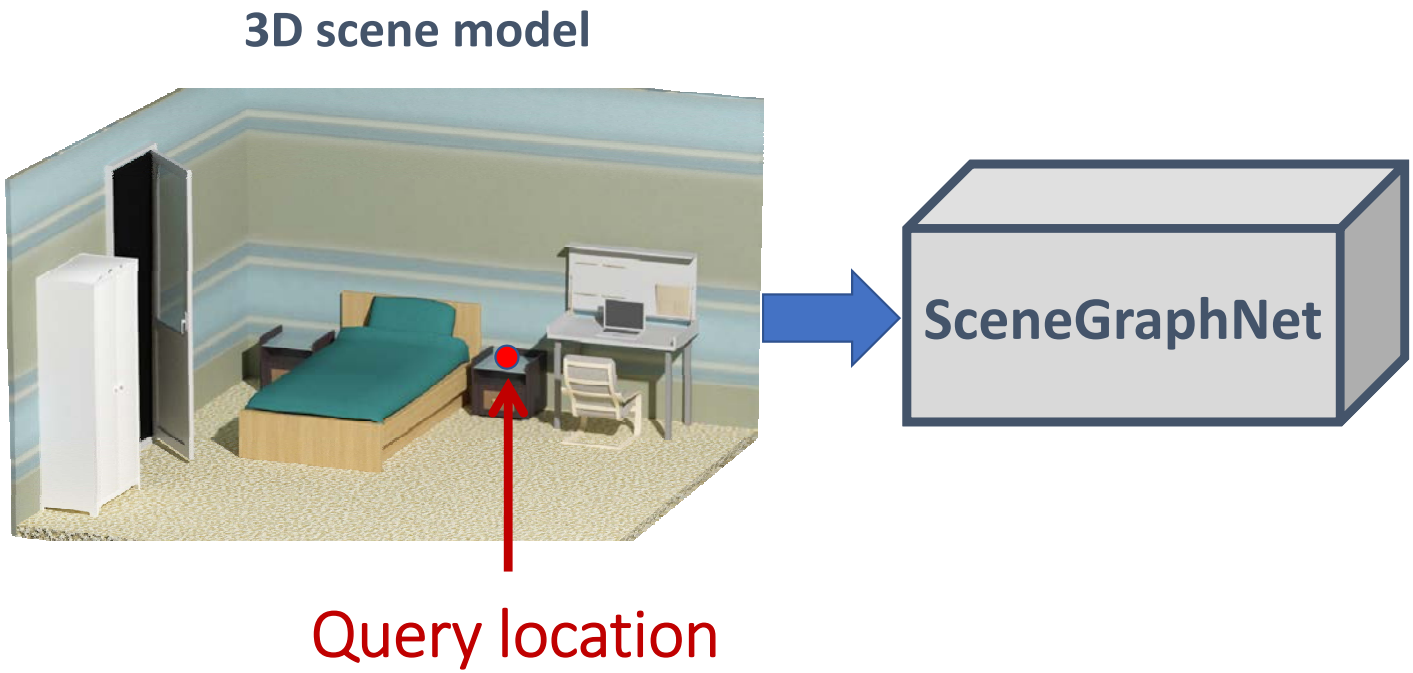

**Evangelos Kalogerakis**

[based on "SceneGraphNet: Neural Message Passing for 3D Indoor Scene Augmentation", Yang Zhou, Zachary While, Evangelos Kalogerakis, ICCV 2019]

