Shape Descriptors

Ligang Liu
Graphics&Geometric Computing Lab
USTC
http://staff.ustc.edu.cn/~lgliu

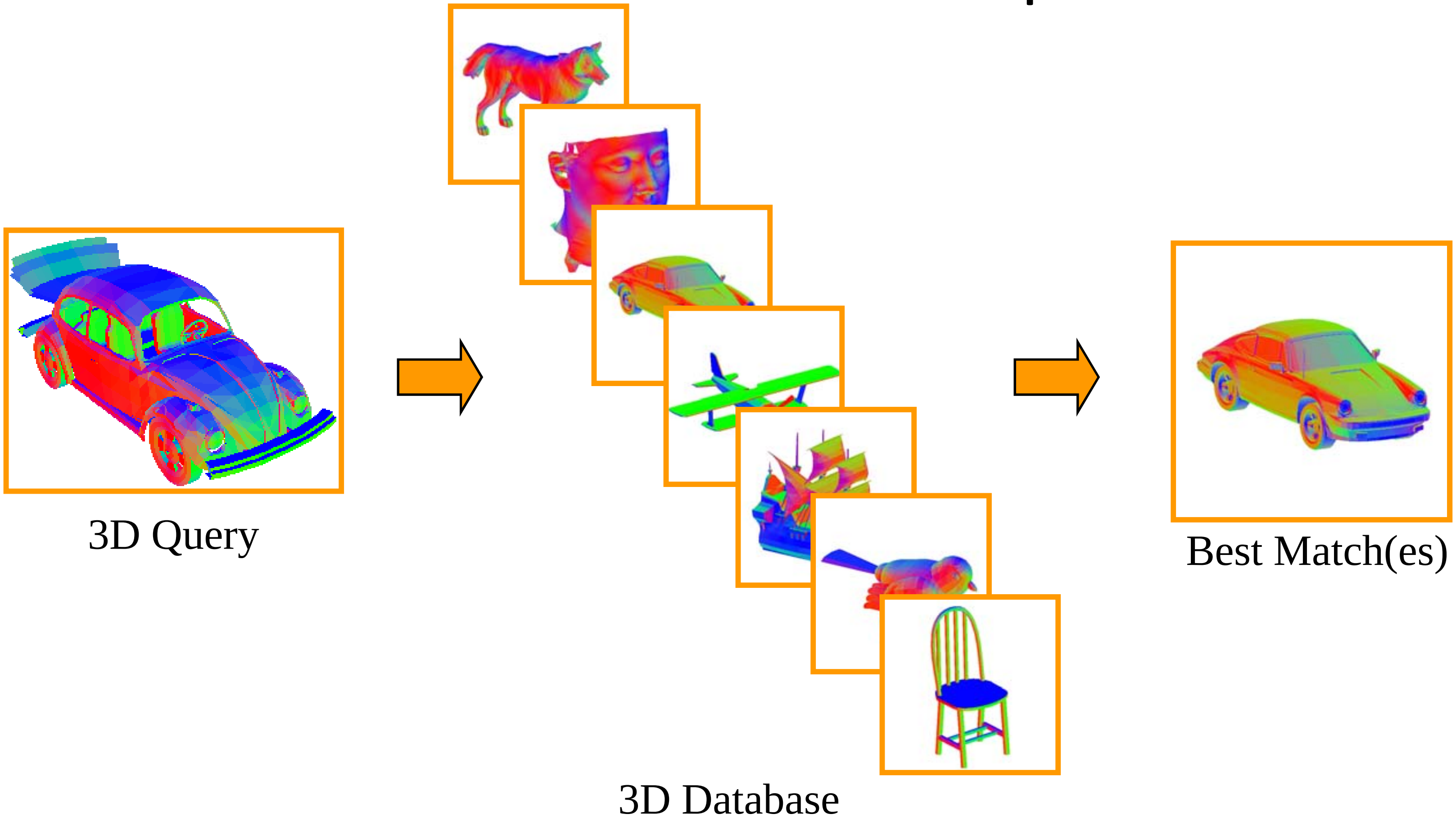
Most of slides are from Thomas Funkhouser

Outline

- Why shape descriptors?
- How do we represent shapes?
- Conclusion

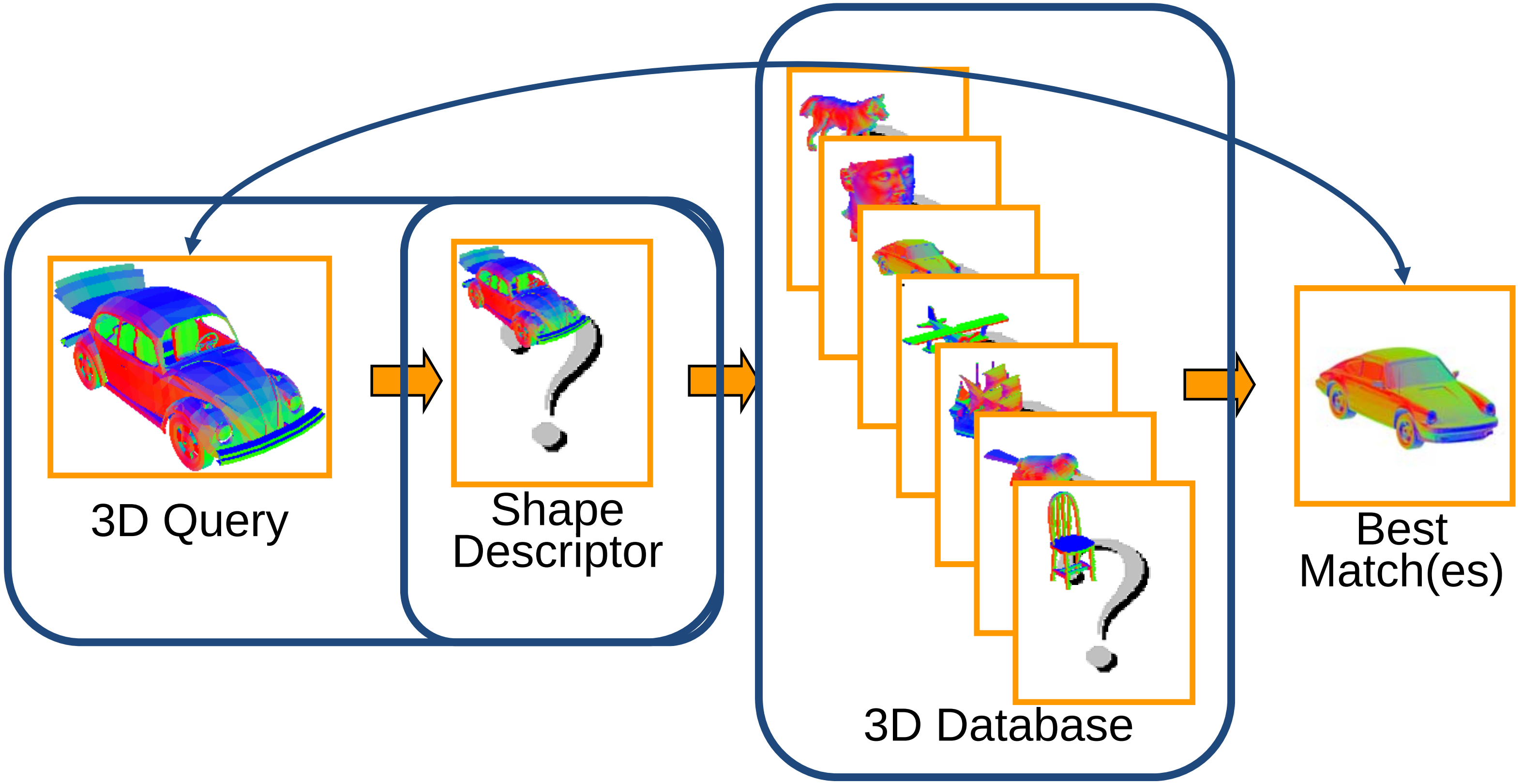
Goal

- Find 3D models with similar shape



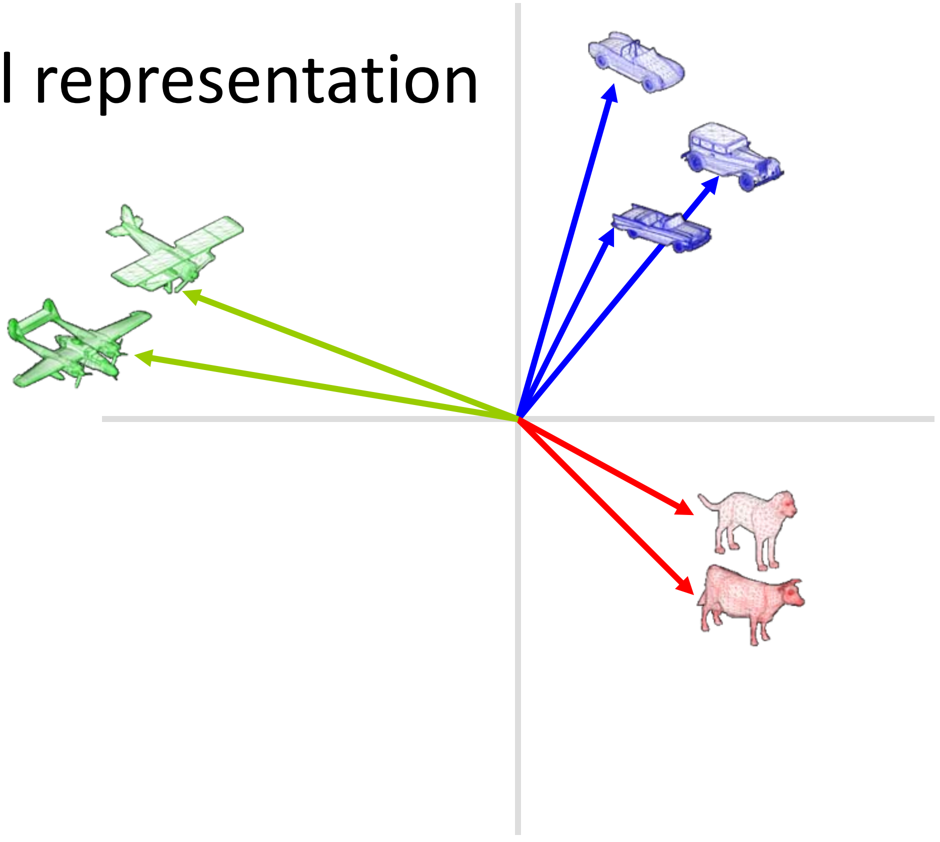
Goal

- Shape Descriptor:
  - Structured abstraction of a 3D model
  - Capturing salient shape information

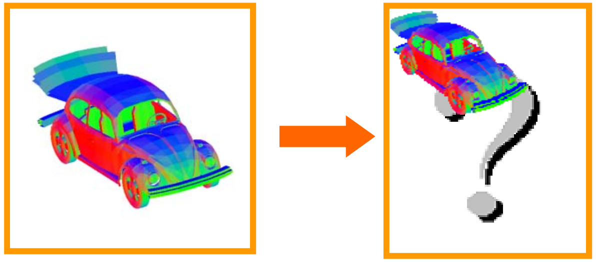


Shape Descriptors

- Shape Descriptors
  - Fixed dimensional vector
  - Independent of model representation
  - Easy to match



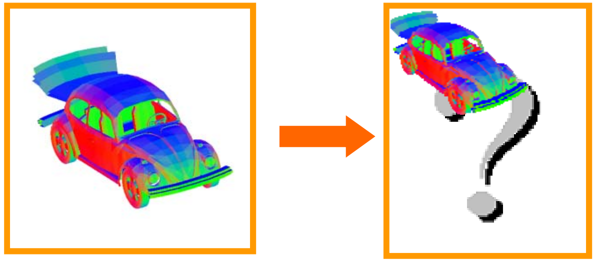
Shape Descriptors



- Representation:
  - What can you represent?
  - What are you representing?
- Matching:
  - How do you align?
  - Part or whole matching?

Shape Descriptors

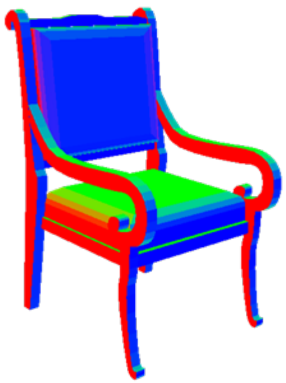
- Representation:
  - What can you represent?
  - What are you representing?



- Matching:
  - How do you align?
  - Part or whole matching?

Point Clouds

Polygon Soups



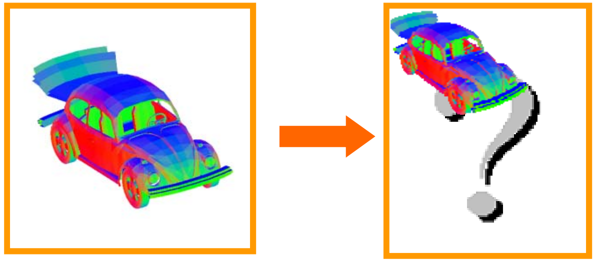
Closed Meshes

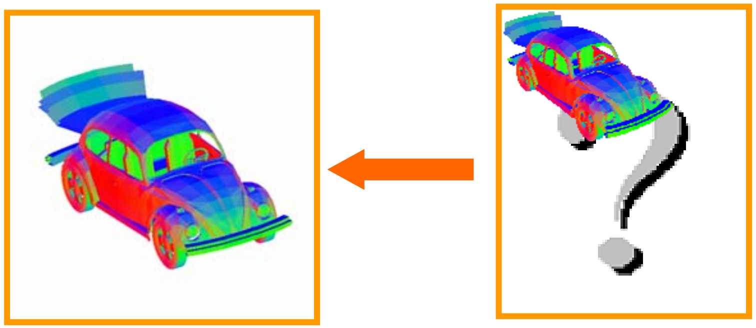
Genus-0 Meshes

Shape Spectrum

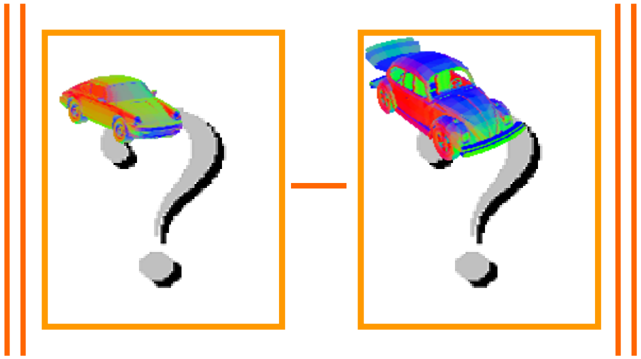
Shape Descriptors

- Representation:
  - What can you represent?
  - What are you representing?
- Matching:
  - How do you align?
  - Part or whole matching?





