CS468: 3D Deep Learning

image → class label
part label

Hao Su
Stanford University
May 10, 2017
Agenda

• Deep Learning Review

• Overview of 3D Deep Learning

• Deep Learning on Multi-view Representation
Agenda

• Deep Learning Review

• Overview of 3D Deep Learning

• Deep Learning on Multi-view Representation
Neural network: A compositional function

Model: Multi-Layer Perceptron (MLP) \[ y' = W_3 f(W_2 f(W_1 x + b_1) + b_2) + b_3 \]

Loss function: L2 loss \[ l(y, y') = (y - y')^2 \]

Optimization: Gradient descent \[ W = W - \eta \frac{\partial L}{\partial W} \]
Universal approximation theorem

A three-layer network approximates any continuous function

Let \( \varphi(\cdot) \) be a nonconstant, **bounded**, and **monotonically-increasing continuous** function. Let \( I_m \) denote the \( m \)-dimensional **unit hypercube** \([0, 1]^m\). The space of continuous functions on \( I_m \) is denoted by \( C(I_m) \). Then, given any function \( f \in C(I_m) \) and \( \varepsilon > 0 \), there exists an integer \( N \), real constants \( v_i, b_i \in \mathbb{R} \) and real vectors \( w_i \in \mathbb{R}^m \), where \( i = 1, \ldots, N \), such that we may define:

\[
F(x) = \sum_{i=1}^{N} v_i \varphi \left( w_i^T x + b_i \right)
\]

as an approximate realization of the function \( f \) where \( f \) is independent of \( \varphi \); that is,

\[
|F(x) - f(x)| < \varepsilon
\]

for all \( x \in I_m \). In other words, functions of the form \( F(x) \) are **dense** in \( C(I_m) \).
A three-layer network approximates any continuous function

Let \( \varphi(\cdot) \) be a nonconstant, \textbf{bounded}, and \textbf{monotonically-increasing} \textbf{continuous} function. Let \( I_m \) denote the \( m \)-dimensional \textbf{unit hypercube} \([0, 1]^m\). The space of continuous functions on \( I_m \) is denoted by \( C(I_m) \). Then, given any function \( f \in C(I_m) \), where \( i = 1, \ldots, k \),

\[
F(x) = \sum_{i=1}^{N} v_i \varphi(w_i \cdot x)
\]

as an approximate realization of the function \( f \) where \( f \) is independent of \( \varphi \); that is,

\[
|F(x) - f(x)| < \varepsilon
\]

for all \( x \in I_m \). In other words, functions of the form \( F(x) \) are \textbf{dense} in \( C(I_m) \).
Reduce the amount of parameters but not expressive power
Why deeper is good?

\[ a_i = \text{ReLU}(x - \theta_i) \]

\[
\text{ReLU}(x) = \begin{cases} 
0 & \text{if } x < 0 \\
 x & \text{if } x \geq 0
\end{cases}
\]

\[ y = \sum_i \text{ReLU}(x - \theta_i) \]

piece-wise linear, 6 knots
Why deeper is good?

\[
a_{1,i} = \text{ReLU}(x - \theta_i) \quad a_{2,i} = \text{ReLU}(x - \phi_i)
\]

\[
y = \sum_{1 \leq i \leq 3} \text{ReLU}([ \sum_{1 \leq j \leq 3} \text{ReLU}(x - \theta_j)] - \phi_i)
\]

piece-wise linear, can have 9 knots!

\[
\theta_i + \phi_j
\]

3 × 3
Why deeper is good?

Interpretation I: With the same number of parameters, create combinatorial data flow
Why deeper is good?

Interpretation I: With the same number of parameters, create combinatorial data flow

Interpretation II: Abstract data progressively
Convolutional Neural Network: LeNet (1998 by LeCun et al.)

One of the first successful applications of CNN.
Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]
Optimization

• **Basics:** Gradient descent, SGD, mini-batch SGD, Momentum, Adam, learning rate decay

• **Other Ingredients:** Data augmentation, Regularization, Dropout, Xavier initialization, Batch normalization
Summary

• Can approximate arbitrarily complicated continuous function
  • classification, segmentation, …

• Because neural networks have high capacity (millions of parameters)

• To estimate so many parameters, big data is needed

• Deep architecture allows building expressive models with less params
• Deep Learning Review

• Overview of 3D Deep Learning

• Deep Learning on Multi-view Representation
Broad applications of 3D data

Robotics
Broad applications of 3D data

Robotics

Augmented Reality
Broad applications of 3D data

Robotics

Augmented Reality

Autonomous driving
Broad applications of 3D data

Robotics

Augmented Reality

Autonomous driving

Medical Image Processing
History of 3D deep learning

• A field with very short history — starting from 2015
• But very active due to huge industry interests!
Fundamental challenges of 3D deep learning

Can we directly apply CNN on 3D data?
Fundamental challenges of 3D deep learning

Can we directly apply CNN on 3D data?

