Manifold learning from a corpus for 3D model retrieval

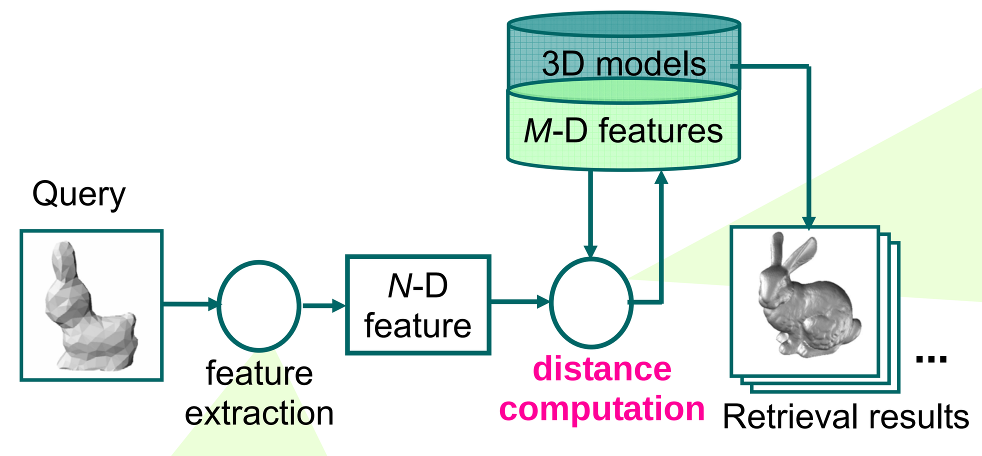
Ryutarou Ohbuchi and Jun Kobayashi
University of Yamanashi, Japan

Introduction

Insufficient retrieval performance!

- **Improve feature extraction**
  - A better feature.
  - A combination of features.
  - ...
- **Improve distance computation**
  - Adapt to a person/occasion
    - Relevance feedback (e.g., [Leifman05])
  - Adapt to a database
    - Find a subspace of 3D model features.
      - Linear subspace (PCA, ICA, MDS, …)
      - Non-linear subspace (Laplacian Eigenmaps, Locally Linear Embedding, …)

A traditional 3D model retrieval pipeline



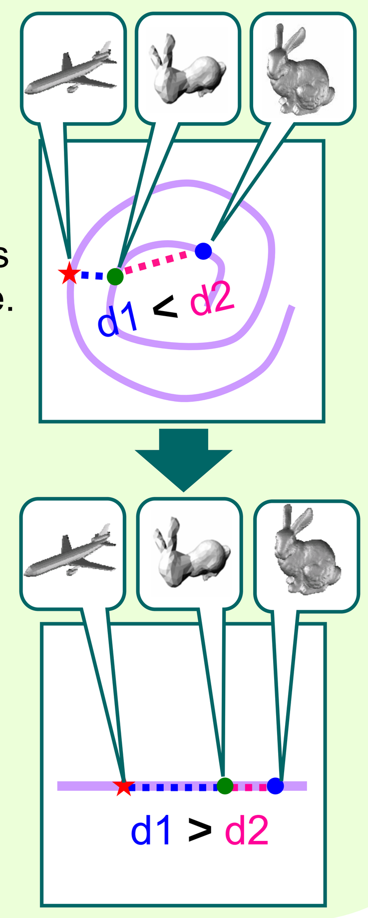
Better 3D shape features
Concise, expressive,…
Transformation invariant.
Robust (geometrical and topological noise.)
Details v.s overall shape.
…

Geodesic distance on a manifold

Intrinsically 1D features embedded in 2D space.

- **Estimate the manifold.**
- **Project the feature onto the manifold.**
- **Compute distance on the manifold.**

The features on 1D manifold.



Proposed method

Compute distances on a manifold learned from a corpus of 3D models.