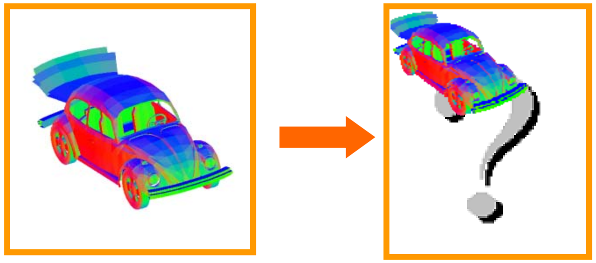
Is the descriptor invertible?



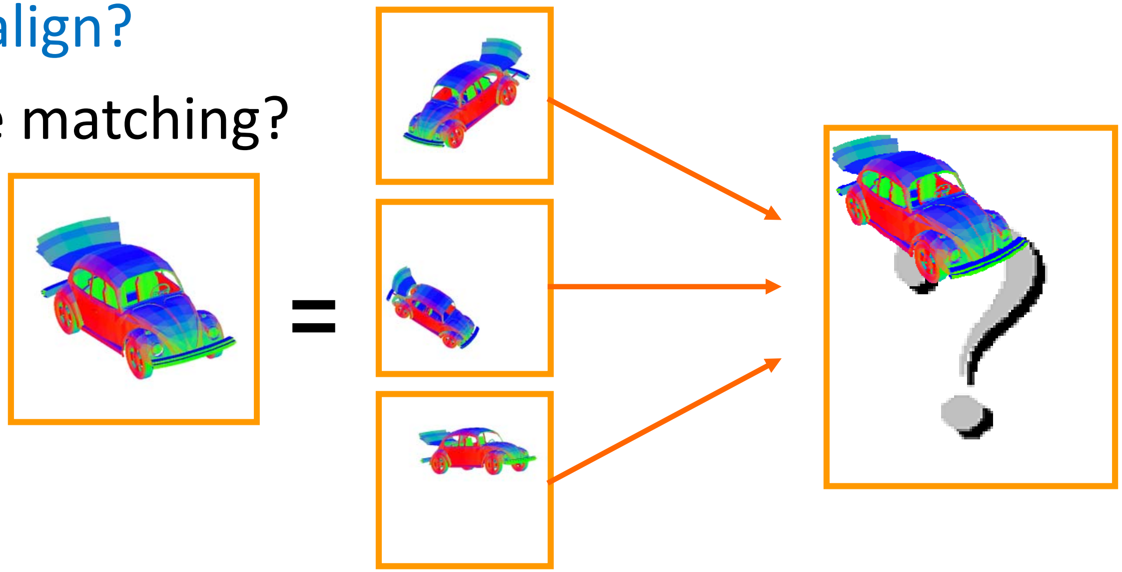
What is represented by the difference in descriptors?

Shape Descriptors

- Representation:
  - What can you represent?
  - What are you representing?



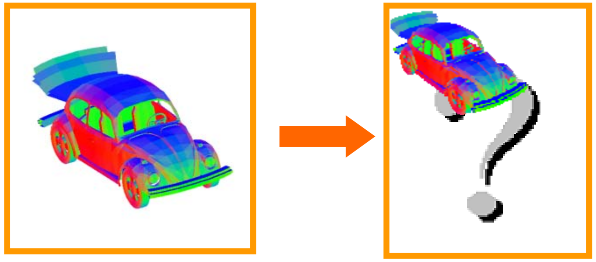
- Matching:
  - How do you align?
  - Part or whole matching?

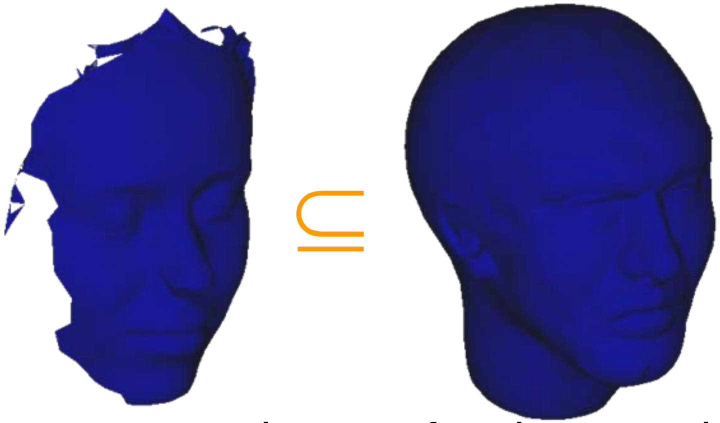


How do you represent models across the space of transformations that don't change the shape?

Shape Descriptors

- Representation:
  - What can you represent?
  - What are you representing?
- Matching:
  - How do you align?
  - Part or whole matching?





Can you match part of a shape to the whole shape?

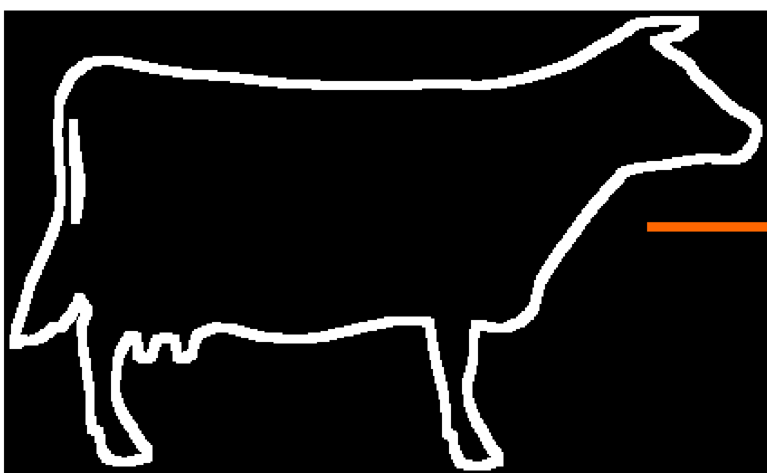
Outline

- Why shape descriptors?
- How do we represent shapes?
  - Volumetric Representations
  - Surface Representations
  - View-Based Representations
- Conclusion

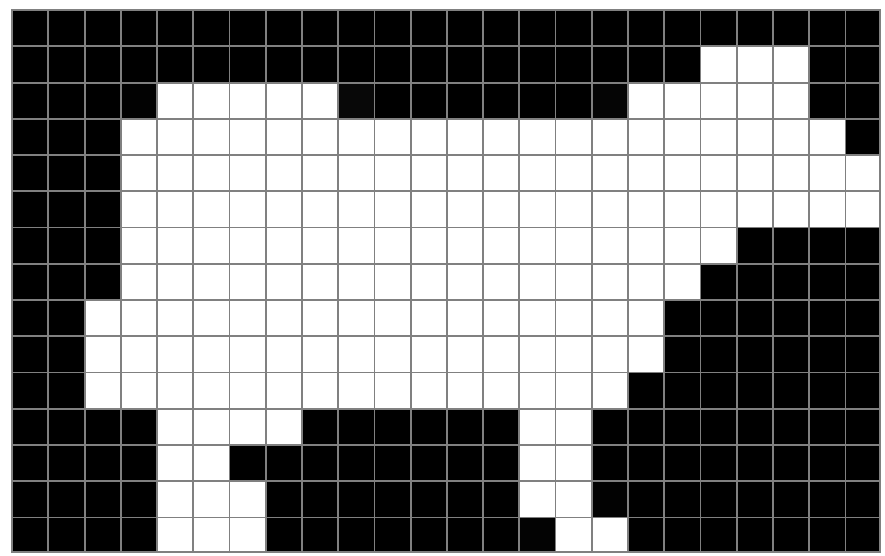
Volumetric Representations

Represent models by the volume that they occupy:

- Rasterize the models into a binary voxel grid
  - A voxel has value 1 if it is inside the model
  - A voxel has value 0 if it is outside



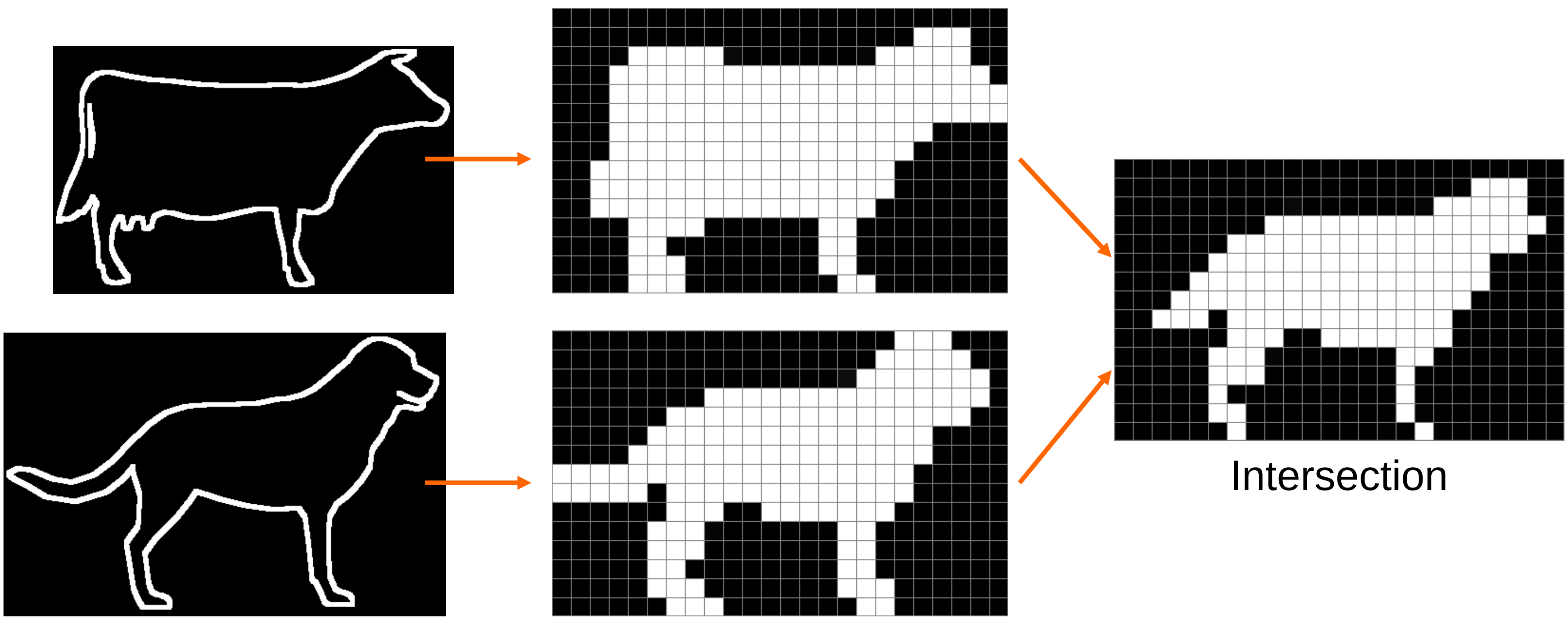
Model



Voxel Grid

Volumetric Representations

- Compare models by measuring the overlaps of their volumes
  - Similarity is measured by the size of the intersection



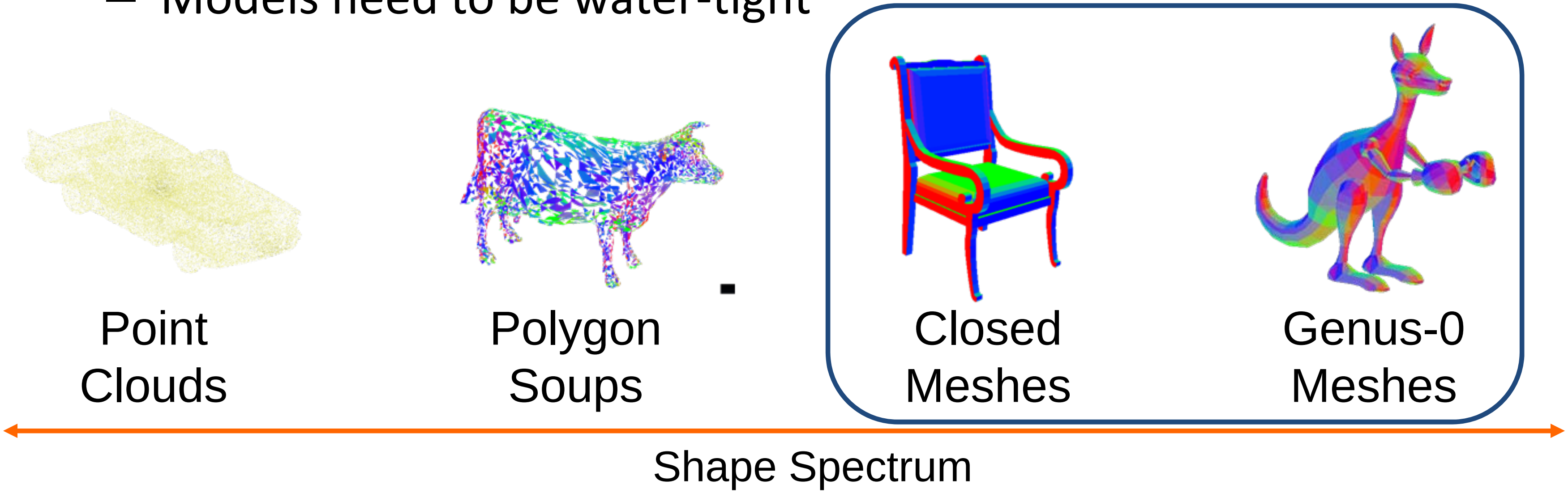
Intersection

Model

Voxel Representation

Volumetric Representations

- Properties:
  - Invertible
  - 3D array of information
  - Comparison gives the measure of overlap
- Limitations:
  - Models need to be water-tight



Outline

- Why shape descriptors?
- How do we represent shapes?
  - Volumetric Representations
  - Surface Representations
    - Spherical Parameterization
    - Extended Gaussian Image
    - Shape Histograms (Sectors + Shells)
    - Gaussian EDT
  - View-Based Representations
- Conclusion

