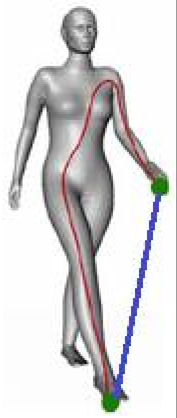
CENG 789 – Digital Geometry Processing

05- Mesh Comparison
(Distance, Descriptor and Sampling)

Prof. Dr. Yusuf Sahillioğlu

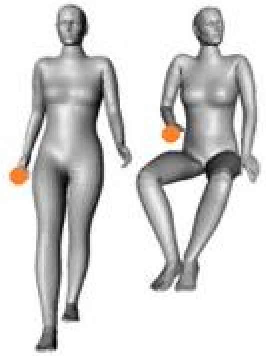
Computer Eng. Dept, MIDDLE EAST TECHNICAL UNIVERSITY, Turkey

Distances



- ✓ Euclidean vs. Geodesic distances.
- ✓ Euclidean geometry: Distance between 2 points is a line.
- ✓ Non-Euclidean geom: Distance between 2 points is a curvy path along the object surface.

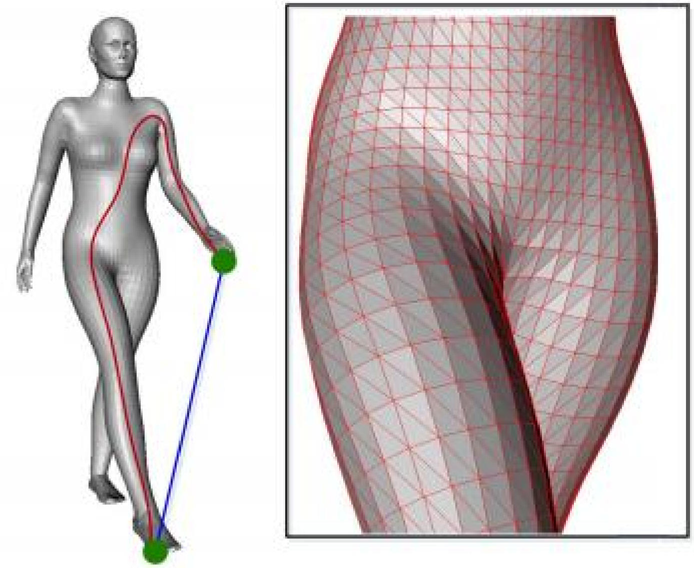
- ✓ Geometry of the objects that we deal with is non-Euclidean.
- ✓ Same intrinsically, different extrinsically:



- ✓ We use intrinsic geometry to compare shapes as the understanding of a finger does not change when you bend your arm.

Geodesic Distance

- ✓ Intrinsic geometry is defined by geodesic distances: length of the shortest (curvy) path (of edges) between two points.



Geodesic Distance

- ✓ Compute with Dijkstra's shortest path algorithm since the mesh is an undirected graph.

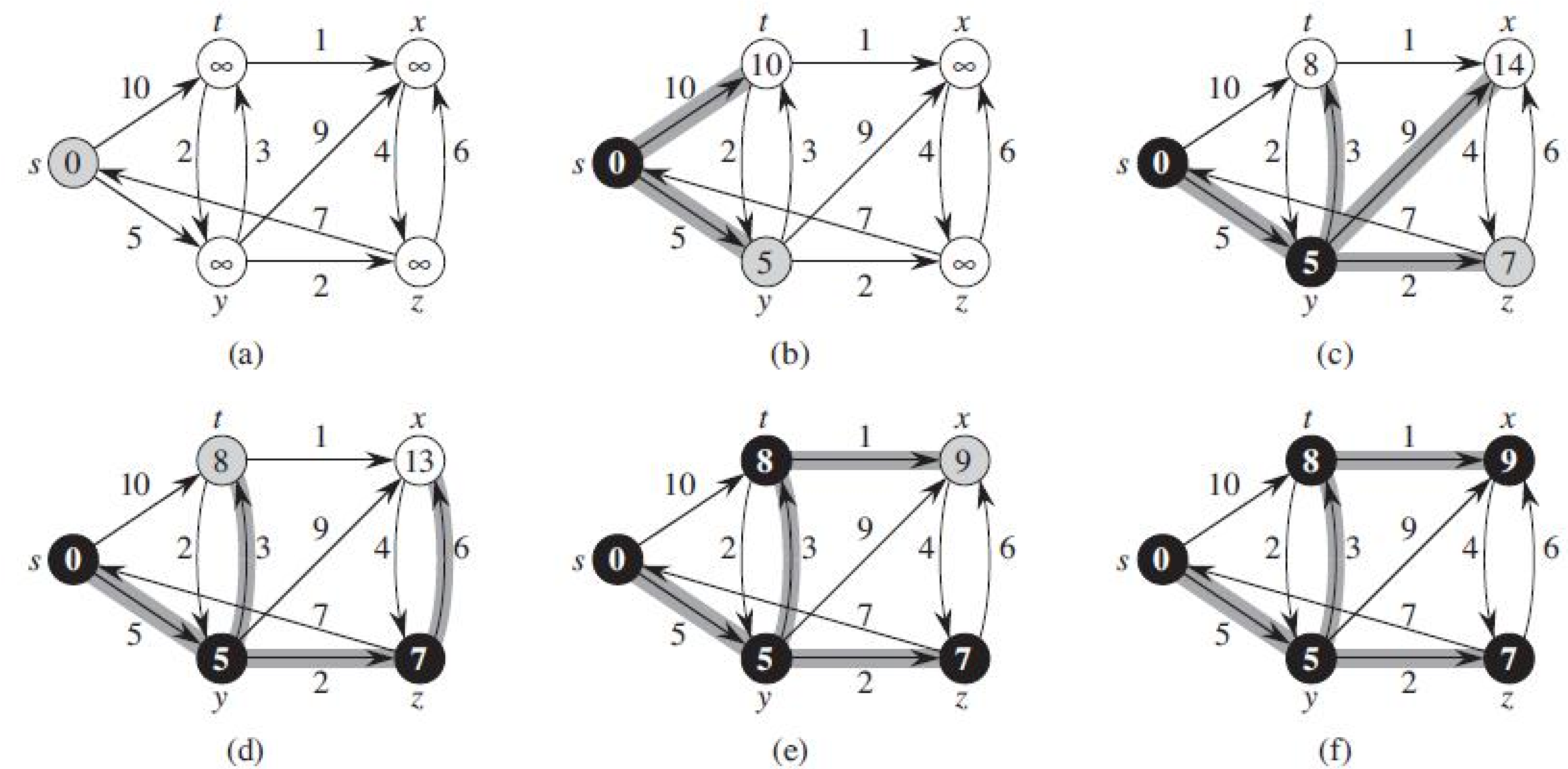
```
DIJKSTRA(G, w, s)
INITIALIZE-SINGLE-SOURCE(G, s)
S = Ø
Q = G.V
while Q ≠ Ø
    u = EXTRACT-MIN(Q)
    S = S ∪ {u}
    for each vertex v ∈ G.Adj[u]
        RELAX(u, v, w)
```

```
RELAX(u, v, w)
if v.d > u.d + w(u, v)
    v.d = u.d + w(u, v)
    v.π = u
```

- ✓ v.d is the shortest-path estimate (from source s to v).
- ✓ v.pi is the predecessor of v.
- ✓ Q is a min-priority queue of vertices, keyed by their d values.
- ✓ S is a set of vertices whose final shortest paths from s is determined.
- ✓ O(VlogV + E) w/ min-heap; O(V^2 + E) w/ array; E = O(V), discard.

Geodesic Distance

✓ The execution of Dijkstra's algorithm (on a directed graph):



Geodesic Distance

- ✓ A* search is an interesting alternative to Dijkstra in which shortest-path estimate value + a heuristic value towards the goal is considered.

  f(n) = g(n) + h(n)

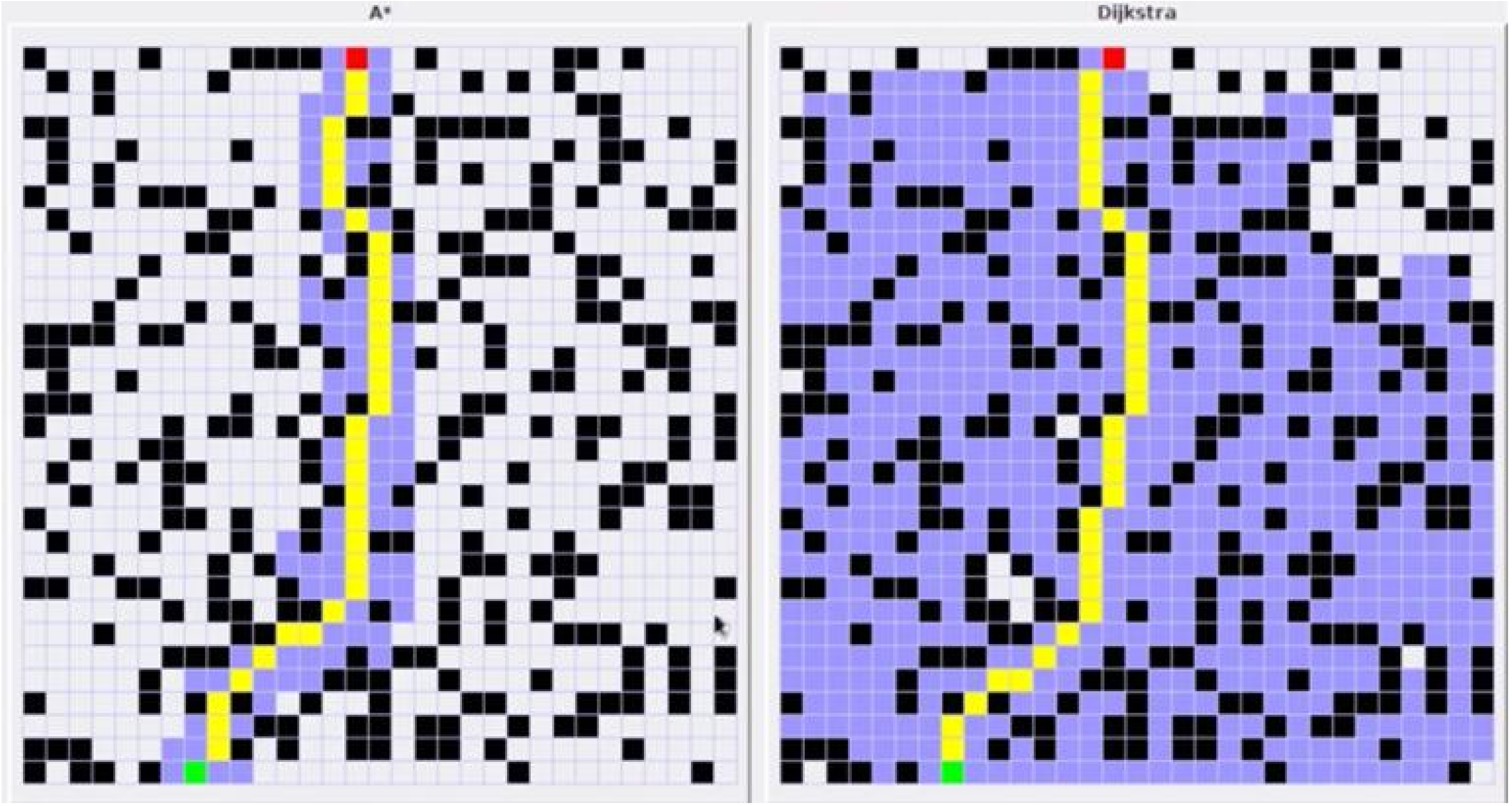
  g(n) := cost to get from initial state to **n**

  h(n) := heuristic estimate of cost to get from **n** to goal

- ✓ See https://youtu.be/ySN5Wnu88nE for a gentle introduction.

