End-to-End In Situ Data Processing and Analytics

Extreme-scale Distribution-based Data Analysis

Han-Wei Shen
Professor
Department of Computer Science and Engineering
The Ohio State University

THE OHIO STATE UNIVERSITY

In Situ Processing and Visualization

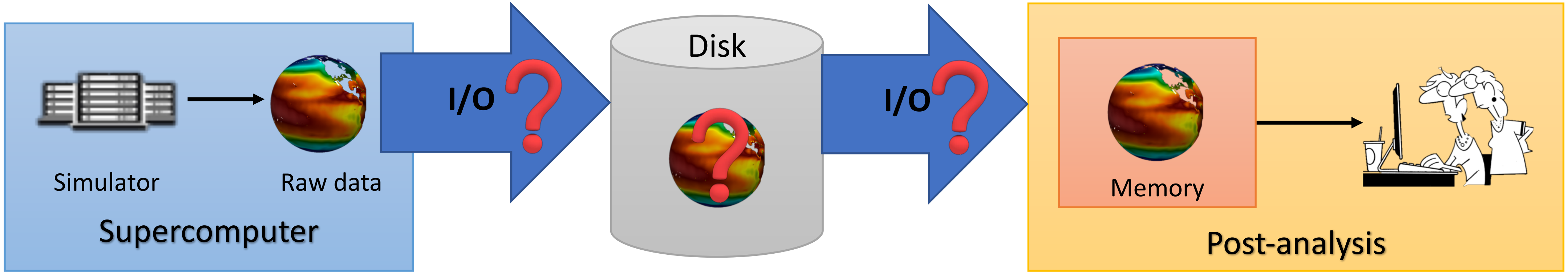
- ExaFLOPs supercomputers is becoming a reality (exa = 1,000,000,000,000,000,000)
  - Number of cores per processor will increase
  - Memory per core will decrease
- The speed and size of memory and I/O devices cannot keep pace with the increase of compute power
  - Cost of moving data will increase
- It will be very difficult for scientists to store and analyze even a small portion of their simulation output

In situ Visualization

Generating Visualization While the Simulation is Still Running

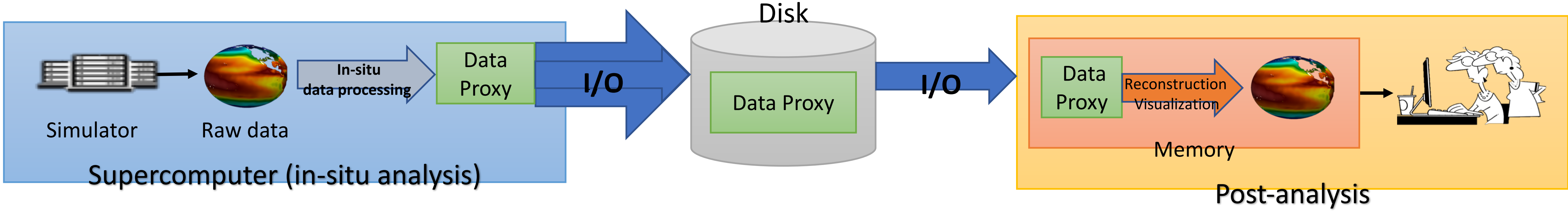
Characteristics of In Situ Visualization

- Data are transient; only available for a short time
- Mainly batch mode processing; Interactive exploration is not possible
- Need to know what is needed a priori; Salient information might not be found
- Limited parameters to explore; Sophisticated visualization is not possible



In Situ Visualization Strategies

- Generate images from preselect parameters (e.g. Catalyst, Libsim)
- Database from a large collection of images (e.g. Cinema Project)
- Visualization with explorable contents (e.g. Explorable Images)
- Feature extraction (e.g. Contour trees, flowlines)
- Data Reduction – Compact data representation or representative samples or time steps (e.g. compression, key time steps)



In Situ Visualization Software

- Application aware vs. not
- Tightly or loosely coupled
  - Shallow or deep copy
  - Space or time share
  - Data synchronization and communication
- Software control (automatic or human control)
- Proximity: Same or different machines
- Single or multi purpose (e.g. ADIOS) APIs
- Types of output (data, images, etc)

Distribution-based In Situ Analytics @ OSU

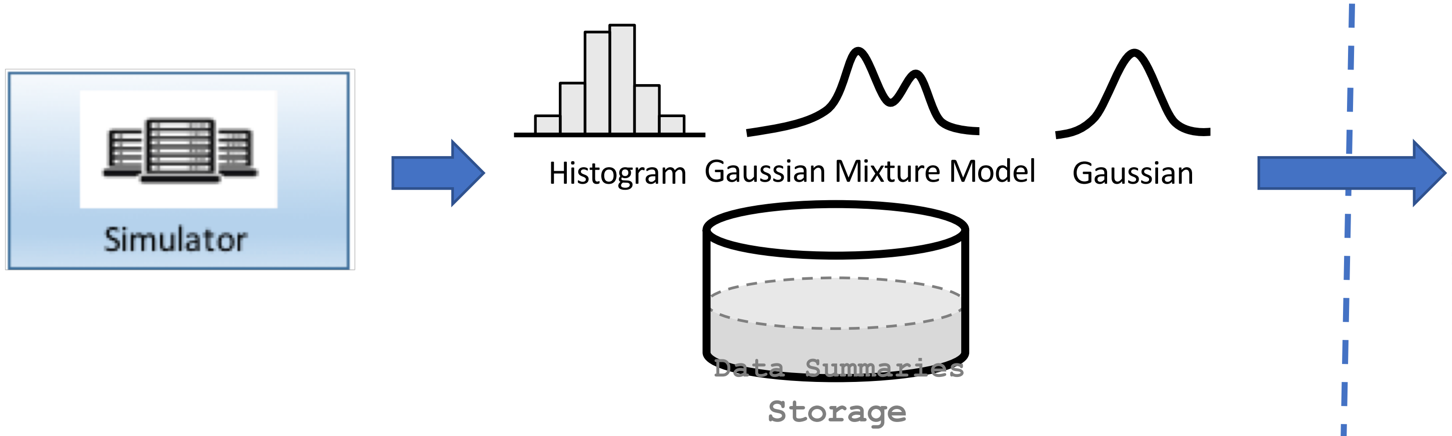
Approaches

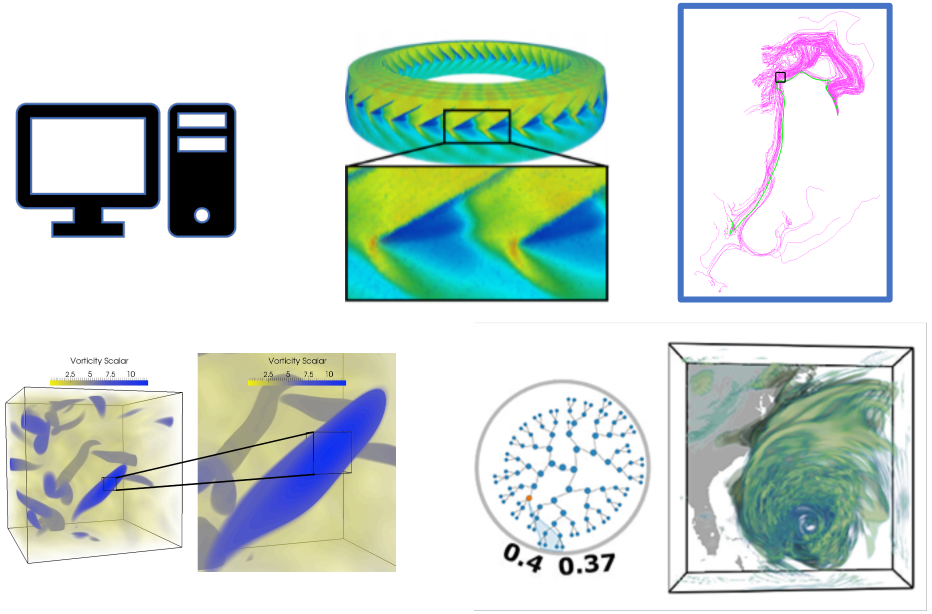
- Probability Distributions collected as in situ time
  - Block or particle based
  - Histograms, GMMs
  - Multivariate
- Distribution-based post-hoc analysis
  - Resampling based visualization
  - Direct inference based on distributions
  - Interactive data queries

Goals

- Preserve
  - Important data characteristics
  - Field values and feature locations
- Allow
  - Post-hoc analysis with standard visualization capabilities
  - Quantitative analysis of quality of uncertainty
  - Interactive data driven queries
- Predict
  - Results of simulations with novel parameter configurations

In Situ Research @OSU





In Situ Data Reduction and Transformation

Post-Hoc Analysis and Visualization

- Distribution Modeling:
  - Spatial Partition
  - Field and particle data
  - Image space (View dependent)
  - Object space
  - Multivariate
  - Time-varying
  - Ensemble data

- Visualization and Analytics:
  - Sampling
  - Scalar data visualization algorithms
  - Vector data visualization algorithms
  - Feature tracking
  - Distribution Exploration
  - Distribution Search
  - Ensemble data analysis

View Dependent Distributions Proxy

Motivations

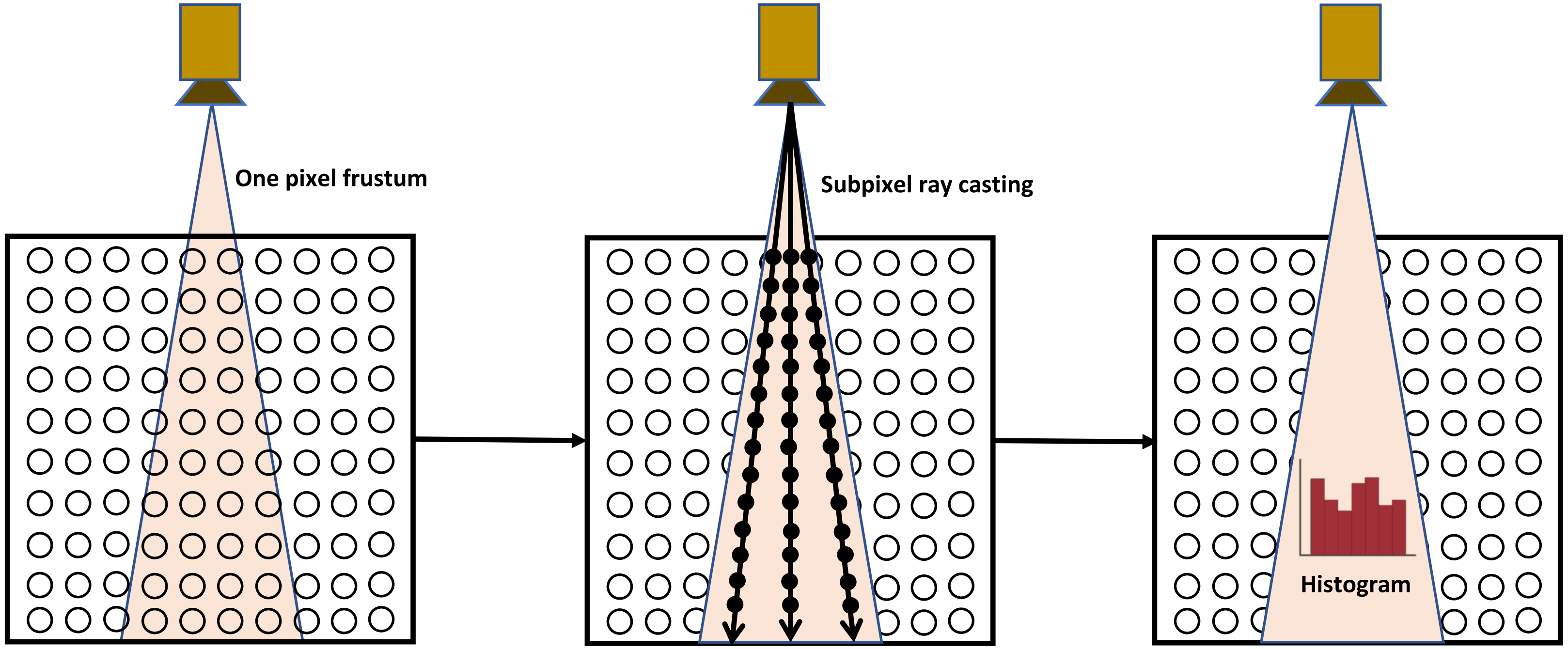
- Image space approaches have emerged as a promising method
  - The scale of data defined in image space (~ 10^6 pixels) is relatively smaller than in object space (~ $10^{9\sim15}$ voxels)
- Freely explore the occluded features
  - Existing image-based approaches have limited ability to explore the occluded features
- Inevitable data loss in the compact representation

Methods

- Collects samples during volume ray casting
- Allows change of transfer functions in post-hoc analysis
- Errors are constrained in the depth dimension
- Warping the samples to different views are possible

