Data-Driven Geometry Processing
3D Deep Learning II

Qixing Huang
March 28th 2017
3D Surface Representations

- Triangular mesh
- Implicit surface
- Light Field Representation
- Part-based models
- Point cloud
Matching in Embedding Spaces
[CVPR’ 16]
Existing methods usually follow a two-step approach (e.g., SIFT flow)

- Local descriptor computation

- Dense pixel labeling via MRF inference
  - Preserve descriptors
  - Preserve smoothness
Issues of such two-step approach

Partial similarity

Inefficient when matching multiple objects
Embedding --- establishing correspondences in the embedding space

Spectral embedding [Liu et al. 06]

Sensitive to 1) partial similarity, and 2) geometric and topological changes
Properties of the desired embedding space

Corresponding points are matched in the embedding space.

Embedding preserves continuity.
The benefits of object embedding

- Correspondences become nearest neighbor query
  - Efficiency for multiple object matching
    $O(n)$ embeddings + $O(n^2)$ queries
  - Partial similarity
  - Fuzzy correspondences
The biggest message of deep neural networks

- Approximate any function given sufficient data
Focus on depth images

- Scanning devices generate depth images

- Complete shape embedding are aggregated from depth image embeddings
  - 3D convolution is not ready yet
Architecture

<table>
<thead>
<tr>
<th>layer</th>
<th>image</th>
<th>conv</th>
<th>max</th>
<th>conv</th>
<th>max</th>
<th>$2 \times$ conv</th>
<th>conv</th>
<th>max</th>
<th>$2 \times$ conv</th>
<th>int</th>
<th>conv</th>
</tr>
</thead>
<tbody>
<tr>
<td>filter-stride</td>
<td>-</td>
<td>11-4</td>
<td>3-2</td>
<td>5-1</td>
<td>3-2</td>
<td>3-1</td>
<td>3-1</td>
<td>3-2</td>
<td>1-1</td>
<td>-</td>
<td>3-1</td>
</tr>
<tr>
<td>channel</td>
<td>1</td>
<td>96</td>
<td>96</td>
<td>256</td>
<td>256</td>
<td>384</td>
<td>256</td>
<td>256</td>
<td>4096</td>
<td>4096</td>
<td>16</td>
</tr>
<tr>
<td>activation</td>
<td>-</td>
<td>relu</td>
<td>lrm</td>
<td>relu</td>
<td>lrm</td>
<td>relu</td>
<td>idn</td>
<td>relu</td>
<td>idn</td>
<td>relu</td>
<td></td>
</tr>
<tr>
<td>size</td>
<td>512</td>
<td>128</td>
<td>64</td>
<td>64</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>16</td>
<td>16</td>
<td>128</td>
<td>512</td>
</tr>
<tr>
<td>num</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>64</td>
<td>64</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

The input is a depth image

The output is a per-pixel descriptor (dim 16)

Convolution + Deconvolution
Training data

- 4 animation sequences (dense correspondences)
- 2500 shapes from Yobi3D (33 feature points)
Direct versus Indirect

• Descriptor learning (e.g., triplet loss [Schroff et al. 15])

• Classification loss (e.g., the second last layer of AlexNet)
We employ a classification loss

Classes are defined in terms of super-patches

We use multiple segmentations --- so the probability of two points belong to the same segment is related to their distance
We employ the classification loss

\[
\{w_i^*\}, w^* = \arg \min_{\{w_i\}, w} \sum_{i=1}^{M} l(w_i, w)
\]
Evaluation on the FAUST dataset

Cumulative error distribution, intra-subject
Evaluation on the FAUST dataset

Cumulative error distribution, inter-subject
Multi-view 3D Models from Single Images
With a Convolutional Network [ECCV’ 16]
Fig. 5. Depth map predictions (top row) and the corresponding ground truth (bottom row). The network correctly estimates the shape.
Multi-view 3D Models from Single Images with a Convolutional Network

Maxim Tatarchenko, Alexey Dosovitskiy, Thomas Brox

Department of Computer Science
University of Freiburg
{tatachmn, dosovits, brox}@cs.uni-freiburg.de

ECCV 2016
Perspective Transformer Nets: Learning Single-View 3D Object Reconstruction without 3D Supervision [Yan et al. 16]
Figure 1: (a) Understanding 3D object from learning agent’s perspective; (b) Single-view 3D volume reconstruction with perspective transformation. (c) Illustration of perspective projection. The minimum and maximum disparity in the screen coordinates are denoted as $d_{min}$ and $d_{max}$.

\[
\mathcal{L}_{vol}(I^{(k)}) = \left\| f(I^{(k)}) - V \right\|_2^2
\]

\[
\mathcal{L}_{proj}(I^{(k)}) = \sum_{j=1}^{n} \mathcal{L}_{proj}^{(j)}(I^{(k)}; S^{(j)}, \alpha^{(j)}) = \frac{1}{n} \sum_{j=1}^{n} \left\| P(f(I^{(k)}); \alpha^{(j)}) - S^{(j)} \right\|_2^2
\]

\[
\mathcal{L}_{comb}(I^{(k)}) = \lambda_{proj} \mathcal{L}_{proj}(I^{(k)}) + \lambda_{vol} \mathcal{L}_{vol}(I^{(k)})
\]
Learning Semantic Deformation Flows with 3D Convolutional Networks [Yumer and Mitra 2016]
Semantic Scene Completion from a Single Depth Image [Song et al. 17]
A Point Set Generation Network for 3D Object Reconstruction from a Single Image [Fan, Su, Guibas, 2017]
Input

Reconstructed 3D point cloud
Network Architecture
Distance Metrics

• Chamfer distance

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

• Earth Mover’s distance

\[ d_{EMD}(S_1, S_2) = \min_{\phi : S_1 \rightarrow S_2} \sum_{x \in S_1} \| x - \phi(x) \|_2 \]

\[ \phi : S_1 \rightarrow S_2 \text{ is a bijection} \]
Visual results
CD (Left) versus EMD (Right)
GRASS: Generative Recursive Autoencoders for Shape Structures [Li, Xu, Chaudhuri, Yumer, Zhang, Guibas, SIGGRAPH’ 17]
Huge Variety of (Attributed) Graphs

- Arbitrary numbers/types of vertices (parts), arbitrary numbers of connections (adjacencies/symmetries)

- For linear graphs (chains) of arbitrary length, we can use a recurrent neural network (RNN/LSTM)
Key Insight

- Edges of a graph can be collapsed sequentially to yield a hierarchical structure

- Looks like a parse tree for a sentence!
Recursive Neural Network (RvNN)

- Repeatedly merge two nodes into one
- Each node has an n-D feature vector, computed recursively

\[ p = f(W [c_1; c_2] + b) \]
Recursively Merging Parts
Different types of merges, varying cardinalities!

- Adjacency
- Translational symmetry
- Rotational symmetry
- Reflectional symmetry
Training with Reconstruction Loss

- Learn weights from a variety of randomly sampled merge orders for each box structure
Results: Shape interpolation
Discussion