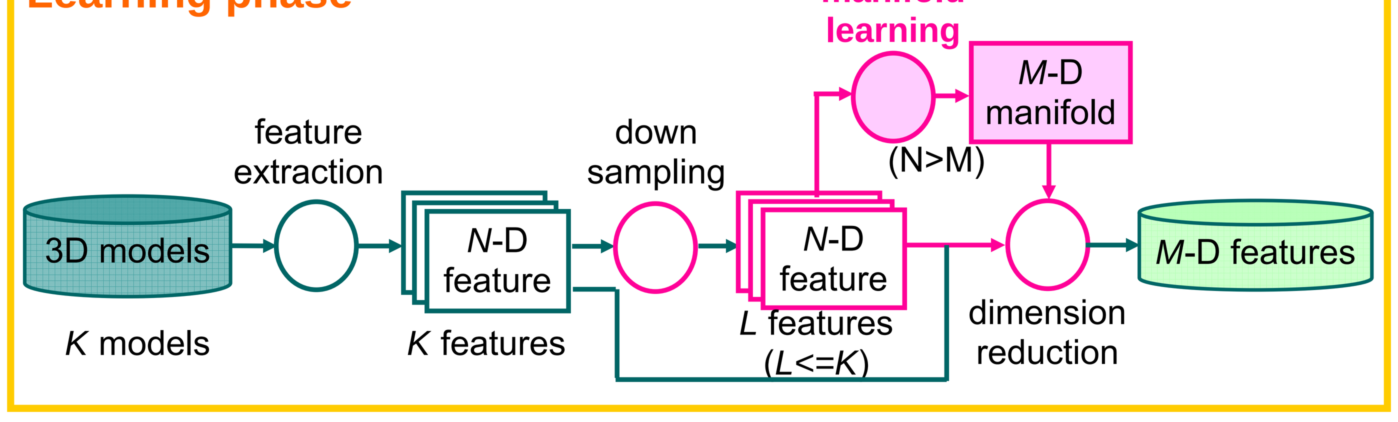
(1) Learning Phase

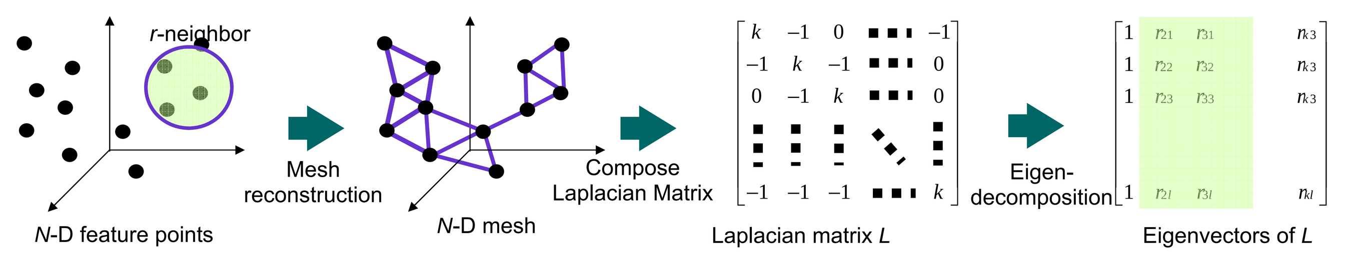
- Find the *M*-D manifold of 3D model features
  - *Unsupervised learning* from a corpus of 3D models.
    - *Laplacian Eigenmaps* (LE) by Belkin, et al. [Belkin02].
      - Reconstruction of an *N*-D mesh from the features.
      - Mesh-spectral analysis if the *N*-D mesh shape.
- Approximate the *M*-D manifold.
  - RBF network regression for a continuous mapping.
    - LE defined only at input points.
- Compute distance on the manifold.