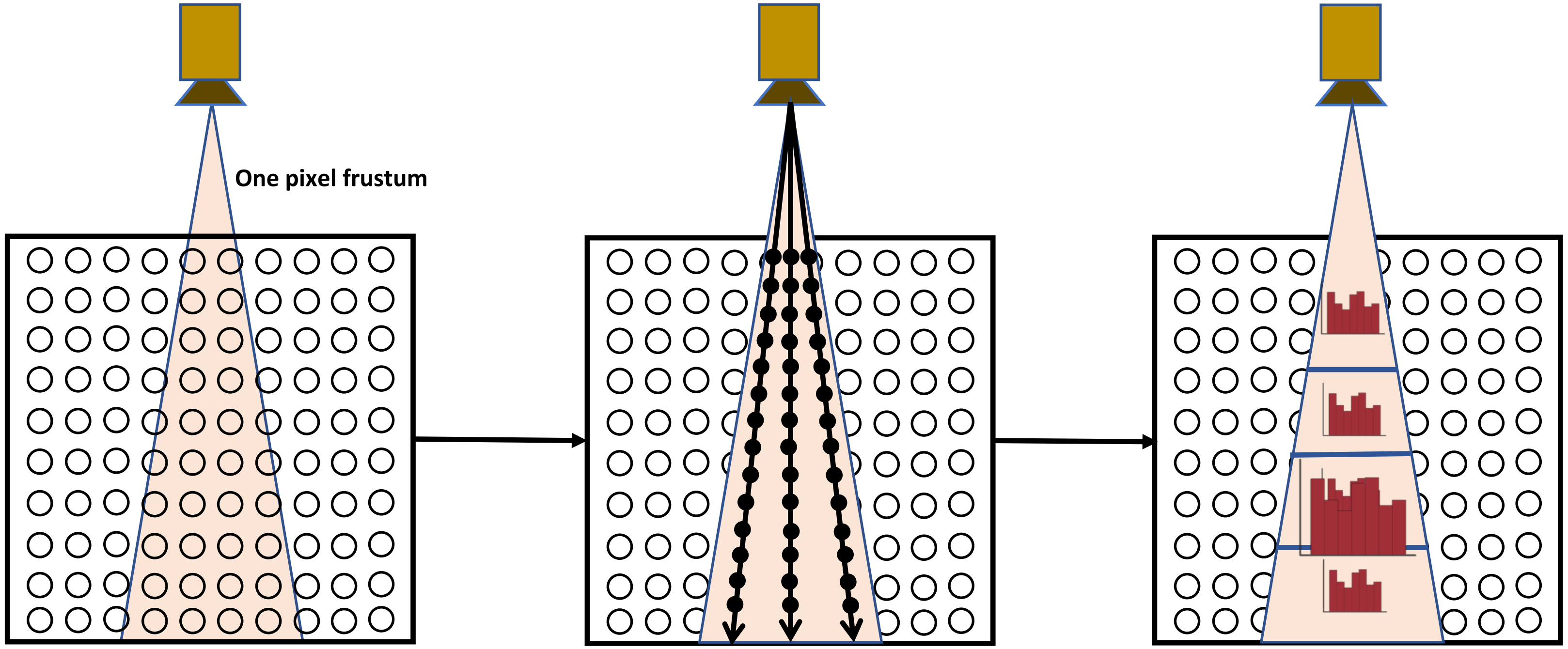
View Dependent Proxy Construction

- Image-based proxy is constructed at each selected view
- Subpixel ray casting to collect samples in the pixel frustum
- Histogram is used to statistically summarize data in the pixel frustum

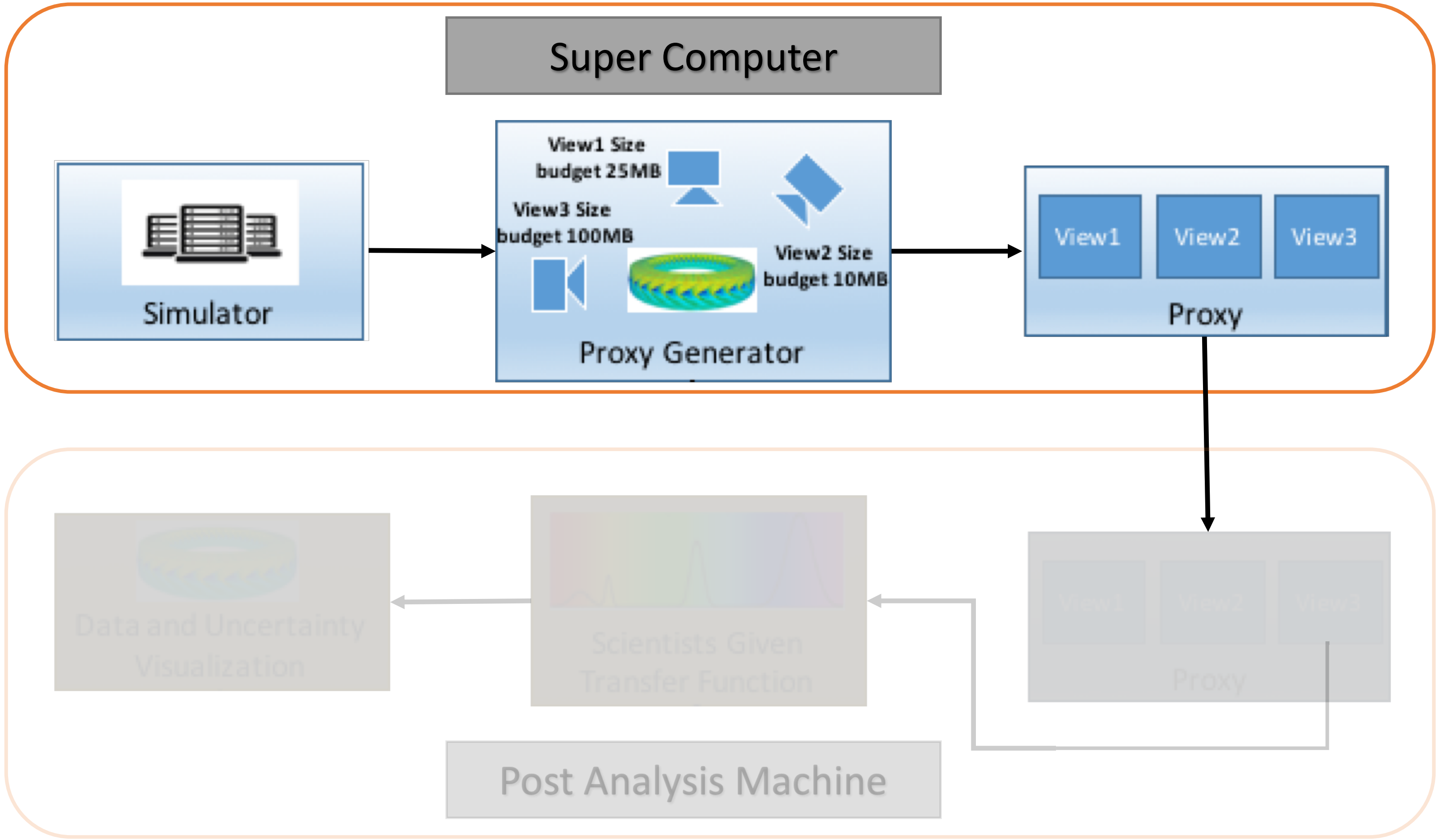


Irregular Frustum Subdivision

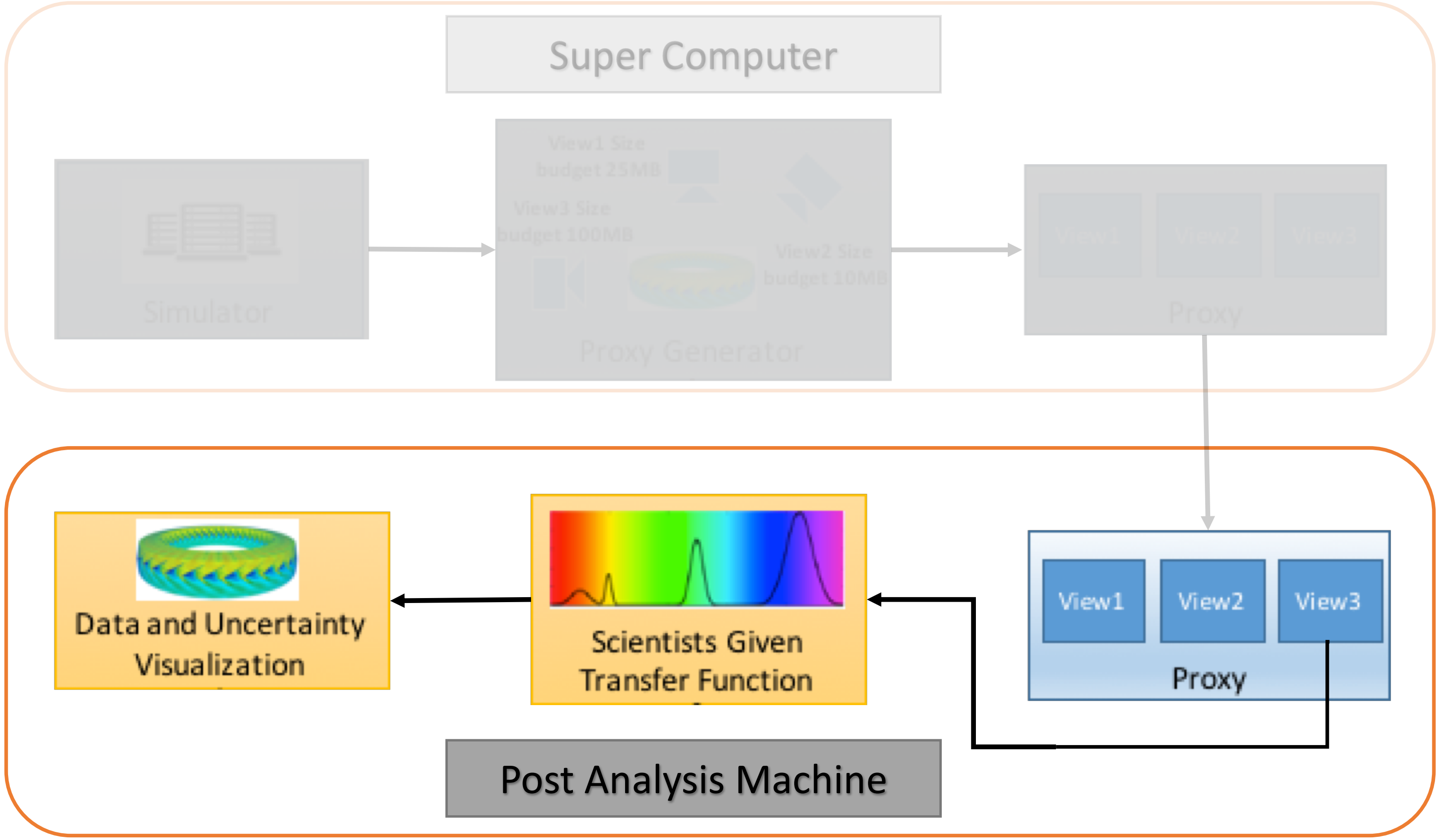
- Histogram does not keep samples’ order in the pixel frustum
  - Samples‘ order is critical to provide depth cue in rendering
- A pixel frustum is sub-divided into sub-frustums which are summarized by histograms
  - More sub-frustums: more accurate samples’ order and store more histograms



Data Visualization in Post Analysis Machine

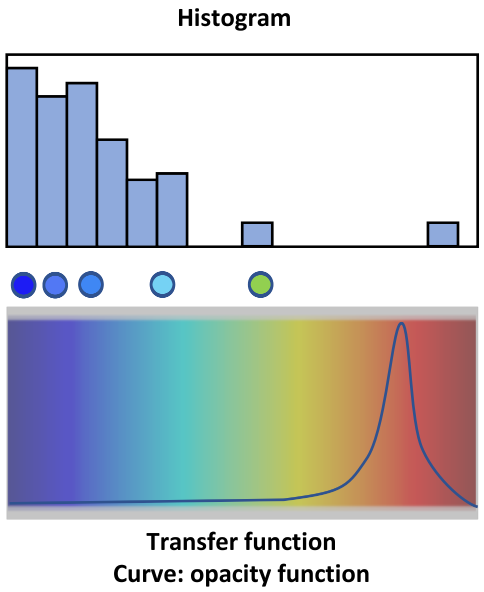


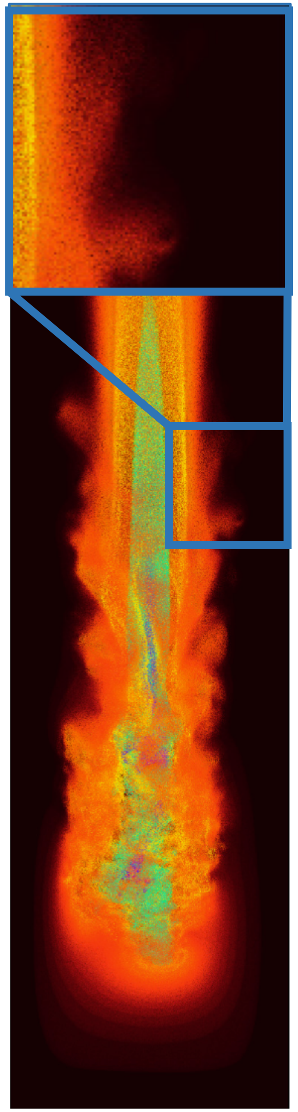
Data Visualization in Post Analysis Machine



