Lecture 7:
Deep Learning on Extrinsic Geometry

Instructor: Hao Su
Jan 30, 2018

slides credits: Justin Solomon, Chengcheng Tang
3D deep learning tasks

3D geometry analysis

Classification

Parsing (object/scene)

Correspondence
3D deep learning tasks

3D synthesis

Monocular 3D reconstruction

Shape completion

Shape modeling
3D deep learning algorithms (by representations)

Multi-view

Volumetric

[Su et al. 2015]
[Kalogerakis et al. 2016]
...

[Maturana et al. 2015]
[Wu et al. 2015] (GAN)
[Qi et al. 2016]
[Liu et al. 2016]
[Wang et al. 2017] (O-Net)
[Tatarchenko et al. 2017] (OGN)
...
3D deep learning algorithms (by representations)

Multi-view

- Point cloud
  - [Qi et al. 2017] (Pointf)
  - [Fan et al. 2017] (Point)

- Mesh (Graph CNN)
  - [Defferard et al. 2016]
  - [Henaff et al. 2015]
  - [Yi et al. 2017] (SyncSpecCNN)

Volumetric

- Part assembly
  - [Tulsiani et al. 2017]
  - [Li et al. 2017] (GRASS)

- 3D shape model rendered with different virtual cameras

- 2D rendered images
Cartesian product space of “task” and “representation”

3D geometry analysis

3D synthesis
Deep Learning on Point Cloud Data
Agenda

• Why point cloud?
• Comparison of point cloud
• Point cloud generation by deep learning
Agenda

- Why point cloud?
- Comparison of point cloud
- Point cloud generation by deep learning
Point Clouds

- Simplest representation: **only points**, no connectivity
- Collection of \((x,y,z)\) coordinates, possibly with normals
- Points with orientation are called **surfels**
Point Clouds

- Simplest representation: **only points**, no connectivity
- Collection of (x,y,z) coordinates, possibly with normals
- Points with orientation are called **surfels**
- Severe limitations:
  - no simplification or subdivision
  - no direct smooth rendering
  - no topological information
Point Clouds

- Simplest representation: only points, no connectivity
- Collection of (x,y,z) coordinates, possibly with normals
- Points with orientation are called surfels
- Severe limitations:
  - no simplification or subdivision
  - no direct smooth rendering
  - no topological information
  - weak approximation power: $O(h)$ for point clouds
    - need square number of points for the same approximation power as meshes
Point Clouds

- Simplest representation: **only points**, no connectivity
- Collection of (x,y,z) coordinates, possibly with normals
- Points with orientation are called **surfels**
- Severe limitations:
  - **no** Simplification or subdivision
  - **no** direct smooth rendering
  - **no** topological information
  - weak approximation power
  - noise and outliers
Why Point Clouds?

1) Typically, that’s the only thing that’s available
2) Isolation: sometimes, easier to handle (esp. in hardware).

Fracturing Solids

Meshless Animation of Fracturing Solids
Pauly et al., SIGGRAPH ‘05

Fluids

Adaptively sampled particle fluids,
Adams et al. SIGGRAPH ‘07
Why Point Clouds?

- Typically, that’s the only thing that’s available
  Nearly all 3D scanning devices produce point clouds
Agenda

• Why point cloud?
• Comparison of point cloud
• Point cloud generation by deep learning
Point cloud as samples

- Point cloud can be thought as a representation of prob. distribution

- Compare point cloud is to compare underlying distributions
Motivating Question

Which is closer, 1 or 2?
Motivating Question

Which is closer, 1 or 2?
Which is closer, 1 or 2?

$p(x, y)$

$p_1(x, y)$  

$p_2(x, y)$

Which is closer, 1 or 2?
Typical Measurement

$p_1(x)$

$p_2(x)$

$p_1(x) - p_2(x)$

$L^p$ norm

KL divergence

d$(p_1, p_2)$
Returning to the Question

Which is closer, 1 or 2?
Returning to the Question

$p(x, y)$

Query

$p_1(x, y)$  1

$p_2(x, y)$  2

Neither!  Equidistant.
What’s Wrong?

