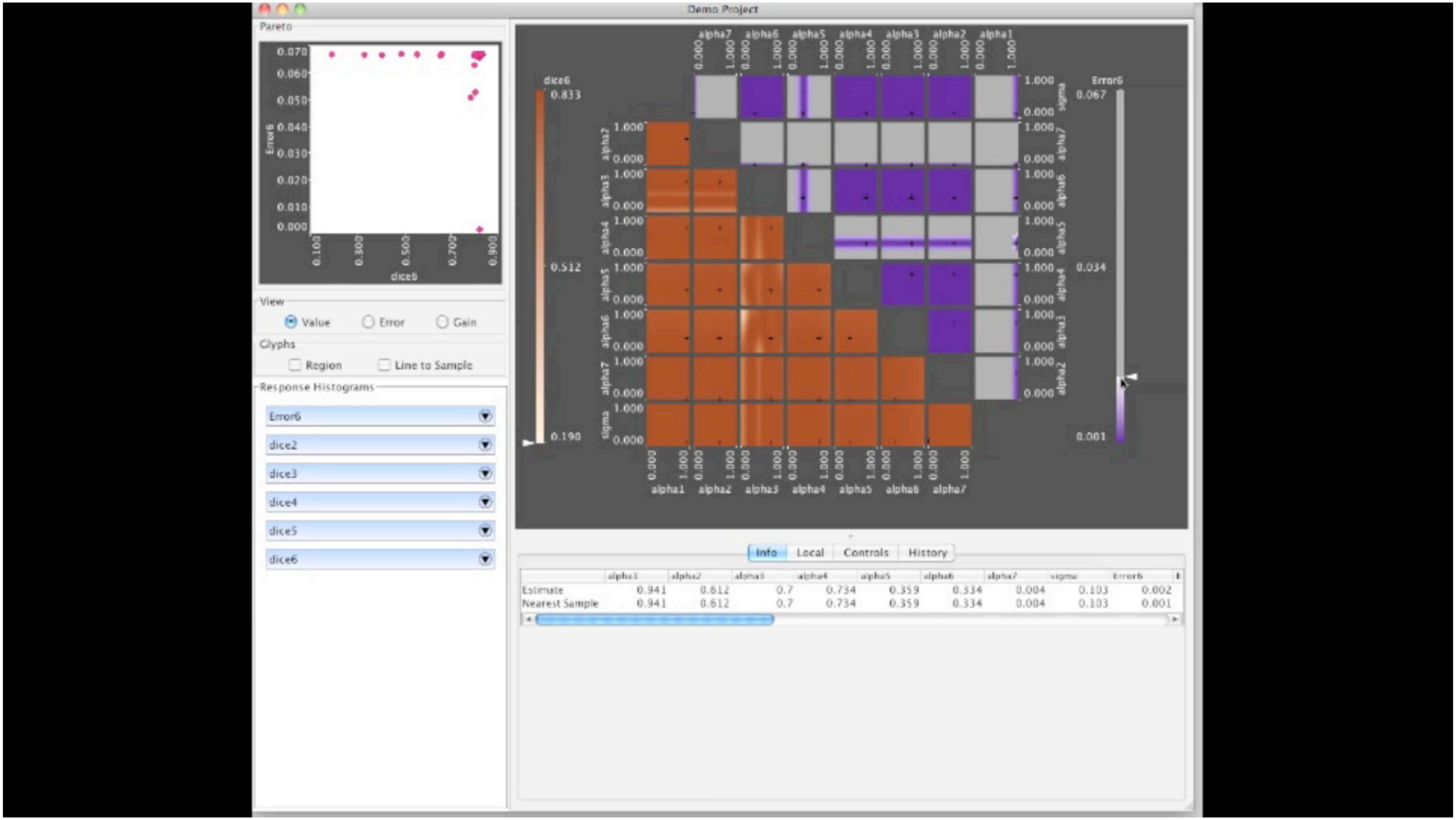
- identical interface to initial sampling
- clicking "Run" button runs samples through black box code
- GP model is automatically rebuilt