Importance Sampling

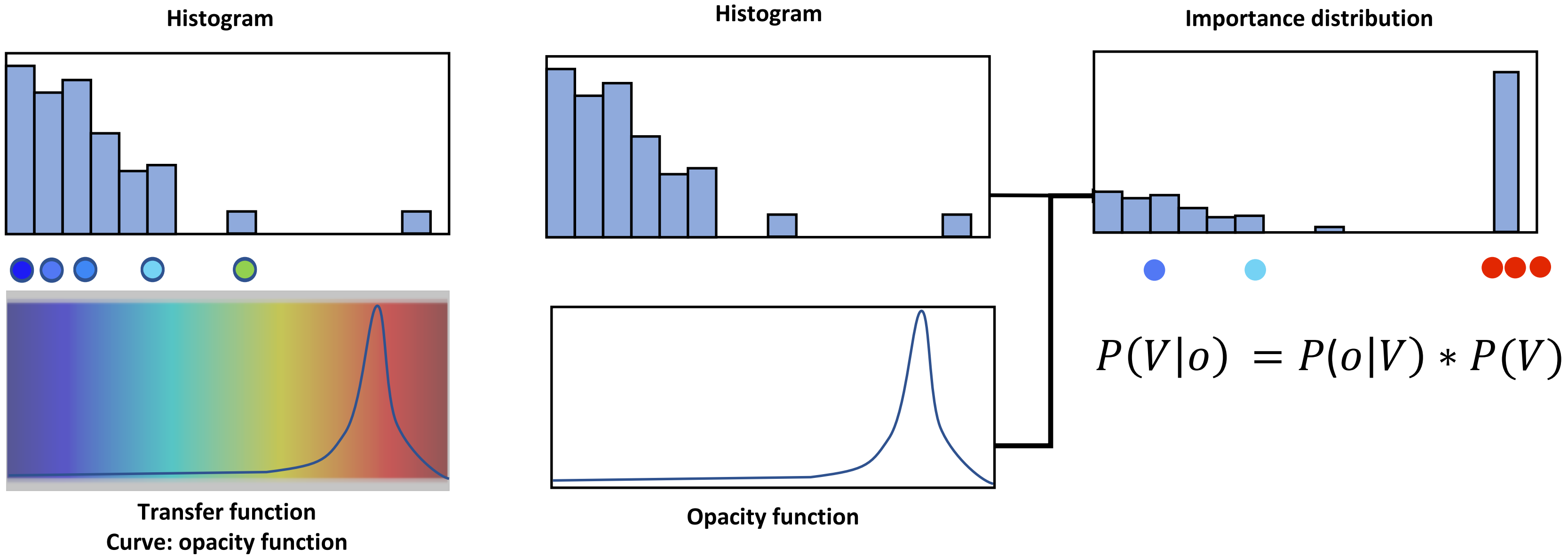
- Samples drawn from a histogram are biased towards to the value with high frequency
  - Samples with high frequency may have low opacity
  - Interesting features consist of samples with high opacity
- Importance sampling
  - Combine histogram and opacity function

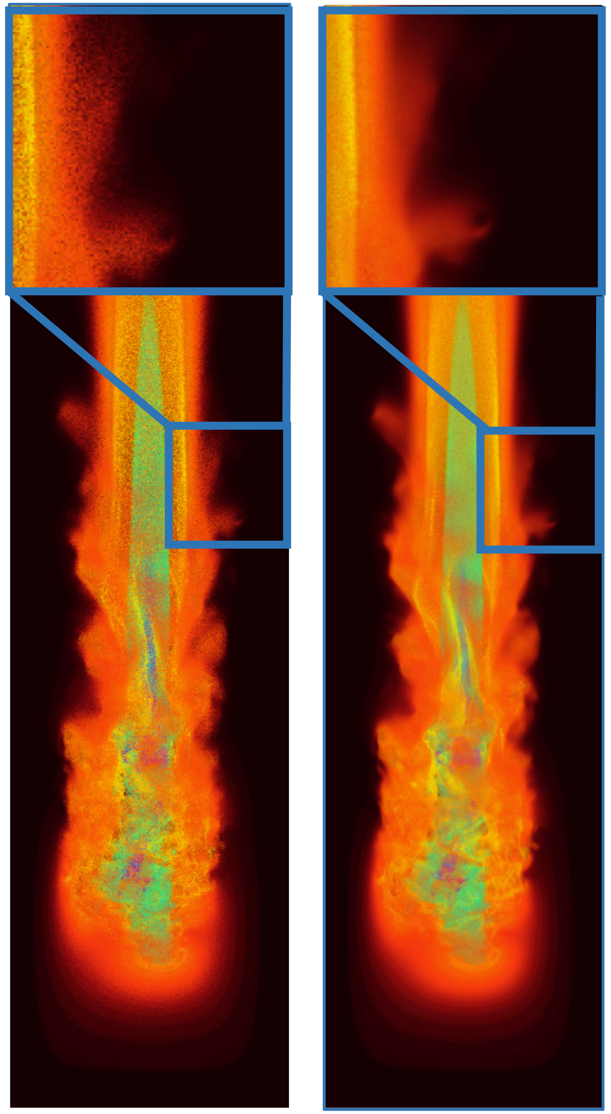




Importance Sampling

- Samples drawn from a histogram are biased towards to the value with high frequency
  - Samples with high frequency may have low opacity
  - Interesting features consist of samples with high opacity
- Importance sampling
  - Combine histogram and opacity function





Quality and Storage

- Turbine dataset
- 50 time steps
- 6 views proxy
- Budget: 50MB

(per view and time step)

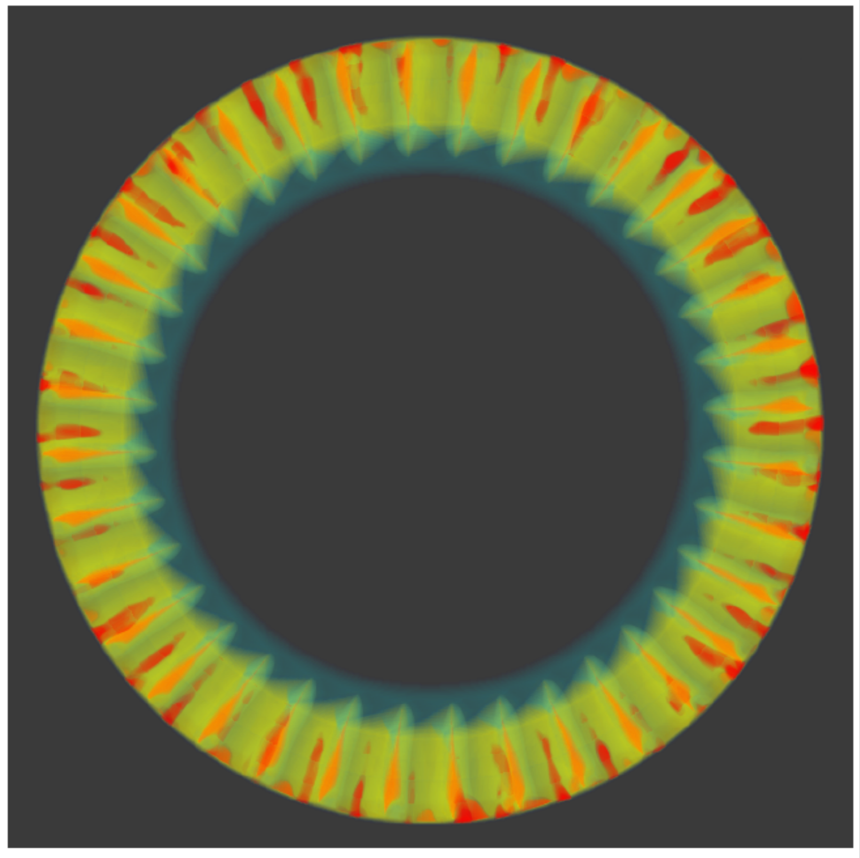


Image from Raw Data
271GB

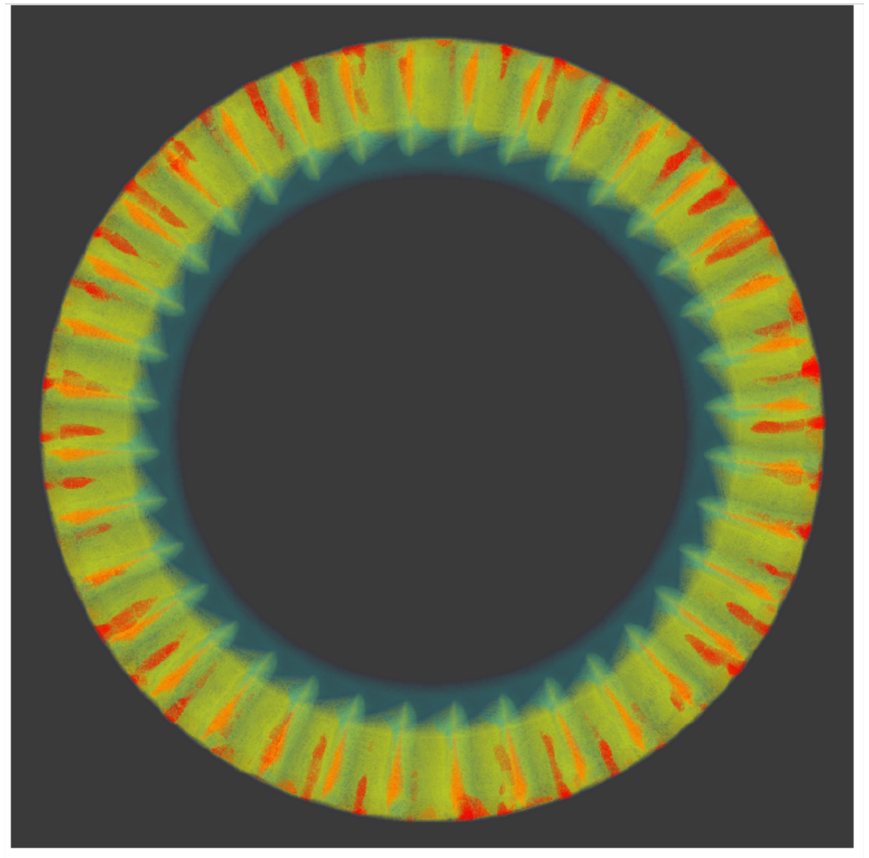
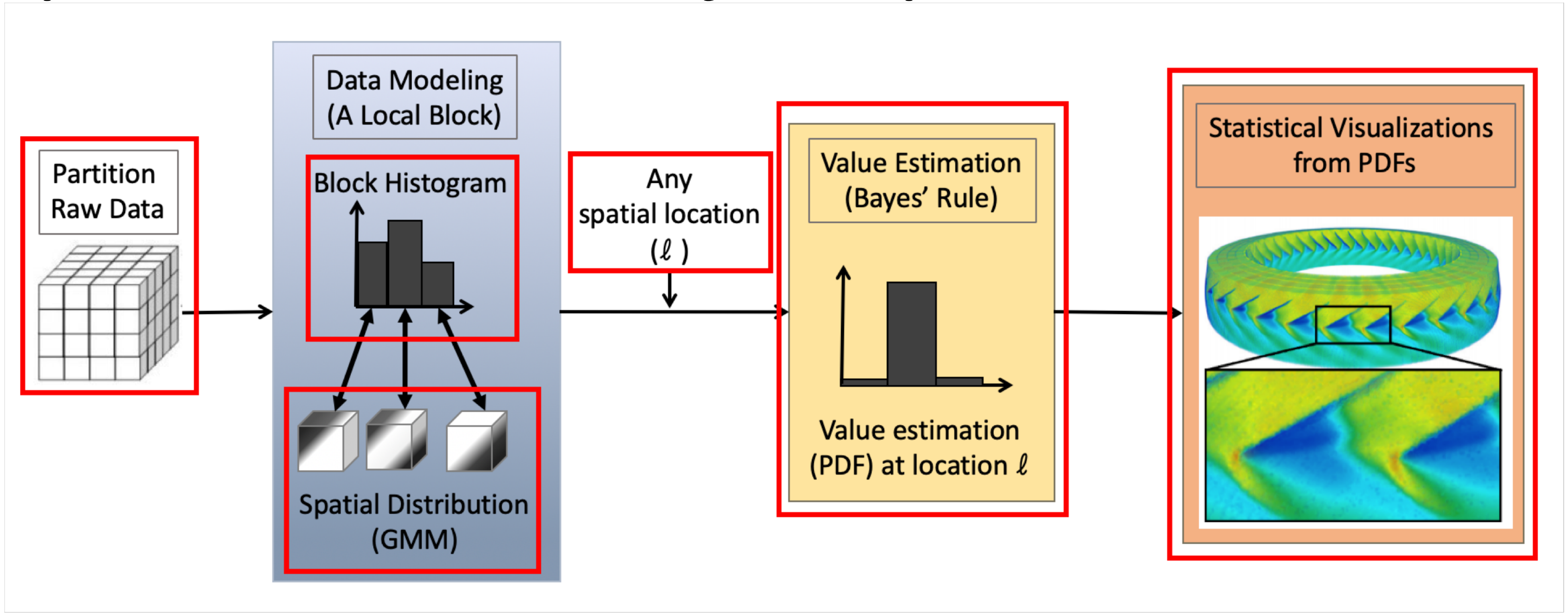