Geodesic Distance

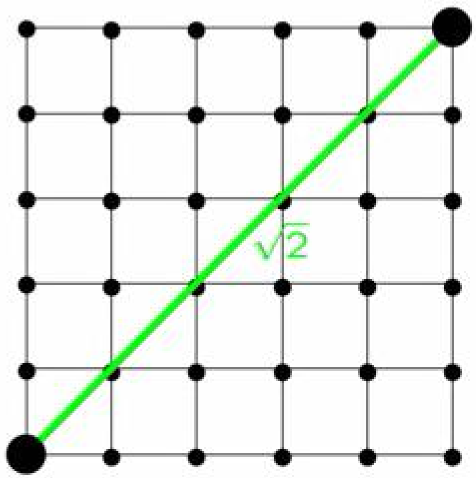
- ✓ A* search is an interesting alternative to Dijkstra in which shortest-path estimate value + a heuristic value towards the goal is considered.



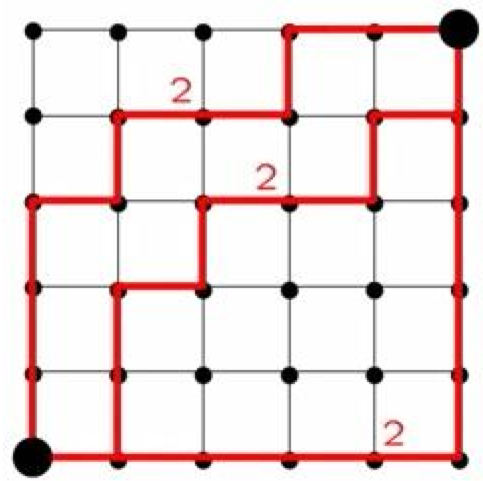
- ✓ Heuristic: Euclidean dist to the red target. Start: green. Visited: purple.

Geodesic Distance

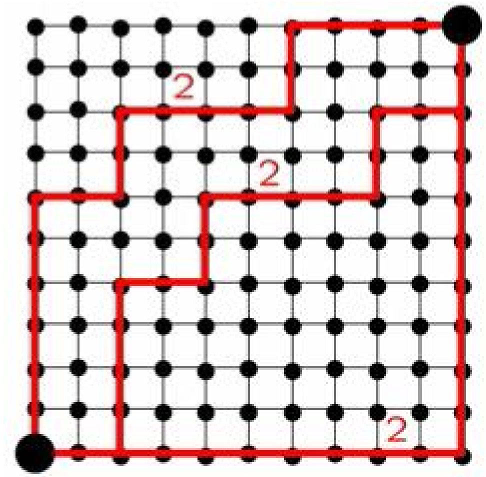
- ✓ Does the path really stay on the surface?
- ✓ Yes, but could have been better if it could go through faces.



Euclidean distance



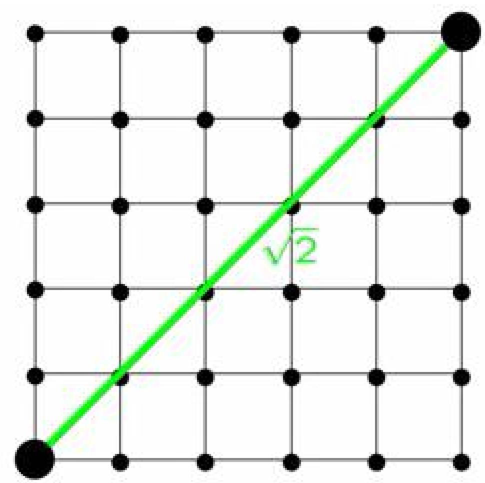
Geodesic distance



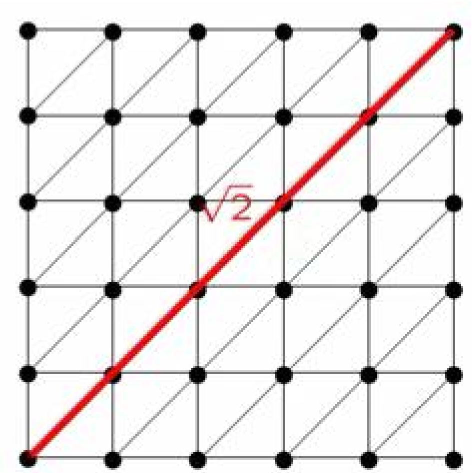
Geodesic on high-reso

Geodesic Distance

- ✓ Does the path really stay on the surface?
- ✓ Yes, but could have been better if it could go through faces.



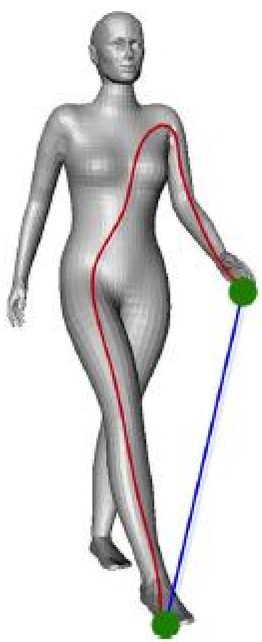
Euclidean distance

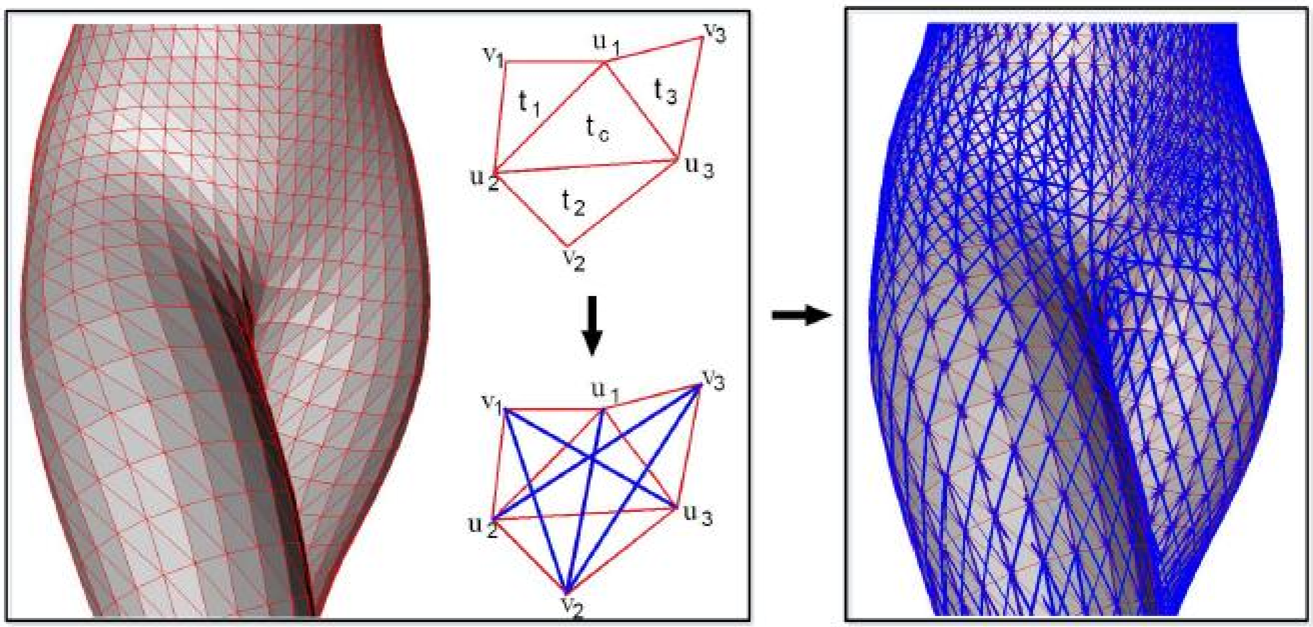


Geodesic distance

Geodesic Distance

- ✓ Does the path really stay on the surface?
- ✓ Yes, but could have been better if it could go through faces.

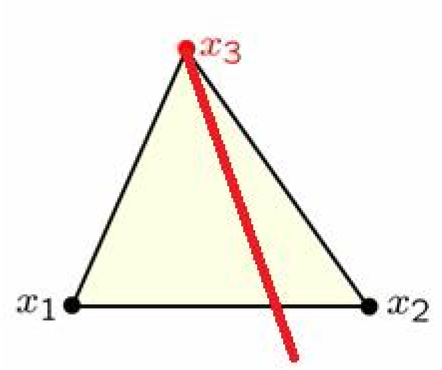




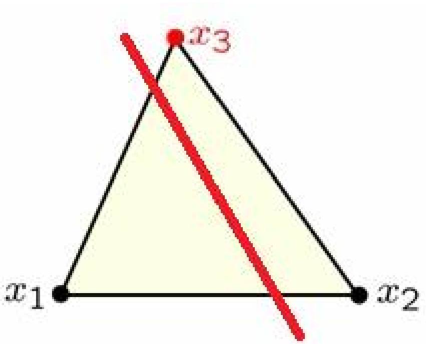
Try to add virtual (blue) edges if the 4 triangles involved are nearly co-planar.

Geodesic Distance

- ✓ Fast marching algorithm for more flexible (and accurate) face travels.
  - ✓ R. Kimmel and J. A. Sethian. Computing geodesic paths on manifolds, 1998.