(2) Retrieval PhaseP

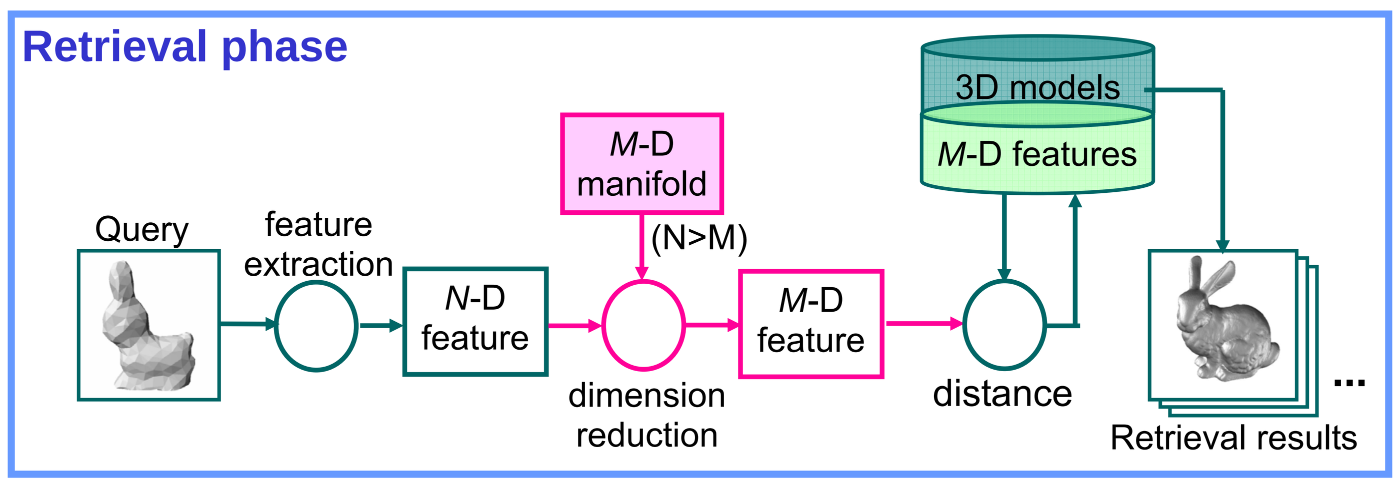
- Compute distance on the manifold.
  - Retrieve the top matches.

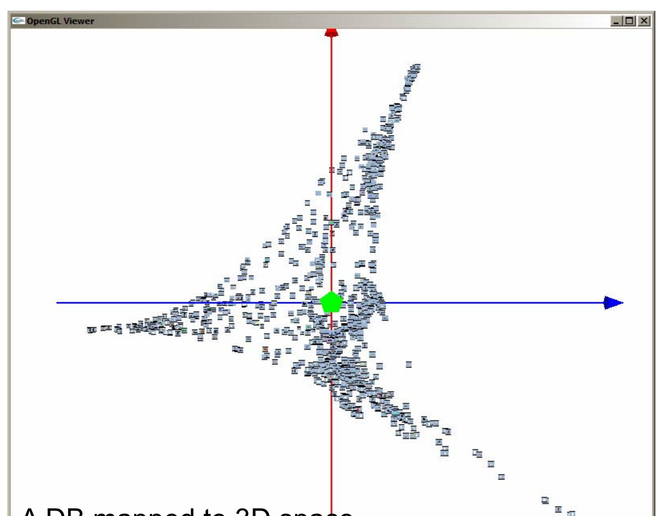
Learning phase



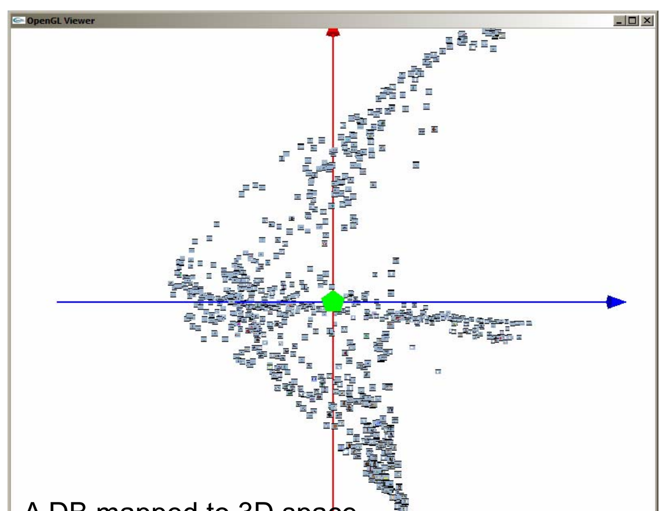


Retrieval phase





A DB mapped to 3D space.
Original feature: AAD, 256D.