Results

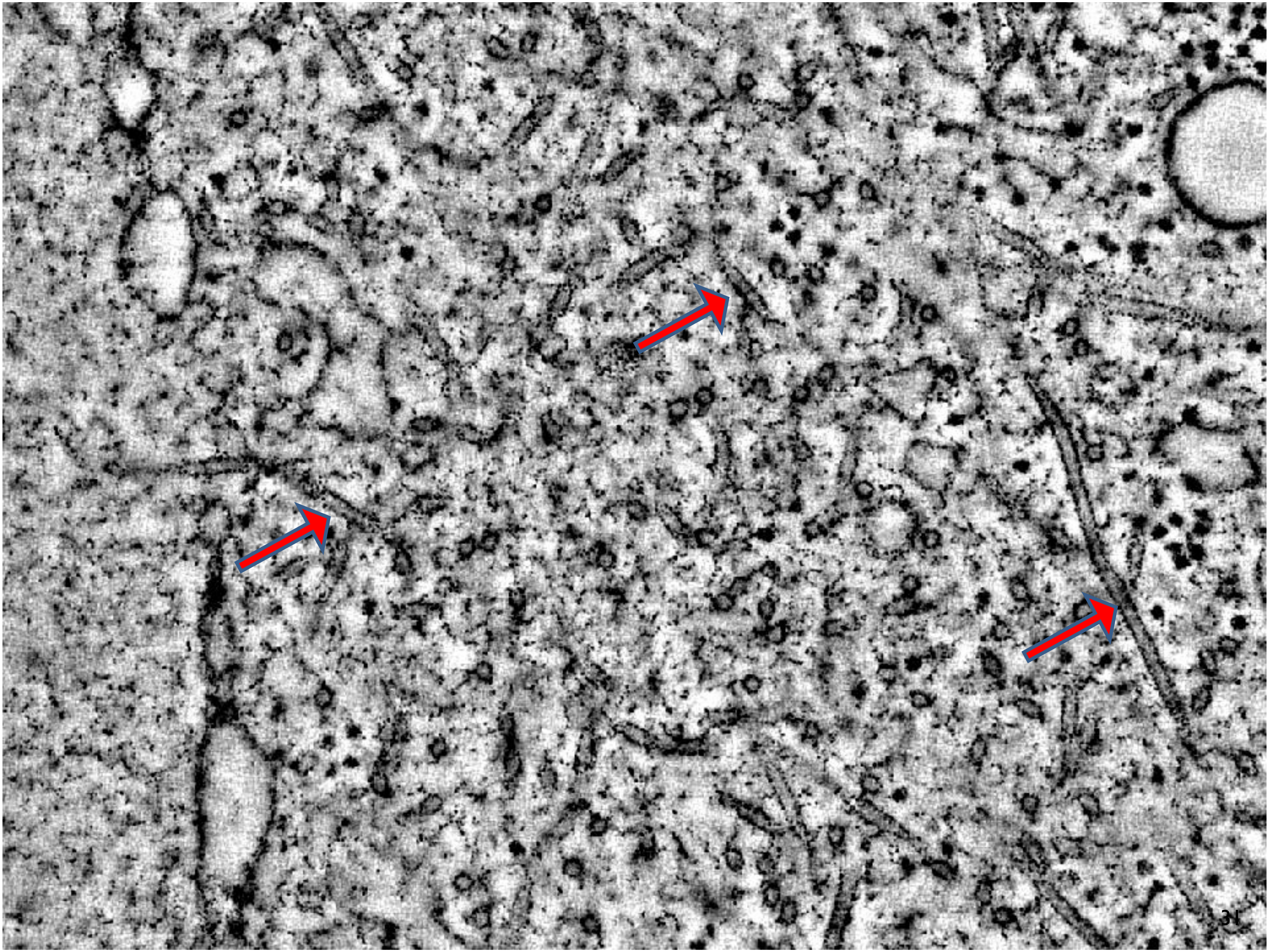
Two scenarios

- 8d dPET image segmentation
- 3d microtubule tracing algorithm

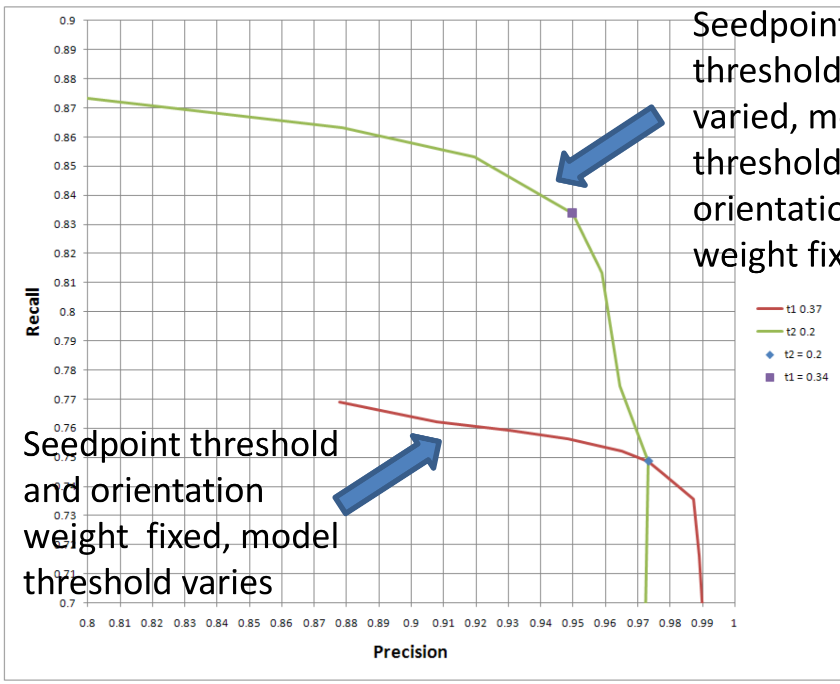

Demo Project
Pareto
Error6
dice6
View
Value
Error
Gain
Glyphs
Region
Line to Sample
Response Histograms
Error6
dice2
dice3
dice4
dice5
dice6
Info
Local
Controls
History
Estimate 0.941 0.612 0.7 0.734 0.359 0.334 0.004 0.103 0.002