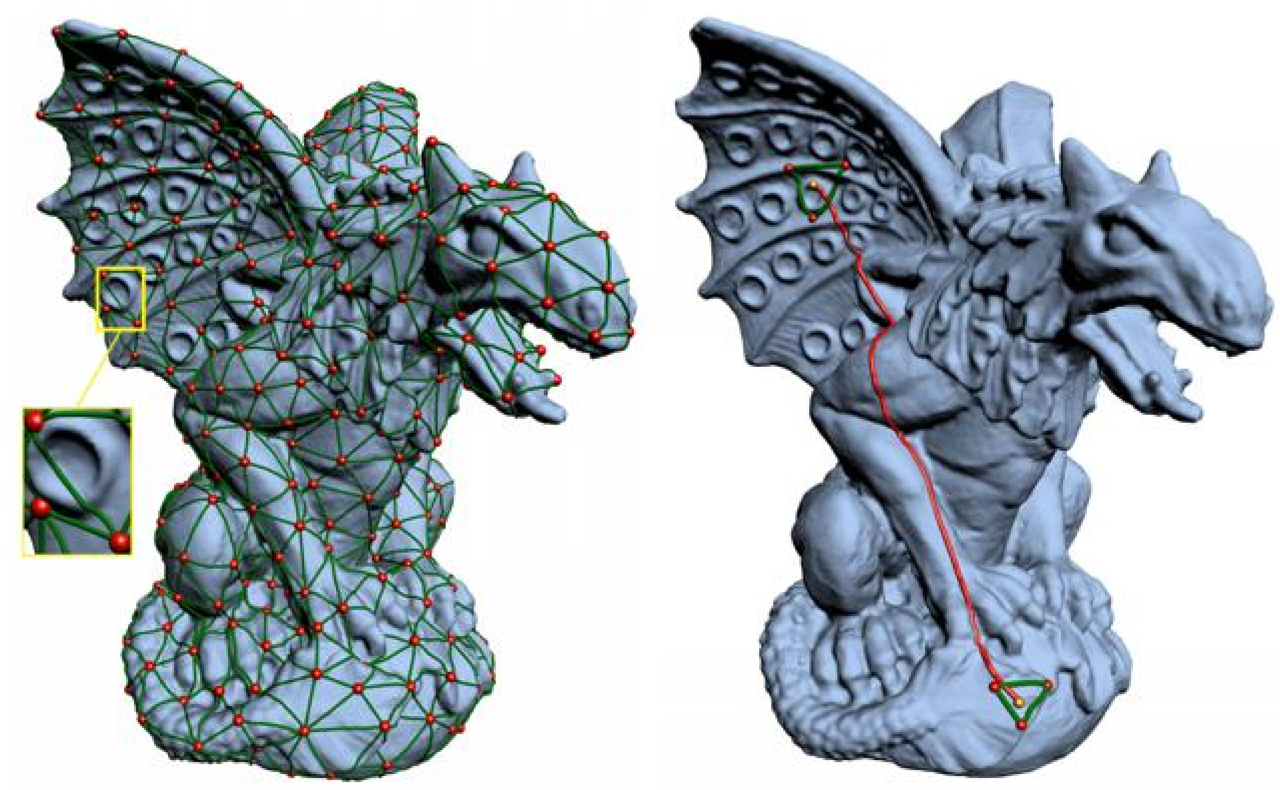
vs.



- ✓ Shortcut edge (blue on prev slide) allows face travel starting from a face vertex.
- ✓ Fast marching allows face travel starting from an arbitrary pnt on the face edge.

Geodesic Distance

- ✓ Exact and approximate algorithms are emerging each year.
- ✓ See Geodesics in Heat, Saddle Vertex Graph, Wavefront Propagation.

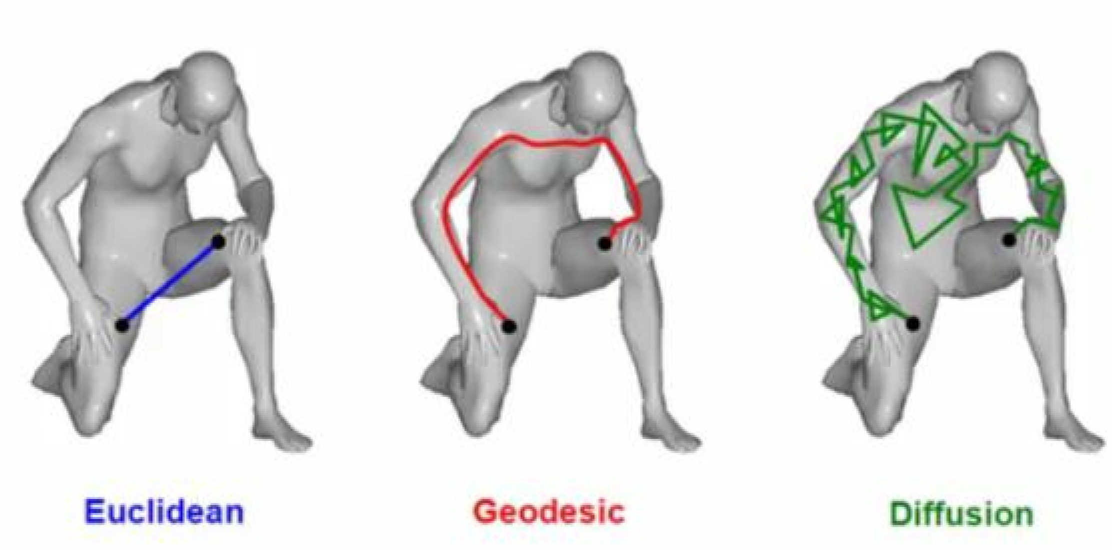


Geodesic Distance

✓ Main disadvantage: Topological noise changes geodesics drastically.

Diffusion Distance

- ✓ Principle: diffusion of heat after some time t.
  - ✓ Geodesic: length of the shortest path.
  - ✓ Diffusion: average over all paths of length t.



Diffusion Distance

- ✓ Principle: diffusion of heat after some time t.
- ✓ Take a hot needle; touch it to a surface point; watch how heat diffuses out over a time t.

Diffusion Distance

- ✓ Principle: diffusion of heat after some time t.
- ✓ $k_t(x,y)$ = probability of Brownian motion of heat starting at point x to be at point y after some time t.

Diffusion Distance

- ✓ Principle: diffusion of heat after some time t.
- ✓ $k_t(x,y)$ = amount of heat transferred from x to y in time t.

Diffusion Distance

- ✓ Principle: diffusion of heat after some time t.
- ✓ $k_t(x,y)$ = amount of heat transferred from x to y in time t.
- ✓ $k_t(x,x)$ = amount of heat remaining at x after time t.

$$k_t(x,y) = \sum_i \exp(-t\lambda_i)\phi_i(x)\phi_i(y)$$

λ_i, ϕ_i eigenvalues/eigenfunctions of the LB operator.

Diffusion Distance

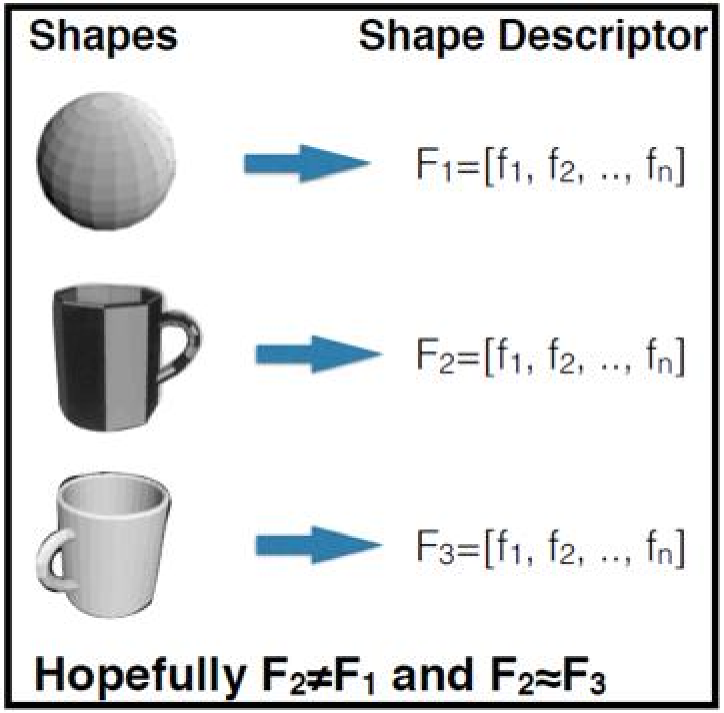
- ✓ Principle: diffusion of heat after some time t.
- ✓ $k_t(x,y)$ = amount of heat transferred from x to y in time t.
- ✓ Intuitively, $k_t(x,y)$, aka the heat kernel, is a weighted average over all paths between x and y possible in time t, which should not be affected by local perturbations of the surface (small topology noise).

Geodesic Distance

- ✓ What can we do with this geodesic?
- ✓ Descriptor.
- ✓ Sampling.
- ✓ Similarity comparisons.
- ✓ ..

Shape Descriptors

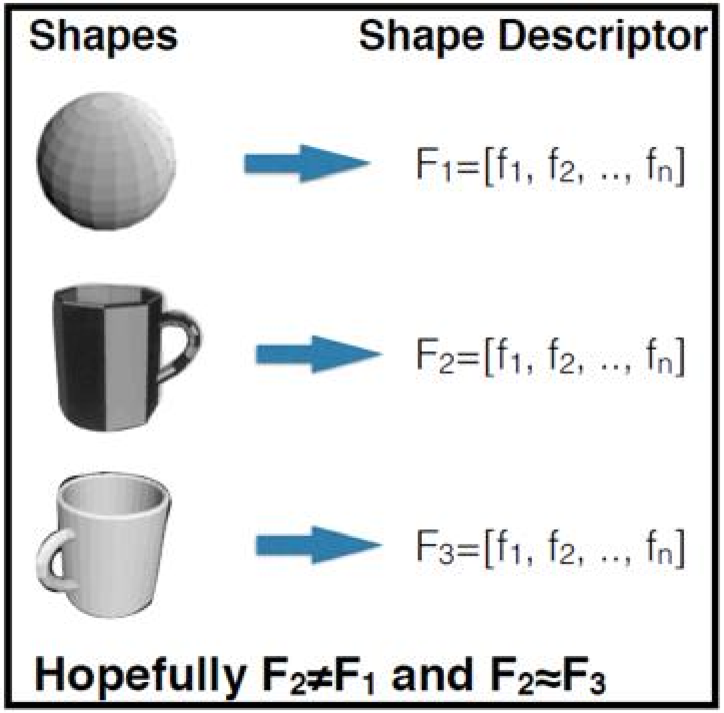
- ✓ Local (vertex-based) vs. Global (shape-based) shape descriptors.



- ✓ Desired properties:
  - ✓ Automatic
  - ✓ Fast to compute and compare
  - ✓ Discriminative
  - ✓ Invariant to transformations (rigid, non-rigid, ..)