Spherical Parameterization

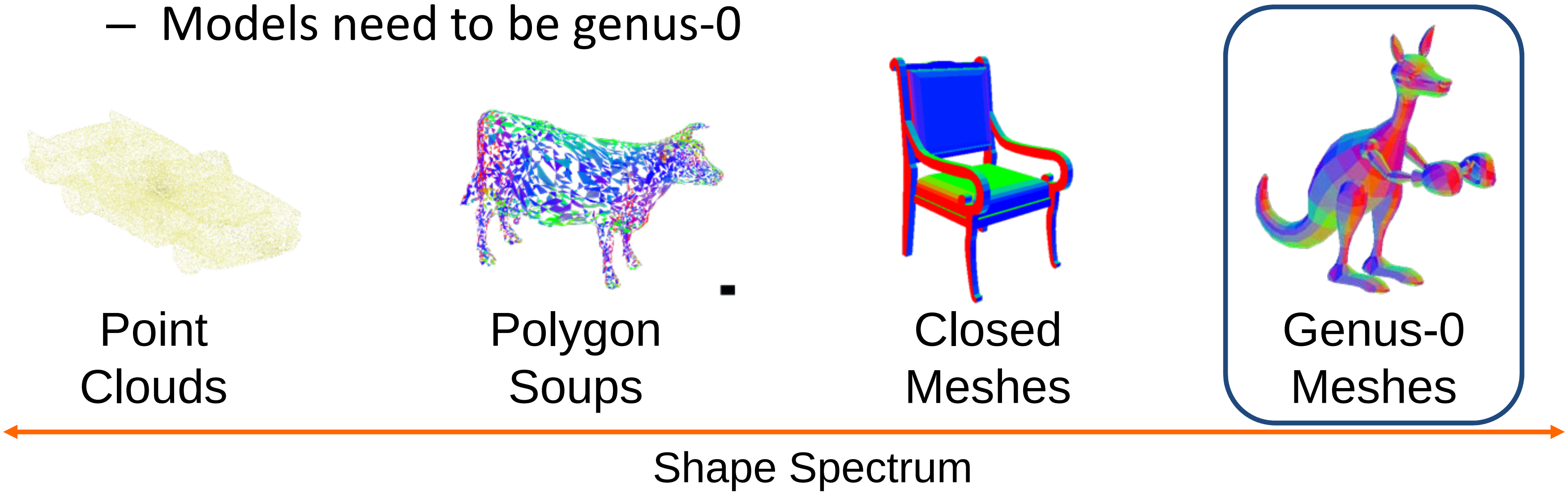
- Create a 1-to-1 mapping between points on the surface of the model and points on the surface of the sphere.
- Compare two models by comparing the distances between two points on the models that map to the same point on the sphere

Model

Spherical Parameterization

Spherical Parameterization

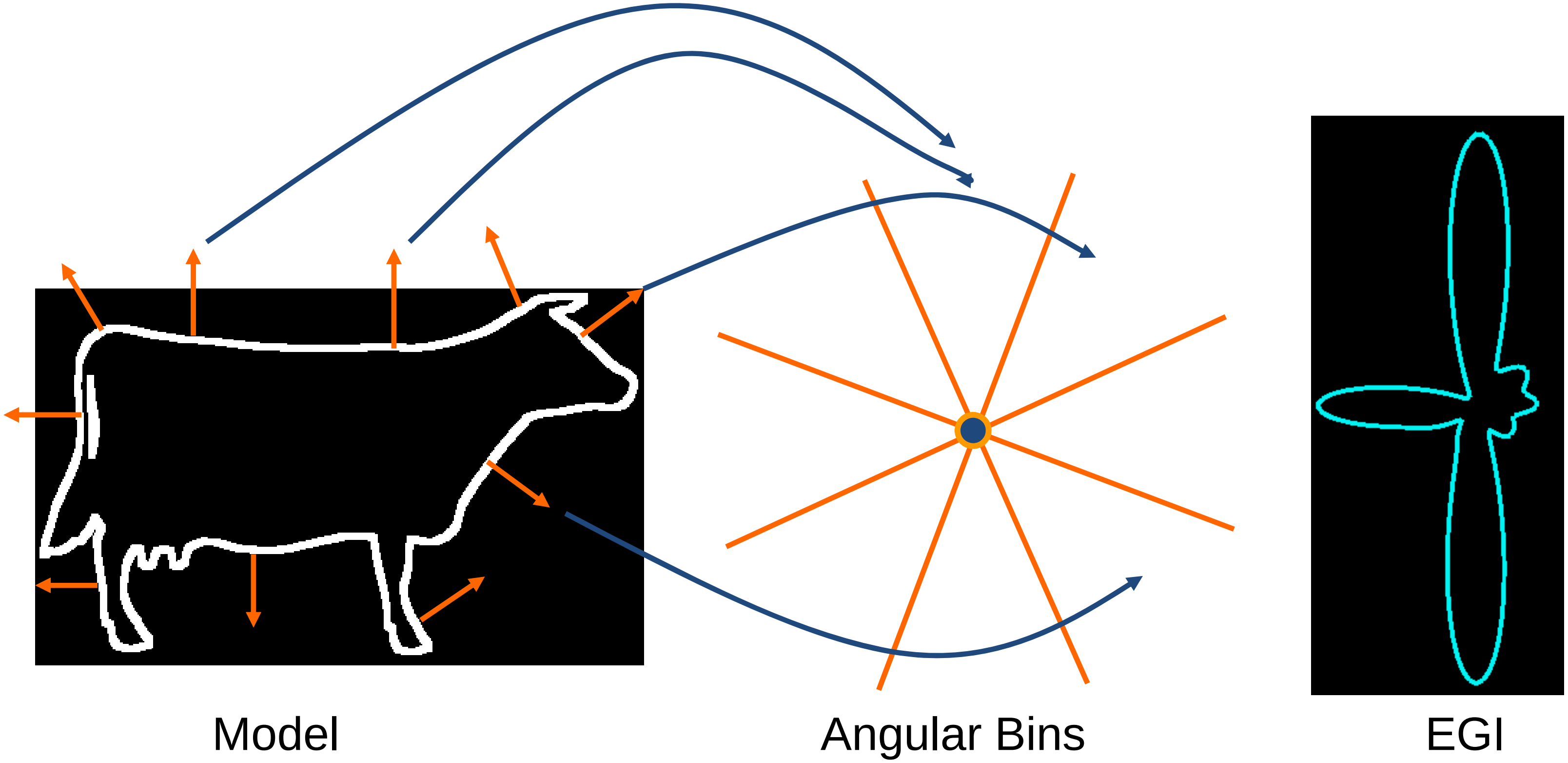
- Properties:
  - Invertible
  - 2D array of information
  - Comparison gives the distance between surfaces
- Limitations:
  - Models need to be genus-0



Extended Gaussian Image

[Horn, 1984]

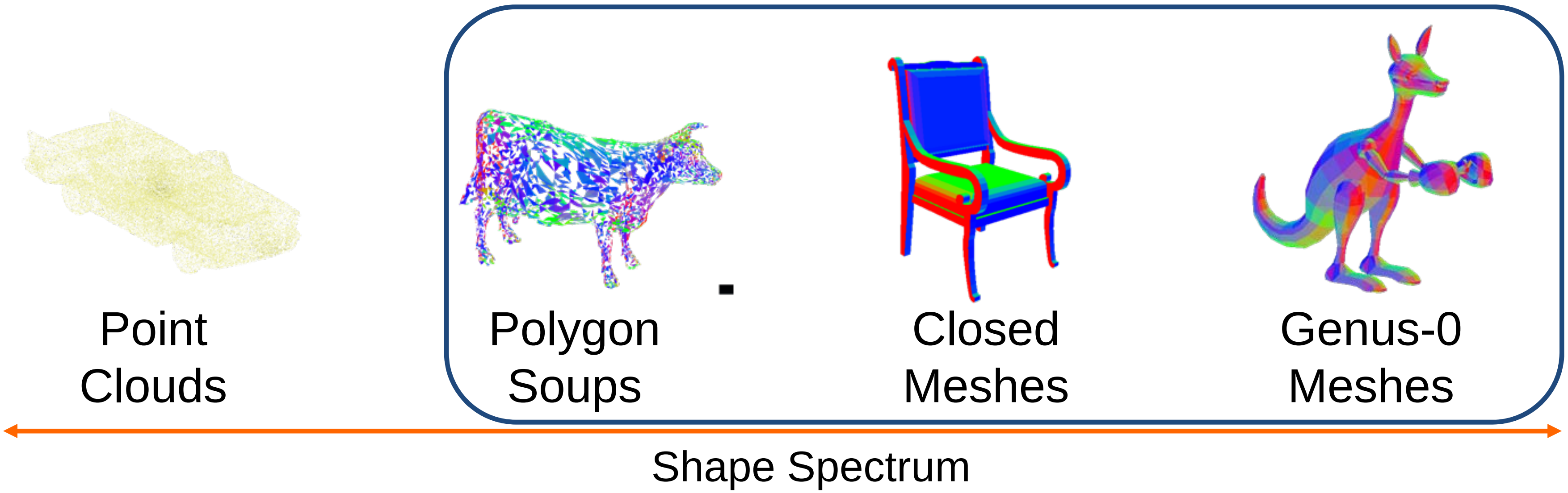
- Represent a model by a spherical function by binning surface normals



Extended Gaussian Image

[Horn, 1984]

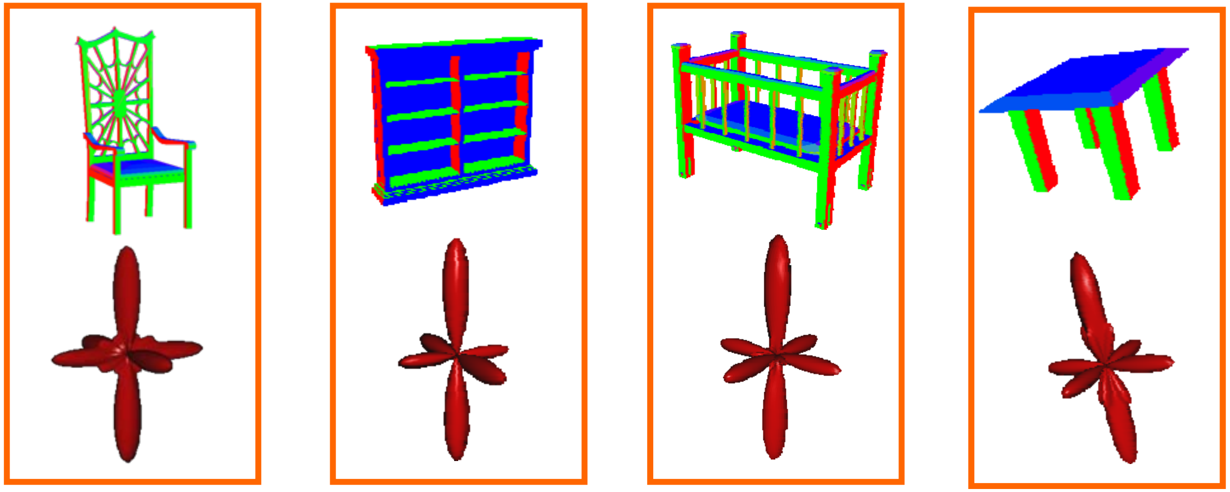
- Properties:
  - Invertible for convex shapes
  - 2D array of information
  - Can be defined for most models



Extended Gaussian Image

[Horn, 1984]

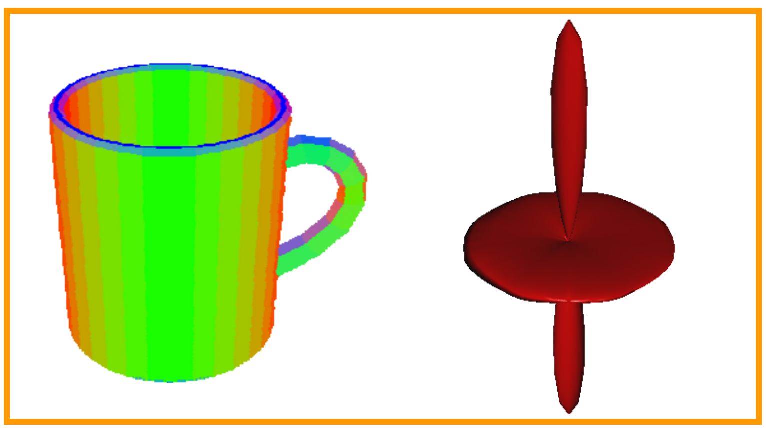
- Properties:
  - Invertible for convex shapes
  - 2D array of information
  - Can be defined for most models
- Limitations:
  - Too much information is lost
  - Normals are sensitive to noise



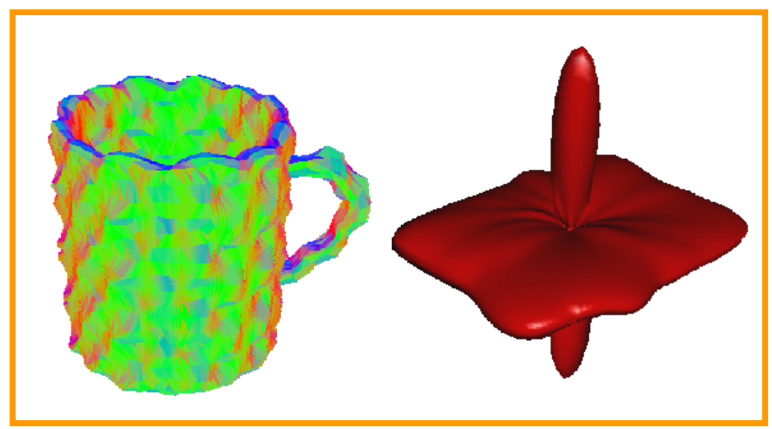
Extended Gaussian Image

[Horn, 1984]

- Properties:
  - Invertible for convex shapes
  - 2D array of information
  - Can be defined for most models
- Limitations:
  - Too much information is lost
  - Normals are sensitive to noise