Nearest Sample 0.941 0.612 0.7 0.734 0.359 0.334 0.004 0.103 0.001

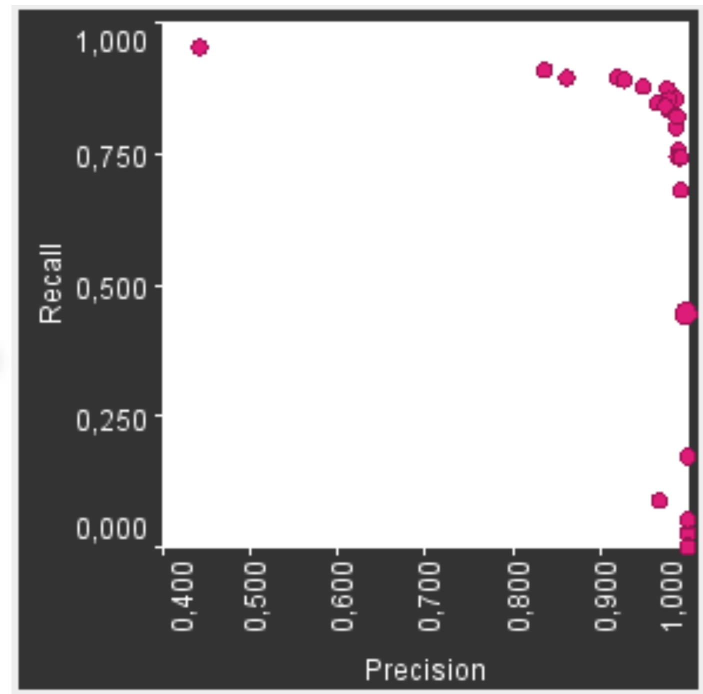
Electron Microscopy

- Samples: eggs of C.Elegans during mitosis
- Preparation: Samples are dry frozen, stained, embedded in plastic and sectioned into ~300nm thick slices
- Image acquisition: Volume is reconstructed from a series of projections in different tilt angles taken with a transmission electron microscope (TEM)