Shape Descriptors

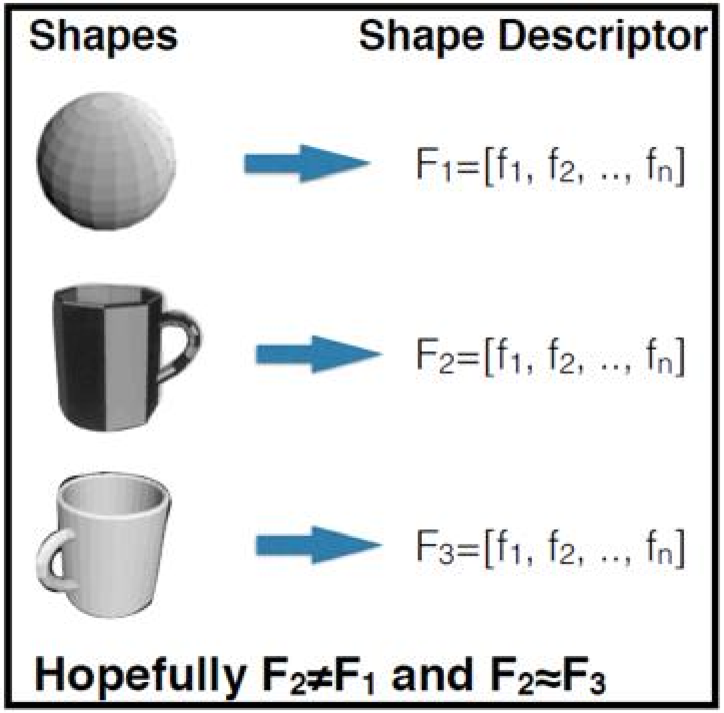
- ✓ A toy global descriptor: F = [x1, y1, z1, x2, y2, z2, .., xn, yn, zn].



- ✓ Desired properties:
  - ✓ Automatic??
  - ✓ Fast to compute and compare??
  - ✓ Discriminative??
  - ✓ Invariant to transformations (rigid, non-rigid, ..)??

Shape Descriptors

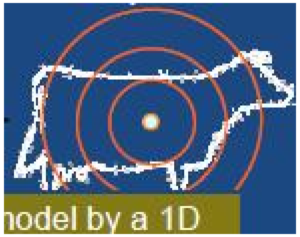
- ✓ A toy global descriptor: F = [numOfHandles]. //computable by Euler's formula.

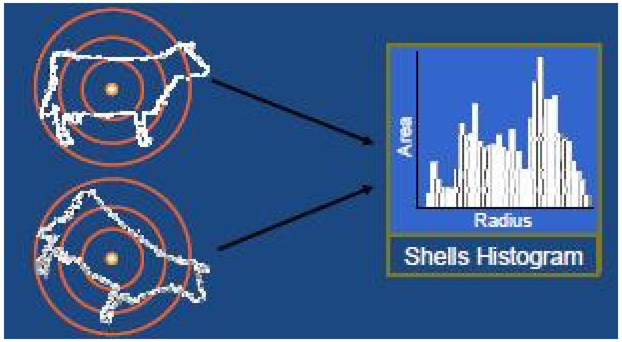


- ✓ Desired properties:
  - ✓ Automatic??
  - ✓ Fast to compute and compare??
  - ✓ Discriminative??
  - ✓ Invariant to transformations (rigid, non-rigid, ..)??

Shape Descriptors

✓ Global descriptors.

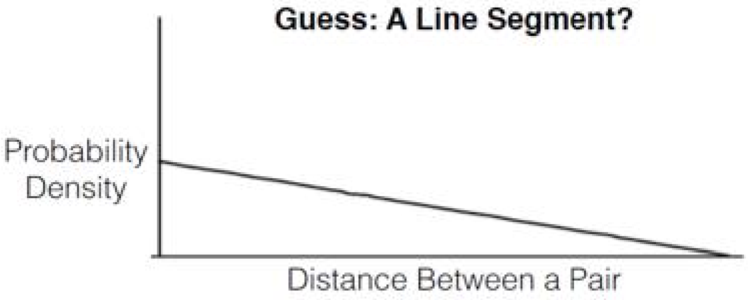


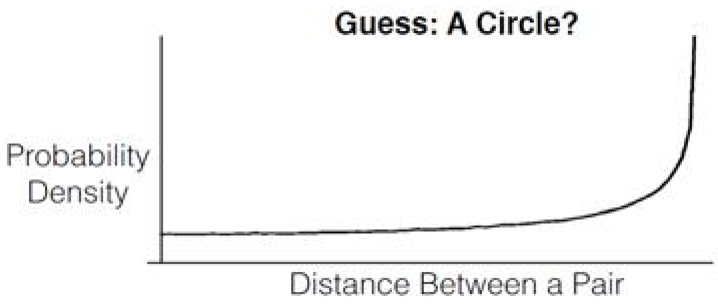


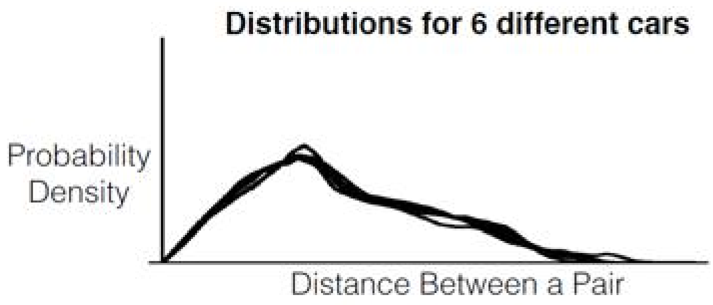
shape histograms
area/volume intersected
by each region stored in a
histogram indexed by radius

Shape Descriptors

- ✓ Global descriptors.







shape distributions

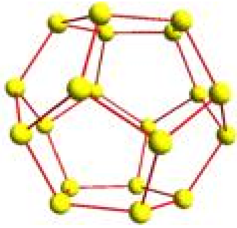
distance between 2 random points on the surface

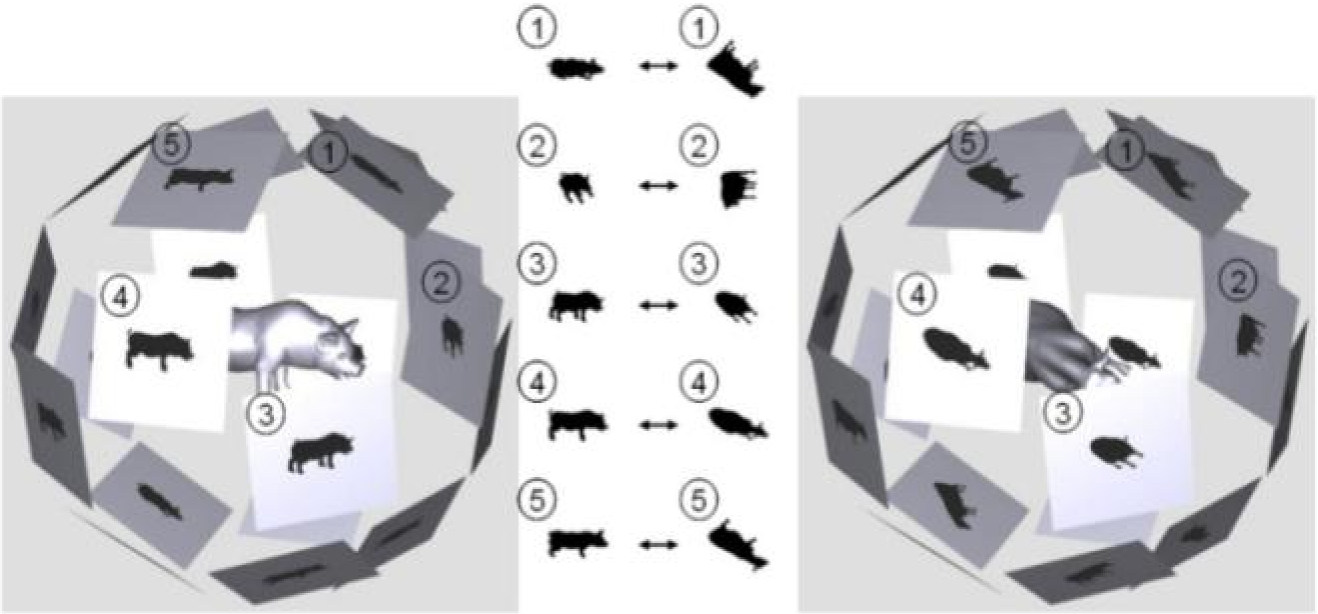
angle between 3 random points on the surface

..

Shape Descriptors

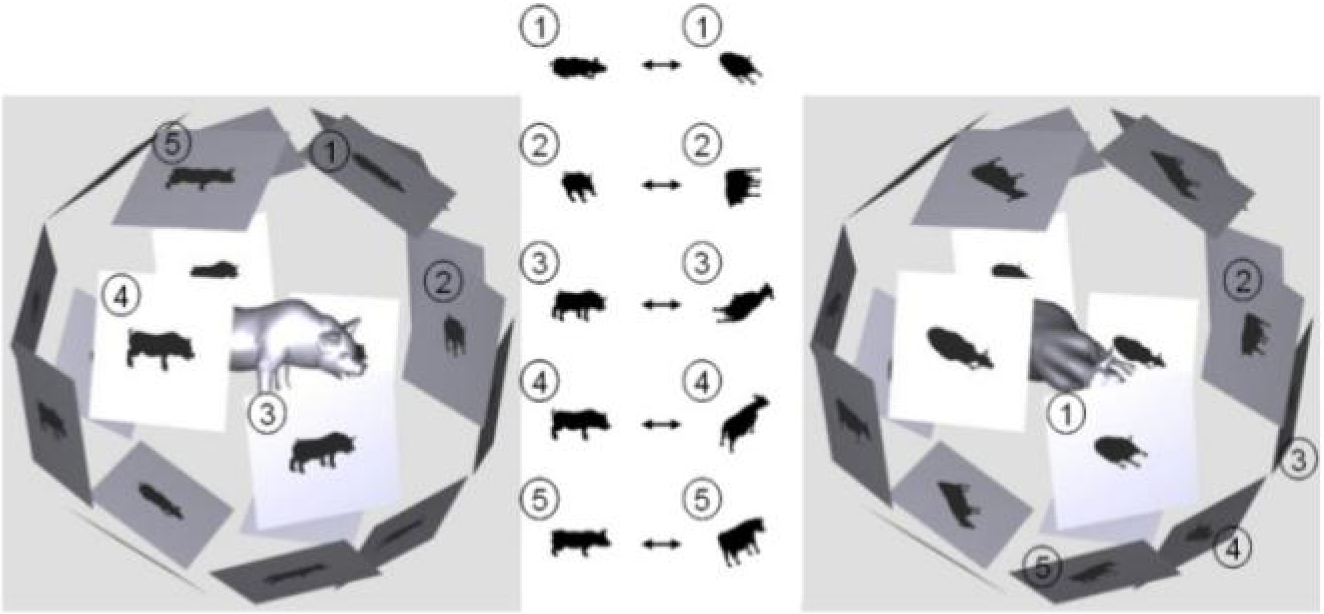
- ✓ Yet another global descriptor: LightField Descriptor
- ✓ 10 silhouette images rendered from vertices of a hemisphere.
  - ✓ Rotation-invariant.