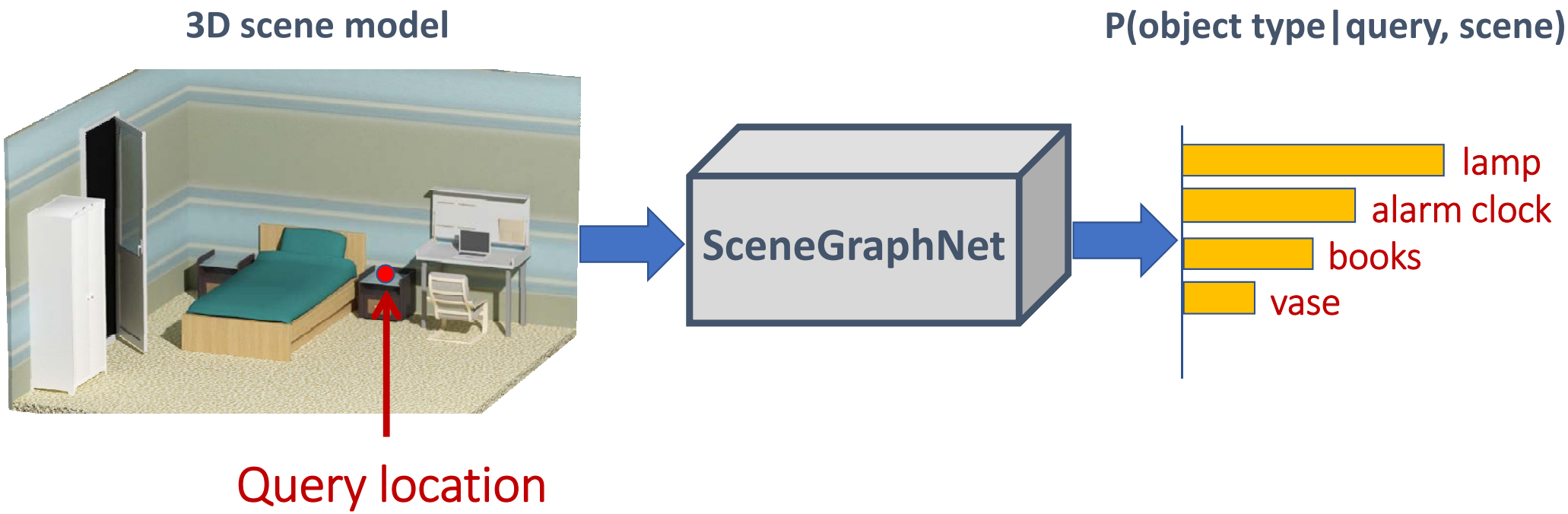

**Evangelos Kalogerakis**

**[based on "SceneGraphNet: Neural Message Passing for 3D Indoor Scene Augmentation", Yang Zhou, Zachary While, Evangelos Kalogerakis, ICCV 2019]**

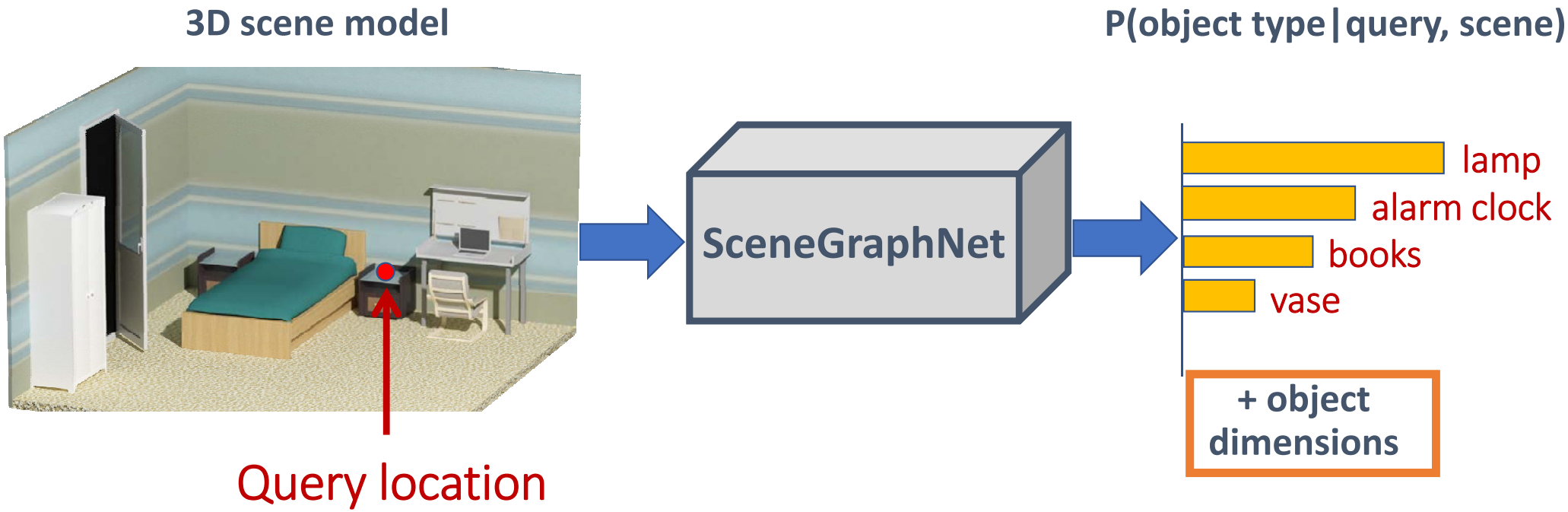
# Goal: predict objects to populate indoor scenes