Image from Proxy (PSNR: 37.07)
15.3GB

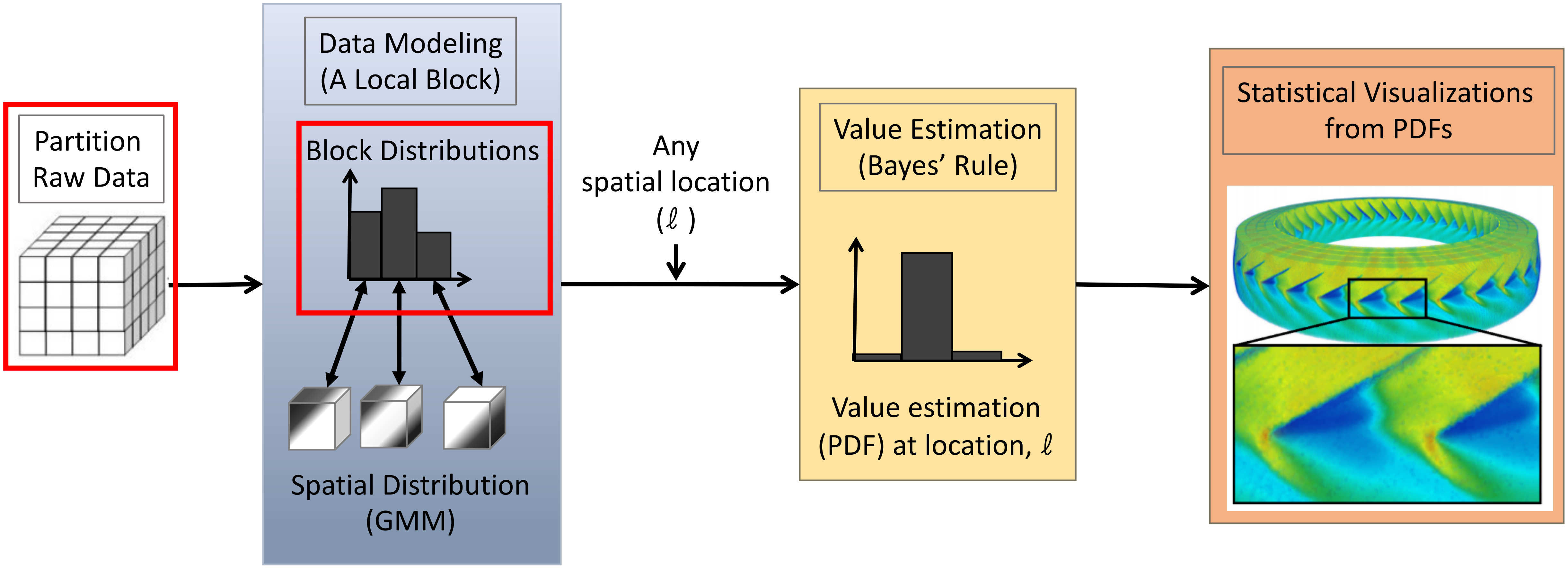
Object Space Distributions Proxy

Arbitrary view exploration

- Option 1: Samples generated from the view dependent proxies can be warped to different views
- Option 2: Create object space distributions

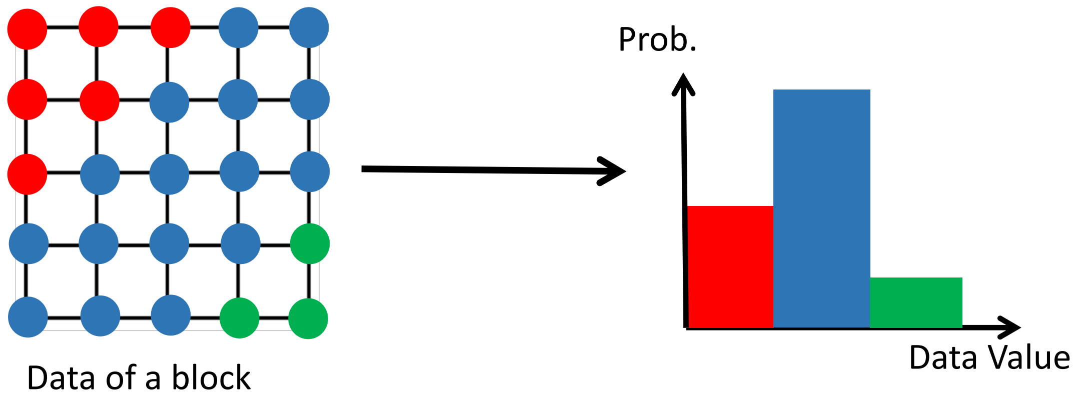


Data Modeling – Block Histogram

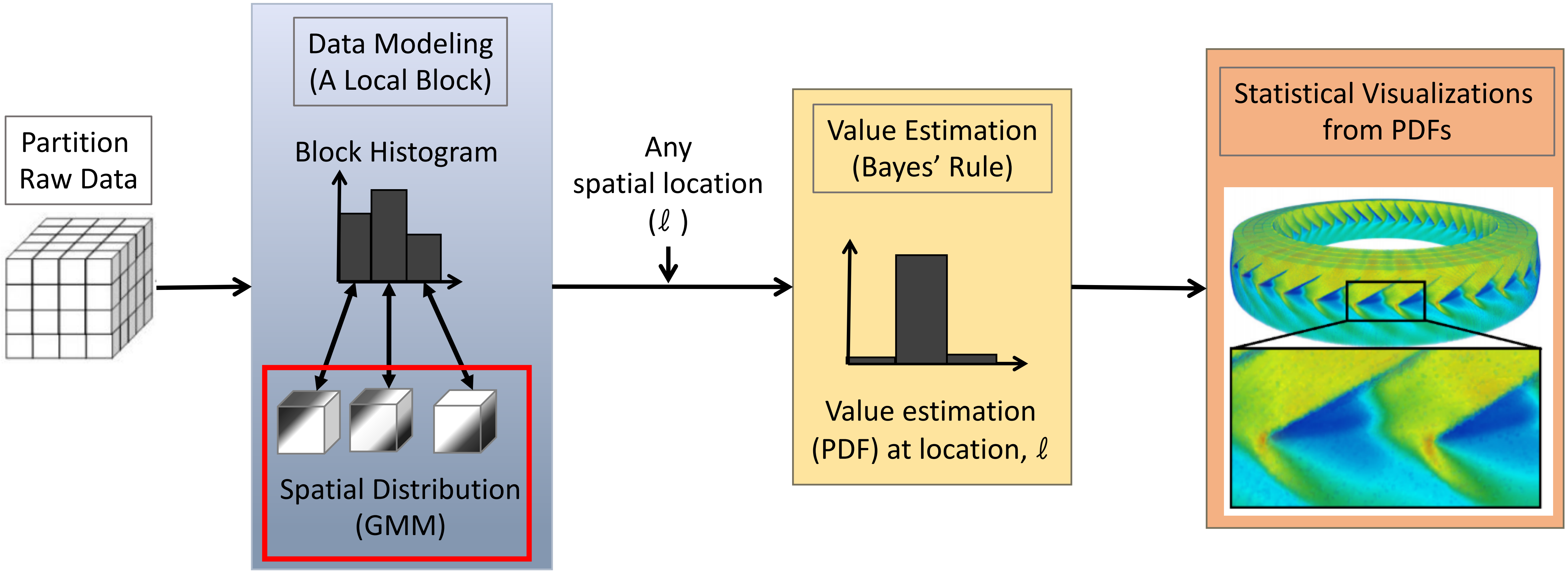


Data Modeling – Block Distributions

- Block histogram or value GMM summarizes data samples in a block
  - Bin b_i represents a continuous data value range $[L_{b_i}, U_{b_i}]$
  - $H(b_i) = \frac{N(b_i)}{\sum_{k=0}^{B-1} N(b_k)}$
    - $N(b_k)$: number of grid points whose values are in range $[L_{b_k}, U_{b_k}]$

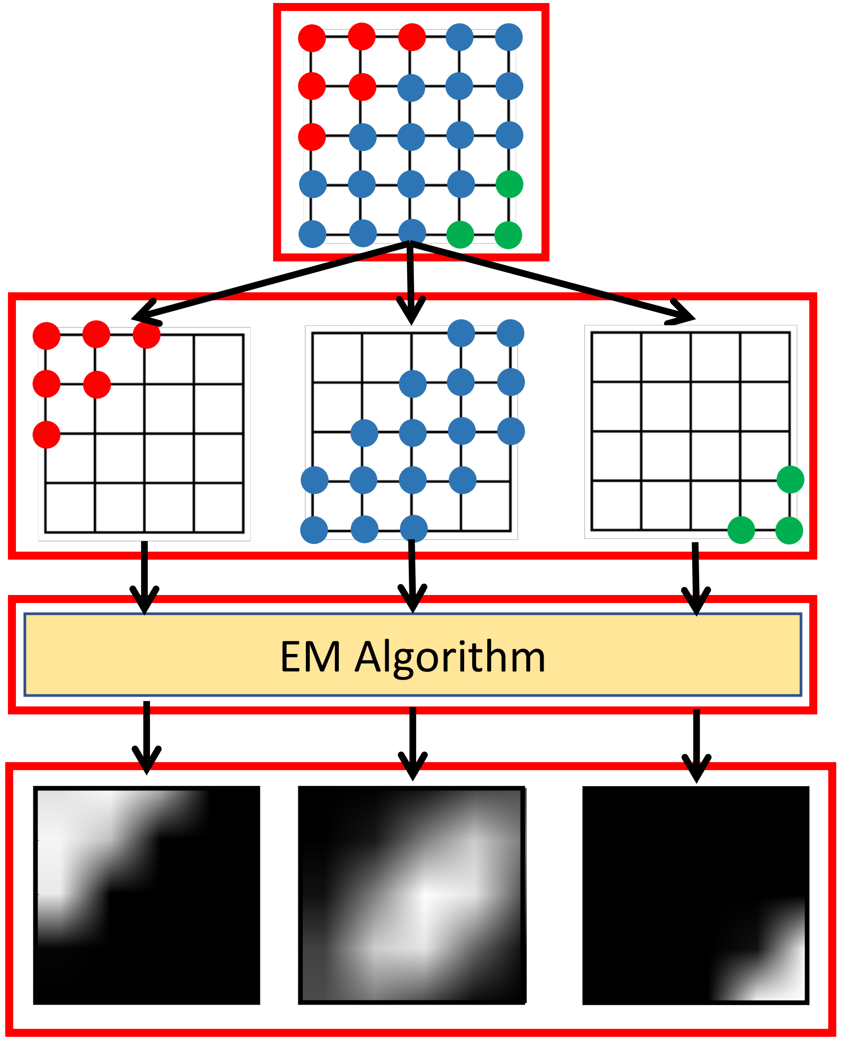


Data Modeling – Spatial Distribution

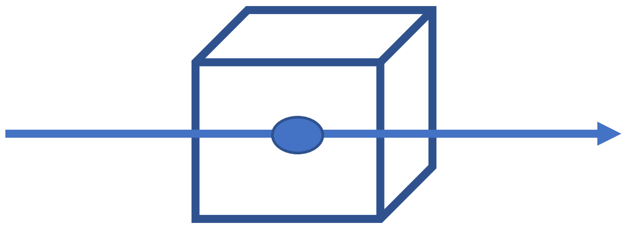


Data Modeling – Spatial Distribution

- Block histogram does not retain samples' locations
- Each bin creates a spatial distribution: $\{S_0, S_1, \ldots S_{B-1}\}$
  - S_{b_i} : maps a spatial location (ℓ) to a probability
    - how likely ℓ has a sample whose value within the range of b_i
    - Estimated by a multivariate GMM (Spatial GMM)
- Spatial GMM modeling
  - Collects coordinates of all grid points assigned to bin b_i
  - Uses EM algorithm to estimate the parameters of the GMM
  - Repeat the process for each bin