Convolution needs an underlying structure

\[(f * g)[n] = \sum_{m=-M}^{M} f[n-m]g[m]\]
Fundamental challenges of 3D deep learning

Images: Regular data

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>14</th>
<th>23</th>
<th>20</th>
<th>12</th>
<th>33</th>
<th>44</th>
</tr>
</thead>
<tbody>
<tr>
<td>51</td>
<td>17</td>
<td>16</td>
<td>40</td>
<td>32</td>
<td>46</td>
<td>48</td>
<td>28</td>
</tr>
<tr>
<td>29</td>
<td>63</td>
<td>60</td>
<td>3</td>
<td>34</td>
<td>43</td>
<td>38</td>
<td>10</td>
</tr>
<tr>
<td>52</td>
<td>53</td>
<td>22</td>
<td>41</td>
<td>49</td>
<td>55</td>
<td>61</td>
<td>36</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>24</td>
<td>19</td>
<td>11</td>
<td>34</td>
<td>43</td>
<td>10</td>
</tr>
<tr>
<td>57</td>
<td>18</td>
<td>9</td>
<td>37</td>
<td>42</td>
<td>25</td>
<td>21</td>
<td>27</td>
</tr>
<tr>
<td>30</td>
<td>13</td>
<td>56</td>
<td>50</td>
<td>64</td>
<td>4</td>
<td>59</td>
<td>6</td>
</tr>
<tr>
<td>58</td>
<td>47</td>
<td>45</td>
<td>31</td>
<td>39</td>
<td>15</td>
<td>62</td>
<td>54</td>
</tr>
</tbody>
</table>
Fundamental challenges of 3D deep learning

3D: Irregular data

- unstructured point clouds
- parametric patches
- meshes
Fundamental challenges of 3D deep learning

3D: Irregular data

Need to design novel architectures!

unstructured point clouds

parametric patches

meshes
3D deep learning algorithms (by representations)

Multi-view

Volumetric

[Socher et al. 2012]
[Su et al. 2015]
[Wei et al. 2016]
[Kalogerakis et al. 2016]

[Maturana et al. 2015]
[Wu et al. 2015]
[Qi et al. 2016]
3D deep learning algorithms (by representations)

**Multi-view**

- Socher et al. 2012
- Su et al. 2015
- Wei et al. 2016
- Kalogerakis et al. 2016

**Volumetric**

- Maturana et al. 2015
- Wu et al. 2015
- Qi et al. 2016

**Point cloud**

- Qi et al. 2017
- Fan et al. 2017

**Graph CNN**

- Defferard et al. 2016
- Henaff et al. 2015
- Yi et al. 2016

**Part assembly**

- Tulsiani et al. 2017
3D deep learning algorithms (by tasks)

3D geometry analysis

It is a chair!

Classification

Segmentation

Correspondence
3D deep learning algorithms (by tasks)

3D geometry synthesis

Monocular 3D reconstruction

Shape completion
This course proceeds by 3D representations

Two types of architectures

3D geometry analysis

3D geometry synthesis
• Deep Learning Review

• Overview of 3D Deep Learning

• Deep Learning on Multi-view Representation
General idea

- Convert irregular (3D) to regular (images)
- Leverage the huge CNN literature in image analysis
• Deep Learning Review

• Overview of 3D Deep Learning

• Deep Learning on Multi-view Representation
  • Classification
  • Segmentation
  • Reconstruction
Task: 3D classification

This is a chair!
Given an input shape
Render with multiple virtual cameras

Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "Multi-view Convolutional Neural Networks for 3D Shape Recognition", Proceedings of ICCV 2015
Traditional approach: feature+linear classifier

\[ y_c = w_c^T x_i + b_c \]

\[ p_{i,c} = \frac{\exp(w_c^T x_i + b_c)}{\sum \exp(w_j^T x_i + b_j)} \]

maximize \( \sum_w \sum_i \log p_{i,l_i} \)
The rendered images are passed through CNN\textsubscript{1} for image features

\textit{CNN\textsubscript{1}}: a ConvNet extracting image features

Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "Multi-view Convolutional Neural Networks for 3D Shape Recognition", Proceedings of ICCV 2015
All image features are combined by view pooling

...
... and then passed through $\text{CNN}_2$ and to generate final predictions

$\text{CNN}_1$.

$\text{CNN}_2$: a second ConvNet producing shape descriptors.

Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "Multi-view Convolutional Neural Networks for 3D Shape Recognition", Proceedings of ICCV 2015
Learning by fine-tuning

• Neural network optimization is non-convex

• In general, training from more data converges at a better local minima

• However, what if your training dataset $D$ is not big?
Learning by fine-tuning (cont.)

Pre-training
- Find a source of massive data $D'$ with similar statistics
- Learn the network parameters from $D'$

Fine-tuning
- Starting from the learned parameters on $D'$, minimize the network loss on $D$

A technique for transfer learning, quite effective in practice
Training: network parameters are pre-trained on image classification ...

