What can go here?

Evangelos Kalogerakis

[based on “SceneGraphNet: Neural Message Passing for 3D Indoor Scene Augmentation”, Yang Zhou, Zachary While, Evangelos Kalogerakis, ICCV 2019]
What can go here?

Evangelos Kalogerakis

[based on “SceneGraphNet: Neural Message Passing for 3D Indoor Scene Augmentation”, Yang Zhou, Zachary While, Evangelos Kalogerakis, ICCV 2019]
Goal: predict objects to populate indoor scenes
Goal: predict objects to populate indoor scenes

3D scene model

SceneGraphNet

$P(\text{object type} | \text{query, scene})$

- lamp
- alarm clock
- books
- vase

Query location
Goal: predict objects to populate indoor scenes

3D scene model

Query location

SceneGraphNet

P(object type|query, scene)

+ lamp

+ alarm clock

+ books

+ vase

+ object dimensions
Alternative goal: iterative scene synthesis
Alternative goal: iterative scene synthesis
Alternative goal: iterative scene synthesis
Motivation: AR/VR applications for indoor design
Prior work: learning-based scene completion

- Probabilistic grammars e.g., [Fisher et al. 12, Qi et al. 18, Purkait et al. 19]
Prior work: learning-based scene completion

• Probabilistic grammars e.g., [Fisher et al. 12, Qi et al. 18, Purkait et al. 19]

• View-based, image convolutional methods e.g., [Wang et al. 18, Ritchie et al. 19]
Prior work: learning-based scene completion

- Probabilistic grammars e.g., [Fisher et al. 12, Qi et al. 18, Purkait et al. 19]
- View-based, image convolutional methods e.g., [Wang et al. 18, Ritchie et al. 19]
- Volumetric methods e.g., [Song et al. 16]
Prior work: learning-based scene completion

• Probabilistic grammars e.g., [Fisher et al. 12, Qi et al. 18, Purkait et al. 19]

• View-based, image convolutional methods e.g., [Wang et al. 18, Ritchie et al. 19]

• Volumetric methods e.g., [Song et al. 16]

• GANs on matrix representation of scene objects e.g., [Zhang et al. 20]
Prior work: learning-based scene completion

- Probabilistic grammars e.g., [Fisher et al. 12, Qi et al. 18, Purkait et al. 19]
- View-based, image convolutional methods e.g., [Wang et al. 18, Ritchie et al 19]
- Volumetric methods e.g., [Song et al. 16]
- GANs on matrix representation of scene objects e.g., [Zhang et al. 20]
- Recursive autoencoders on trees graph representations e.g., [Li et al. 19]
Prior work: learning-based scene completion

• Probabilistic grammars e.g., [Fisher et al. 12, Qi et al. 18, Purkait et al. 19]

• View-based, image convolutional methods e.g., [Wang et al. 18, Ritchie et al 19]

• Volumetric methods e.g., [Song et al. 16]

• GANs on matrix representation of scene objects e.g., [Zhang et al. 20]

• Recursive autoencoders on trees graph representations e.g., [Li et al. 19]

• GNNs on object relation graphs [Wang et al. 19]
Key idea: capture surrounding object relations
Key idea: capture surrounding object relations

Short-range relations?
Key idea: capture surrounding object relations

Short-range relations?
Key idea: capture surrounding object relations

Short-range relations
+
Longer-range relations
Key idea: capture surrounding object relations

**Short-range relations**

+ 

**Longer-range relations**
Key Challenges

Short-range relations + Longer-range relations

How to learn these relations?
Key Challenges

Short-range relations
+
Longer-range relations

How to learn these relations?

How to weigh & aggregate them?
Pipeline

Scene Graph

Neural Message Passing
Neural Message Passing Pipeline

Scene Graph

Neural Message Passing
Scene graph construction
Nodes represent existing objects + queries
Edges represent relationships
Bed: “I am also here – I may affect decisions, too”
More nodes / edges

'next to'
'side table'
'on top of'

bed
More nodes / edges

side table 'next to'

side table 'next to'

bed

'next to'

'on top of'

?
Many more nodes / edges
Multiple edges for each pair of objects
More queries

- Bed
  - Side table
    - Desk
      - Laptop
    - Lamp
      - Lamp
  - Desk
    - Desk
Denser graphs yielded better predictions
Edge types

- “On top of”
- “Under”
- “On top of”
- “Under”
- “Surrounded-by”
- “Surrounding”
Edge types

- “On top of”
- “Under”
- “Surrounded-by”
- “Surrounding”
- “Next-to”
Edge types

- “On top of”
- “Under”
- “Surrounded-by”
- “Surrounding”
- “Next-to”
- “Co-occurring”
Scene Graph

Neural Message Passing

Inspired by [Gilmer et al. 2017]
Node representation

lamp
Node representation

lamp

$h_{\text{lamp}}$
Raw node features

{(x,y,z), size, object category, shape features}
Initial node representation

\[ h_{\text{lamp}}^{(\text{init})} \]

\{(x,y,z), \text{size, object category, shape features}\}
Initial node representation
Messages between objects
Messages between objects

- Lamp
- 'on top of'
- Side table

$h^{(init)}_{\text{lamp}}$ 

MLP$_{\text{msg}}$

Message $m_{\text{table} \rightarrow \text{lamp}}$
Edge weights

\{(x,y,z), size, object category, shape features\}

MLP_{att}

weight \ a_{table,lamp}

\{(x,y,z), size, object category, shape features\}

'on top of'

lamp

side table
Node update

- Lamp placed on top of side table.
- GRU node updated with input $m_{\text{table} \rightarrow \text{lamp}}^{(1)}$. 