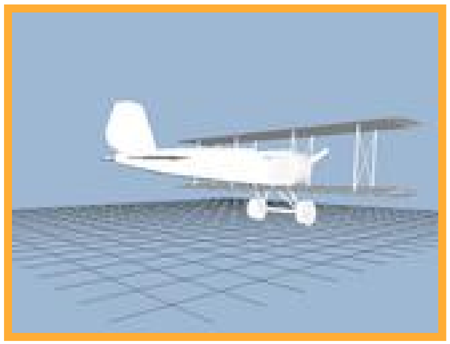
Initial Model

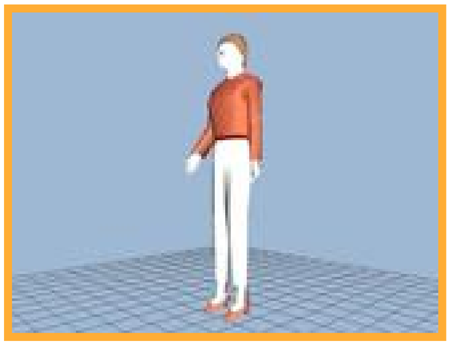


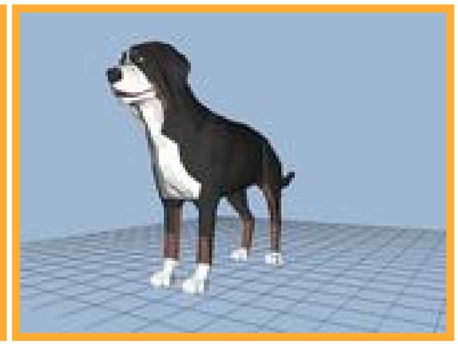
Noisy Model

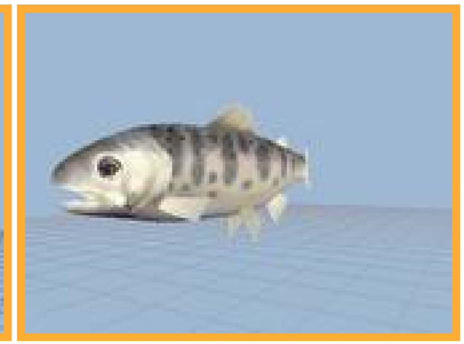
Retrieval Results

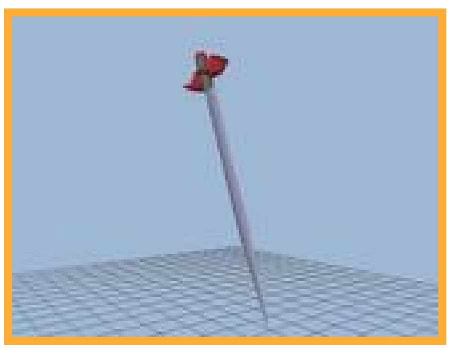
- Princeton Shape Benchmark
  - ~900 models, 90 classes

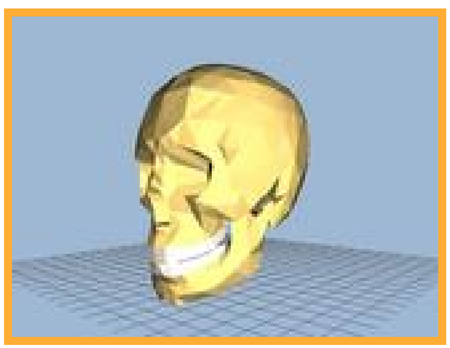

14 biplanes


50 human bipeds


7 dogs


17 fish


16 swords

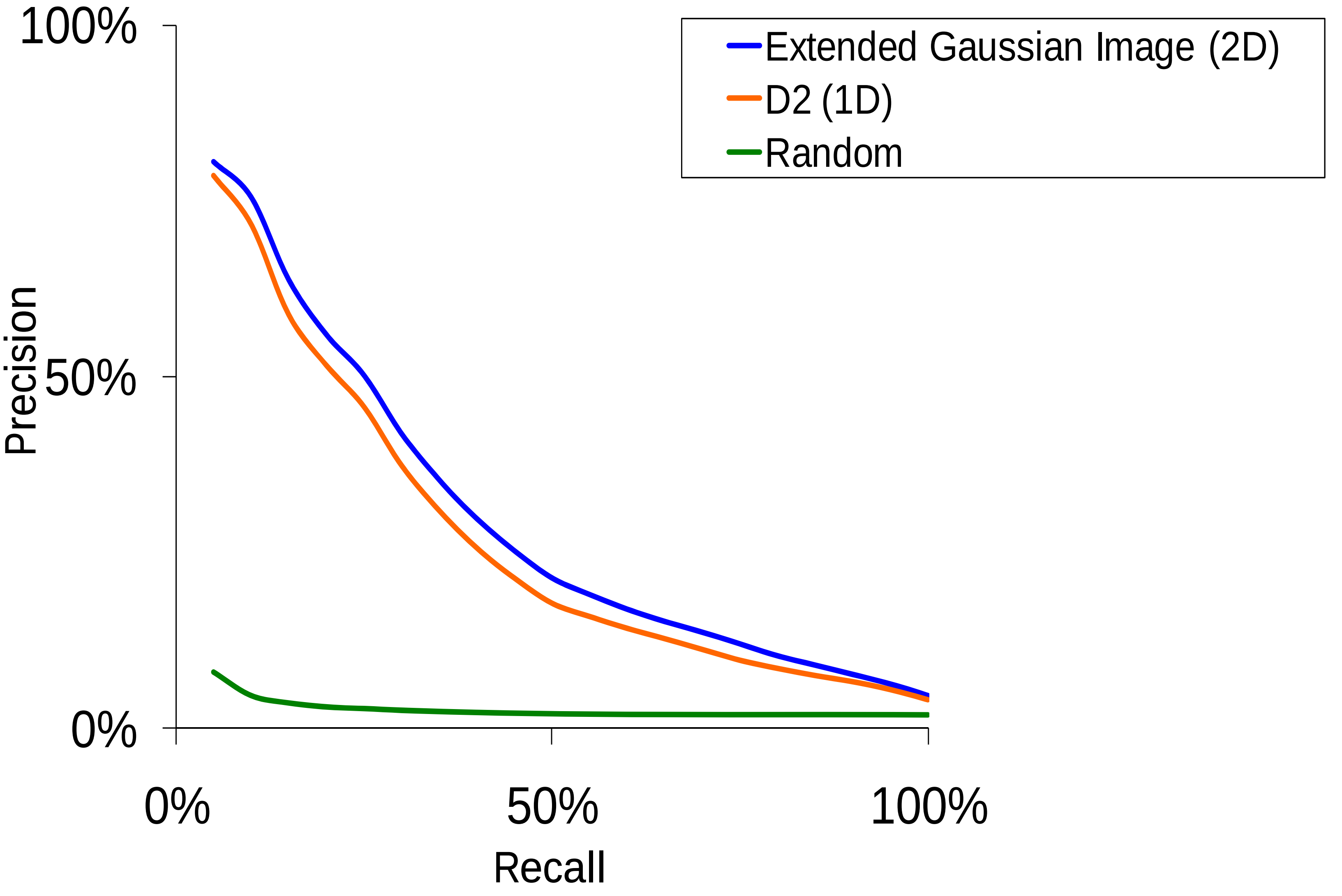

6 skulls


15 desk chairs


13 electric guitars

http://www.shape.cs.princeton.edu/benchmark/

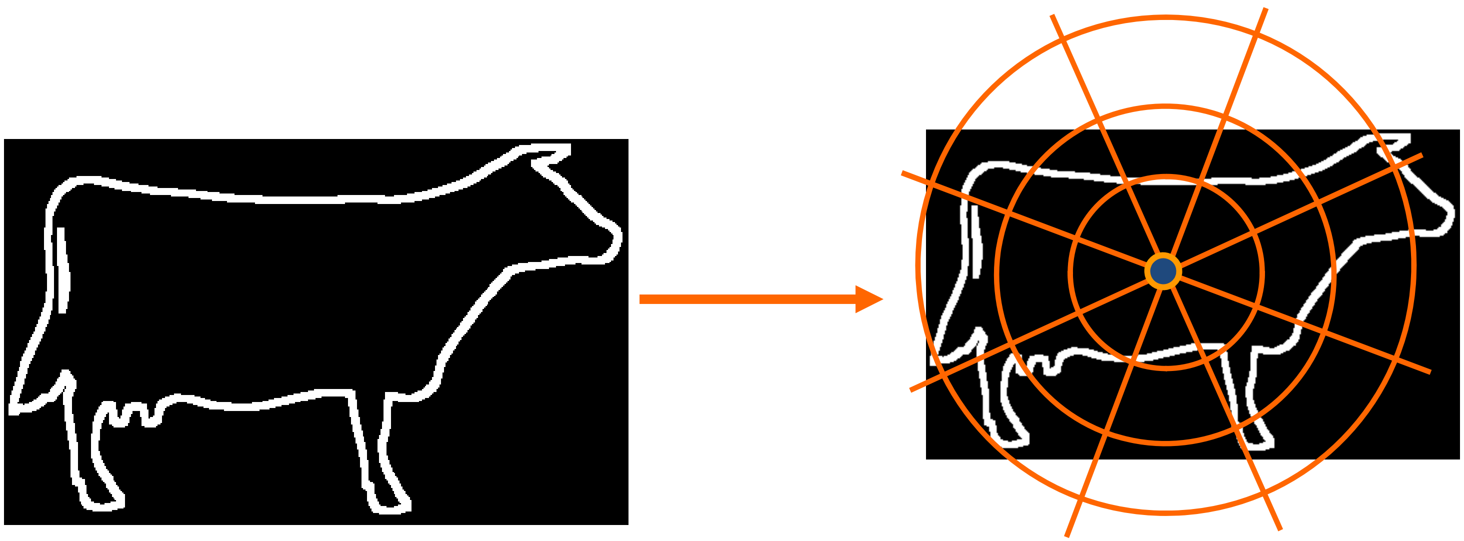
Retrieval Results



Shape Histograms

[Ankerst *et al.*, 1999]

- Shape descriptor stores a histogram of how much surface resides at different bins in space

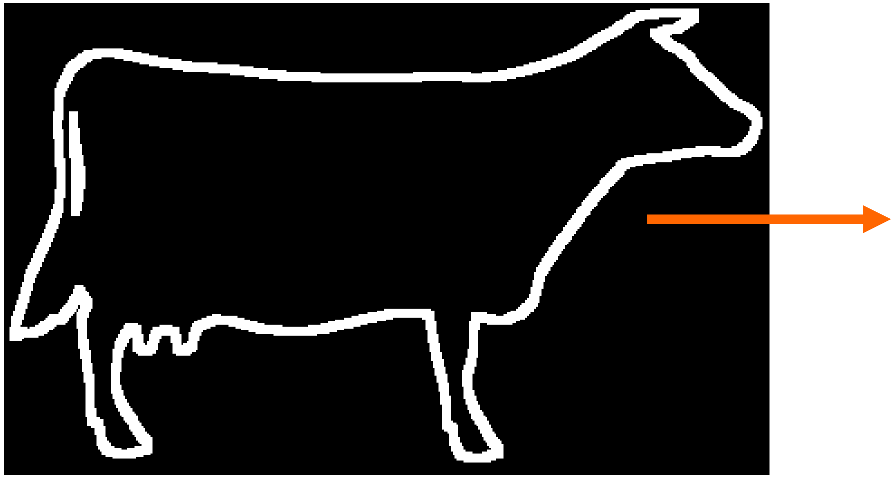


Model

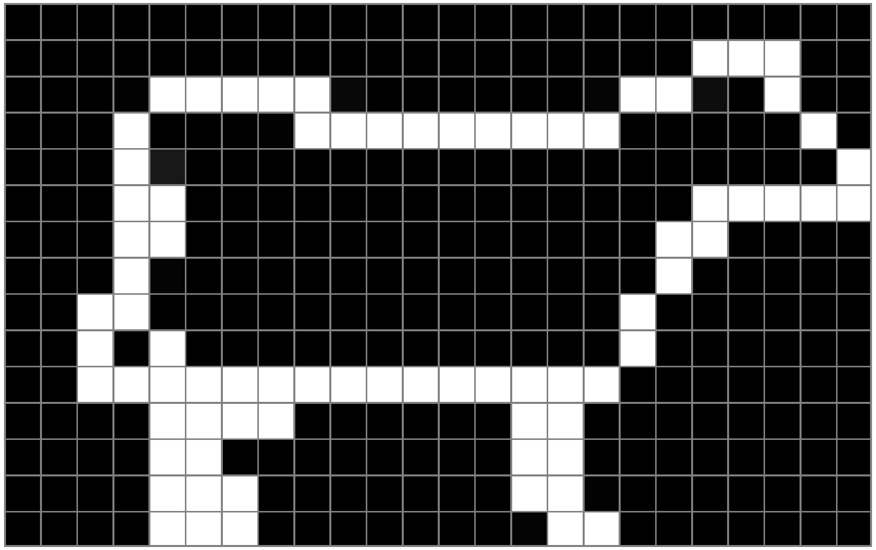
Shape Histogram
(Sectors + Shells)

Boundary Voxel Representation

- Represent a model as the (anti-aliased) rasterization of its surface into a regular grid:
  - A voxel has value 1 (or area of intersection) if it intersects the boundary
  - A voxel has value 0 if it doesn't intersect