A DB mapped to 3D space.
Original feature: SPRH, 625D.

Experiments and Results

(1) Manifold dimension and retrieval performance

- No clear intrinsic dimension found.
  - AAD [Ohbuchi05] increasing monotonously.
  - SPRH [Wahl03] peaking out at about 200 dim.

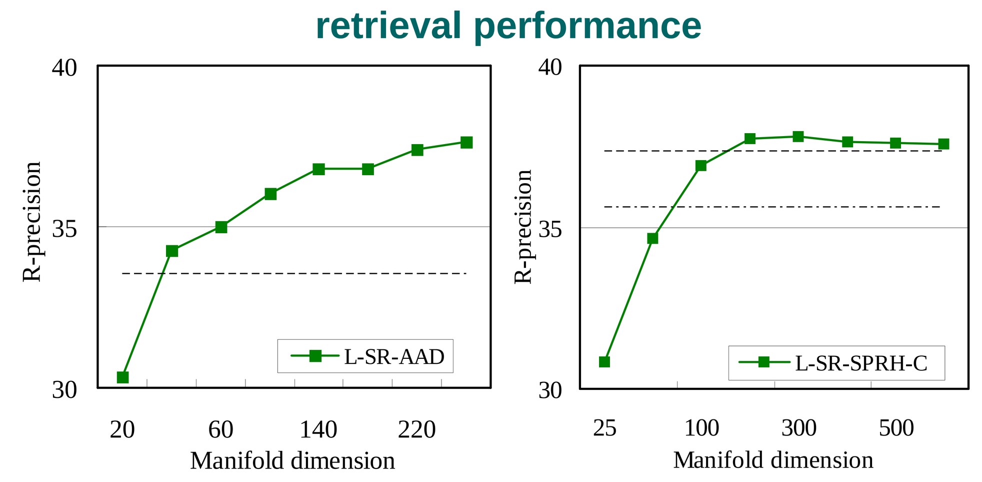
(2) Training set size and retrieval performance

- Learned versions outperform the originals.
  - About 5% gain due to learning.
  - e.g., Trained, multiresolution version of SPRH outperforms LFD [Chen03].

Base feature	MR/ SR	Train?	Feature name	*L*	RP [%]	11P [%]
AAD	SR	No	SR-AAD	-	33.5	36.3
		Yes	L-SR-AAD	4000	37.1	39.5
	MR	No	MR-AAD	-	40.3	43.9
		Yes	L-MR-AAD	4000	43.9	46.8
SPRH	SR	No	SR-SPRH-K	-	37.4	40.1
		No	SR-SPRH-C	-	35.6	38.4
		Yes	L-SR-SPRH-C	5000	37.8	38.7
	MR	No	MR-SPRH-C	-	42.5	45.7
		Yes	L-MR-SPRH-C	5000	47.5	50.1
LFD	-	-	-	-	45.9	49.3

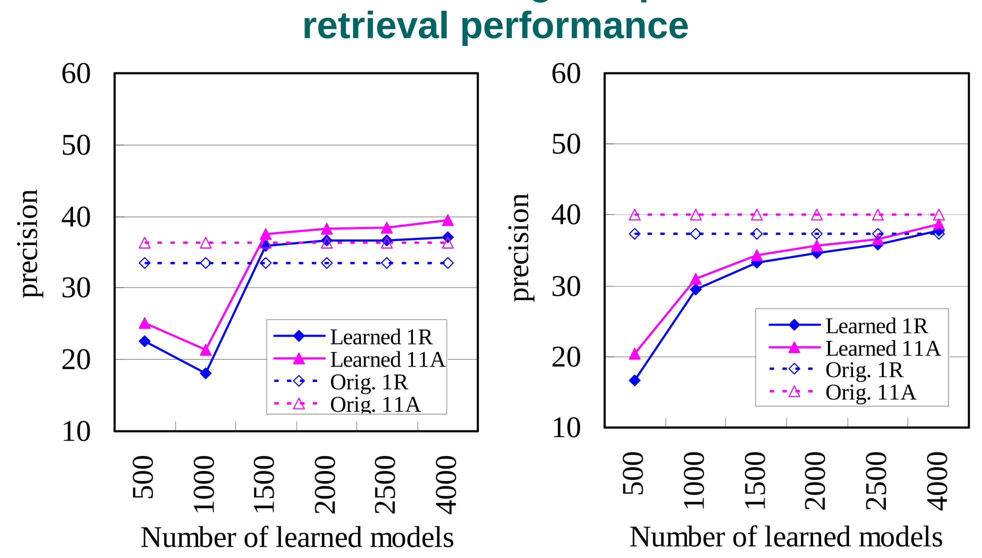
SR: Single Resolution MR: Multi Resolution
L: Number of training samples.