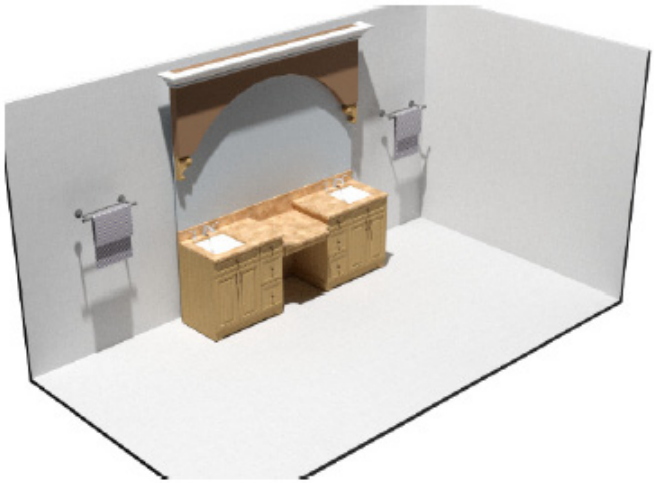
# Goal: predict objects to populate indoor scenes



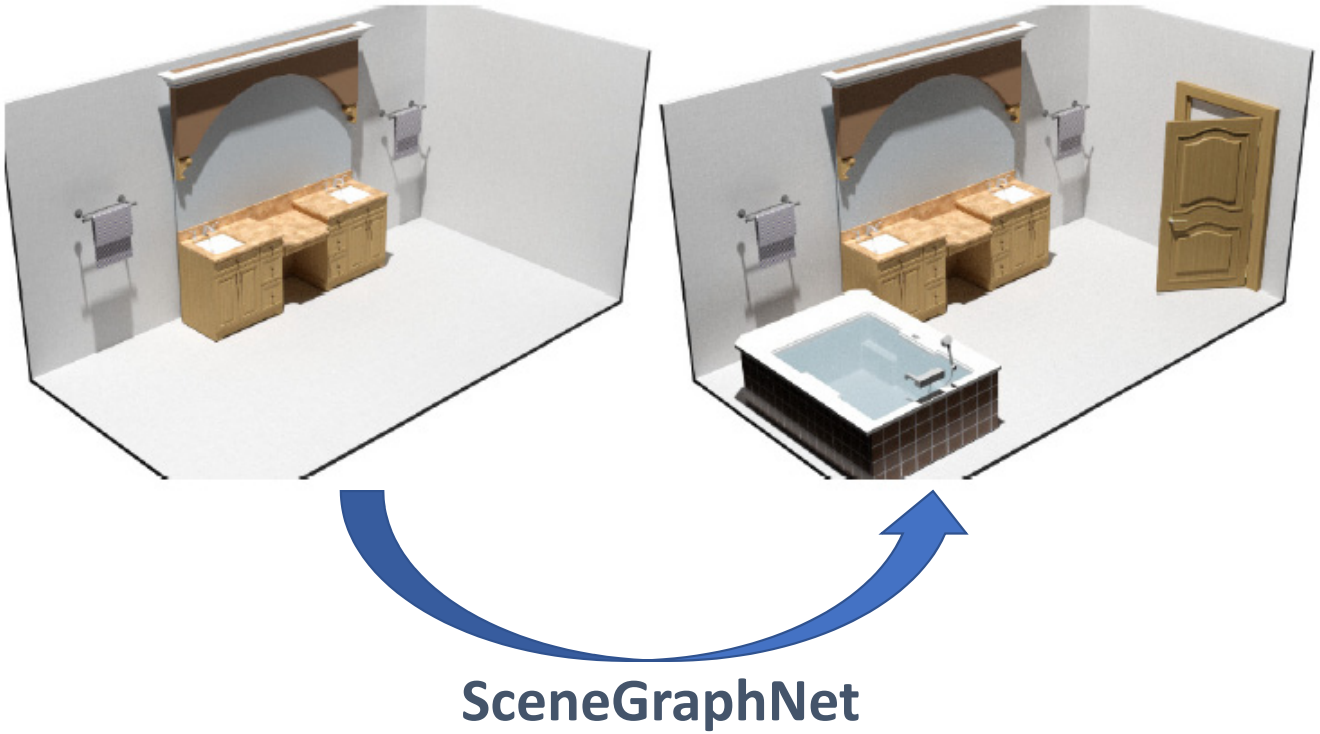
# Goal: predict objects to populate indoor scenes