Measured overlap, not displacement.
Optimal Transport

Geometric theory of probability

Image courtesy M. Cuturi

[McCann’95] Interpolant

Over a metric space
Compare in this direction

Not in this direction
Alternative Idea

Match mass from the distributions
Transportation Matrix

- **Supply** distribution $p_0$
- **Demand** distribution $p_1$

\[
\begin{align*}
T & \geq 0 \\
T1 &= p_0 \\
T^\top 1 &= p_1
\end{align*}
\]
Earth Mover’s Distance

\[
\min_T \sum_{i,j} T_{ij} d(x_i, x_j) \\
s.t. \sum_j T_{ij} = p_i \\
\sum_i T_{ij} = q_j \\
T \geq 0
\]

- Starts at \( p \)
- Ends at \( q \)
- Positive mass

\( m \cdot d(x, y) \)
EMD is a metric when \( d(x,y) \) satisfies the triangle inequality.

“The Earth Mover's Distance as a Metric for Image Retrieval”

Revised in:
“Ground Metric Learning”
Cuturi and Avis; JMLR 15 (2014)
Comparing histogram descriptors

http://web.mit.edu/vondrick/ihog/
Discrete Perspective

Matching in disguise? Min-cost flow
Algorithm for Small-Scale Problems

- **Step 1:** Compute $D_{ij}$

- **Step 2:** Solve linear program
  - Simplex
  - Interior point
  - Hungarian algorithm
  - ...

Transportation Matrix Structure

Matches bins

Underlying map!
$p$-Wasserstein Distance

$$W_p(\mu, \nu) \equiv \min_{\pi \in \Pi(\mu, \nu)} \left( \int \int_{X \times X} d(x, y)^p \, d\pi(x, y) \right)^{1/p}$$

- Shortest path distance
- Expectation

General cost: “Monge-Kantorovich problem”

Continuous analog of EMD

Agenda

- Why point cloud?
- Comparison of point cloud
- Point cloud generation by deep learning
3D perception from a single image
Monocular vision

a typical prey

Pigeon

Binocular vision

Monocular vision

a typical predator

Owl

Cited from https://en.wikipedia.org/wiki/Binocular_vision
A psychological evidence – mental rotation

Mental Images and Their Transformations
Shepard

by Roger N. Shepard, National Science Medal Laurate
and Lynn Cooper, Professor at Columbia University

Hao Su
Visual cues are complicated

- contrast
- color
- texture
- motion
- symmetry
- part
- category-specific 3D knowledge

......
Status review of monocular vision algorithms

- Shape from X (texture, shading, …)

[Horn, 1989]

[Kender, 1979]
Status review of monocular vision algorithms

- **Shape from X** (texture, shading, …)
  - [Horn, 1989]
  - [Kender, 1979]

- **Learning-based** (from small data)
  - Hoiem et al, ICCV’05
  - Saxena et al, NIPS’05
  - …
  - large planes
  - fine structure
  - topological variation
  - …

Hao Su
Lecture 7 - 42
Status review of monocular vision algorithms

- Shape from X (texture, shading, …)
- Learning-based (from small data)

Hoiem et al, ICCV’05
Saxena et al, NIPS’05

[Strong assumption]
[Not robust]

[Horn, 1989]
[Kender, 1979]

- large planes
- fine structure
- topological variation
- …
Data-driven 2D-3D lifting

Many 3D objects

A priori knowledge of the 3D world
Our result: 3D reconstruction from real Images

CVPR 2017, A Point Set Generation Network for 3D Object Reconstruction from a Single Image

Input

Reconstructed 3D point cloud

CVPR '17, Point Set Generation
Our result: 3D reconstruction from real Images

CVPR 2017, A Point Set Generation Network for 3D Object Reconstruction from a Single Image