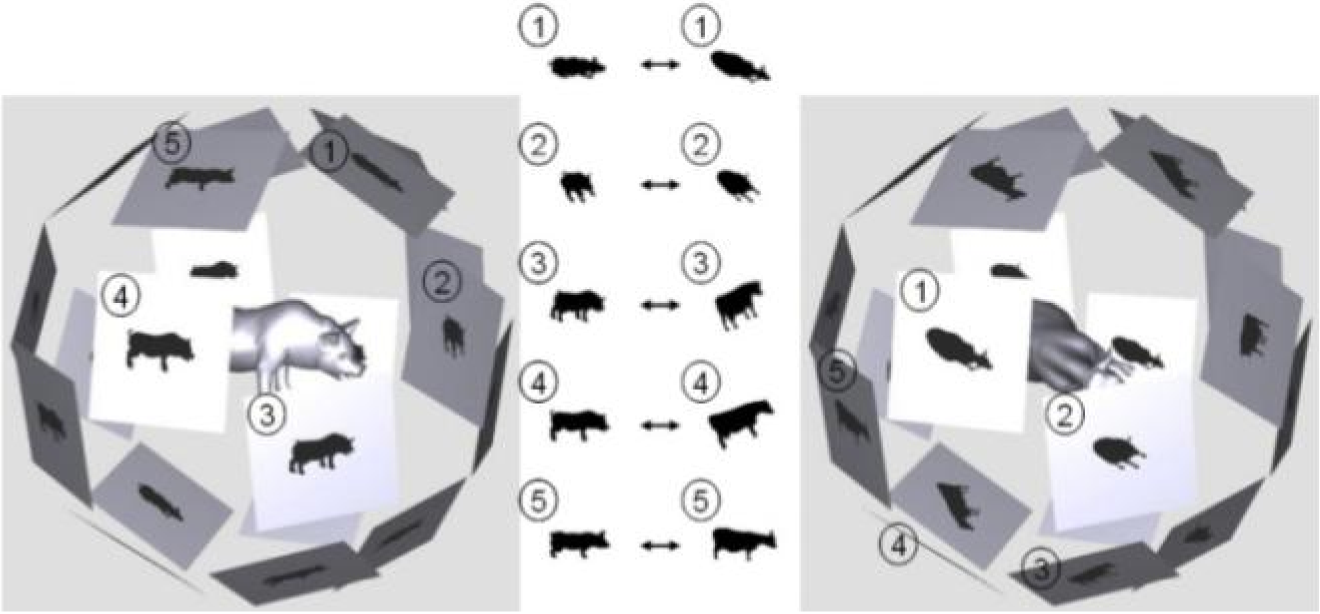
Shape Descriptors

- ✓ Yet another global descriptor: LightField Descriptor
- ✓ 10 silhouette images rendered from vertices of a hemisphere.
  - ✓ Rotation-invariant.



Shape Descriptors

- ✓ Yet another global descriptor: LightField Descriptor
- ✓ 10 silhouette images rendered from vertices of a hemisphere.
  - ✓ Rotation-invariant.

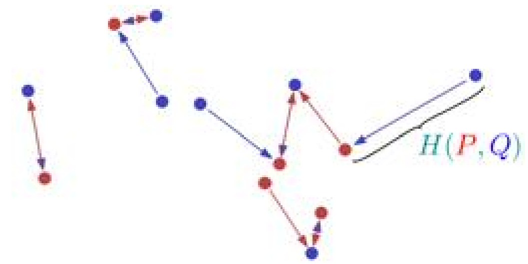


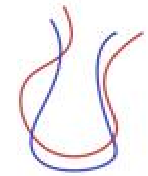
- ✓ Dissimilarity b/w two 3D models: $D_A = \min_i \sum_{k=1}^{10} d(I_{1k}, I_{2k})$.

Shape Descriptors

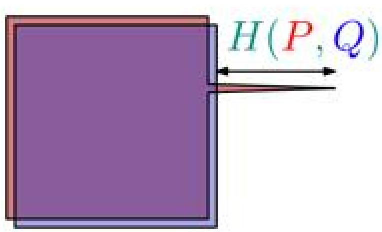
- ✓ Global descriptor for pairwise sim.: Haussdorf distance b/w 2 models.
- ✓ Find closest pnts of each red (min), and then each blue (min). Pick the max of the most widely separated closest pnts (max).

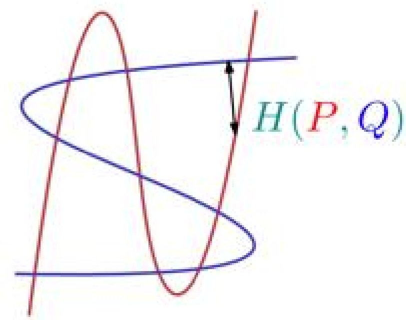
$$H(P,Q) = \max\left(\max_{p\in P} : \min_{q\in Q} : |pq|\right.$$
$$\left., \max_{q\in Q} : \min_{p\in P} : |pq|\right)$$

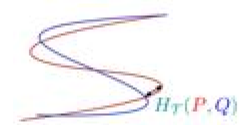




- ✓ Advantage: considers all boundary points, mutual proximity.
- ✓ Disadv: sensitive to outliers; sometimes unnatural (not rigid invariant).



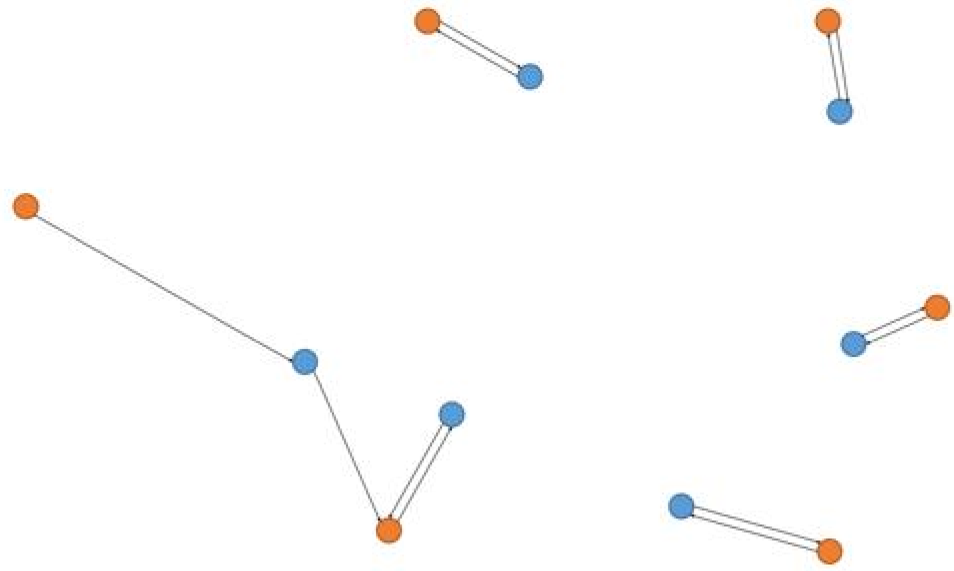




- ✓ Can be made transf-invariant if minimized through all transformations.

Shape Descriptors

- ✓ Global descriptor for pairwise sim.: Chamfer distance b/w 2 models.
- ✓ Find closest pnts of each red (min), and then each blue (min). Pick the sum of these sums, i.e., replace max of Haussdorf w/ additions.



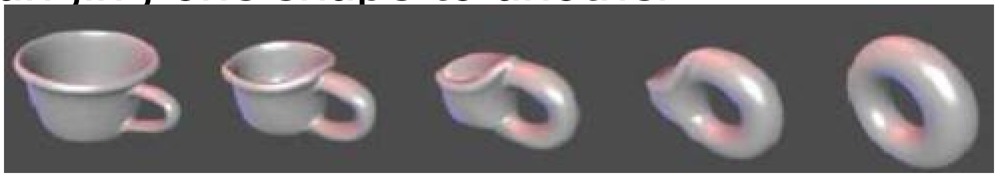
a.k.a Chamfer distance (CD)

$$d_{CD}(S_1, S_2) = \sum_{x \in S_1} \min_{y \in S_2} \|x - y\|_2^2 + \sum_{y \in S_2} \min_{x \in S_1} \|x - y\|_2^2$$

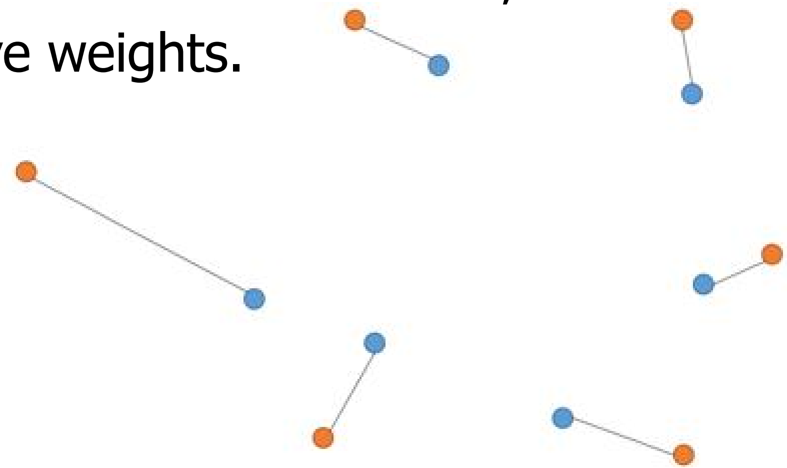
- ✓ Intuitively may appear more robust to outliers, still quite sensitive.

Shape Descriptors

- ✓ Global descriptor for pairwise sim.: Earth Mover's dist. b/w 2 models.
- ✓ Cost of changing one shape to another.



- ✓ Measure total length of matched dists, minimized over all bijections.
- ✓ May be more natural than Hausdorff, but harder to compute.
- ✓ Points may have weights.