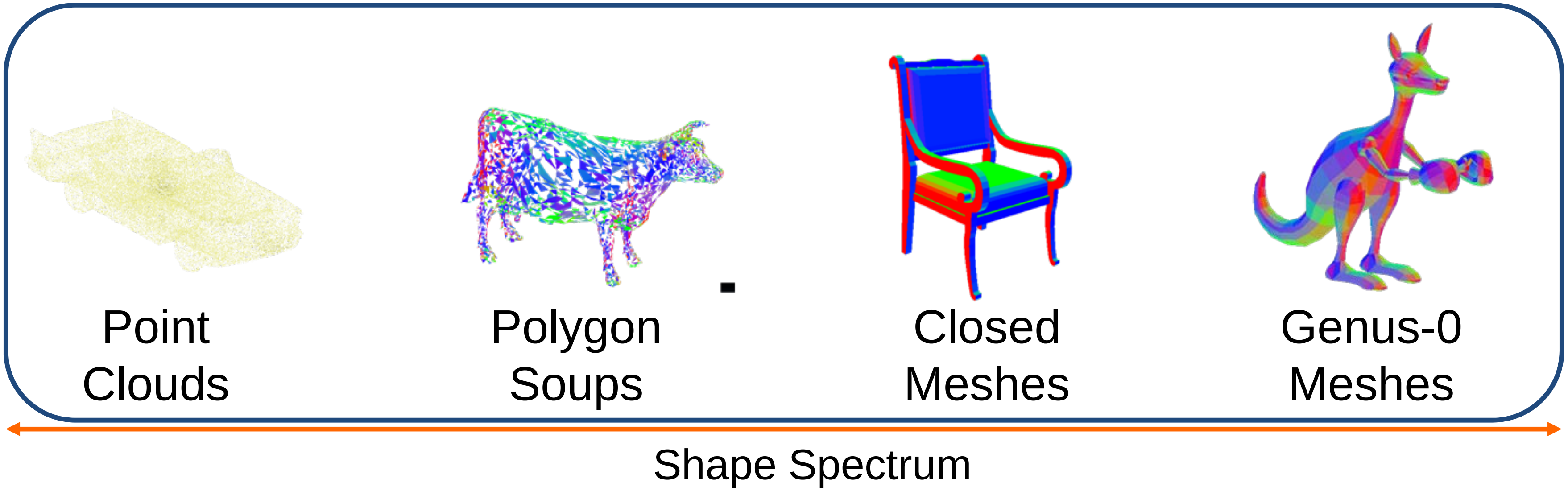
Model



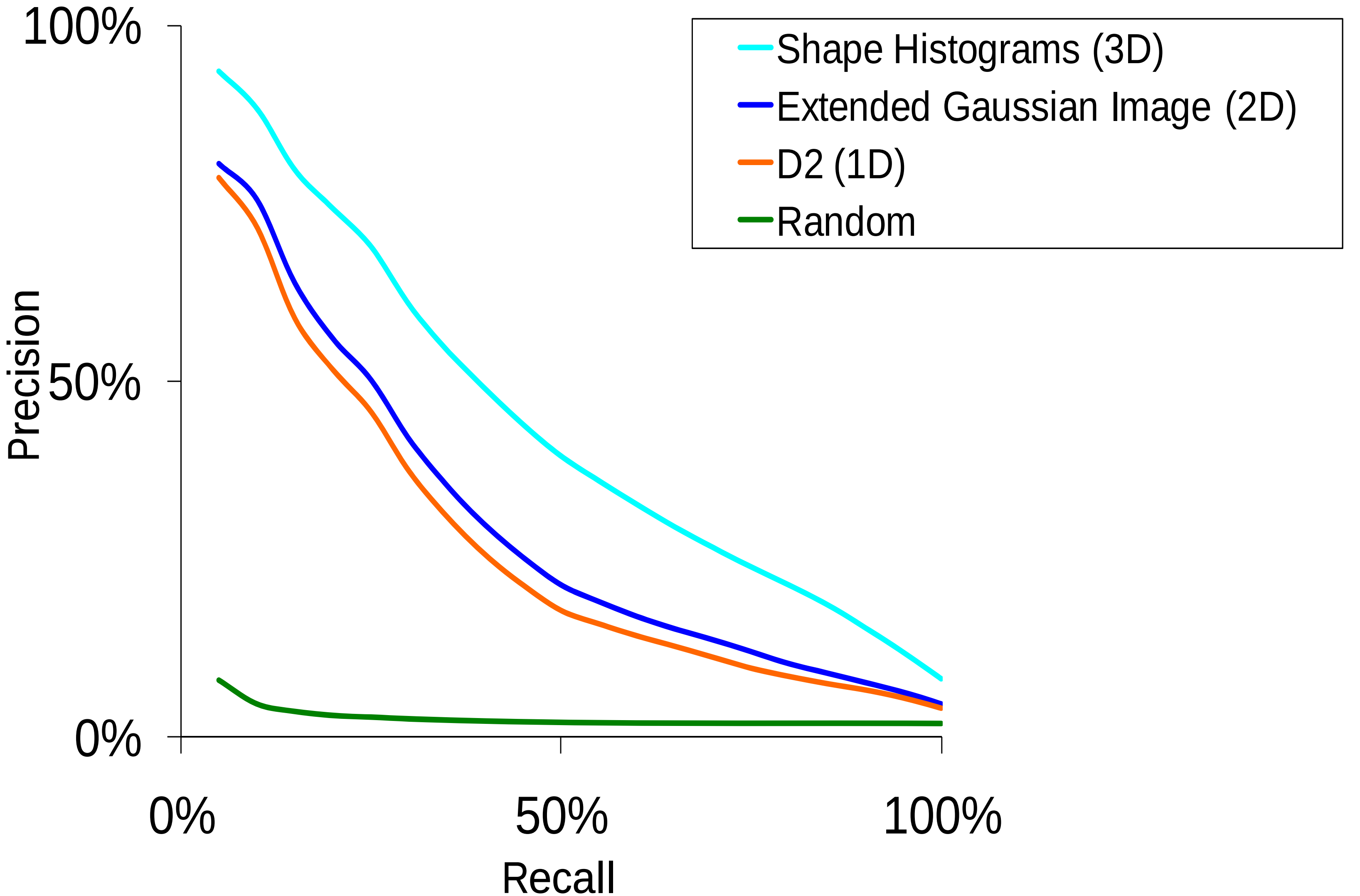
Voxel Grid

Boundary Voxel Representation

- Properties:
  - Invertible
  - 3D array of information
  - Can be defined for any model

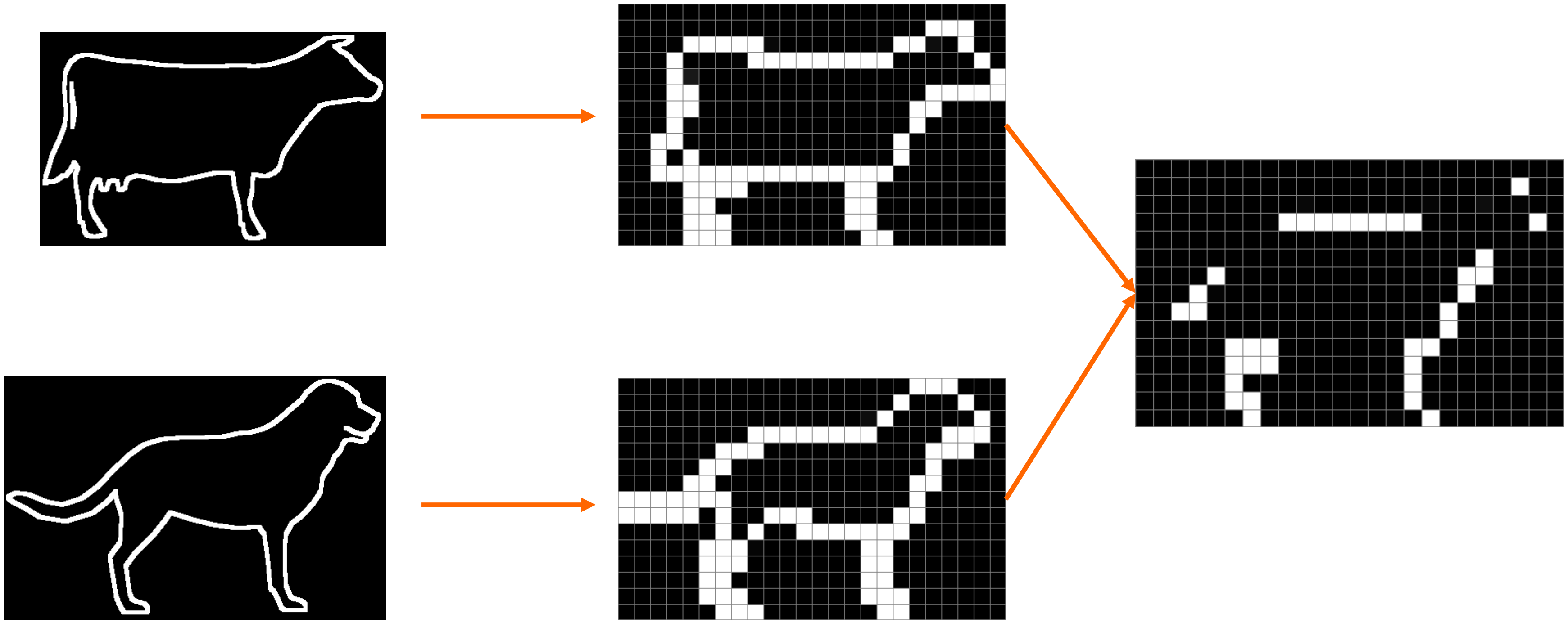


Retrieval Results



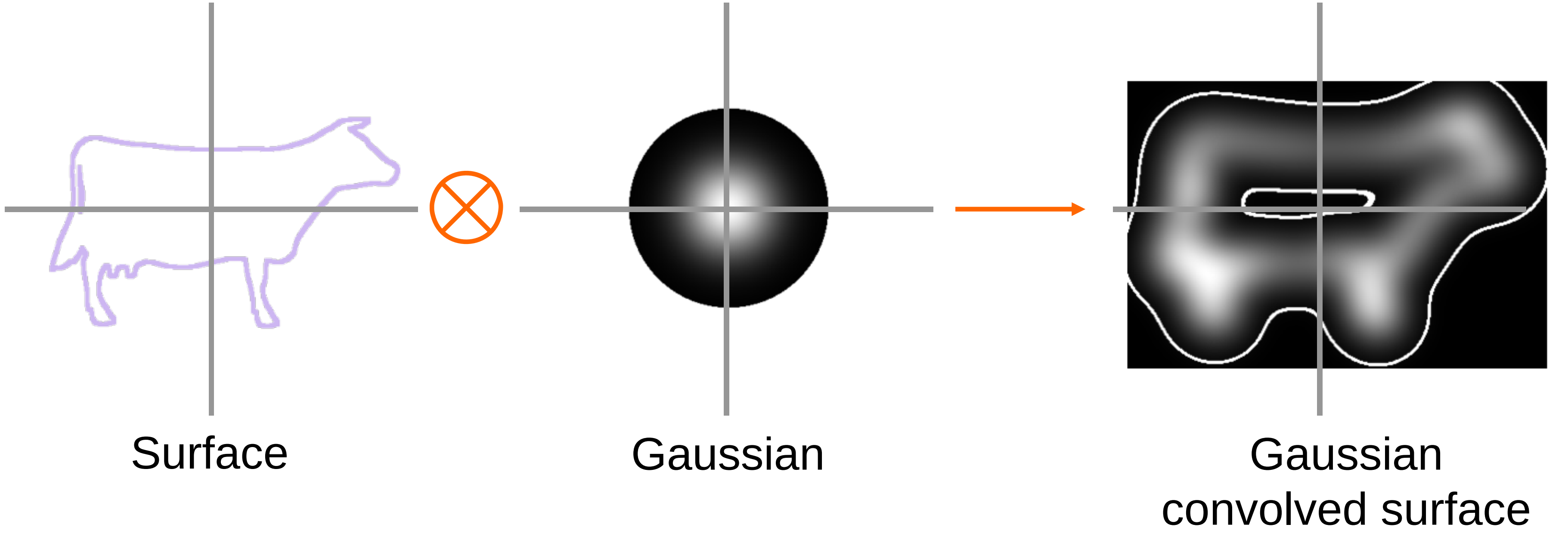
Histogram Representations

- Challenge:
  - Histogram comparisons measure overlap, not proximity.



Convolving with a Gaussian

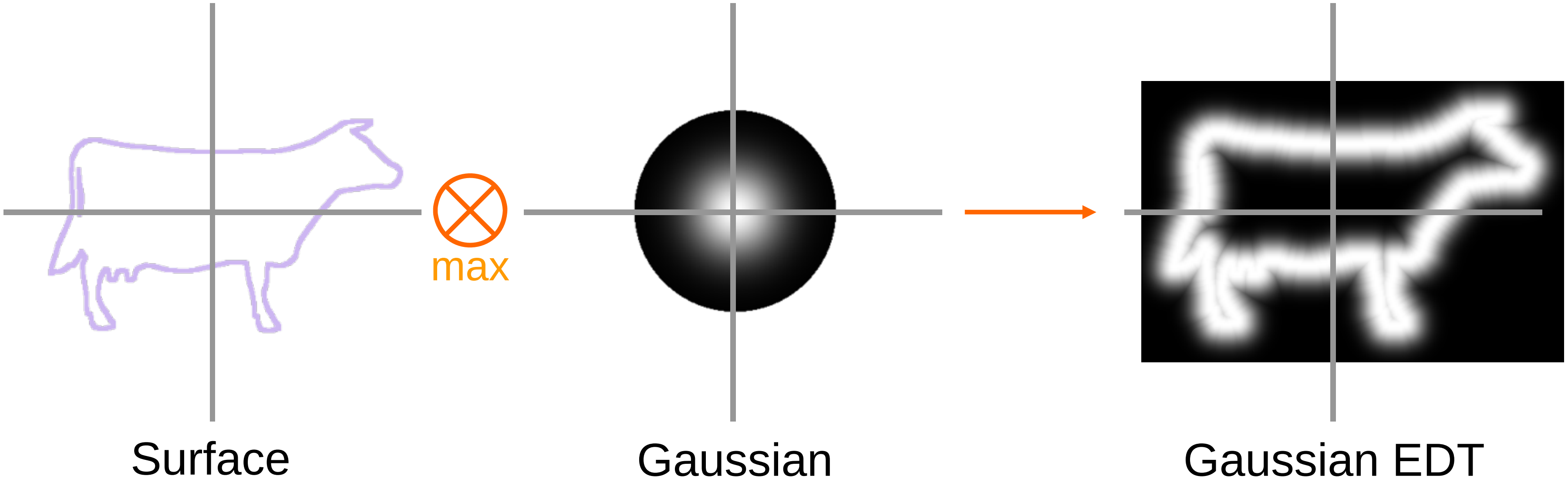
- The value at a point is obtained by summing Gaussians distributed over the surface of the model.
  - ✓ Distributes the surface into adjacent bins
  - ✗ Blurs the model, loses high frequency information



Gaussian EDT

[Kazhdan *et al.*, 2003]

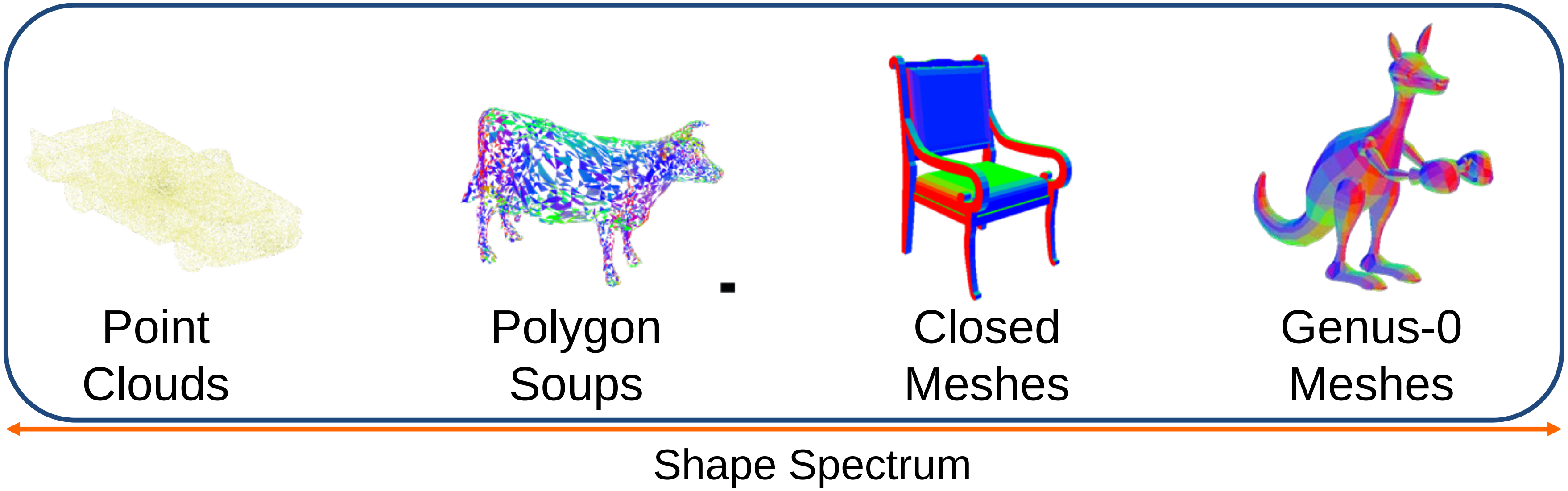
- The value at a point is obtained by summing the Gaussian of the closest point on the model surface.
  - ✓ Distributes the surface into adjacent bins
  - ✓ Maintains high-frequency information