Input

Reconstructed 3D point cloud

CVPR '17, Point Set Generation
3D point clouds

✓ Flexible
  • a few thousands of points can precisely model a great variety of shapes

CVPR '17, Point Set Generation
3D point clouds

✓ Flexible
  • a few thousands of points can precisely model a great variety of shapes

✓ Geometrically manipulable
  • deformable
  • interpolable, extrapolable
  • convenient to impose structural constraints

CVPR ’17, Point Set Generation
Pipeline

CVPR '17, Point Set Generation
Pipeline

2K object categories
200K shapes
~10M image/point set pairs

Groundtruth point set

CVPR ’17, Point Set Generation
Pipeline

Shape predictor

Prediction
\[
\begin{aligned}
(x'_1, y'_1, z'_1) \\
(x'_2, y'_2, z'_2) \\
\vdots \\
(x'_n, y'_n, z'_n)
\end{aligned}
\]

Groundtruth point set

CVPR ’17, Point Set Generation
Pipeline

Shape predictor ($f$)

Prediction

$\{ (x'_1, y'_1, z'_1) \}
\{ (x'_2, y'_2, z'_2) \}
\ldots$

$\{ (x'_n, y'_n, z'_n) \}$

A set is invariant up to permutation

Groundtruth point set

CVPR ’17, Point Set Generation

Hao Su

Lecture 7 - 52
Pipeline

Shape predictor $(f)$

Prediction

\[
\begin{align*}
(x'_1, y'_1, z'_1) \\
(x'_2, y'_2, z'_2) \\
&\quad \cdots \\
(x'_{n}, y'_{n}, z'_{n})
\end{align*}
\]

Loss on sets $(L)$

Groundtruth point set

CVPR '17, Point Set Generation

Hao Su
Pipeline

Shape predictor $(f)$

Prediction

\[
\begin{align*}
& (x'_1, y'_1, z'_1) \\
& (x'_2, y'_2, z'_2) \\
& \vdots \\
& (x'_n, y'_n, z'_n)
\end{align*}
\]

Loss on sets $(L)$

Groundtruth point set

CVPR ’17, Point Set Generation

Hao Su

Lecture 7 - 54
Set comparison

Given two sets of points, measure their discrepancy

CVPR ’17, Point Set Generation
Set comparison

Given two sets of points, measure their discrepancy

Key challenge: correspondence problem

CVPR ’17, Point Set Generation
Correspondence (I): optimal assignment

Given two sets of points, measure their discrepancy

a.k.a Earth Mover’s distance (EMD)

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

where \( \phi : S_1 \rightarrow S_2 \) is a bijection.

CVPR ’17, Point Set Generation
Correspondence (II): closest point

Given two sets of points, measure their discrepancy

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

a.k.a Chamfer distance (CD)

CVPR '17, Point Set Generation
Required properties of distance metrics

Geometric requirement

Computational requirement

CVPR ’17, Point Set Generation
Required properties of distance metrics

Geometric requirement

• Reflects natural shape differences
• Induce a nice space for shape interpolations

Computational requirement

CVPR ’17, Point Set Generation
How distance metric affects learning?

A fundamental issue: inherent ambiguity in 2D-3D dimension lifting

CVPR ’17, Point Set Generation
How distance metric affects learning?

A fundamental issue: inherent ambiguity in 2D-3D dimension lifting
How distance metric affects learning?

A fundamental issue: inherent ambiguity in 2D-3D dimension lifting

CVPR '17, Point Set Generation
How distance metric affects learning?