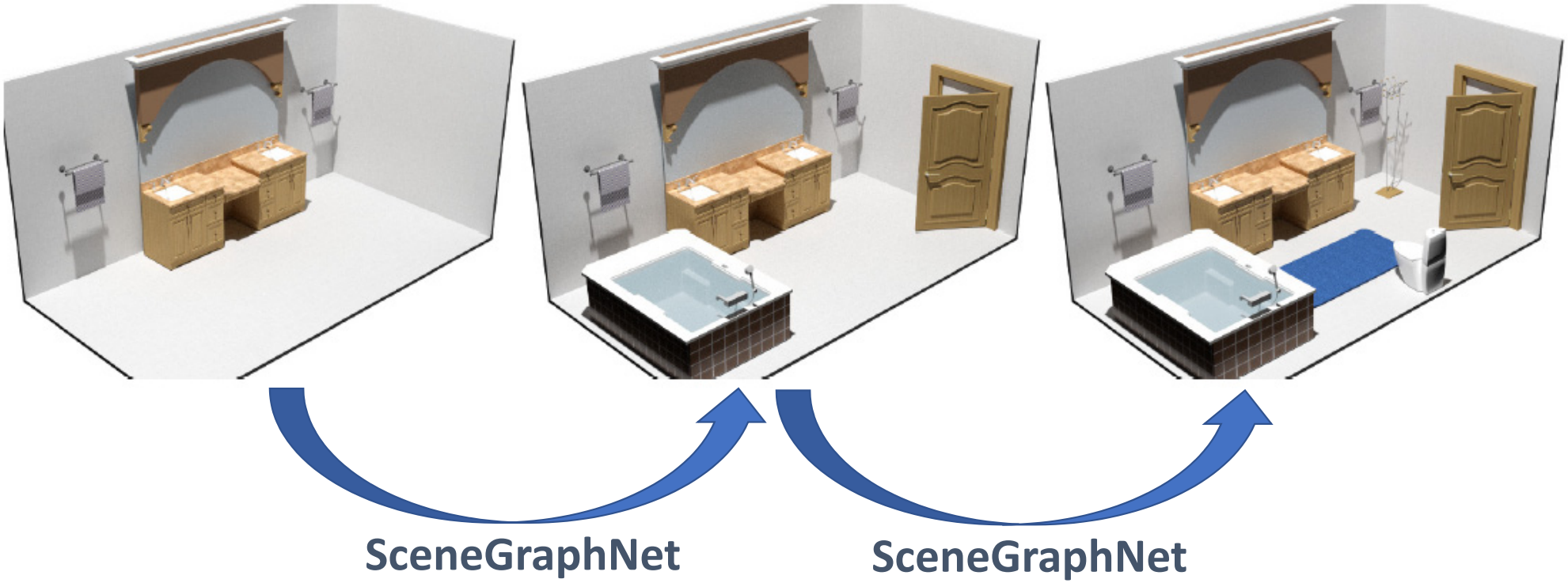
# Alternative goal: iterative scene synthesis