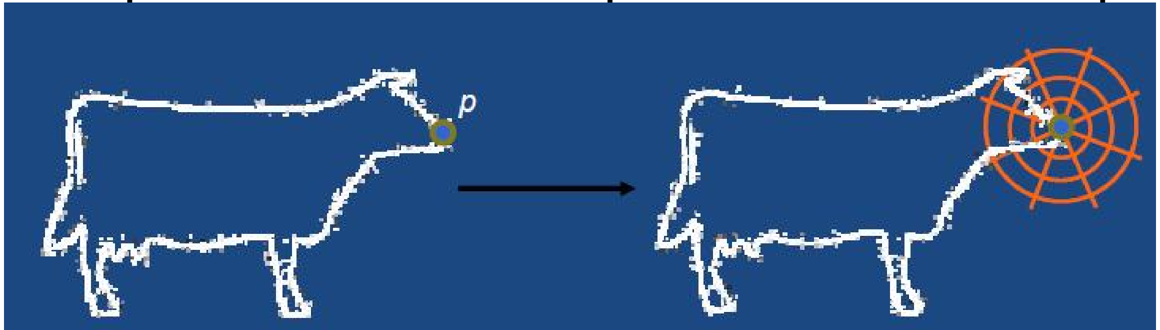


a.k.a Earth Mover's distance (EMD)

$$d_{EMD}(S_1, S_2) = \min_{\phi: S_1 \to S_2} \sum_{x \in S_1} \|x - \phi(x)\|_2 \quad \text{where } \phi : S_1 \to S_2 \text{ is a bijection.}$$

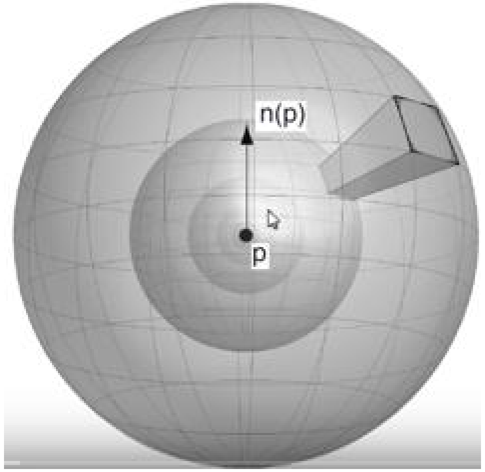
Shape Descriptors

- ✓ Global descriptor to a local descriptor: center it about p.



Shape histograms as local shape descriptors

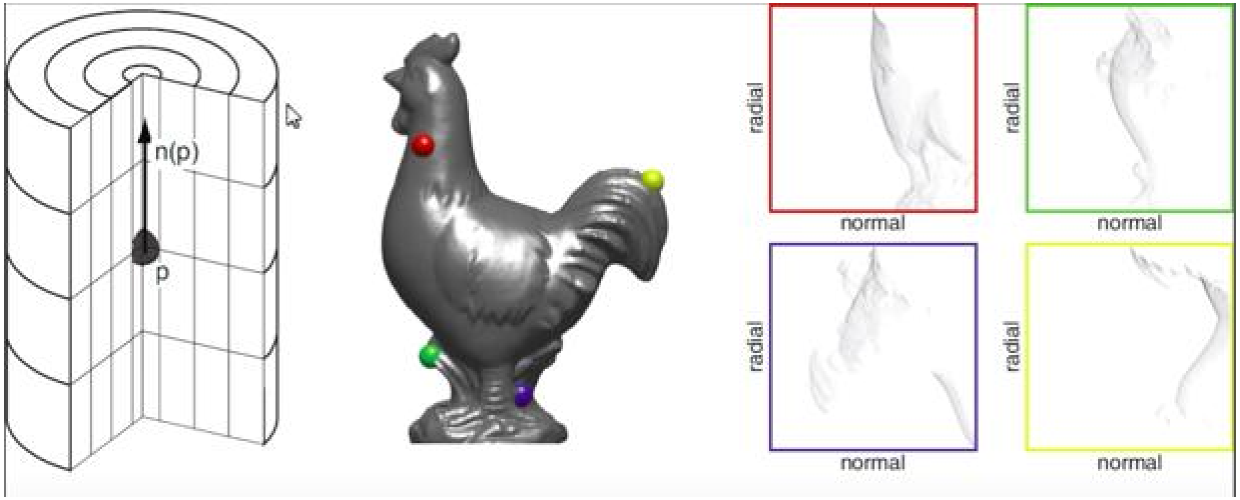
- ✓ Count points in different sectors of a sphere: shape contexts.



Shape Descriptors

- ✓ Global descriptor to a local descriptor: center it about p.

- ✓ Spin Images: Count 3D points falling inside that solid ring.
  - ✓ Go this far in normal direction, and this far in radial direction.

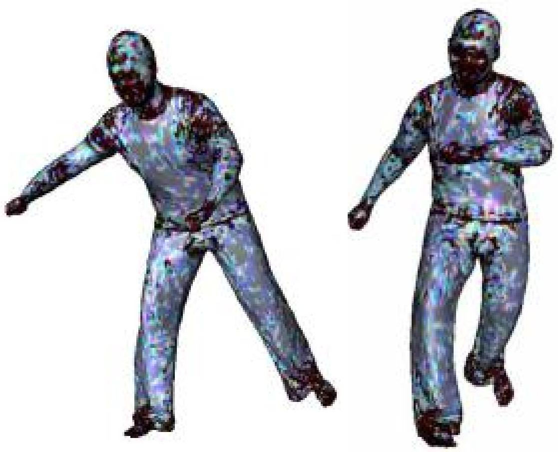


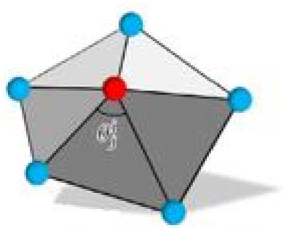
- ✓ Shape contexts in the following part of this video

Shape Descriptors

✓ Local shape descriptors.

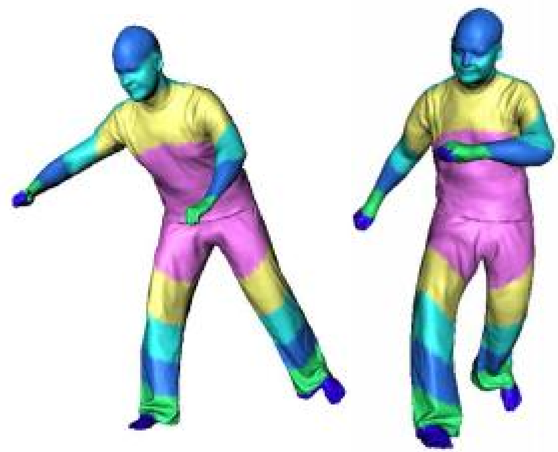
Curvature





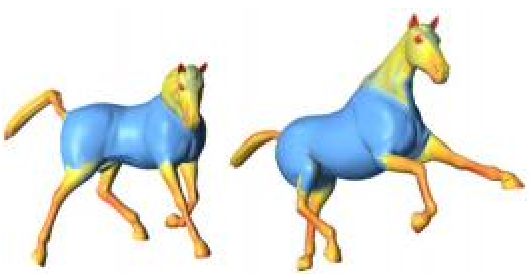
$$K(v_i) = \left(2\pi - \sum_{j: v_j \in N_1(v_i)} \theta_j^i\right)$$

Average Geodesic Distance



sum of geodesics from
v to all other vertices.

Shape Diameter (SDF)

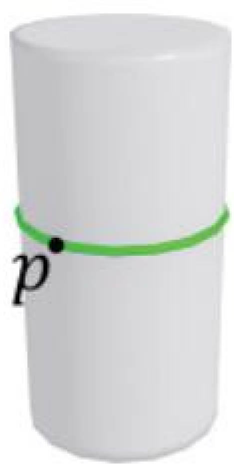


sum of ray lengths from v.

Shape Descriptors

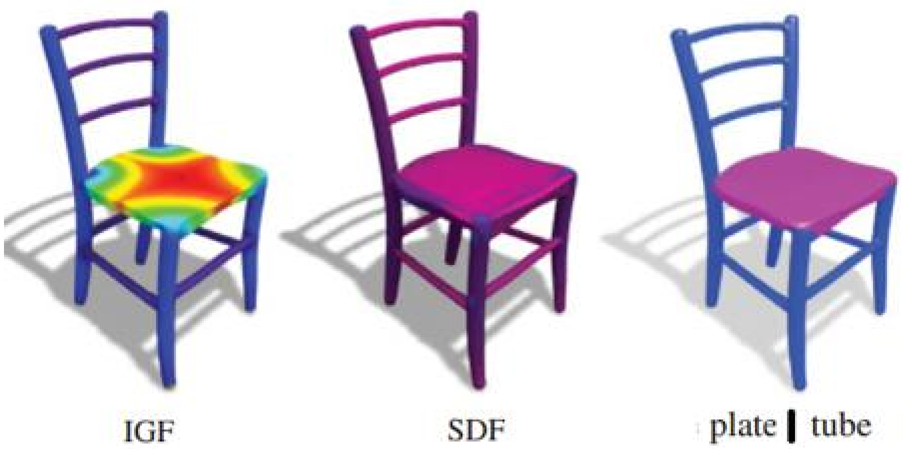
✓ Local shape descriptors.

<u>Intrinsic Girth Function</u>



Shortest geodesic
starts & ends at p.