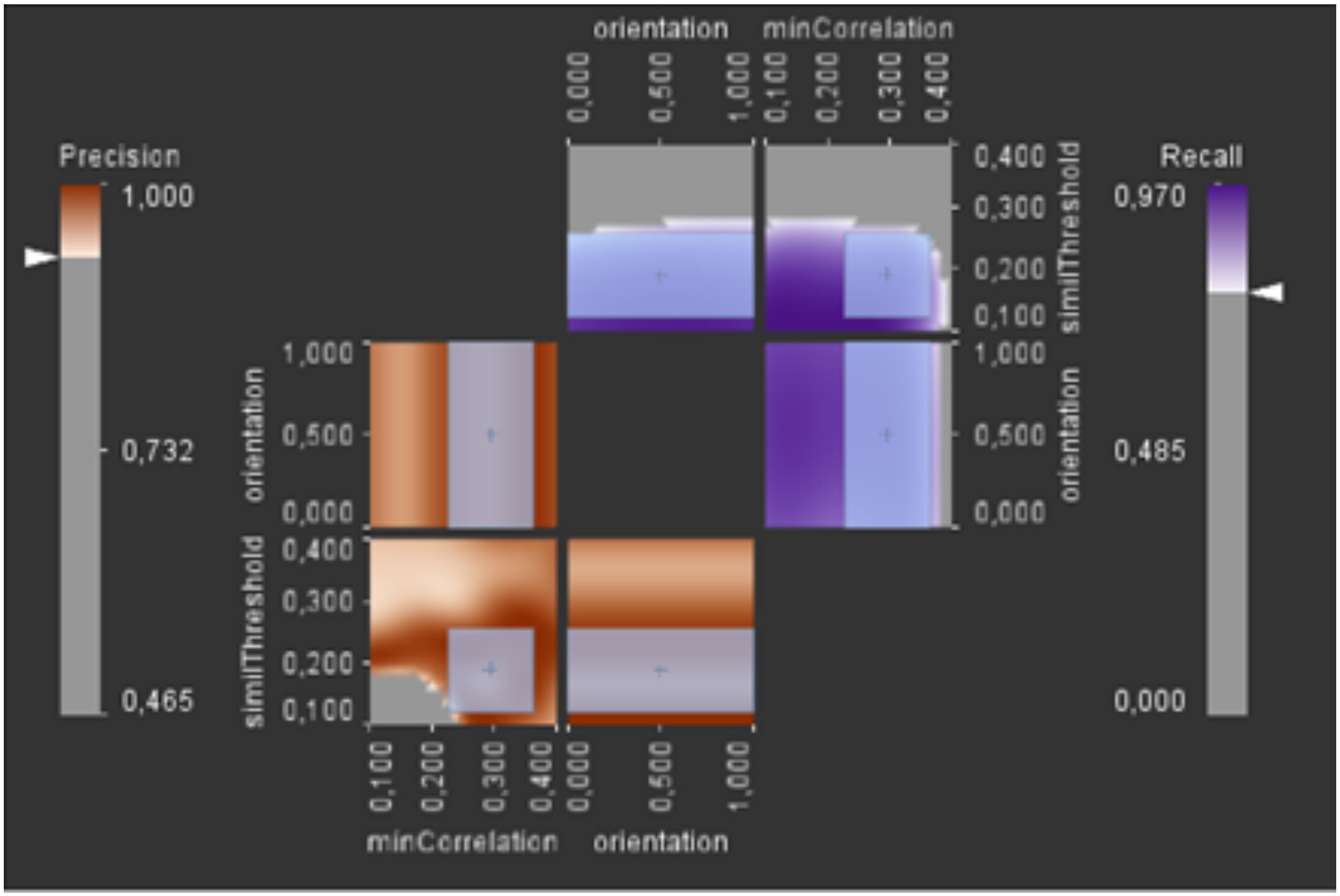
Microtubule tracing





“No more making stupid plots”

Microtubule tracing



Summary

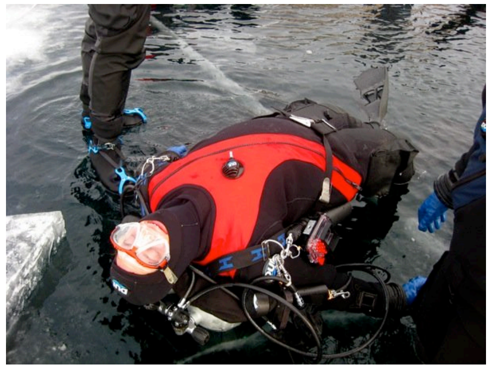
- principled way of exploring multi-d parameter space
- understanding trade-off of multiple objectives
- “This reduced the work of days to a couple of hours.”
- lots of things to improve!
- not everything is an optimization

BC INNOVATION COUNCIL

NSERC CRSNG

Canada Foundation for Innovation
Fondation canadienne pour l'innovation

Acknowledgments



Thomas-Torsney Weir
SFU

Steven Bergner
SFU

Ahmed Saad
SFU

Britta Weber
Zuse I Berlin

Hans-Christian Hege
Zuse I Berlin

Jean-Marc Verbavatz
MPI-CBG Dresden

biovis

IEEE Symposium on
Biological Data Visualization
23-24 Oct 2011 / Providence, RI, USA @ IEEE VisWeek 2011

www.biovis.net

early registration deadline - **16 September**

20 papers + **27** abstracts

Visualization Challenge (focus on eQTL)
  Deadline Sep 7th

Keynote by Lynda Chin

Invited Session by Arthur Olson, Cydney Nielsen, Willy Supatto

Tutorial by Larry Hunter, Kun Huang

IEEE   IEEE Computer Society   IEEE vgtc

Questions?

torsten@sfu.ca

http://gruvi.cs.sfu.ca