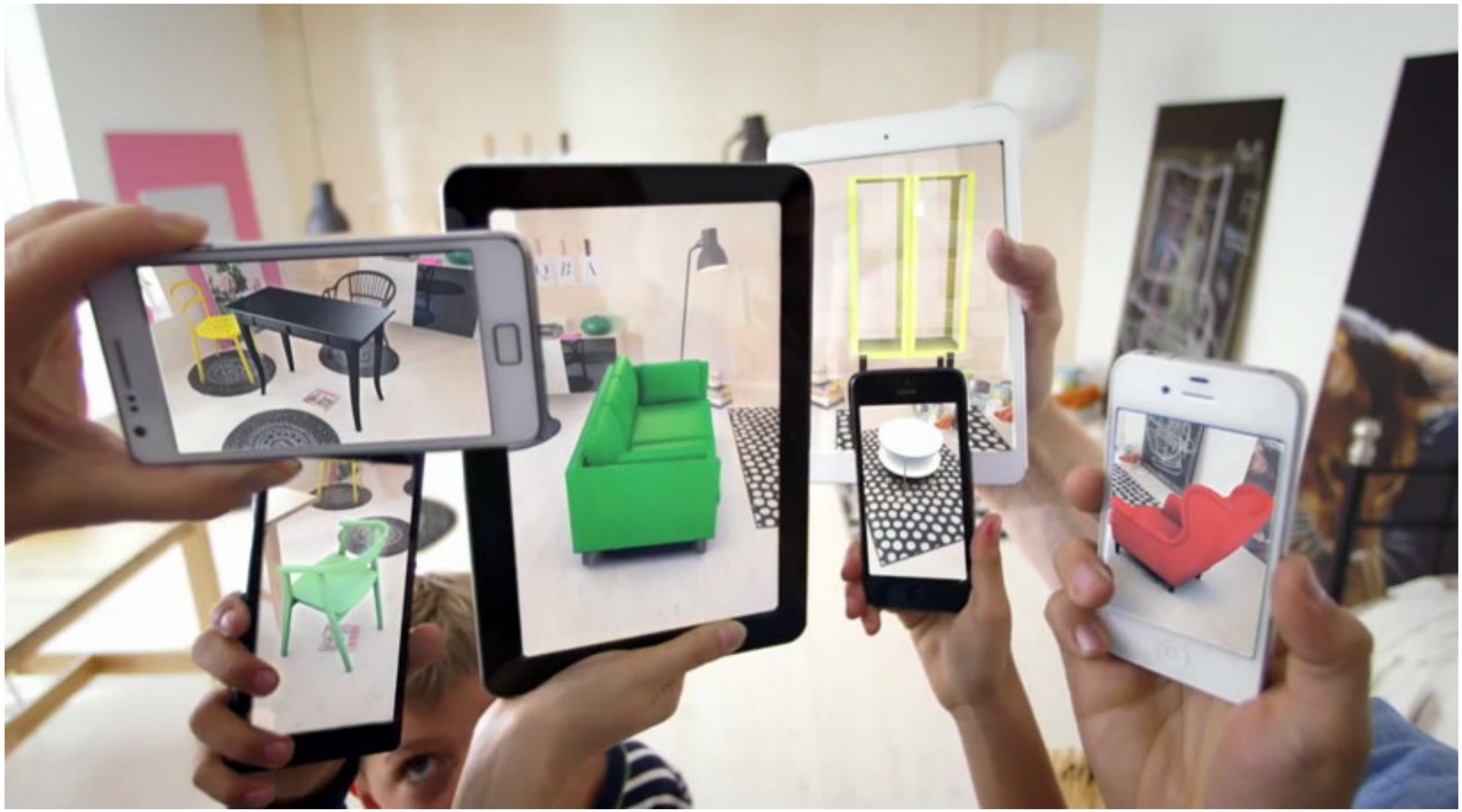
# Alternative goal: iterative scene synthesis



# 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]**

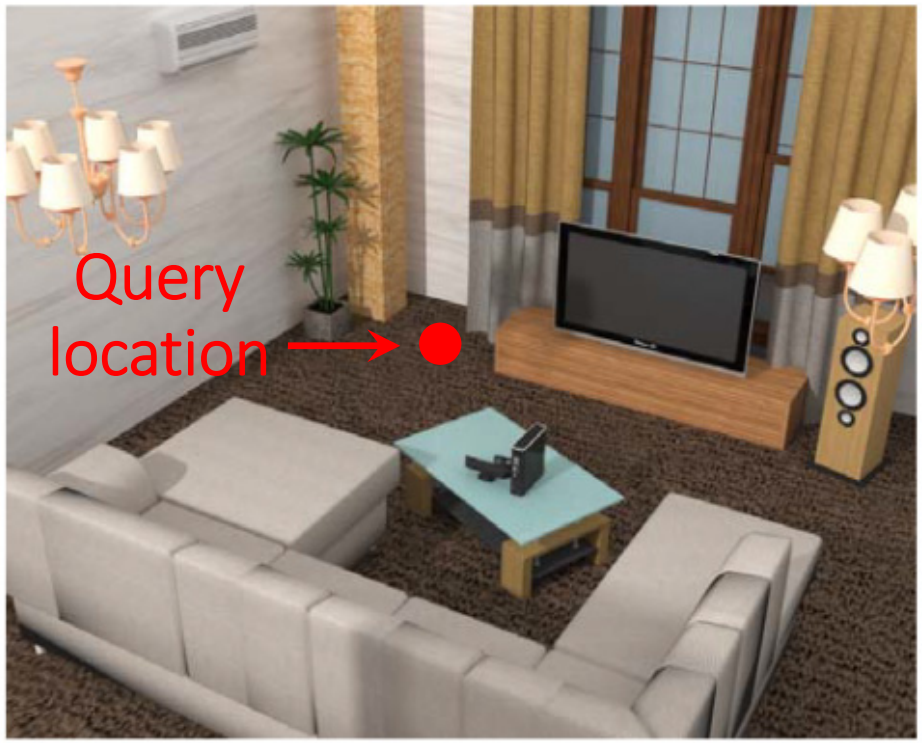
# Prior work: learning-based scene completion