Combine with SDF to detect plate/tube

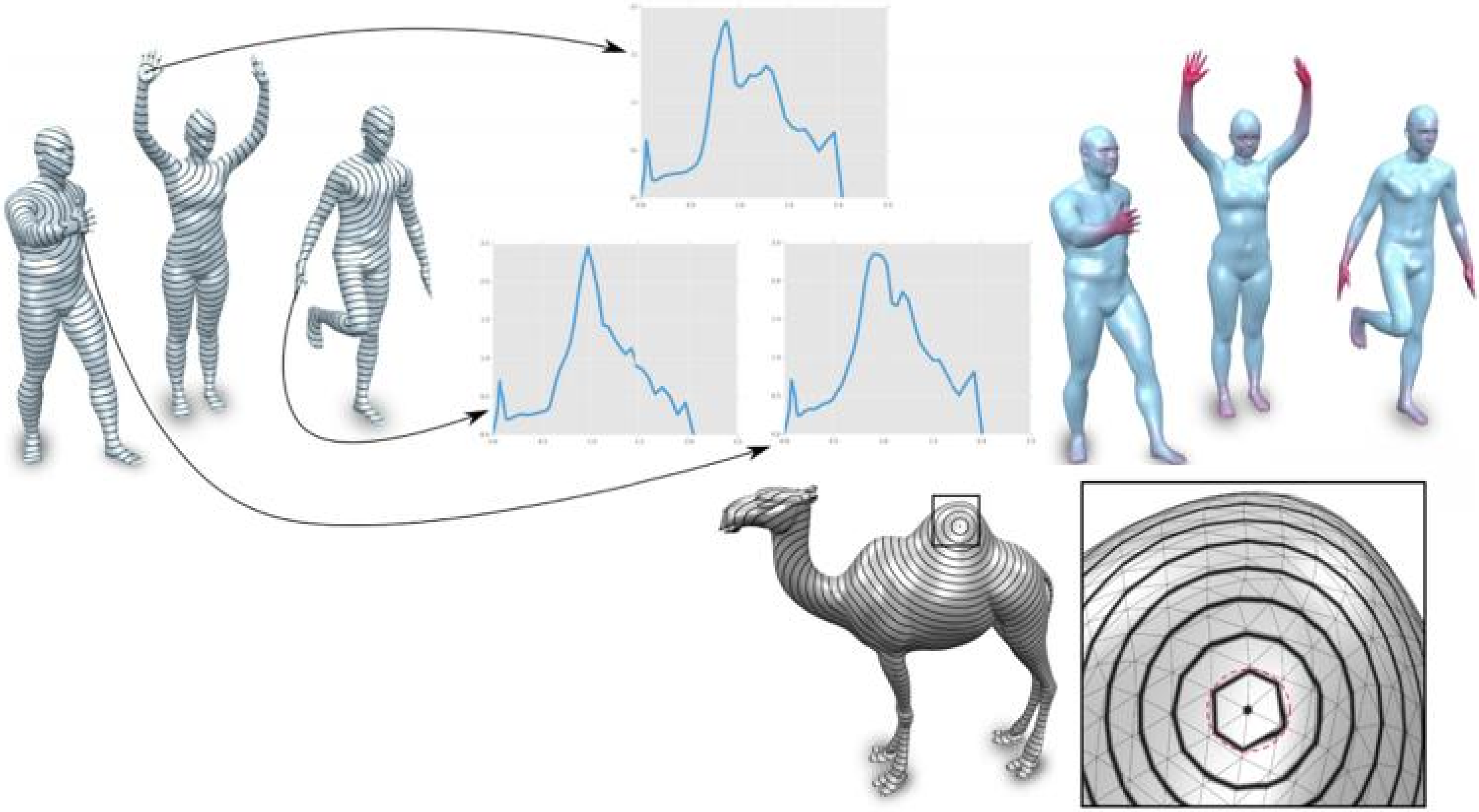


Big IGF(p), small SDF(p) ➔ plate
Small IGF(p), small SDF(p) ➔ tube

Shape Descriptors

✓ Local shape descriptors.

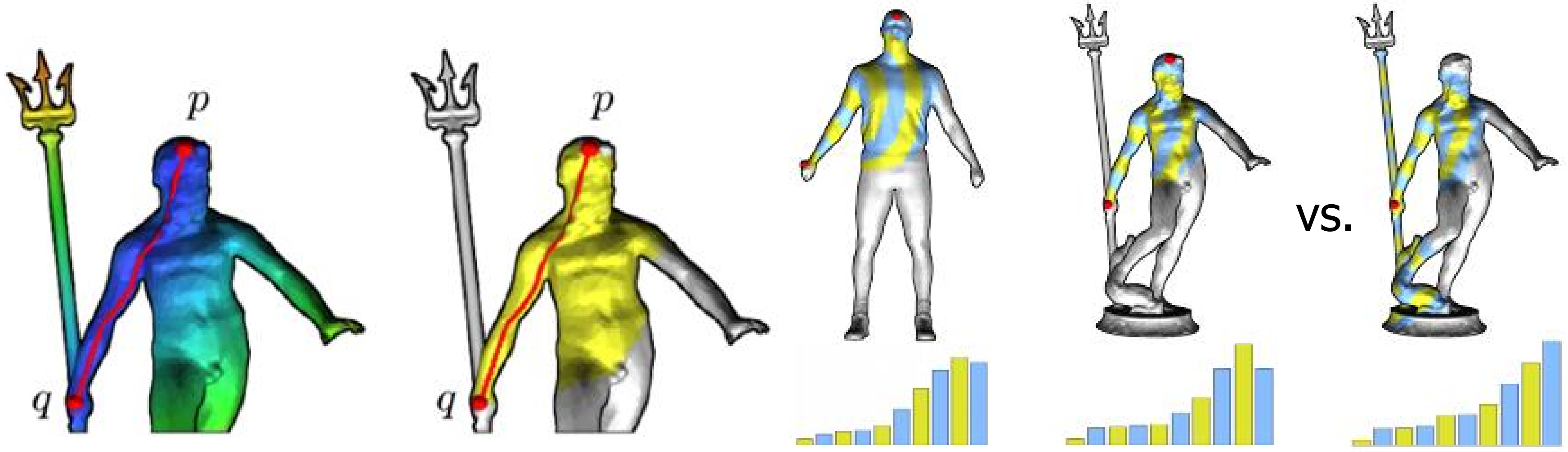
Geodesic Iso-Curve Signature



Shape Descriptors

✓ Local shape descriptors.

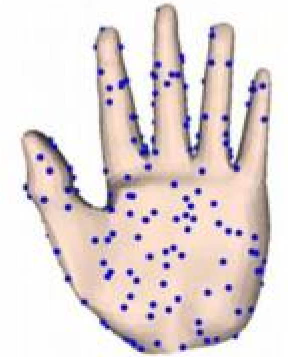
<u>Bilateral Maps</u>

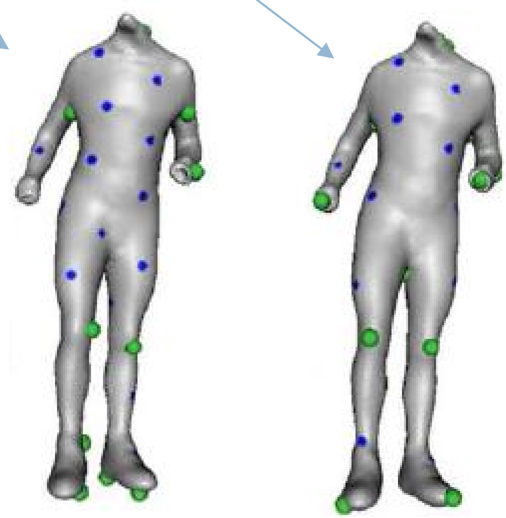


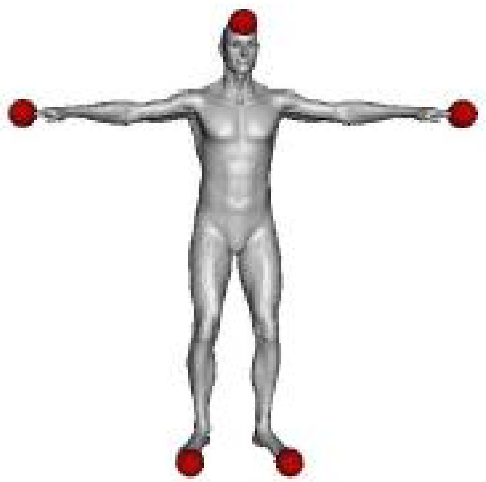
Sampling

✓ Subset of vertices are sampled via

Uniform sampling | Evenly-spaced sampling | Curvature-oriented evenly-spaced sampling | Farthest-point sampling (FPS)





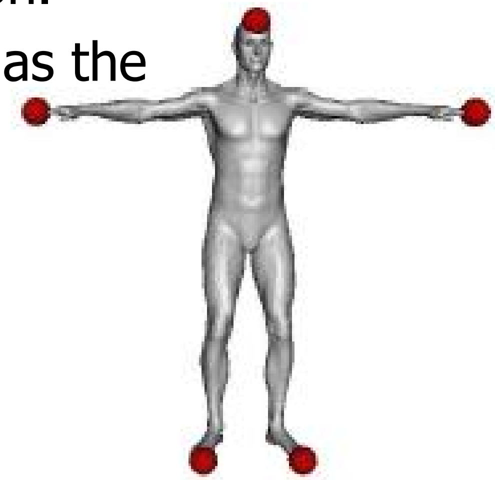


✓ Uniform sampling: Pick triangles w/ probabilities proportional to their areas; then put a random sample inside the picks.

✓ FPS: Eldar et al., The farthest point strategy for progressive image sampling.

Sampling

- ✓ FPS idea.
- ✓ You have V vertices and N samples; want to sample the (N+1)st.
- ✓ V-N candidates.
- ✓ Each candidate is associated with the closest existing sample.
  - ✓ Remember the distance used in this association.
- ✓ As your (N+1)st sample, pick the candidate that has the max remembered distance.



- ✓ If geodesic distance is in use (more accurate):
  - ✓ O(NV + NVlogV) = O(NVlogV) complexity.
- ✓ Else if Euclidean distance is in use:
  - ✓ O(NV) complexity.
  - ✓ Cost of finding the max distance for each new sample: O(NV), no additional Dijkstra geodesic computation.

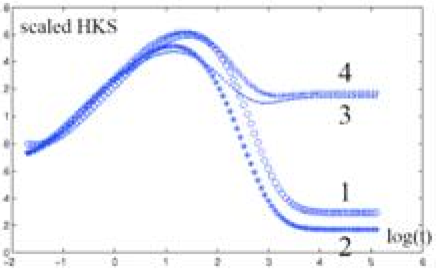
Sampling

- ✓ Can we get this sampling via FPS?

Sampling

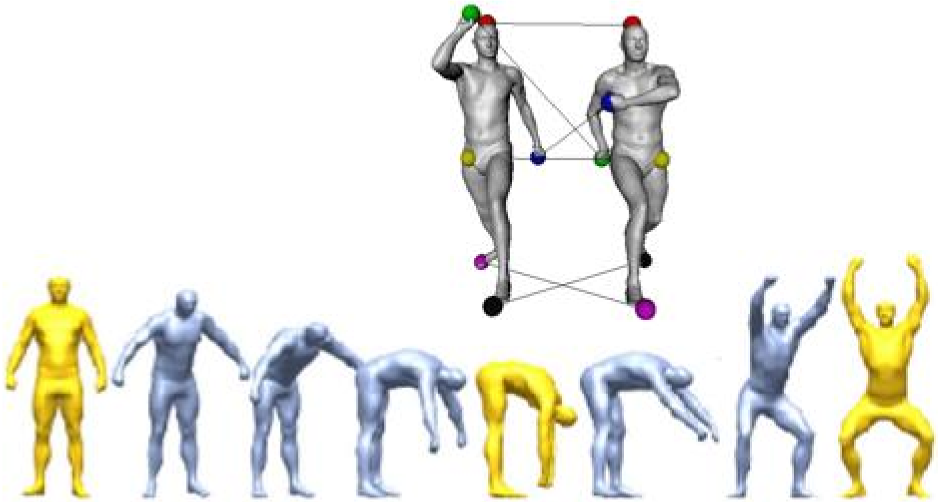
- ✓ Can we get this sampling via FPS?