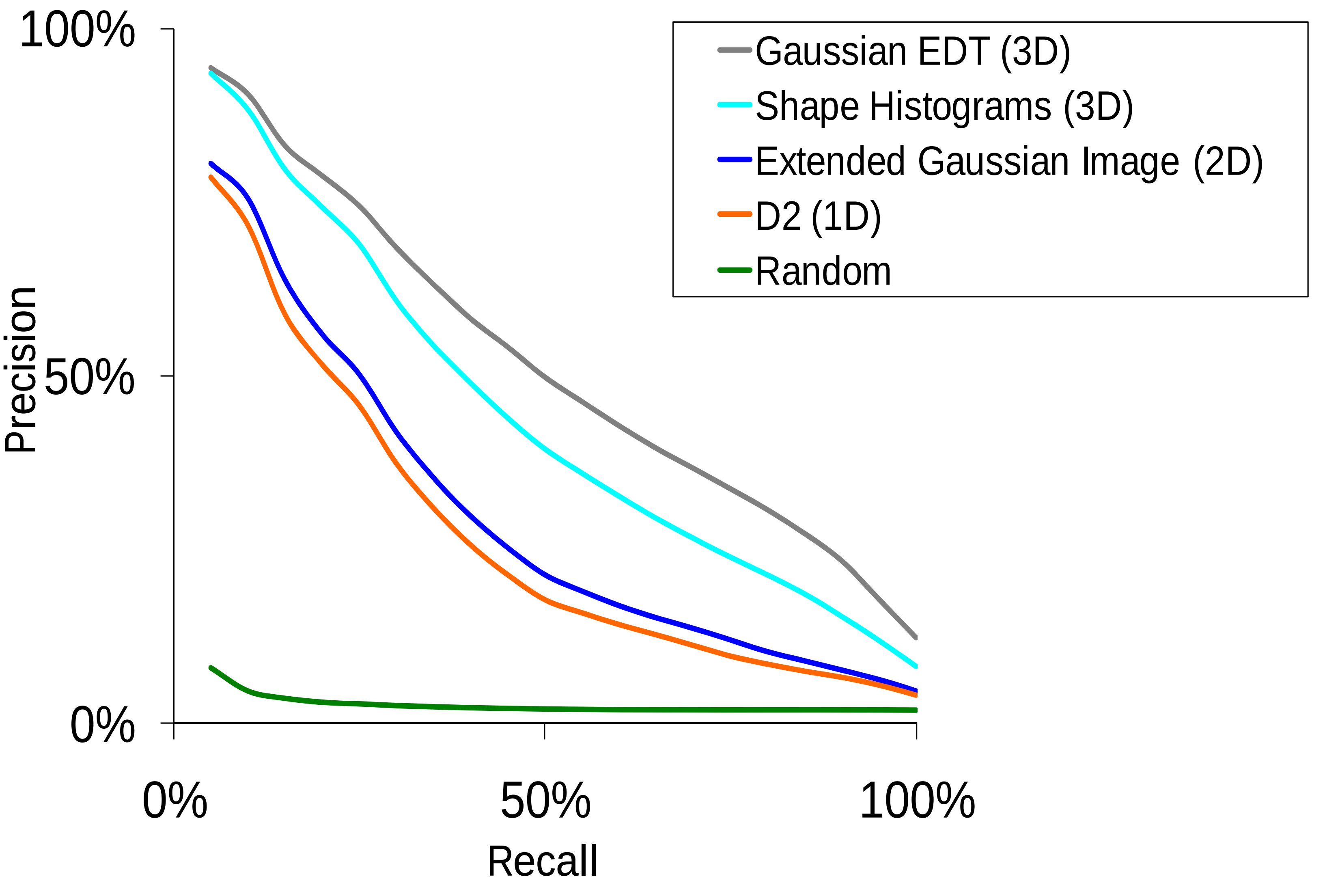
Gaussian EDT

[Kazhdan *et al.*, 2003]

- Properties:
  - Invertible
  - 3D array of information
  - Can be defined for any model
  - Difference measures proximity between surfaces



Retrieval Results



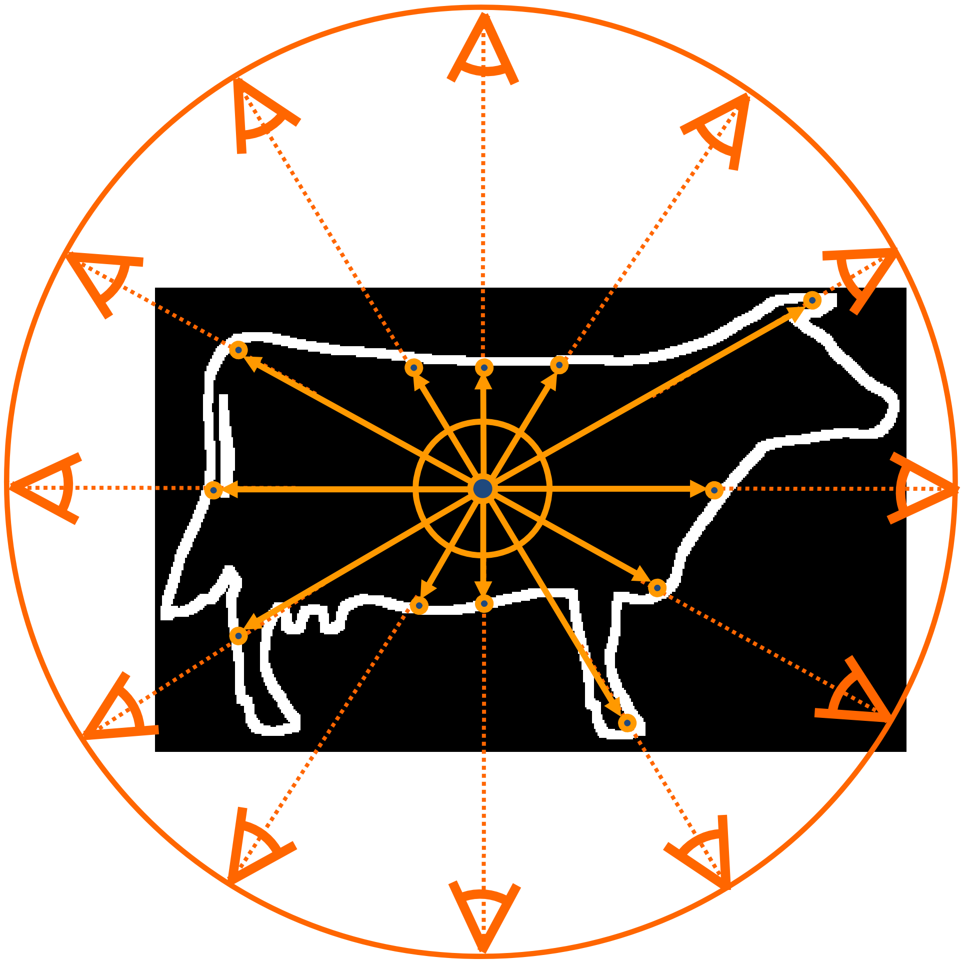
Outline

- Why shape descriptors?
- How do we represent shapes?
  - Volumetric Representations
  - Surface Representations
  - View-Based Representations
    - Spherical Extent Function
    - Light Field Descriptor
- Conclusion

Spherical Extent Function

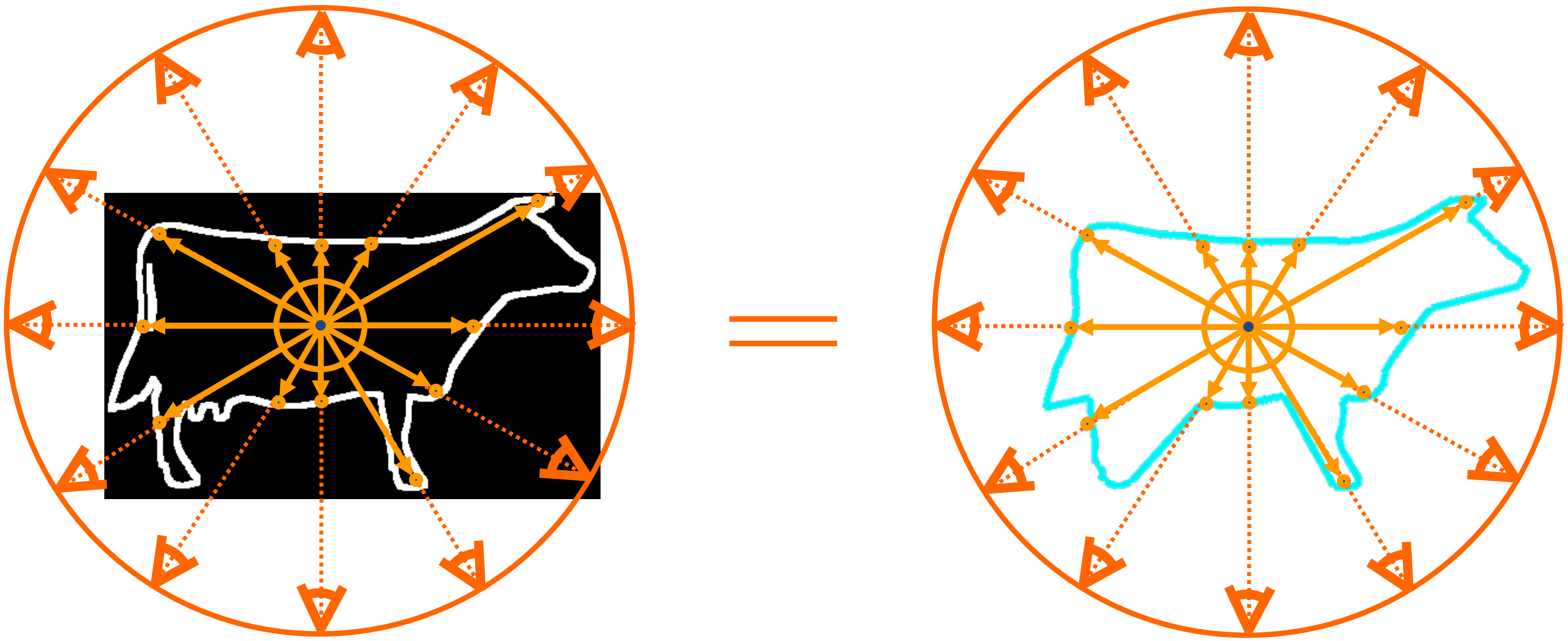
[Vranic *et al.* 2002]

- For every view direction, store the distance to the first point a viewer would see when looking at the origin.



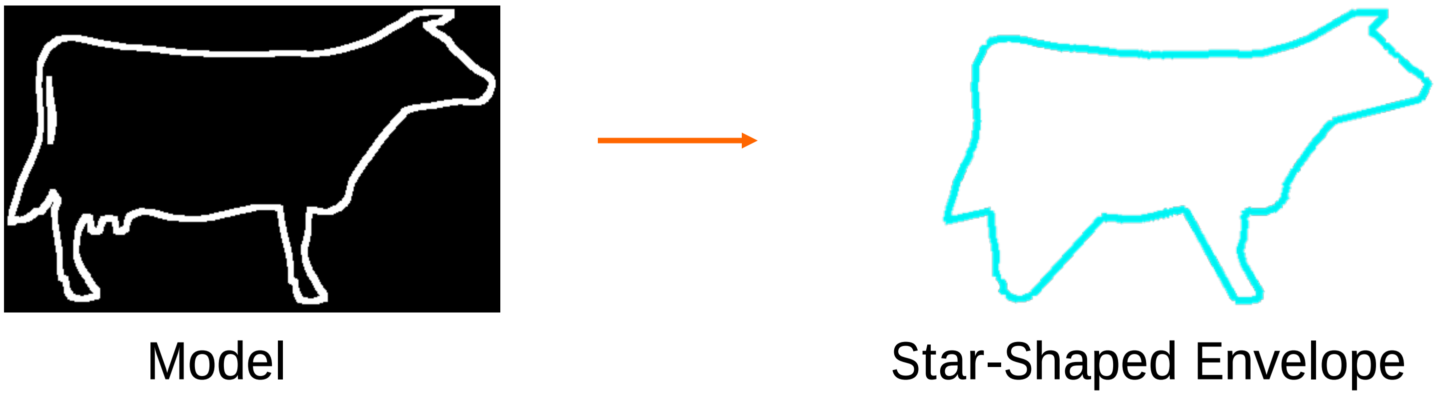
Spherical Extent Function

- A model is represented by its star-shaped envelope:
  - The minimal surface containing the model with the property that the center sees every point on the surface
  - Transforms arbitrary genus models to genus-0 surfaces