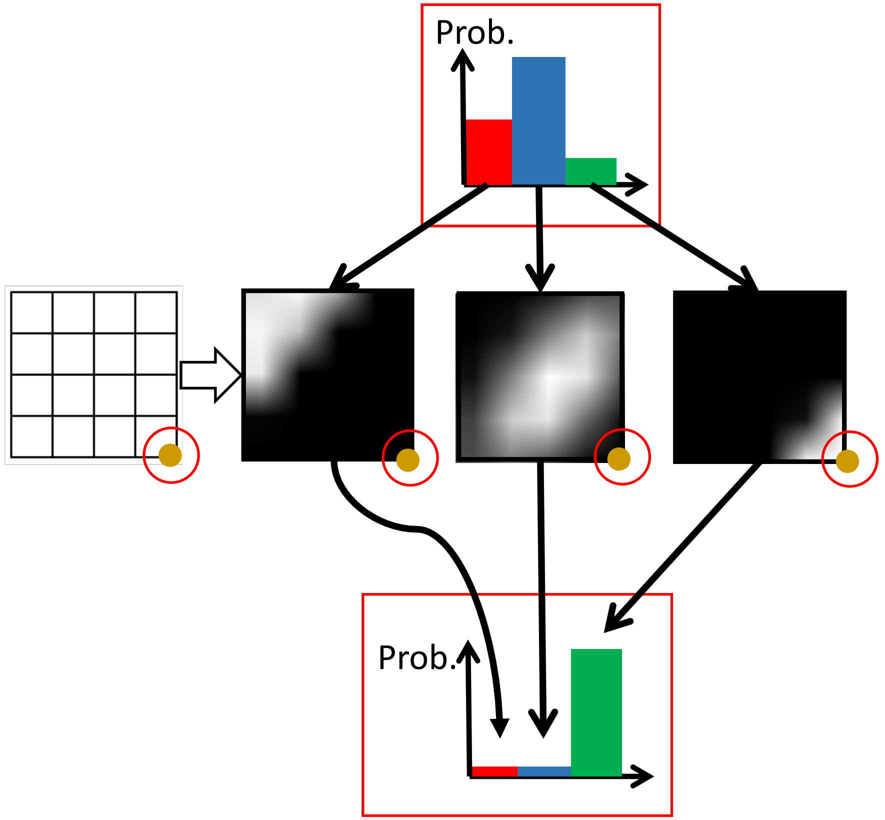
Value Estimation at a location **X**



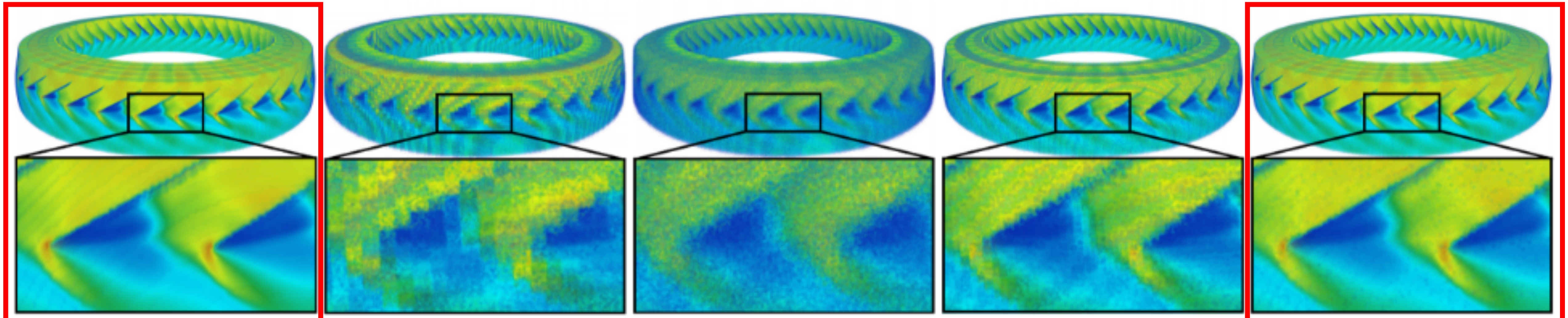
- Spatial GMMs to model spatial probability density function for each value interval (V)

- Bayes' rule
  - The prior is adjusted by the related evidences
  - Prior P(V) : block distribution/ histogral
  - Evidences: probabilities of spatial GMMs at
  - Posterior: estimated PDF at **x**

$$P(v|\mathbf{x}) \sim P(\mathbf{x}|v) * P(v)$$



Post-Hoc Analysis
Sampling-based Volume Rendering



Raw data
Size: 10871MB

Block histogram
Size: 131.4MB
Block size: 22^3

Block histogram w/ interpolation
Size: 131.4MB
Block size: 22^3

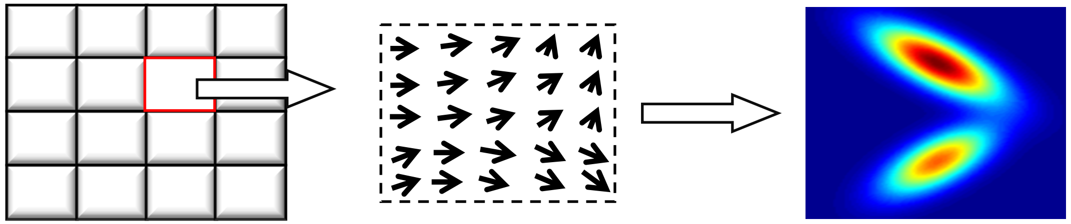
Block GMM
Size: 163.71MB
Block size: 10^3

Our approach
Size: 151.54MB
Block size: 32^3
Number of Gaussians: 4

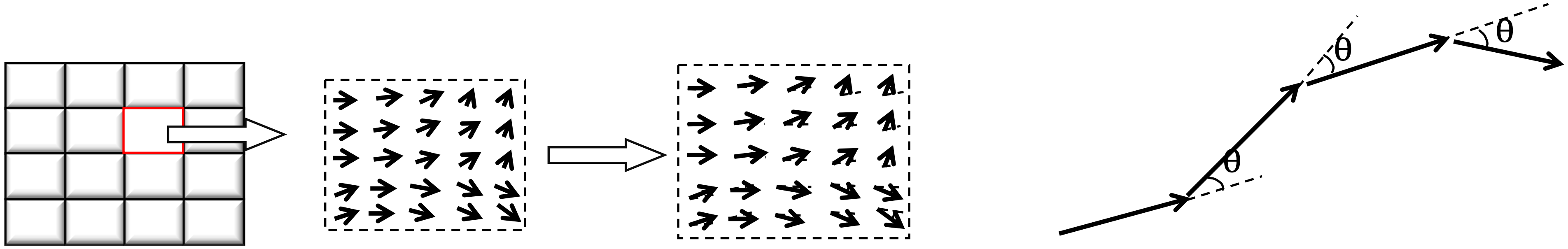
Volume rendering from the reconstructed volume of Turbine pressure variable

Particle Tracing in Distribution Fields

- Representing the vectors in the block using Gaussian mixture model (GMM):
  $g(\vec{v}) = \sum_{j=1}^{K} \omega_j N(\vec{v}|\mu_j, \Sigma_j)$



- The vector transition information can also be represented by GMMs of winding angle: GMM $h(\theta) = \sum_{j=1}^{K} \omega_j N(\theta|\mu^{\theta}{}_j, \Sigma^{\theta}{}_j)$



Particle Tracing in Distribution Fields

- What to do with vector GMM of vector $g(\vec{v}) = \sum_{j=1}^{K} \omega_j N(\vec{v}|\mu_j, \Sigma_j)$
  - Use Monte Carlo sampling to trace a bundle of traces
  - Use the mean vector to trace a single trace
- $g(\vec{v})$ is an unconditional distribution

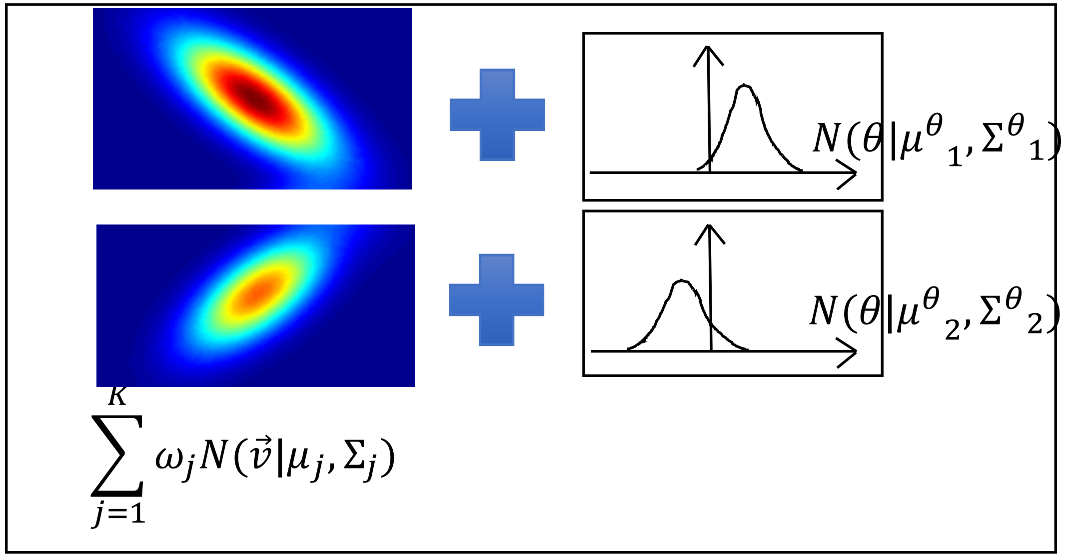
- Condition of $g(\vec{v})$?
  - Have already traced the particle for k steps, by $\{\vec{v}_0, \dots, \vec{v}_{k-1}\}$
  - Conditional distribution $g(\vec{v}|\vec{v}_0, \dots, \vec{v}_{k-1})$
  - Assume a Markov model
  - Conditional distribution $g(\vec{v}|\vec{v}_{k-1})$

Particle Tracing in Distribution Fields

- Conditional distribution $g(\vec{v}|\vec{v}_{k-1})$
  - Bayes Theorem
    - $g(\vec{v}|\vec{v}_{k-1}) = \alpha * g(\vec{v}) * g(\vec{v}_{k-1}|\vec{v})$
  - Replace $\vec{v}_{k-1}$ with its angle with $\vec{v}$: $\theta(\vec{v}_{k-1}, \vec{v})$
    - $g(\vec{v}|\vec{v}_{k-1}) = \alpha * g(\vec{v}) * g(\theta(\vec{v}_{k-1}, \vec{v})|\vec{v})$
- As a result
  - $g(\vec{v}|\vec{v}_{k-1}) = \alpha * \sum_{j=1}^{K} \left(\omega_j N\left(\theta(\vec{v}_{k-1}, \mu_j)\middle|{\mu^\theta}_j, {\Sigma^\theta}_j\right)\right) N(\vec{v}|\mu_j, \Sigma_j)$

Particle Tracing in Distribution Fields

- Conditional distribution $g(\vec{v}|\vec{v}_{k-1})$
  - Unconditional $g(\vec{v}) = \sum_{j=1}^{K} \omega_j N(\vec{v}|\mu_j, \Sigma_j)$
  - Conditional $g(\vec{v}|\vec{v}_{k-1}) = \alpha * \sum_{j=1}^{K} \left(\omega_j N\left(\theta(\vec{v}_{k-1}, \mu_j)\middle|\mu^{\theta}{}_j, \Sigma^{\theta}{}_j\right)\right) N(\vec{v}|\mu_j, \Sigma_j)$



Tracing Method

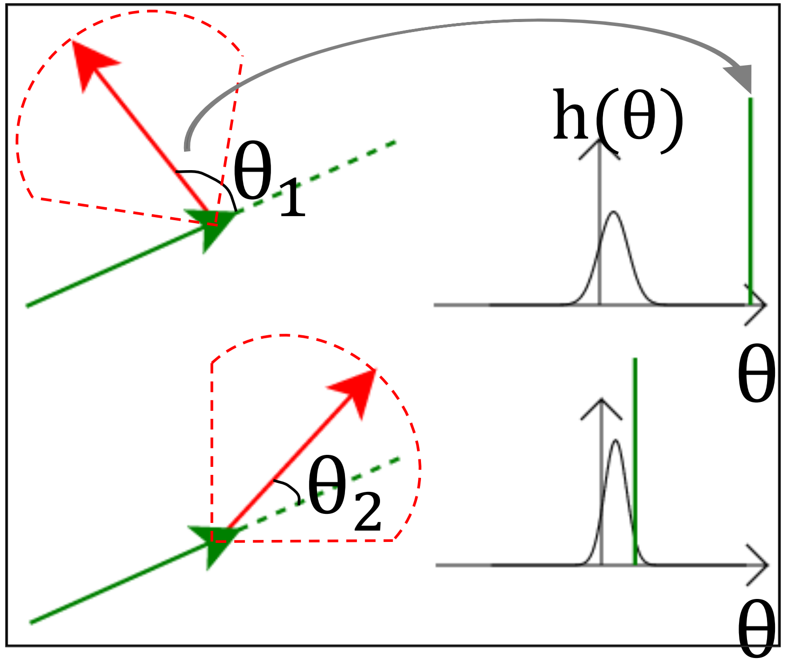
- Tracing with the conditional distribution $g(\vec{v}|\vec{v}_{k-1})$
  - Use Monte Carlo sampling to trace a bundle of traces – sample from $g(\vec{v}|\vec{v}_{k-1})$
    - *Conditional Monte Carlo (CMC)*
    - Use $g(\vec{v}|\vec{v}_{k-1})$ from the second step
  - Use the mean vector to trace a single trace – mean of $g(\vec{v}|\vec{v}_{k-1})$
    - *Conditional Mean Vector (CMV)*
    - Use $g(\vec{v}|\vec{v}_{k-1})$ from the second step
    - Use $g(\vec{v}|\vec{v}_{k-1})$ only when the mean of the winding angle distribution has an absolute value larger than a threshold

Qualitative Comparison

- Comparison - *Conditional Monte Carlo (CMC)*
  - Reward the Gaussian component that better fits the angle pattern

Baseline Monte Carlo

Conditional Monte Carlo



Cost and Performance

- Cost of using conditional distribution
  - Extra storage:
    - $g(\vec{v}) = \sum_{j=1}^{K} \omega_j N(\vec{v}|\mu_j, \Sigma_j)$, plus $h(\theta) = \sum_{j=1}^{K} \omega_j N(\theta|\mu^{\theta}{}_j, \Sigma^{\theta}{}_j)$
    - 33% extra storage

	Data Reduction		Single Line Tracing		Monte Carlo Tracing	
	Baseline	Our Method	Baseline	CMV	Baseline	CMC
Time (s)	73.35	76.53	0.1003	0.1080	3.307	5.480