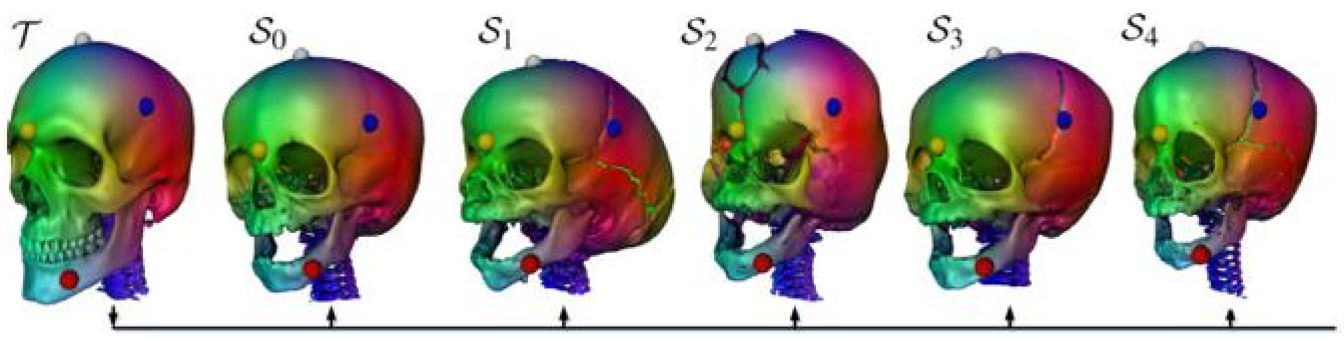
- ✓ No.
- ✓ Use local maxima of Average Geodesic Distance descriptor
- ✓ Use local maxima of heat kernel signature (Slide 16), a multiscale descriptor similar to Geodesic Iso-Curves.



An Application: Shape Correspondence

✓ Feature-based and/or distance-based algorithms for shape correspondence, an important problem in computer graphics.





An Application: Shape Correspondence

- ✓ Feature-based: match the points with close feature descriptors.
- ✓ Pointwise term, i.e., involves 1 point at a time.

$$\mathrm{Dis}_l(\varphi) = \sum_{x \in X} d_F\left(f^X(x), f^Y(\varphi(x))\right)$$

An Application: Shape Correspondence

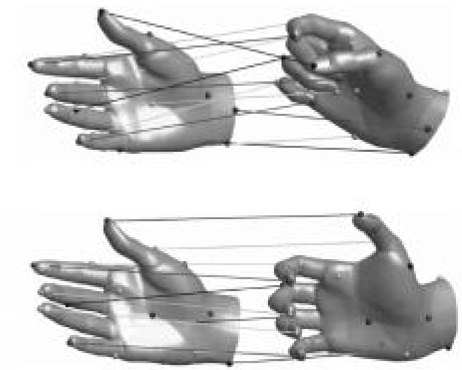
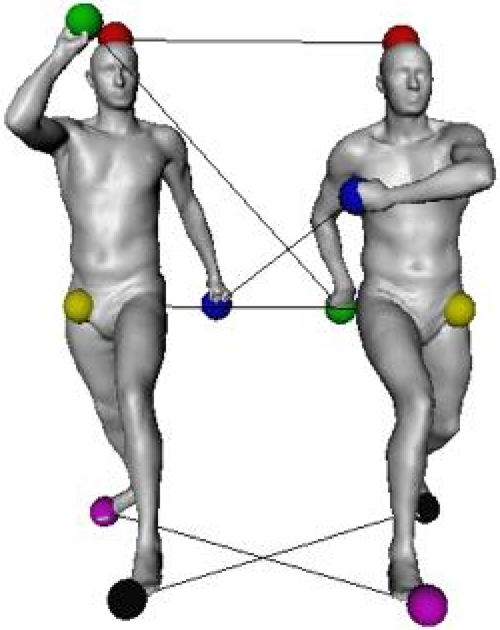
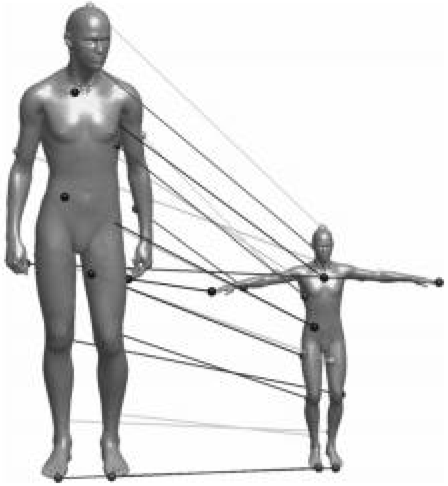
- ✓ Distance-based: match 2 points with compatible distances.
- ✓ Pairwise term, i.e., involves 2 points at a time.

$$\mathrm{Dis}_q(\varphi) = \sum_{x,\tilde{x} \in X} |d_X(x,\tilde{x}) - d_Y(\varphi(x), \varphi(\tilde{x}))|$$

An Application: Shape Correspondence

✓ Feature- and distance-based: a combination of both.

$$\mathrm{Dis}(\varphi) = \sum_{x \in X} d_F(f^X(x), f^Y(\varphi(x))) + \lambda \cdot \sum_{x, \tilde{x} \in X} |d_X(x, \tilde{x}) - d_Y(\varphi(x), \varphi(\tilde{x}))|$$

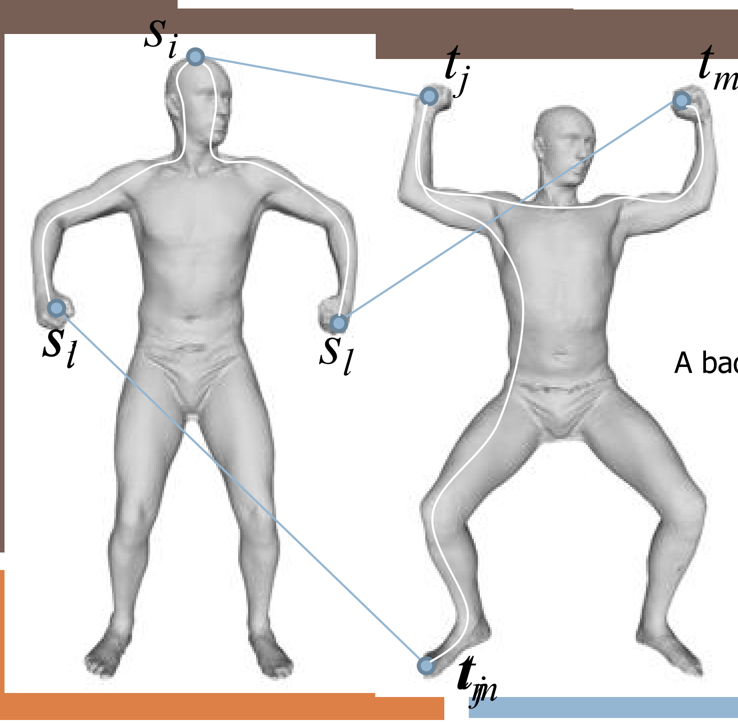




An Application: Shape Correspondence

- ✓ Let's see a useful pairwise term in action.
  - ✓ Copied from my paper talk: A Genetic Isometric Shape Correspondence Algorithm with Adaptive Sampling, SIGGRAPH Asia, 2018.

$$\mathcal{D}_{\text{iso}}(\phi) = \frac{1}{|\phi|} \sum_{(s_i, t_j) \in \phi} \left(\frac{1}{|\phi'|} \sum_{(s_l, t_m) \in \phi'} \underbrace{|d_g(s_i, s_l) - d_g(t_j, t_m)|}_{|.34\ -\ .98|\ =\ .64} \right)$$

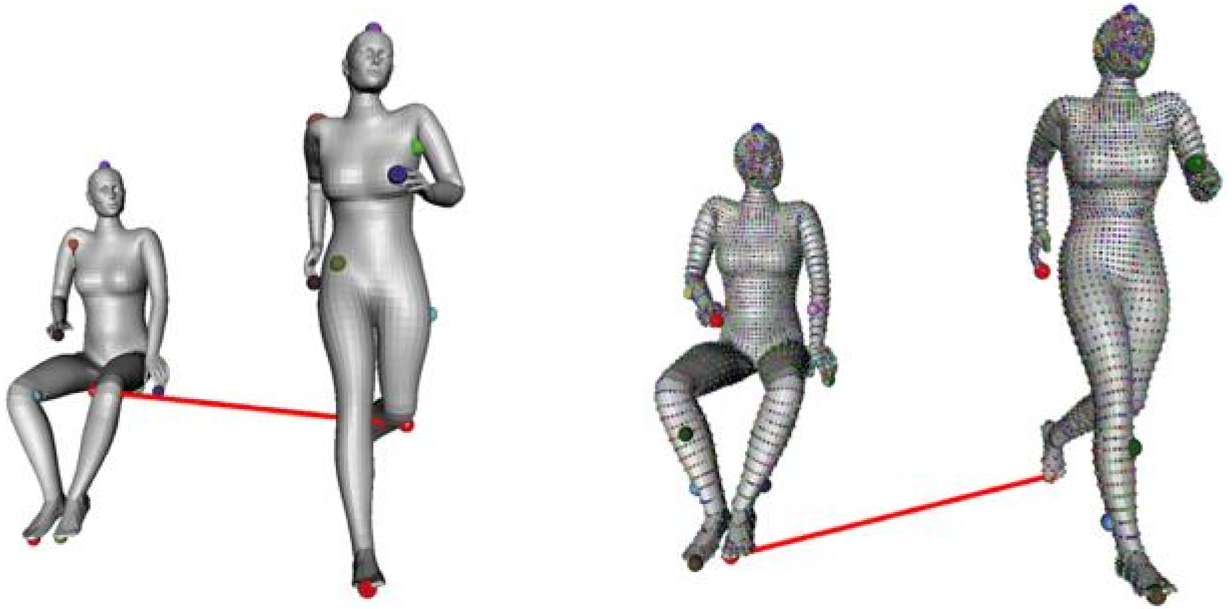
☹



A bad/high-distortion map.

An Application: Shape Correspondence

- ✓ Interpolation of a sparse correspondence into a dense one.
- ✓ Compute for each vertex on M1 the geodesic distances to all sparse correspondence points on M1, and then establish a correspondence to the vertex on M2 with the most similar distances to sparse correspondence points on M2.



Potential Project Topics

- ✓ Implement paper: Constant-Time All-Pairs Geodesic Distance Query On Triangle Meshes (Figure in Slide 12).
- ✓ Implement paper: Intrinsic Girth Function for Shape Processing.
- ✓ Given the shape, computing the descriptor is easy. How about the inverse problem: given the descriptor(s), compute the shape?
  - ✓ Inspiration: these slides and/or some Shape Registration slides.