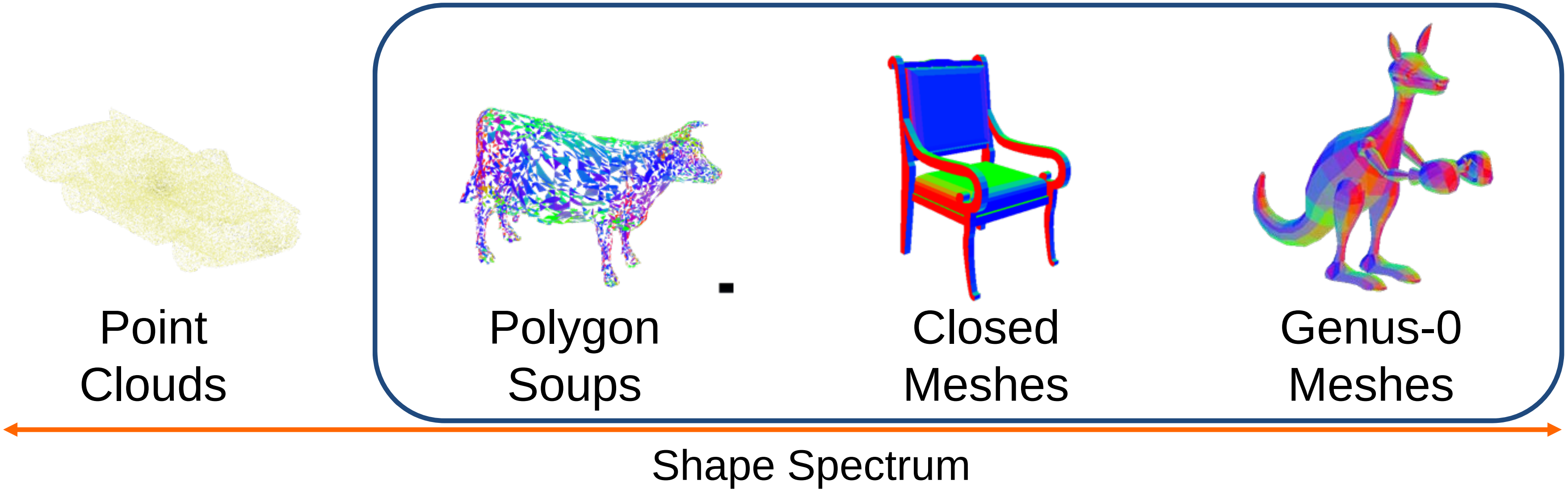
Spherical Extent Function

- A model is represented by its star-shaped envelope:
  - The minimal surface containing the model with the property that the center sees every point on the surface
  - Transforms arbitrary genus models to genus-0 surfaces



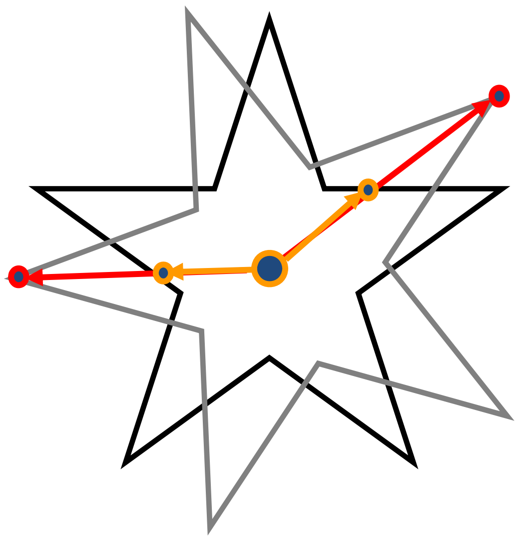
Spherical Extent Function

- Properties:
  - Invertible for star-shaped models
  - 2D array of information
  - Can be defined for most models

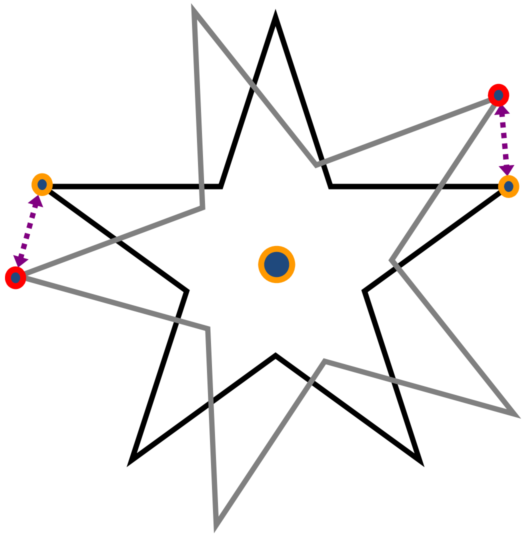


Spherical Extent Function

- Properties:
  - Can be defined for most models
  - Invertible for star-shaped models
  - 2D array of information
- Limitations:
  - Distance only measures angular proximity

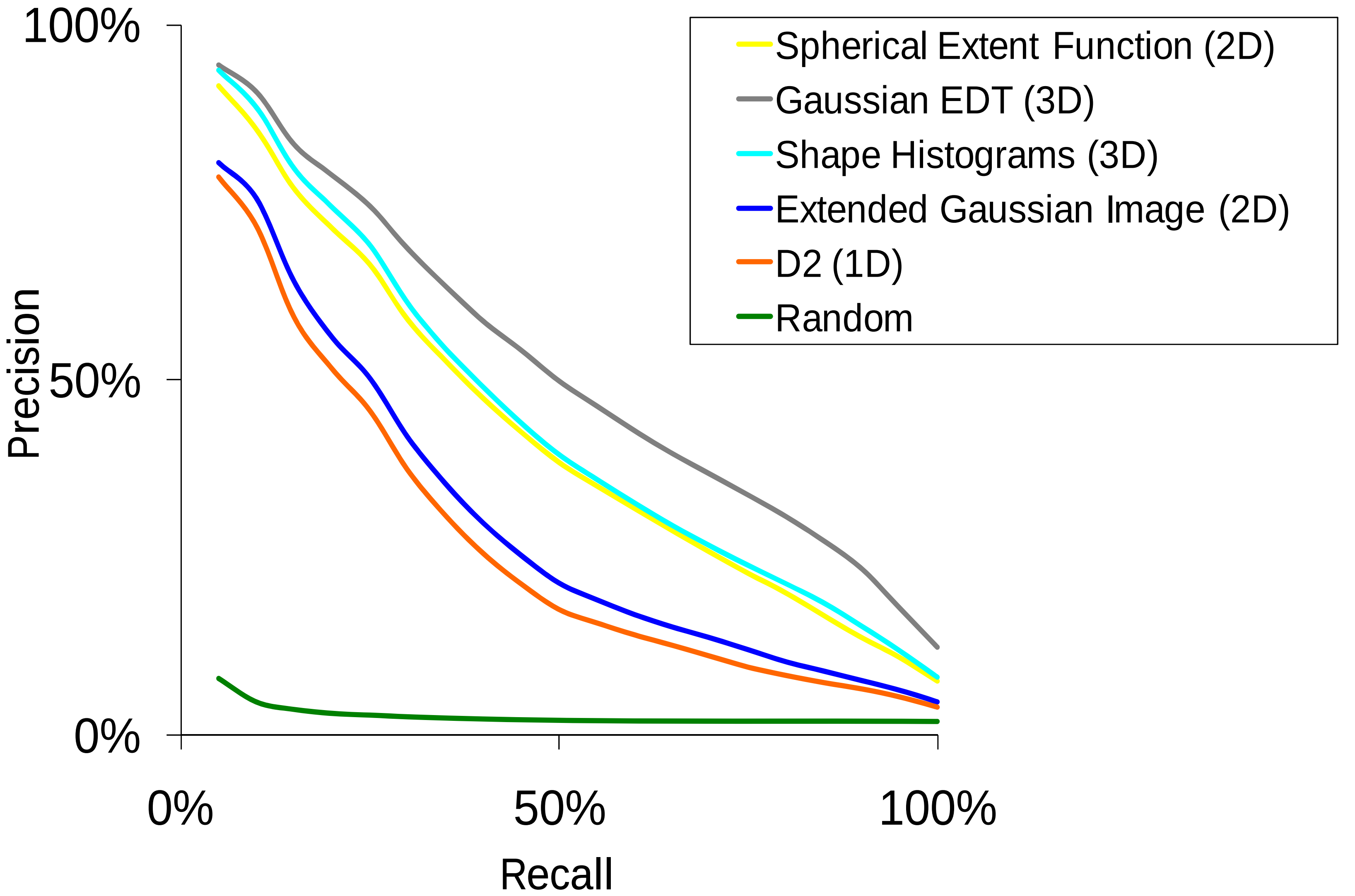


Spherical Extent Matching



Nearest Point Matching

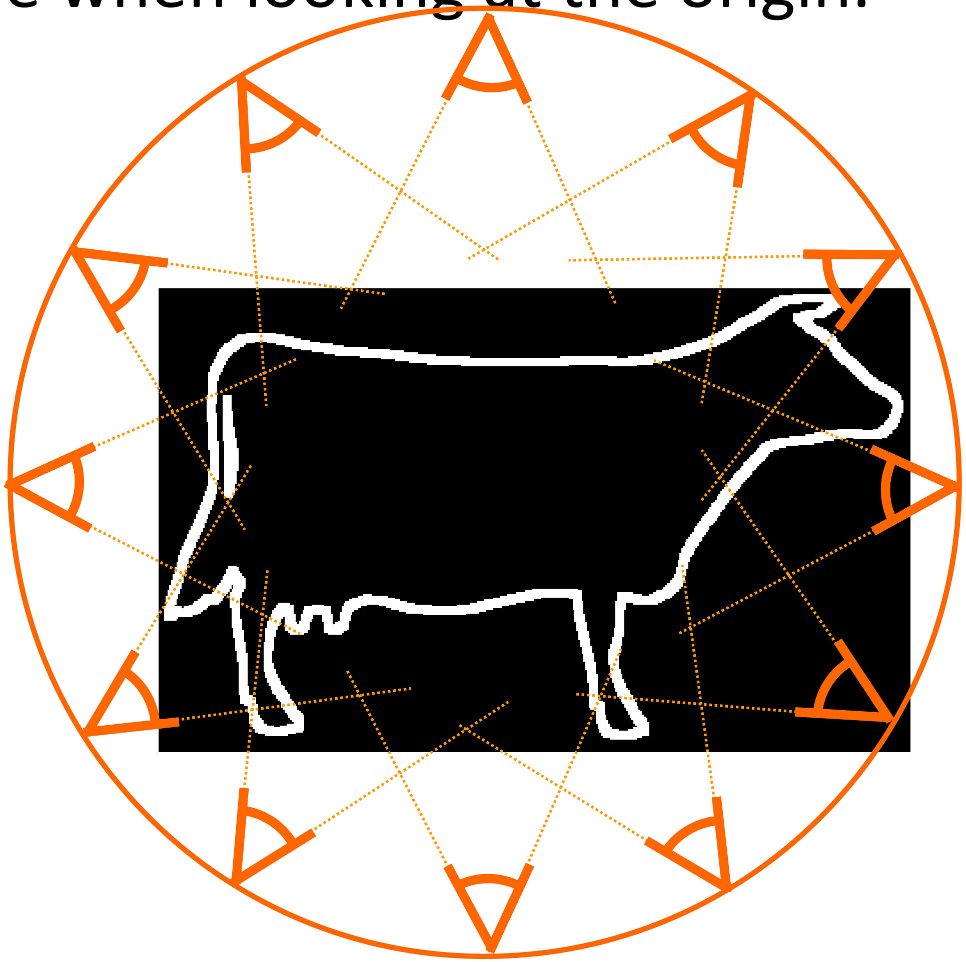
Retrieval Results



Light Field Descriptor

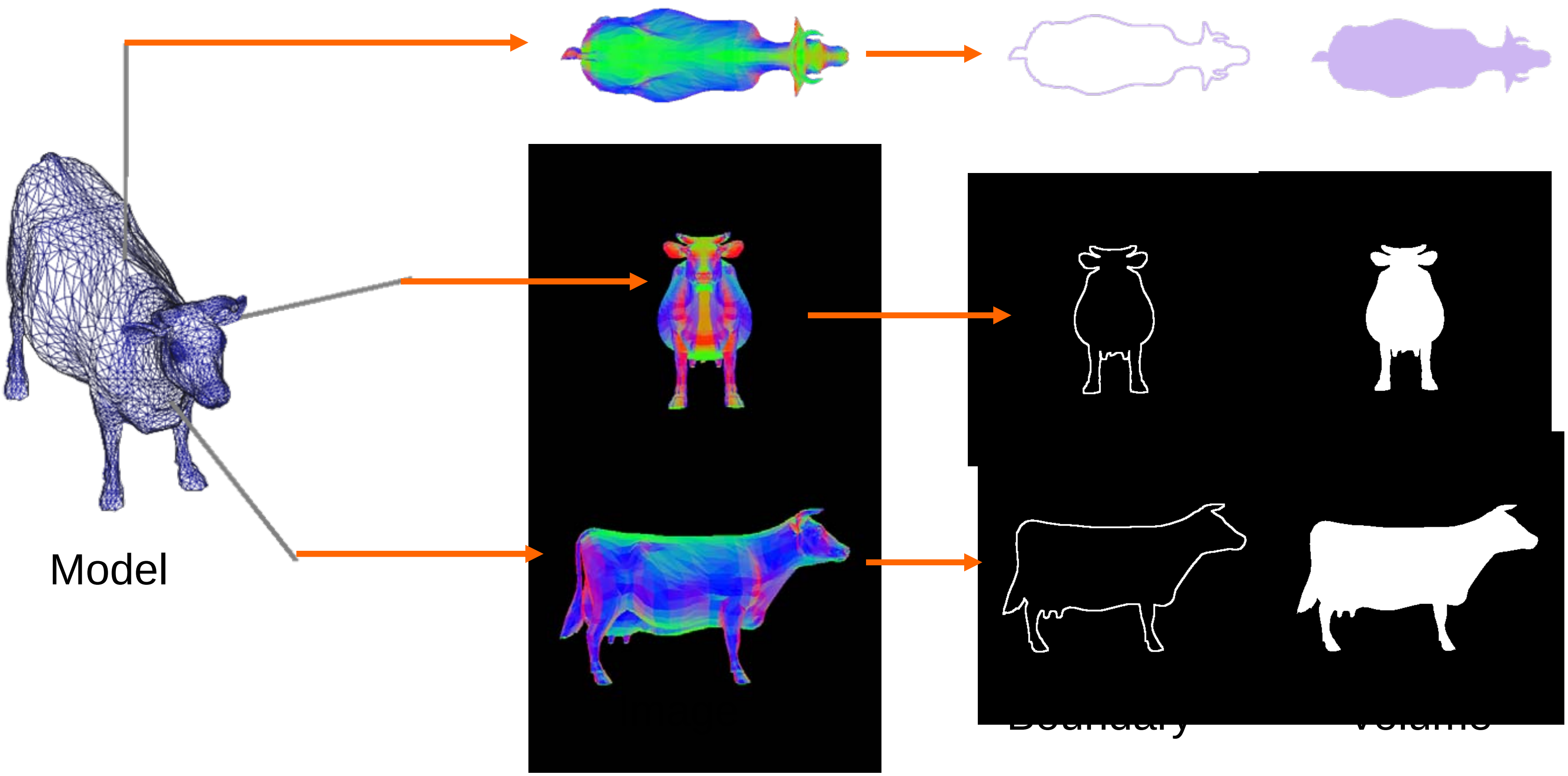
[Chen *et al.* 2003]

- For every view direction, store the image the viewer would see when looking at the origin.