- 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



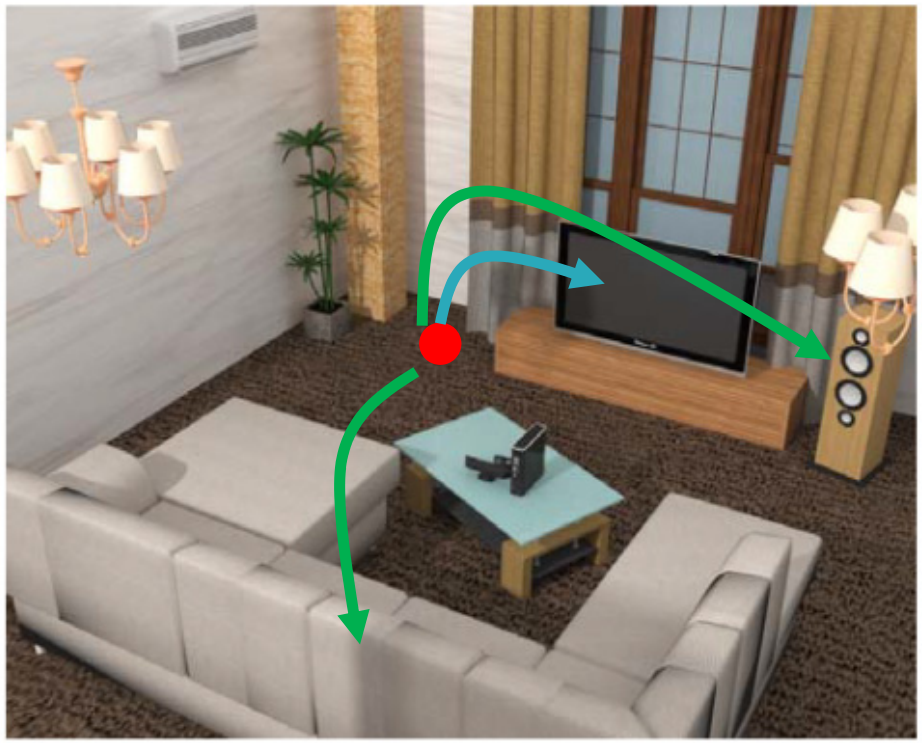
# Key idea: capture surrounding object relations