Manifold dimension vs. retrieval performance

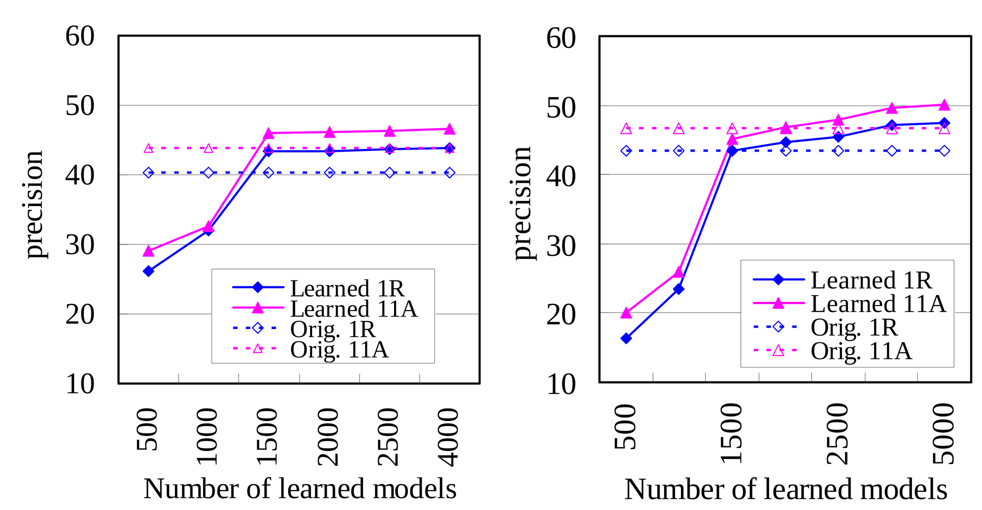


(a) SR-AAD (b) SR-SPRH

Number of training samples vs. retrieval performance



(a) SR-AAD (b) SR-SPRH



(c) MR-AAD (d) MR-SPRH

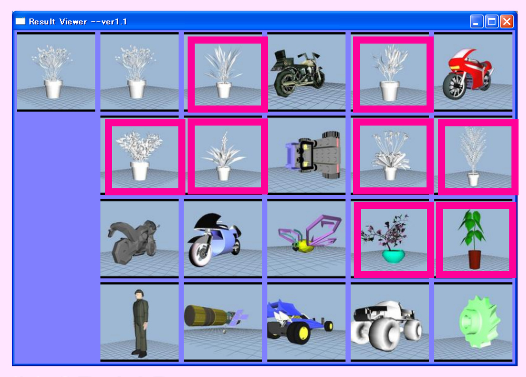
Conclusion and Future Work

- Learning the subspace of 3D model features improves 3D model retrieval.
  - For some shape features, for a learning algorithm.
- Future work
  - Other shape features?
  - Different subspace learning algorithms (e.g., LLE, Isomap, etc.)?
  - Larger training set.
    - Currently limited by (spatial) computational cost.