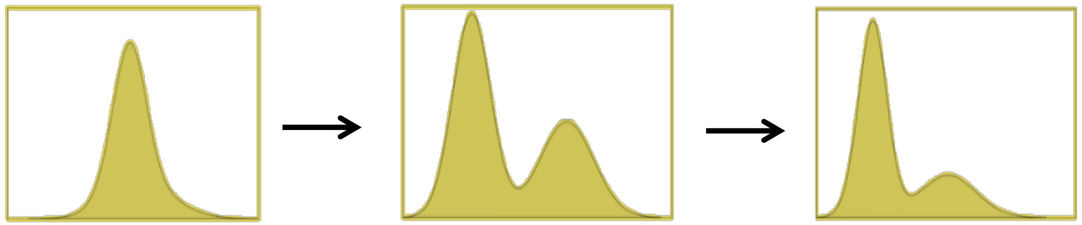
Distributions Based Feature Tracking

Probabilistic Data Modeling

- A block-wise data modeling approach
  - Each block is represented by a mixture of Gaussians (GMM)
- Probability density of a GMM is expressed as:

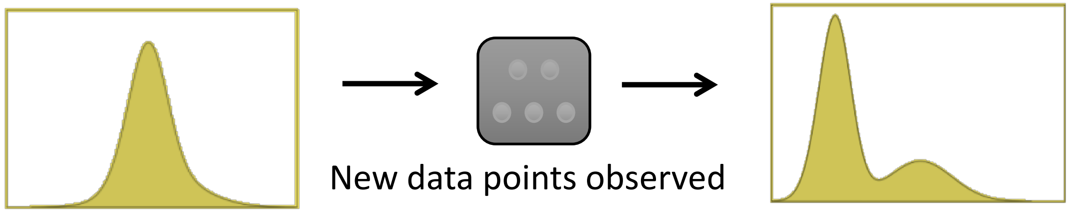
$$p(X) = \sum_{i=1}^{K} \omega_i * N(X \mid \mu_i, \sigma_i)$$

Incremental estimation of temporal data distribution



Update block distributions incrementally

Incremental Distribution Update for Time-Varying Fields



GMM before update at t = t_0 Distribution after update at t = t_1

- Update mean and standard deviation as:

$$\mu_{i,t} = (1-\beta)\mu_{i,t-1} + \beta\mu_{i,t}$$

$$\sigma^2_{i,t} = (1-\beta)\sigma^2_{i,t-1} + \beta(\mu_{i,t} - x_{i,t})^2$$

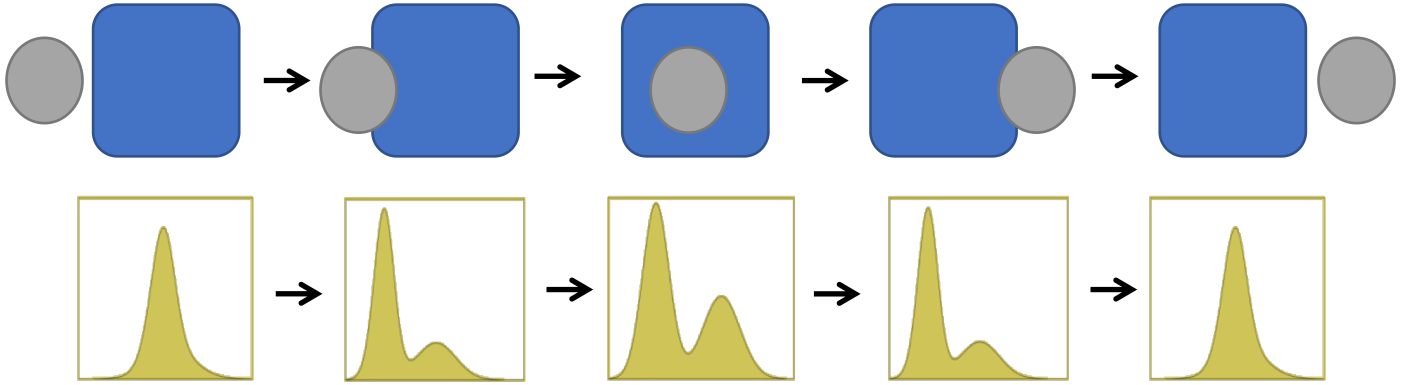
- Update weight as:

$$\omega_{i,t} = (1-\beta)\omega_{i,t-1} + \beta(I_{i,t})$$

Classification Using Foreground Detection

- A block is classified as foreground if new data
  - do not match any existing Gaussians
  - match with a newly created Gaussian

$$Possibility_{foreground,t}(b_{i,t}) = q_{i,t} / n_{i,t}$$

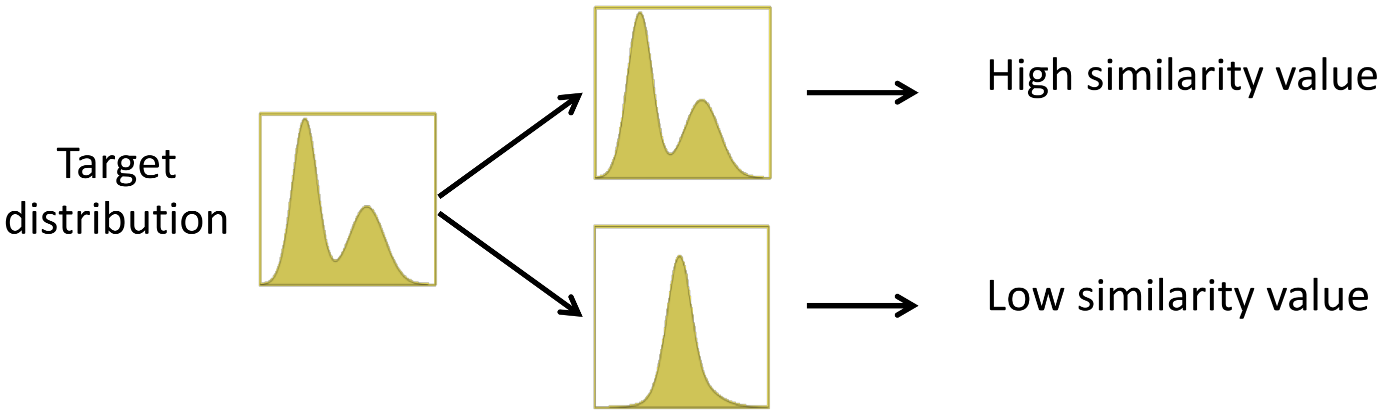


Similarity Based Classification

- Similarity of a block with the target GMM is estimated by Bhattacharya distance:

$$\psi(p, p') = \sum_{i=0}^{n} \sum_{j=0}^{m} \omega_i \omega_j' \xi(p_i, p_j')$$

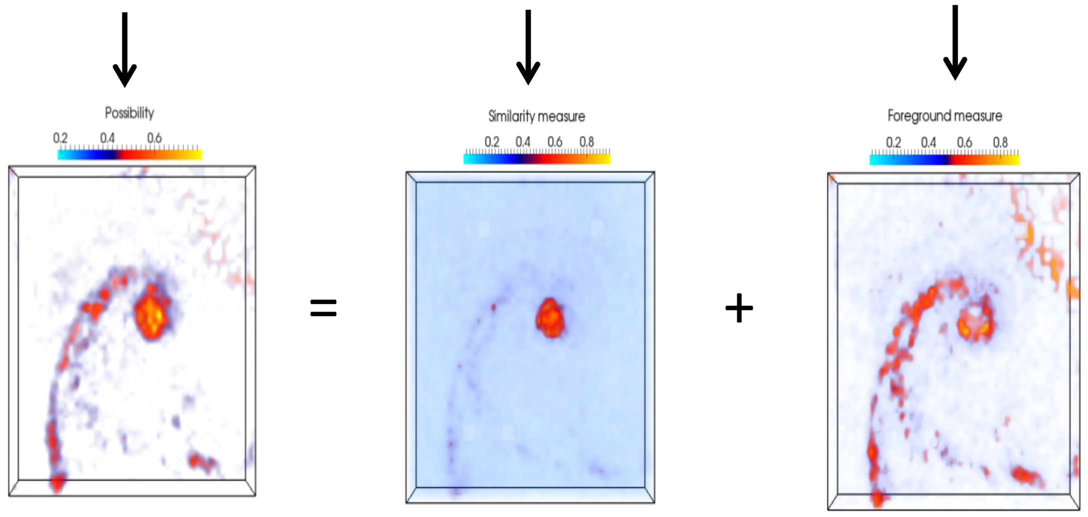
$$Possibility_{similarity,t}(b_{i,t}) = 1 - \psi_{norm}(b_{i,t}, f_t)$$



Feature-aware Classification Field

- Linear combination of foreground information and similarity measure

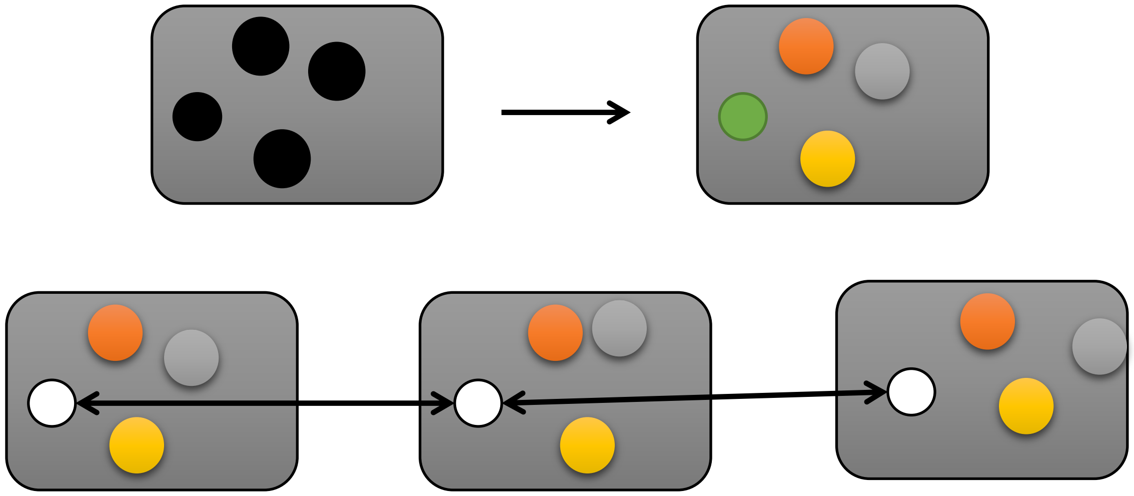
$$Possibility_{feature}(b_i) = \gamma * Possibility_{similarity}(b_i) + (1-\gamma) * Possibility_{foreground}(b_i)$$



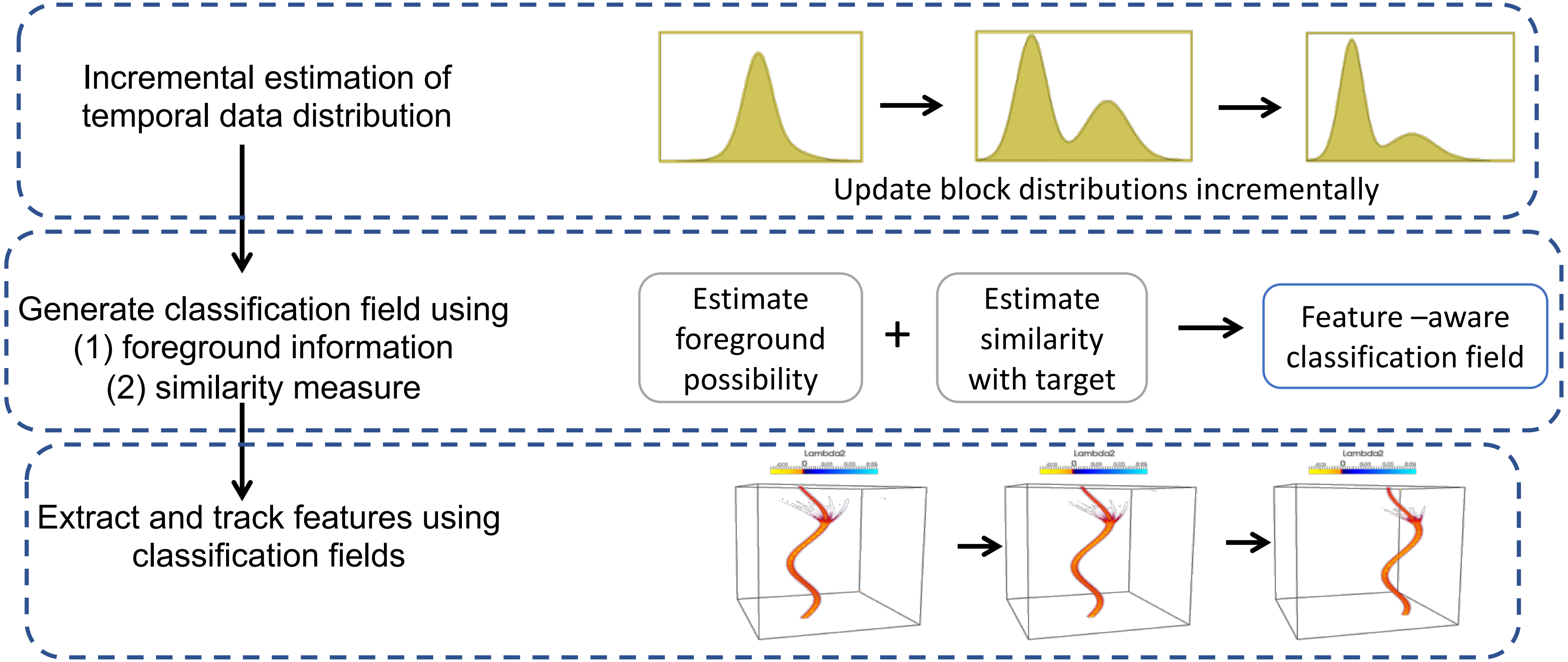
Final combined field Similarity measure Foreground measure