**Short-range relations?**

# Key idea: capture surrounding object relations

**Short-range relations?**

# Key idea: capture surrounding object relations



**Short-range relations**

**+**

**Longer-range relations**

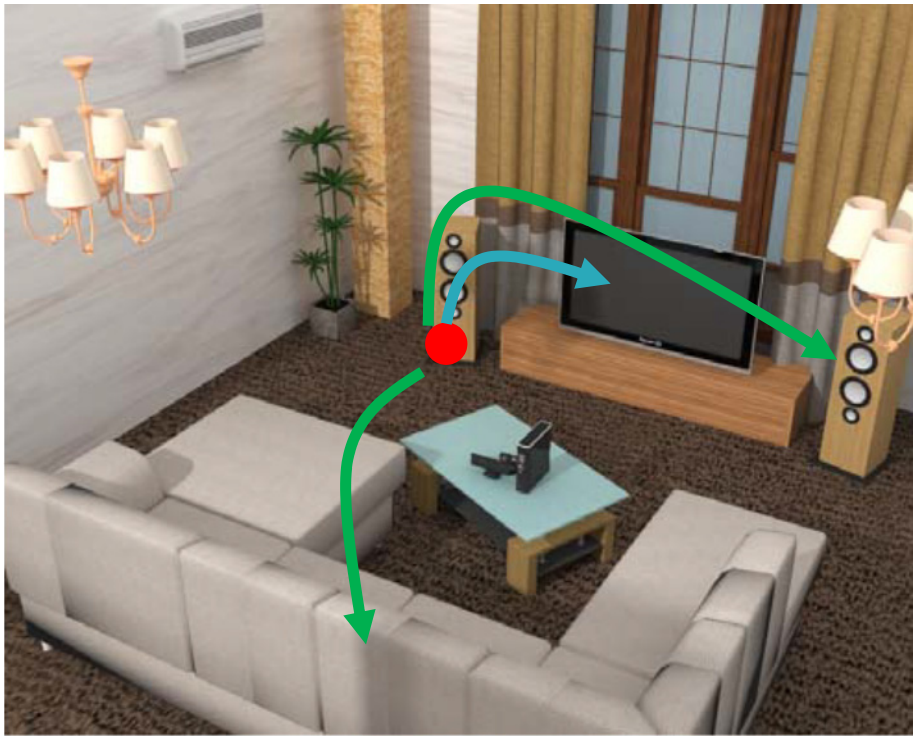
# Key idea: capture surrounding object relations