Diagram illustrates the node update process with a lamp on a side table connected to a GRU node.
Node update

'On top of'

\[ m_{\text{table} \rightarrow \text{lamp}}^{(1)} \]

\[ m_{\text{bed} \rightarrow \text{lamp}}^{(1)} \]
Node update

bed
desk
lamp
side table

'on top of'

\[ m_{table \rightarrow lamp}^{(1)} \]
\[ m_{bed \rightarrow lamp}^{(1)} \]
\[ m_{desk \rightarrow lamp}^{(1)} \]
Node update

- bed
- desk
- lamp

'On top of'

- side table
- table
- bed
- desk

$g_{\text{lamp}}^{(1)}$

$\mathbf{m}_{\text{table}\to\text{lamp}}^{(1)}$

$\mathbf{m}_{\text{bed}\to\text{lamp}}^{(1)}$

$\mathbf{m}_{\text{desk}\to\text{lamp}}^{(1)}$
Node update
Node update

- bed
- desk
- lamp

'On top of'

MLP_{upd}

h^{(1)}_{lamp}

g^{(1)}_{lamp}

GRU

x

m^{(1)}_{table \rightarrow lamp}

m^{(1)}_{bed \rightarrow lamp}

m^{(1)}_{desk \rightarrow lamp}
Messages between objects

MLP_{msg}

$h^{(1)}_{lamp}$

$h^{(1)}_{side\ table}$

message $m^{(2)}_{table\rightarrow lamp}$
Node update

- Bed
- Desk
- Lamp
- Side table

'On top of'

$h_{lamp}^{(1)} \rightarrow$ Lamp

$g_{lamp}^{(2)} \rightarrow$ Lamp

$MLP_{upd} \rightarrow$ Lamp

$GRU \rightarrow$ Lamp

$m_{table \rightarrow lamp}^{(2)} \rightarrow$ Lamp

$m_{bed \rightarrow lamp}^{(2)} \rightarrow$ Lamp

$m_{desk \rightarrow lamp}^{(2)} \rightarrow$ Lamp
Decoding query node representations
Iterative synthesis

Incomplete Scene

Evaluate output probability in a dense grid of locations e.g., chair
Iterative synthesis

Incomplete Scene

Augmented Scene
Application: context-driven object recognition

\[ P(\text{object type} \mid \text{shape} ) \]

(multi-view CNN, Su et al. 2015)
Application: context-driven object recognition

\[
P(\text{object type}| \text{shape}) \quad \text{(multi-view CNN, Su et al. 2015)} \quad \text{P(\text{object type}| \text{query, scene})} \times \quad \text{P(\text{object type}| \text{shape})}
\]
Training

All MLP/GRU modules are learned from a large dataset of indoor rooms (SUNCG).

We remove random objects, then train our network to predict them.

[Song et al. 16]
SceneGraphNet Results
GRAINS Results  [Li et al. 19]
“Deep Priors” Results [Wang et al. 19]
Numerical Evaluation

How well do we predict objects removed from a test scene?

<table>
<thead>
<tr>
<th>Method</th>
<th>Correct object type</th>
</tr>
</thead>
<tbody>
<tr>
<td>“GRAINS” *</td>
<td>44.2%</td>
</tr>
</tbody>
</table>

* GRAINS: Generative Recursive Autoencoders for INdoor Scenes, TOG 2019
Numerical Evaluation

How well do we predict objects removed from a test scene?

<table>
<thead>
<tr>
<th>Method</th>
<th>Correct object type</th>
</tr>
</thead>
<tbody>
<tr>
<td>“GRAINS” *</td>
<td>44.2%</td>
</tr>
<tr>
<td>Wang et al.**</td>
<td>50.9%</td>
</tr>
</tbody>
</table>

* GRAINS: Generative Recursive Autoencoders for INdoor Scenes, TOG 2019
** Deep convolutional priors for indoor scene synthesis, SIGGRAPH 2018
Numerical Evaluation

How well do we predict objects removed from a test scene?