Parameters initialized from VGG-M model [CHATFIELD14]


Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "Multi-view Convolutional Neural Networks for 3D Shape Recognition", Proceedings of ICCV 2015
... and then fine-tuned on 3D datasets

Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "Multi-view Convolutional Neural Networks for 3D Shape Recognition", Proceedings of ICCV 2015
Extract compact shape descriptor for other applications

Shape descriptor can be extracted from $\text{CNN}_2$, and a low-rank metric is learned w/ good&bad pairs
Experiments – classification & retrieval

On **ModelNet40**, compared against:

- 3 existing methods:
  - SPH, LFD, 3D ShapeNets
- 2 strong baselines:
  - Fisher vectors, CNN

<table>
<thead>
<tr>
<th>Method</th>
<th>Classification (Accuracy)</th>
<th>Retrieval (mAP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPH [16]</td>
<td>68.2%</td>
<td>33.3%</td>
</tr>
<tr>
<td>LFD [5]</td>
<td>75.5%</td>
<td>40.9%</td>
</tr>
<tr>
<td>3D ShapeNets [37]</td>
<td>77.3%</td>
<td>49.2%</td>
</tr>
<tr>
<td>FV, 12 views</td>
<td>84.8%</td>
<td>43.9%</td>
</tr>
<tr>
<td>CNN, 12 views</td>
<td>88.6%</td>
<td>62.8%</td>
</tr>
<tr>
<td>MVCNN, 12 views</td>
<td><strong>89.9%</strong></td>
<td>70.1%</td>
</tr>
<tr>
<td>MVCNN+metric, 12 views</td>
<td>89.5%</td>
<td><strong>80.2%</strong></td>
</tr>
<tr>
<td>MVCNN, 80 views</td>
<td>90.1%</td>
<td>70.4%</td>
</tr>
<tr>
<td>MVCNN+metric, 80 views</td>
<td><strong>90.1%</strong></td>
<td><strong>79.5%</strong></td>
</tr>
</tbody>
</table>

[credit: Hang Su]
Visualization of saliency across views

$$[\omega_1, \omega_2 \ldots \omega_K] = \left[ \frac{\partial F_c}{\partial I_1} \bigg|_S, \frac{\partial F_c}{\partial I_2} \bigg|_S \ldots \frac{\partial F_c}{\partial I_K} \bigg|_S \right]$$

[Credit: Hang Su]
How do you use multi-view approach for point cloud?

*Sphere* Rendering *Images*

[credit: CVPR 2016 spotlight]
Practical multi-view CNN

State-of-the-art performance for 3D mesh classification

Issues:

- What viewpoints to select? In particular, where shall we place the camera in a scene?
- What if the input is noisy and incomplete? e.g., point cloud
Agenda

• Deep Learning Review

• Overview of 3D Deep Learning

• Deep Learning on Multi-view Representation
  • Classification
  • **Segmentation**
  • Reconstruction
3D segmentation
Basic architecture

Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri, “3D Shape Segmentation with Projective Convolutional Networks”, CVPR2017
Basic architecture

Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri, “3D Shape Segmentation with Projective Convolutional Networks”, CVPR2017
Basic architecture

Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri, “3D Shape Segmentation with Projective Convolutional Networks”, CVPR2017
Basic architecture

\[ P(R_o) = \frac{1}{Z_o} \prod_f \phi_{\text{unary}}(R_f) \prod_{adj \ f, f'} \phi_{\text{adj}}(R_f, R_{f'}) \prod_{f, f'} \phi_{\text{dist}}(R_f, R_{f'}) \]

Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri, “3D Shape Segmentation with Projective Convolutional Networks”, CVPR2017
Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri, “3D Shape Segmentation with Projective Convolutional Networks”, CVPR2017
Fully Convolutional Network (FCN)

Segmentation:

Learning Deconvolution Network for Semantic Segmentation
Fully convolutional network (FCN) variations

Output scores
HxWxN

upsample

conv

Input image
HxWx3

Output scores
HxWxN

upconv

conv

Input image
HxWx3

Skip links

dilated
conv

Input image
HxWx3

Output scores
HxWxN
Performance

- unary factor only
- full CRF
- ground-truth
However, still worse than pure 3D based segmentation approach (graph CNN)

Challenges:

• Cannot process invisible points
• 2D proximity is different from 3D proximity after perspective projection
• Deep Learning Review

• Overview of 3D Deep Learning

• Deep Learning on Multi-view Representation
  • Classification
  • Segmentation
  • Reconstruction
Maxim Tatarchenko, Alexey Dosovitskiy, Thomas Brox,
“Multi-view 3D Models from Single Images with a Convolutional Network”,
ECCV2016
Results

(a)

(b)

(c)
Problems

• Predicting novel views is not easy
• Combining predictions across views is not easy