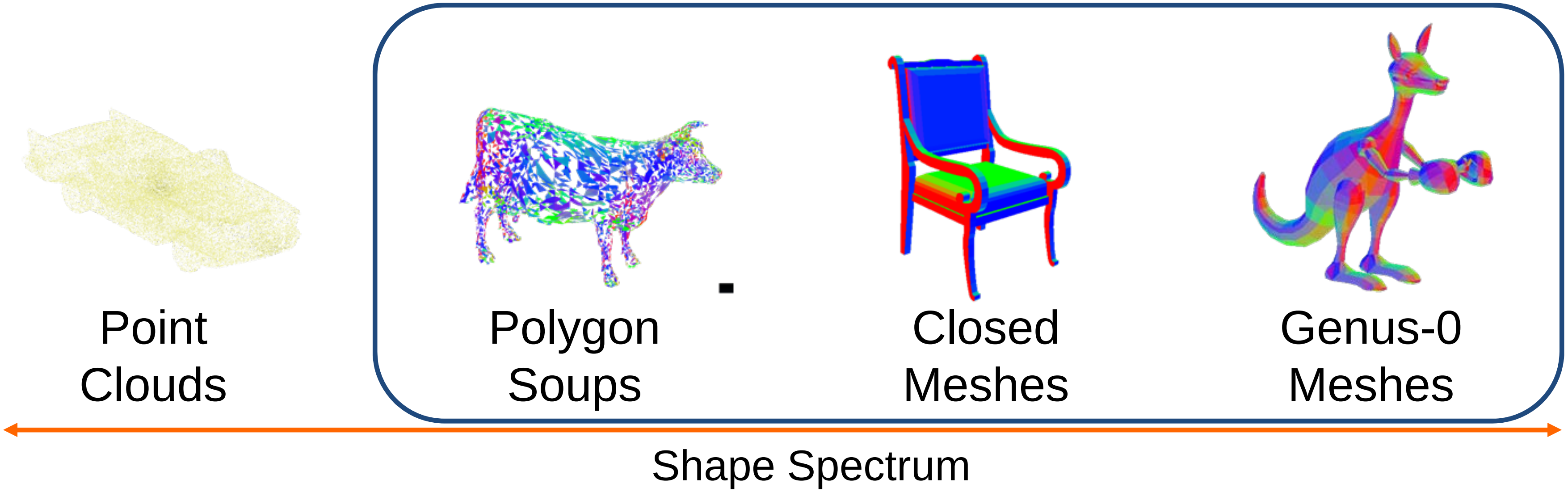
Light Field Descriptor

- Hybrid boundary/volume representation



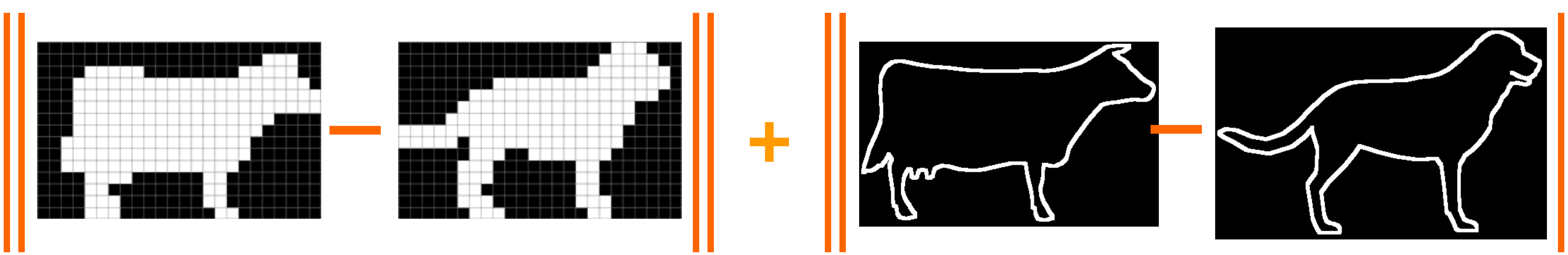
Light Field Descriptor

- Properties:
  - Represents the visual hull of the model
  - 4D array of information
  - Can be defined for most models



Light Field Descriptor

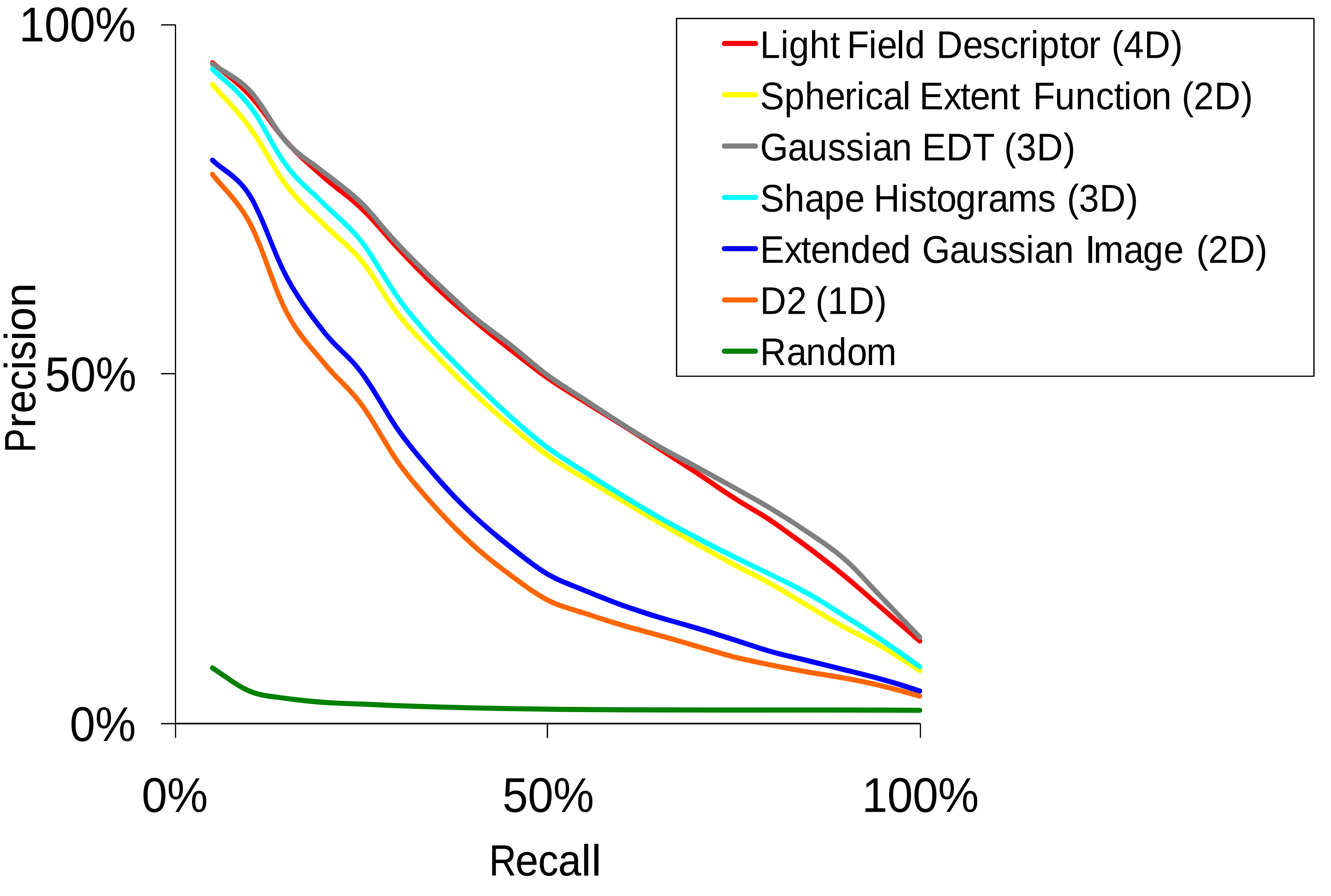
- Properties:
  - Can be defined for most models
  - Invertible for star-shaped models
  - 4D array of information
  - Similarity = sum of area and contour similarities
    - There is a well defined interior
    - Can parameterize contours in 2D



Area Comparison

Contour Comparison

Retrieval Results



高斯不变量描述子

[曹，2011]

- 离散计算

$$GCMI_n = (GCM_0)^{n-1}(GCM_n) = (GCM_0)^{n-1}\iint_S K^n \rho ds \approx (GCM_0)^{n-1}\rho\sum_{i=1}^{N} K_i^n \Delta S_i$$

- 描述子

$$\overrightarrow{GIV} = (GCMI_1, GCMI_2, \cdots, GCMI_n)$$

谱矩不变量

- 基于Diffusion距离

$$NSMId_m = (SMId_m)(SMId_0)^{\frac{m-2}{2}}$$

$$t \in \{[\frac{1}{\lambda_i}], [\frac{1}{\lambda_{i+1}}], [\frac{1}{\lambda_{i+2}}], \cdots, [\frac{1}{\lambda_N}]\}$$

- 基于Commute-time距离

$$NSMIc_m = \frac{SMIc_m}{SMIc_0}$$

- 基于Biharmonic距离

$$NSMIb_m = \frac{SMIb_m}{(MIDb_0)^{\frac{m}{2}+1}}$$

离散计算及描述子

- M阶谱矩不变量离散形式

$$SMI_m = \iint\limits_S \iint\limits_S D(x,y)^m \rho(x)\rho(y) ds_y ds_x \approx \sum_{i=1}^{N}\sum_{j=1}^{N} D_{ij}^m \Delta s_{y_j} \Delta s_{x_i} \rho(x)\rho(y)$$

- 描述子

$$\overrightarrow{MIV} = (NSMI_1, NSMI_2, \cdots, NSMI_n)$$

特征函数矩不变量描述子

- 特征函数矩

$$EM_{(i_k)}^{(m_k)} = \iint\limits_M \prod_{k=1}^{K} (\phi_{i_k}(x))^{m_k} ds \approx \sum_{j=0}^{n-1} \prod_{k=1}^{K} (v_{i_k j})^{m_k} \Delta s_j$$

- 特征函数矩不变量

$$EMI_{(i_k)}^{(m_k)} = S^{\frac{1}{2}\sum_{k=1}^{K} m_k} \sum_{j=0}^{n-1} \prod_{k=1}^{K} (v_{i_k j})^{m_k} \Delta s_j$$

- 描述子

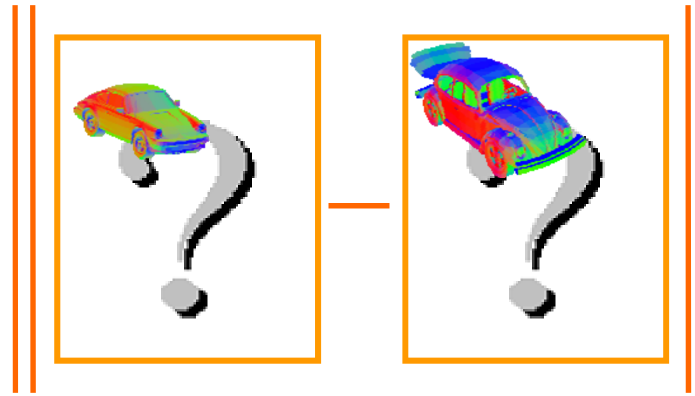
$$\overrightarrow{EIV} = (EMI_1, EMI_2, EMI_3, \cdots, EMI_n)$$

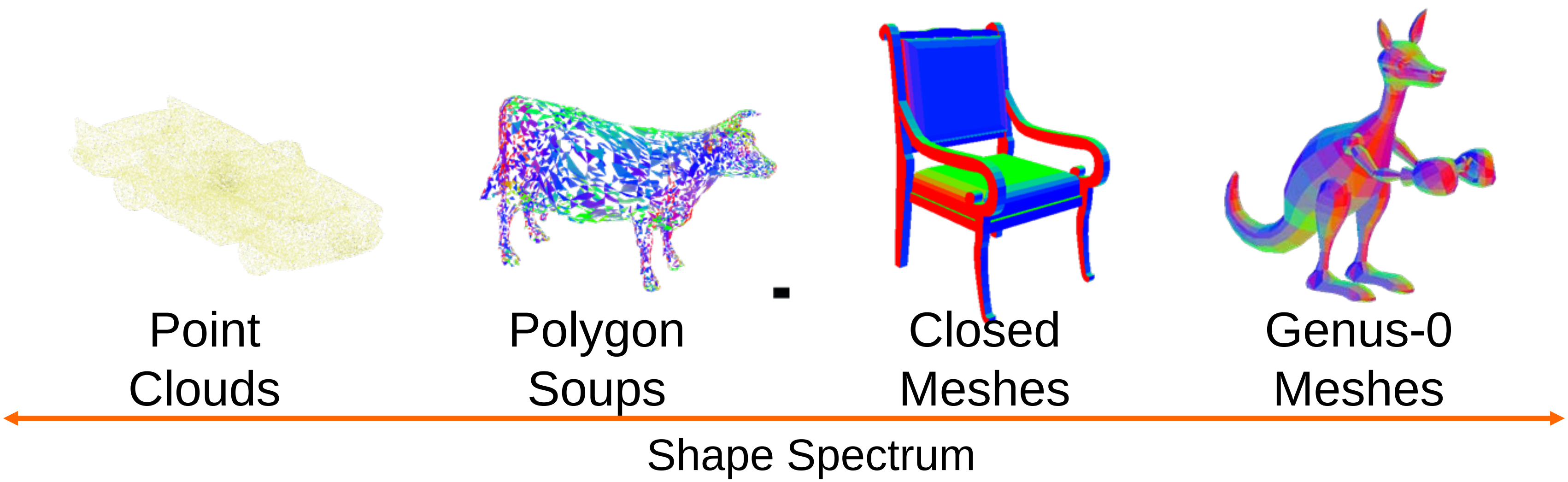
Conclusion

- Extended Gaussian Image
  - Differential properties are not always stable
- Gaussian Euclidean Distance Transform
  - Distributes surface across space without blurring
- Spherical Extent Function
  - Represents arbitrary genus shape by a genus-0 model
- Light Field Descriptors
  - 2D matching allows for volumetric comparisons and silhouette parameterizations

Conclusion

- In designing a shape descriptor, you want to consider:
  - What kind of models can be represented?
  - What kind of shape metric is defined?





Discussion