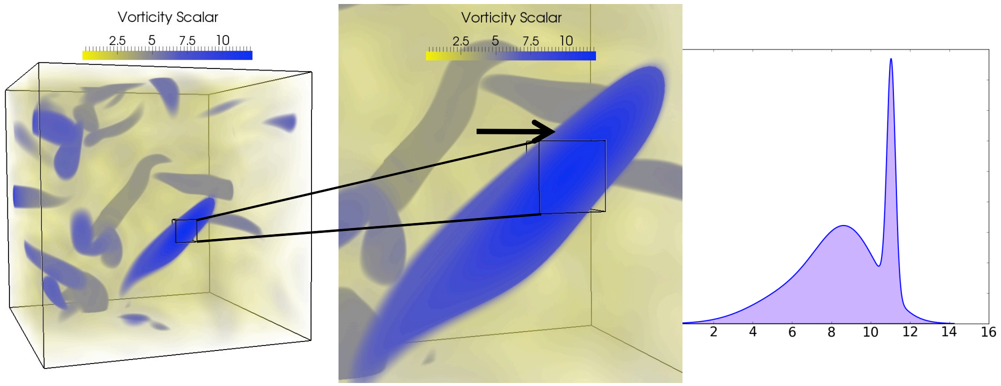
Tracking in Classification Field

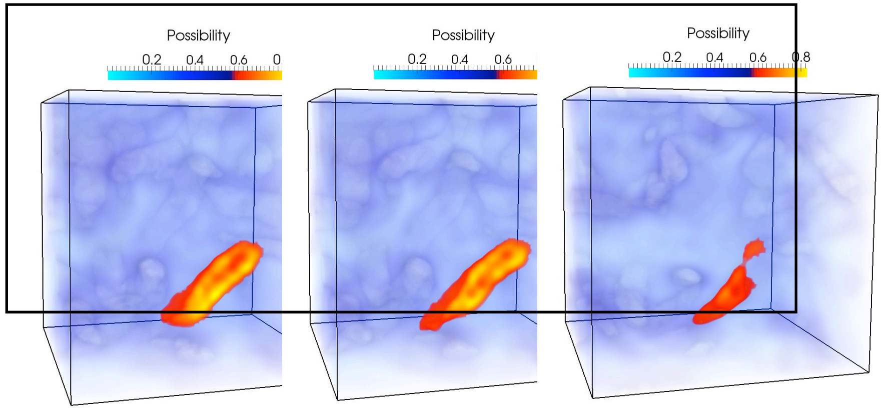
- Given a user specified threshold
  - Segment the data using the threshold
  - Apply Connected Component algorithm



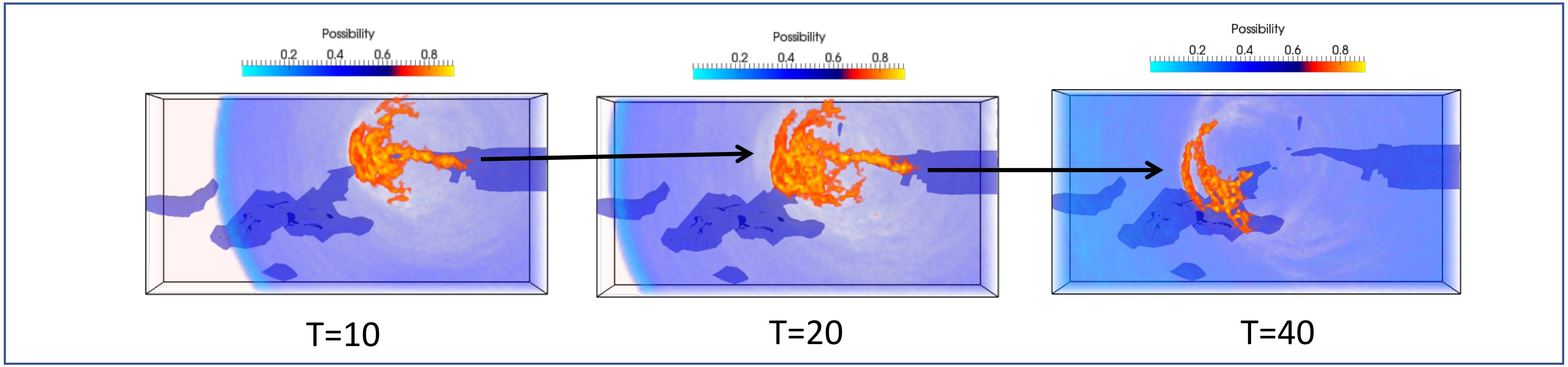
Distribution Driven Feature Tracking

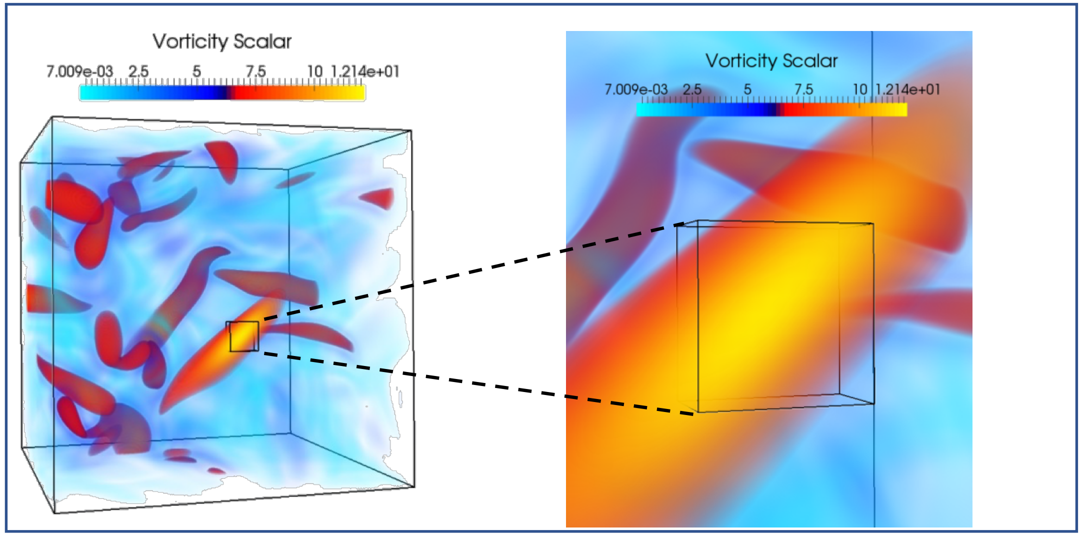


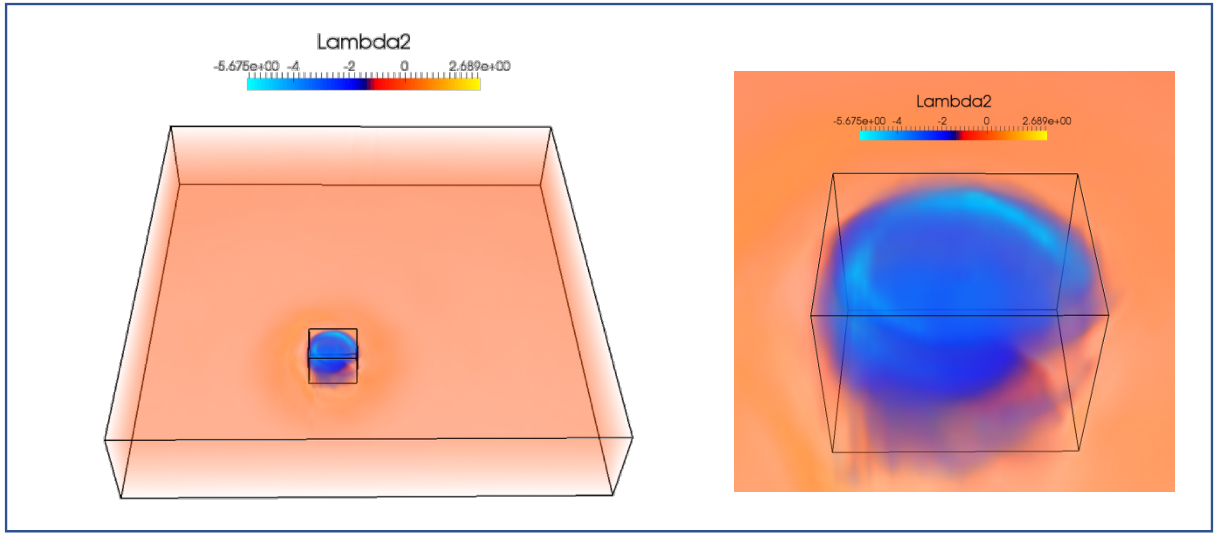




Tracking Examples



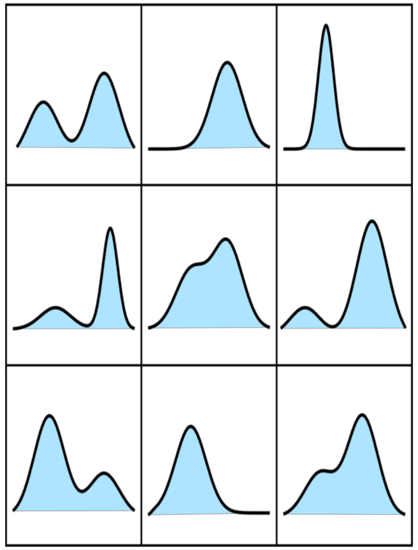




Query and Exploration of Distributions

- Provide an overview of the distribution data without sampling
- Identifying features from distributions directly
- Visualization of probability distribution fields are challenging
  - Visualizing distribution at each data point needs more screen space
  - Overall trend may not be easy to see
- Possible approaches
  - Statistical summaries (e.g. mean)
  - Dissimilarity measures

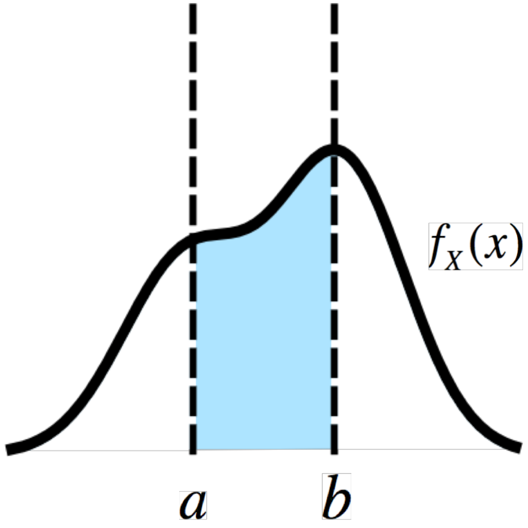
A probability distribution field



Visualizing Cumulative Probabilities

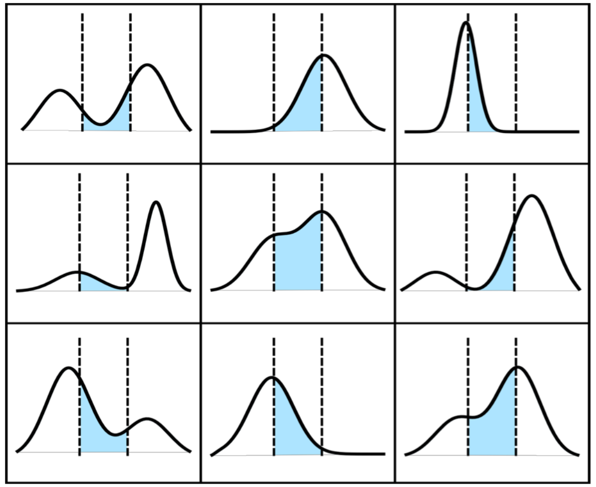
- Visualizing and analyzing distributions with cumulative probabilities over different value ranges
- The cumulative probability of a probability density function $f_X(x)$ for random variable X over a range Γ=(a,b) is defined as

$$\Pr[a \leq X \leq b] = \int_a^b f_X(x)\,dx$$



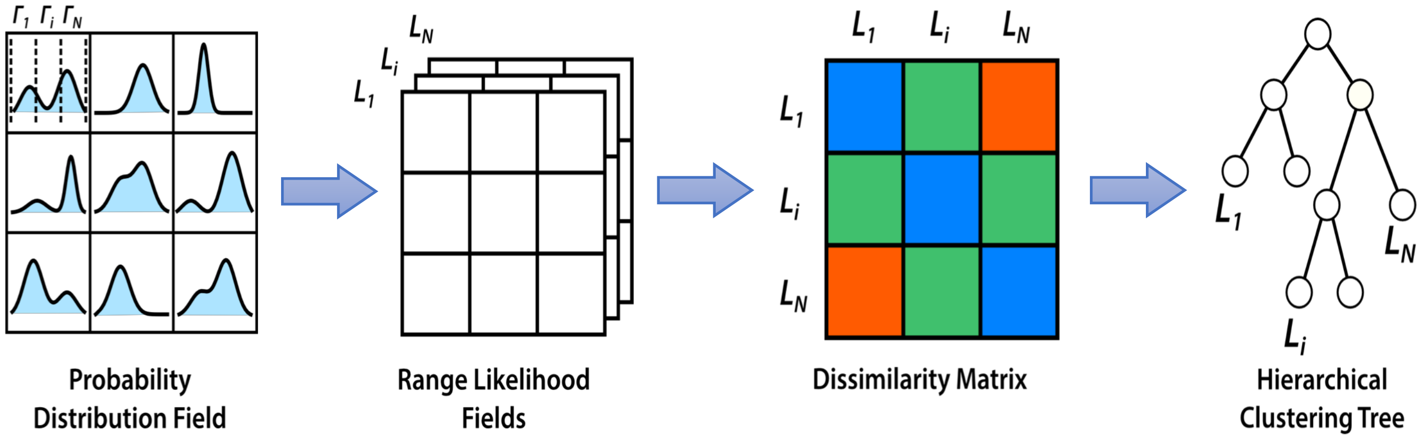
Probability Distribution Field to Cumulative Probability Fields