Retrieval examples

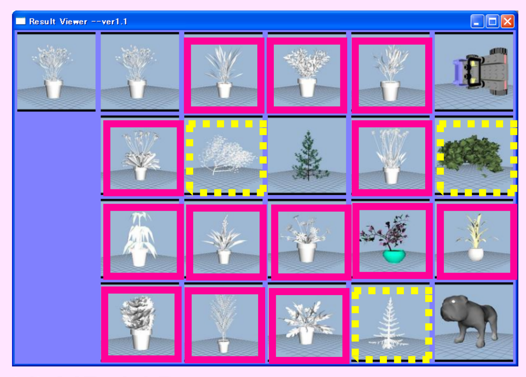
Before learning | After learning

(a) SR-AAD
(HR=4/19, R=0/19, |C|=25)

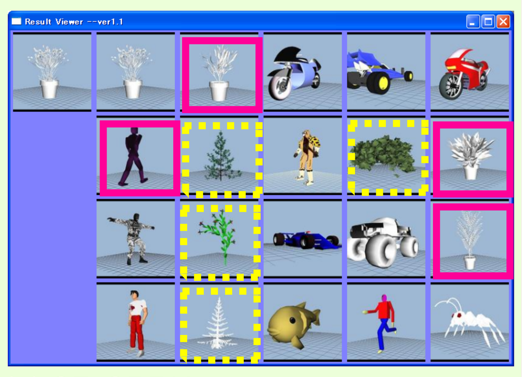


(b) SR-AAD
(HR=8/19, R=0/19, |C|=25.)

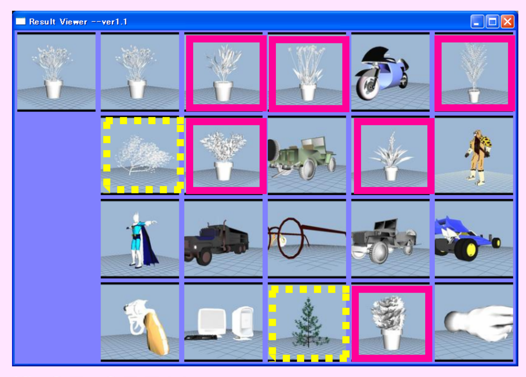
(c) MR-AAD
(HR=9/19, R=13/19, $|C_{HR}|$=25.)



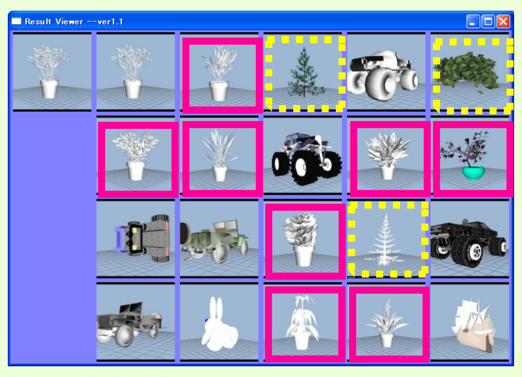
(d) MR-AAD
(HR=13/19, R=16/19, $|C_{HR}|$=25.)



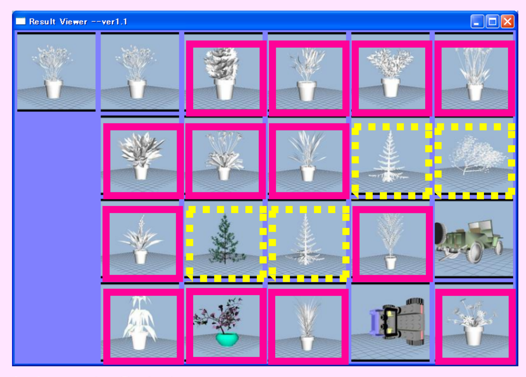
(e) SR-SPRH
(HR=3/19, R=7/19, $|C_{HR}|$=25.)



(f) SR-SPRH
(HR=6/19, R=8/19, $|C_{HR}|$=25.)



(g) MR-SPRH
(HR=8/19, R=11/19, $|C_{HR}|$=25)



(h) MR-SPRH
(HR=13/19, R=17/19, $|C_{HR}|$=25.)