A fundamental issue: inherent ambiguity in 2D-3D dimension lifting

- By loss minimization, the network tends to predict a “mean shape” that averages out uncertainty

CVPR ’17, Point Set Generation
Distance metrics affect mean shapes

The mean shape carries characteristics of the distance metric

\[
\bar{x} = \arg\min_x \mathbb{E}_{s \sim S}[d(x, s)]
\]

continuous hidden variable (radius)

Input  EMD mean  Chamfer mean

CVPR ’17, Point Set Generation
Mean shapes from distance metrics

The mean shape carries characteristics of the distance metric

\[ \bar{x} = \arg\min_x \mathbb{E}_{s \sim S}[d(x, s)] \]

Continuous hidden variable (radius)

Discrete hidden variable (add-on location)

Input

EMD mean

Chamfer mean

CVPR ’17, Point Set Generation
Comparison of predictions by EMD versus CD

Input

EMD

Chamfer

CVPR '17, Point Set Generation
Required properties of distance metrics

Geometric requirement

- Reflects natural shape differences
- Induce a nice space for shape interpolations

Computational requirement

- Defines a loss function that is numerically easy to optimize

CVPR ’17, Point Set Generation
Computational requirement of metrics

To be used as a loss function, the metric has to be

- **Differentiable** with respect to point locations
- **Efficient** to compute
Computational requirement of metrics

- **Differentiable** with respect to point location

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

- Simple function of coordinates
- In general positions, the correspondence is unique
- **With infinitesimal movement, the correspondence does not change**

**Conclusion:** differentiable almost everywhere
Computational requirement of metrics

• Differentiable with respect to point location

• For many algorithms (sorting, shortest path, network flow, ...),

• an infinitesimal change to model parameters

CVPR '17, Point Set Generation
Computational requirement of metrics

- **Efficient** to compute

  Chamfer distance: trivially parallelizable on CUDA
  Earth Mover’s distance (optimal assignment):
  - We implement a **distributed** approximation algorithm on CUDA
  - Based upon [Bertsekas, 1985], \((1 + \epsilon)\)-approximation

*CVPR ’17, Point Set Generation*
Pipeline

Deep network $(f)$

Sample

Prediction

Loss on sets $(L)$

CVPR '17, Point Set Generation

Hao Su

Lecture 7 - 73
Deep neural network

- Universal function approximator
  - A cascade of layers

CVPR '17, Point Set Generation
Deep neural network

A cascade of layers
Each layer conducts a simple transformation (parameterized)

Universal function approximator

CVPR ’17, Point Set Generation
Deep neural network

- A cascade of layers
- Each layer conducts a simple transformation (parameterized)
- Millions of parameters, has to be fitted by many data
Pipeline

Deep network 

\( f \)

Prediction

\[ \begin{cases} 
(x_1, y_1, z_1) \\
(x_2, y_2, z_2) \\
\vdots \\
(x_n, y_n, z_n) 
\end{cases} \]

Los s

\( \mathcal{L} \)

CVPR '17, Point Set Generation
Pipeline

Encoder → shape embedding space → Predictor → \{ (x_1, y_1, z_1), (x_2, y_2, z_2), \ldots, (x_n, y_n, z_n) \}

Los s

CVPR '17, Point Set Generation
Pipeline

Encoder \rightarrow \text{shape embedding space} \rightarrow \text{Predictor} \rightarrow \{ (x_1, y_1, z_1), (x_2, y_2, z_2), \ldots, (x_n, y_n, z_n) \} \rightarrow \text{Loss on sets} (L)

CVPR ’17, Point Set Generation
Pipeline

\[ L \]

Conv

Predictor

\[ \{(x_1, y_1, z_1), (x_2, y_2, z_2), \ldots, (x_n, y_n, z_n)\} \]

Loss on sets

CVPR '17, Point Set Generation

Hao Su

Lecture 7 - 80
Pipeline

CVPR '17, Point Set Generation
Many local structures are common
- e.g., planar patches, cylindrical patches
- strong local correlation among point coordinates
Many local structures are common
- e.g., planar patches, cylindrical patches
- strong local correlation among point coordinates

Also some intricate structures
- points have high local variation

CVPR ’17, Point Set Generation