Geodesic Distance Descriptors
Gil Shamai and Ron Kimmel
sgils@cs.technion.ac.il, ron@cs.technion.ac.il
Computer Science Department, Technion

ABSTRACT
The Gromov-Hausdorff (GH) distance is traditionally used for measuring distances between metric spaces, and is adapted for non-rigid shape comparison and matching of isometric surfaces. Computing the GH distance is a hard combinatorial problem that requires pre-computation and storing of all pairwise geodesic distances for the matched surfaces. Current simplifications of this problem introduce errors due to low rank approximations and relaxations of the permutation matrices. Here we define the Geodesic Distance Descriptor (GDD). The GDD encodes all information of the geodesic distances and can be thought of a generalized canonical form with no flattening errors. We show that the GH related shape matching problem is equivalent to nothing but a rigid matching of these descriptors. We define the geodesic distance basis, which is optimal for compact approximation of geodesic distances. We use the suggested basis to extract the GDD without actually computing and storing all geodesic distances. These observations are used to very simply construct a very simple procedure that improves both accuracy and efficiency of all state-of-the-art shape matching methods.

GEODESIC DISTANCE BASIS

$D$ - some $n \times n$ pairwise geodesic distance matrix.
- Projecting $D$ on the LBO eigenspace:
  $D_Q = \Phi_d A_d \Phi_d^T$ (5)
LBO is best for representing smooth functions, but what if we consider only the (smooth) geodesic distance functions?
- Eigenvalue decomposition of $D$:
  $D = QAQ^T$ (6)
$Q$ is the optimal basis for geodesic distances.
- Projecting $D$ on the optimal basis:
  $D_Q = Q_d A_d Q_d^T$ (7)
The first 10 eigenfunctions of the geodesic distance basis.
To compute $Q$ one must store and decompose $D$ - impractical.
- Recent works (1) approximate $D$ as
  $D = STS^T$ (8)
Using QR factorization and eigendecomposition we derive
  $QR = S, V \Lambda V^T = RTT^T, \hat{Q} = QV \Rightarrow D = \hat{Q} \Lambda \hat{Q}^T$
$\hat{Q}$ approximates the geodesic basis $Q$.

GEODESIC DESCRIPTORS

Denote by $W$ the square root of $\Lambda$, such that
  $W_d = \sqrt{\Lambda_d}$ (9)
Define
  $X = QW$, (10)such that,
  $D = XX^T$. (11)
We term $X$ as the Geodesic Distance Descriptor (GDD).

$GDD$ $LBO$

The Euclidean distance $E_{ij} = \|x_i - x_j\|_2$ VS the geodesic distance $D_{ij}$, for all pairs $i, j$.
- $X$ is complex.
- Holds all information of the geodesic distances.
- The first 50 columns contain almost all information.
- $X$ is a canonical form with no flattening error.
- Can be used for dimensionality reduction tasks.
- Invariant to isometric deformations.
- Unique up to rotations and reflections.
- Row $x^i$ in $X$ is the descriptor of point $i$ in the shape.

THE GROMOV-HAUSDORFF DISTANCE

The GH distance is defined as
  $d_{GH}(S_1, S_2) = \frac{1}{2} \min_{\pi \in \Pi(n)} \max_{(s, q) \in C} \left\{ \|d_1(s, \pi(s)) - d_2(q, \pi(q))\| \right\}$ (1)
where
  $\forall s \in S_1, \exists q \in S_2$ s.t. $(s, q) \in C$, (2)
and
  $\forall q \in S_2, \exists s \in S_1$ s.t. $(s, q) \in C$. (3)
In the discrete domain:
  $\arg\min_{\pi \in \Pi(n)} \|PD_1P^T - D_2\|_{\infty}$. (4)$\pi(n)$ is the set of $n \times n$ permutation matrices.$D_1, D_2$ are $n \times n$ pairwise geodesic distances matrices.

EQUIVALENT FORMULATION

For Isometric shapes:
  $\arg\min_{\pi \in \Pi(n)} \|PD_1P^T - D_2\|_{\infty}$
$D = XX^T$
arg\min_{\pi \in \Pi(n)} \|PX_1X_1^TP^T - X_2X_2^T\|_{\infty}$
arg\min_{\pi \in \Pi(n), C \cap G(n)} \|PX_1C - X_2\|_{\infty}$
Can be solved with Iterative Closest Point procedure (ICP).

EXPERIMENTAL RESULTS

Quantitative evaluation of shape correspondence methods applied to the shapes from the SCAPE dataset, using the protocol of Kim et al.

<table>
<thead>
<tr>
<th>Method</th>
<th>GDD</th>
<th>SGMDS</th>
<th>FMaps</th>
<th>GT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cats</td>
<td>146.4</td>
<td>149.4</td>
<td>202.1</td>
<td>147.6</td>
</tr>
<tr>
<td>Horses</td>
<td>180.7</td>
<td>178.9</td>
<td>178.4</td>
<td>189.7</td>
</tr>
<tr>
<td>Wolves</td>
<td>82.9</td>
<td>83.5</td>
<td>90.3</td>
<td>84.3</td>
</tr>
<tr>
<td>Centaurs</td>
<td>153.3</td>
<td>174.3</td>
<td>151.6</td>
<td>171.1</td>
</tr>
<tr>
<td>Davids</td>
<td>58.6</td>
<td>58.6</td>
<td>66.2</td>
<td>62.1</td>
</tr>
</tbody>
</table>

# of vertices 4344 19248 27894 52565
SGMDS 45 498 1022 4961
FMaps 11 102 79 88
GDD 3 16 22 54

Comparison of computation times (in seconds) of different methods.

REFERENCES
(1) Shamai Gil and Kimmel Ron. Fast Classical Scaling.