<table>
<thead>
<tr>
<th>Method</th>
<th>Correct object type</th>
</tr>
</thead>
<tbody>
<tr>
<td>“GRAINS” *</td>
<td>44.2%</td>
</tr>
<tr>
<td>Wang et al.**</td>
<td>50.9%</td>
</tr>
<tr>
<td>SceneGraphNet</td>
<td>67.3%</td>
</tr>
</tbody>
</table>

* GRAINS: Generative Recursive Autoencoders for INdoor Scenes, TOG 2019
** Deep convolutional priors for indoor scene synthesis, SIGGRAPH 2018
Numerical Evaluation

How well do we predict objects removed from a test scene?

<table>
<thead>
<tr>
<th>Method</th>
<th>Correct object within top-5 predictions</th>
</tr>
</thead>
<tbody>
<tr>
<td>“GRAINS” *</td>
<td>73.6%</td>
</tr>
<tr>
<td>Wang et al.**</td>
<td>79.3%</td>
</tr>
<tr>
<td>SceneGraphNet</td>
<td>89.5%</td>
</tr>
</tbody>
</table>

* GRAINS: Generative Recursive Autoencoders for INdoor Scenes, TOG 2019
** Deep convolutional priors for indoor scene synthesis, SIGGRAPH 2018
Numerical Evaluation

Object classification accuracy averaged over all objects and test scenes

<table>
<thead>
<tr>
<th>Method</th>
<th>Classification accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVCNN *</td>
<td>59.2%</td>
</tr>
<tr>
<td>MVCNN + SceneGraphNet</td>
<td>72.2%</td>
</tr>
</tbody>
</table>

* Multi-view Convolutional Neural Networks for 3D Shape Recognition, ICCV 2015
Summary

• Network for context-based scene completion & iterative scene synthesis

• Key component: neural message passing on a graph whose nodes represent objects and edges represent spatial and structural relationships

• Demonstrated also for context-based object recognition
Limitations

• Graphics-oriented: input scenes are synthetic & 3D models are labeled

• Our graph edges are derived from simple heuristics

• Limited to predicting only object categories and sizes – it does not generate geometry or style!
Shape style transfer
Functionality Preserving Shape Style Transfer, TOG 2016
Challenge: separate style from function

exemplar (style) → target (function) → output
Observation: similar style elements
Observation: similar style elements

similarly shaped elements
Observation: similar style elements

similar dominant curves
Key idea

Search for **compatible** element operations that:

• Maintain gross form & part structure of target shape
  => **functionality preservation**

• Increase style similarity to exemplar
  => **style transfer**
Framework overview

**Input**

- **style**
- **function**

**Segmentation**

**Tabu Search**

- substitution
- deformation
- addition

**Output**
Hierarchical segmentation and extracted curves
Element-level operations
Element-level operations

- part substitution
Element-level operations

• part substitution
• part addition / removal
Element-level operations

- part substitution
- part addition / removal
- curve-based deformation
Unconstrained element edits

- style
- function

output
Compatible element edits

style

function

output
Search for compatible element-level operations

substitution
Search for compatible element-level operations
Search for compatible element-level operations
Search for compatible element-level operations

substitution

deformation

addition
Best solution

style

function

output
Multiple solutions

style

function

outputs

...
Measures

- Style similarity measure from:
  “Elements of Style: Learning Perceptual Shape Style Similarity”, Lun, Kalogerakis, Sheffer, TOG 2015

- Functional compatibility measure

\[ D_{\text{func}}(\text{chair}, \text{chair}) = 1.32 \]
\[ D_{\text{func}}(\text{chair}, \text{chair}) = 0.57 \]
Measures

- **Style similarity measure from:**
  “Elements of Style: Learning Perceptual Shape Style Similarity”, Lun, Kalogerakis, Sheffer, TOG 2015

- **Functional compatibility measure**

\[ D_{\text{func}}(\text{Chair 1}, \text{Chair 2}) = 1.32 \quad D_{\text{func}}(\text{Chair 3}, \text{Chair 4}) = 0.57 \]
Measures

- **Style similarity measure** from: “Elements of Style: Learning Perceptual Shape Style Similarity”, Lun, Kalogerakis, Sheffer, TOG 2015

- **Functional compatibility measure**

  \[ D_{\text{func}}(\text{chair}, \text{chair}) = 1.32 \]  
  \[ D_{\text{func}}(\text{chair}, \text{chair}) = 0.57 \]
Element relation graph

- Adjacency
- Containment
- Symmetry
Results
User study: style similarity

Which of the two shapes on the bottom (B or C) is more similar style-wise to the shape on the top (A)?

- B
- C
- can't tell - Both B and C are very similar to A in style
- can't tell - Neither B nor C are similar to A in style
User study: style similarity

Style Similarity

Our top-ranked synthesized shapes 39.1% 28.4% 32.5% Shapes created by modelers

draw
Summary & Future work

• Algorithm for synthesizing shapes by **transferring style** between objects with **different structure and functionality**

• Develop **style-aware generative models** that are capable of generating plausible shape structure & accurate surface geometric details
Thank you!

Our project web pages:

https://people.umass.edu/~yangzhou/scenegraphnet/

&

http://people.cs.umass.edu/~zlun/papers/StyleTransfer