- By calculating cumulative probabilities over a given value range for distributions on each grid point
  - A scalar field is generated
  - The resulting scalar field is called *range likelihood field (RLF)*



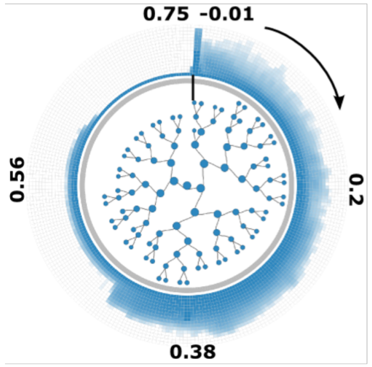
Exploring Value Ranges

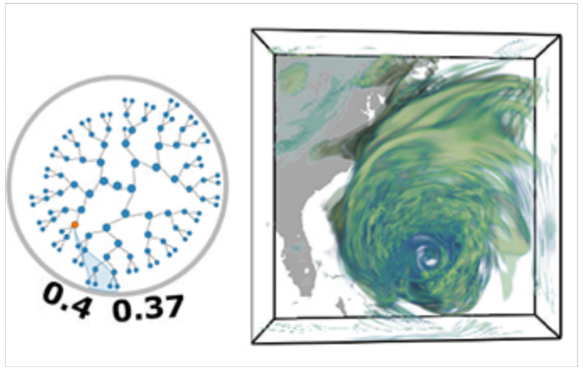
- To select representative value ranges, we
  - partition the value domain into *N* subranges $\Gamma_1, \Gamma_2, \ldots, \Gamma_N$
  - generate *N* RLFs $L_1, L_2, \ldots, L_N$ for the subranges
  - compute distances between every pair of RLFs
  - organize the value ranges and corresponding RLFs into a binary tree using hierarchical clustering

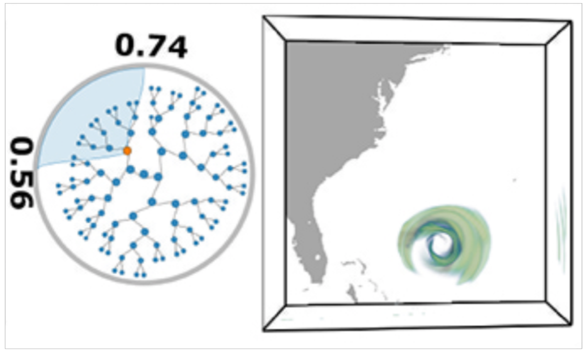


Exploring Value Ranges

- To select representative value ranges, we
  - partition the value domain into *N* subranges $\Gamma_1, \Gamma_2, \ldots, \Gamma_N$
  - generate *N* RLFs $L_1, L_2, \ldots, L_N$ for the subranges
  - compute distances between every pair of RLFs
  - organize the value ranges and corresponding RLFs into a binary tree using hierarchical clustering

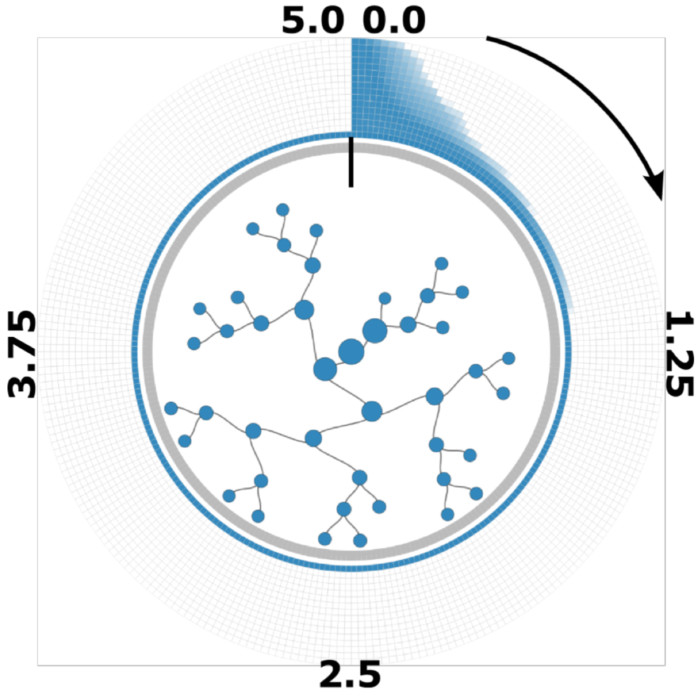






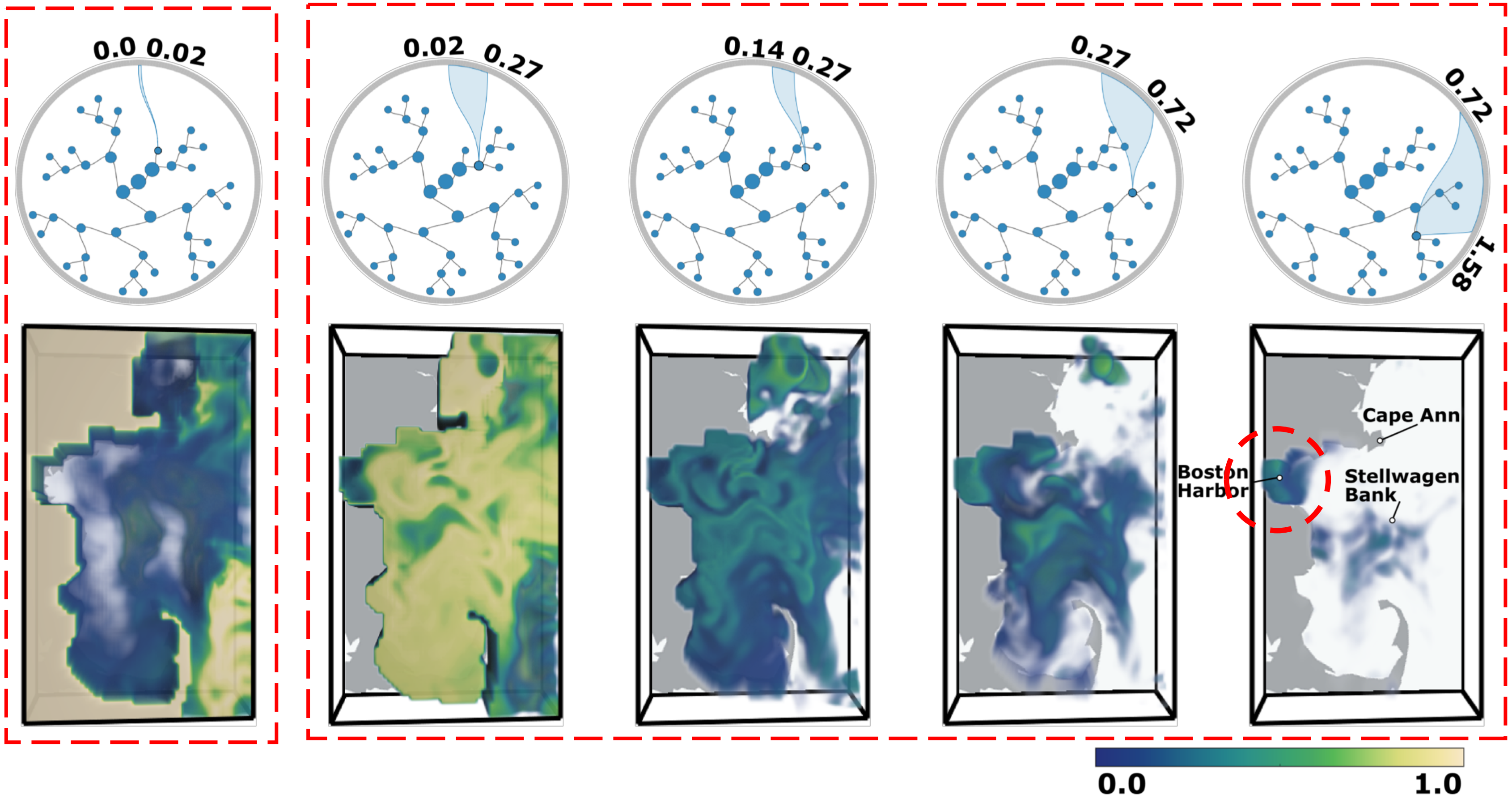
Case Study - Massachusetts Bay Sea Trial Ensemble Dataset

- The probability distribution field
  - Performing kernel density estimation for the variable chlorophyll-a concentration (CHL) on all 600 ensemble members
- The initial RLT view



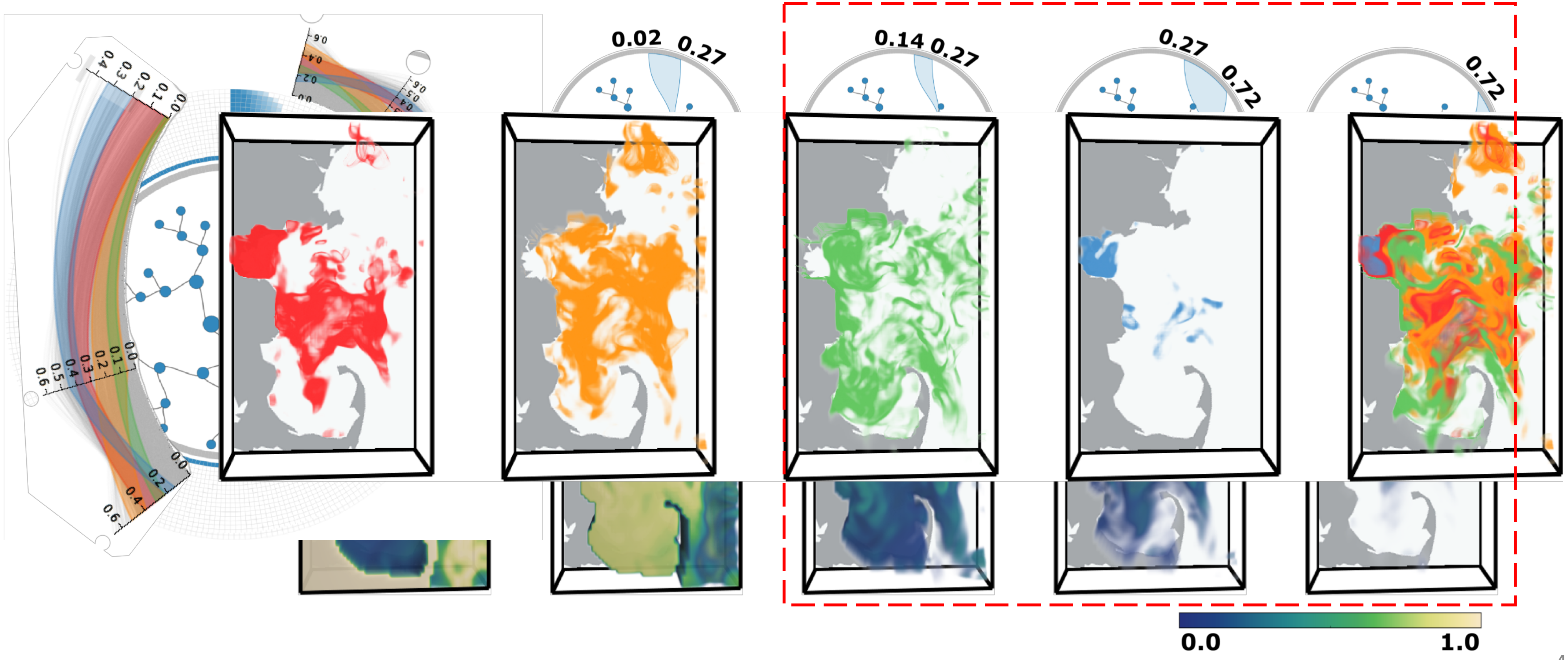
Case Study - Massachusetts Bay Sea Trial Ensemble Dataset

- Visualizing user selected RLFs



Case Study - Massachusetts Bay Sea Trial Ensemble Dataset

- Visualizing and Analyzing Multiple RLFs



Additional Work

- Multivariate distribution modeling using Coupla functions (Vis 17,18)
- Pathline and data modeling for time-varying flow fields (LDAV 16)
- Efficient histogram search (EuroVis 16, Pacific Vis 17)
- Uncertainty and sensitivity simulation parameter analysis (Vis 16, 17)
- Surface density estimation (TVCG 19)
- Ensemble Data Modeling and Reconstruction (Pacific Vis 19)

Future Research Directions