**Short-range relations**

**+**

**Longer-range relations**

# Key Challenges



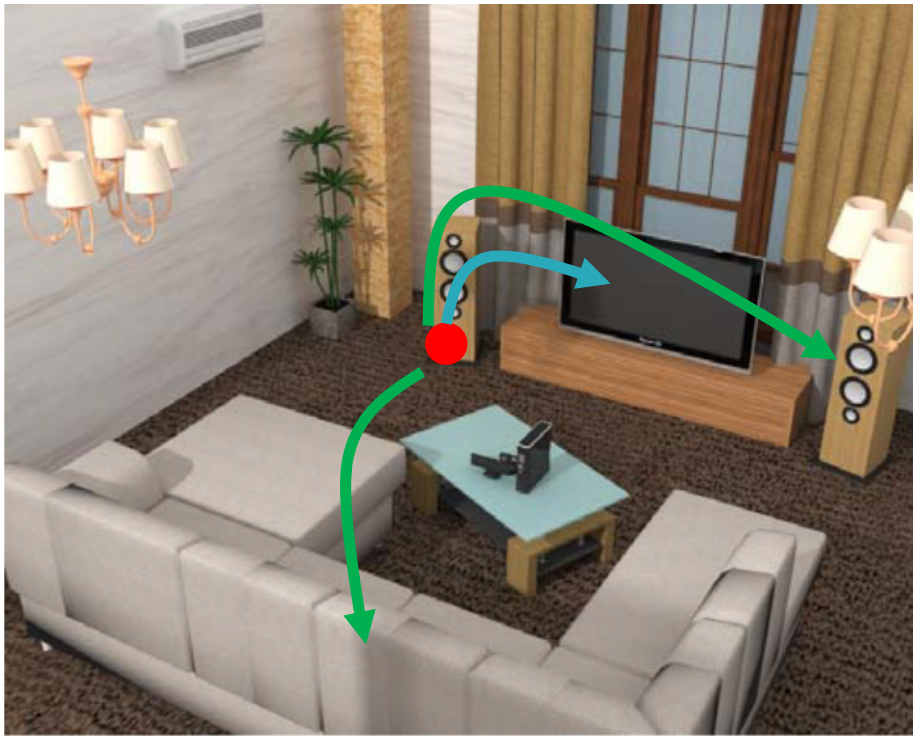
**Short-range relations**

**+**

**Longer-range relations**

**How to learn these relations?**

# Key Challenges



**Short-range relations**

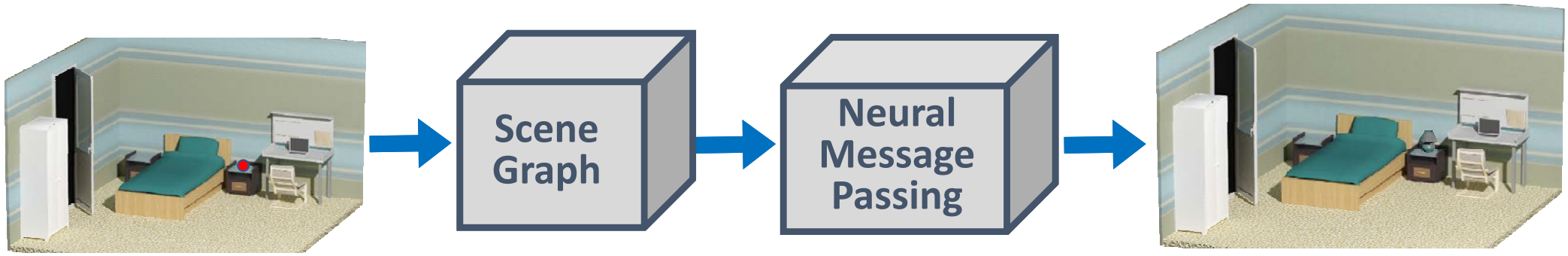
**+**

**Longer-range relations**

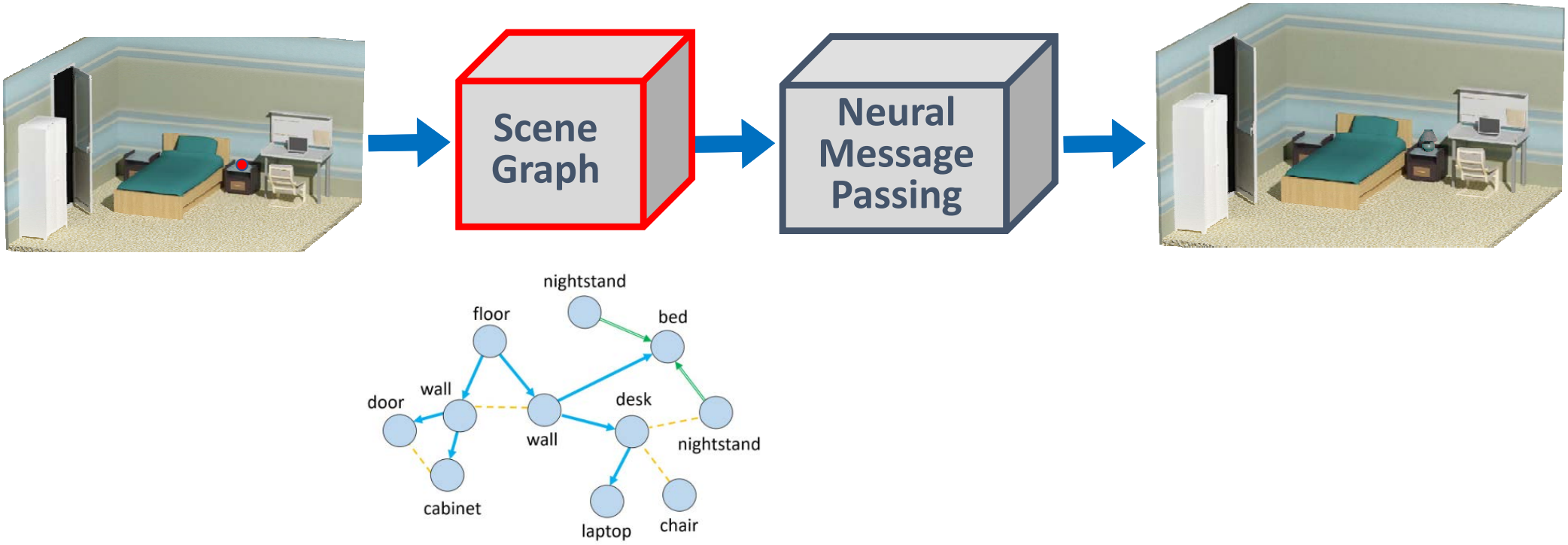
**How to learn these relations?**

**How to weigh & aggregate them?**

# Pipeline



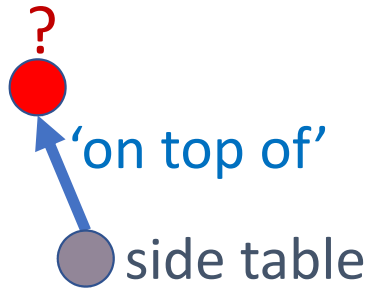
# Pipeline


Scene Graph
Neural Message Passing
nightstand
floor
bed
wall
door
desk
wall
nightstand
cabinet
laptop
chair

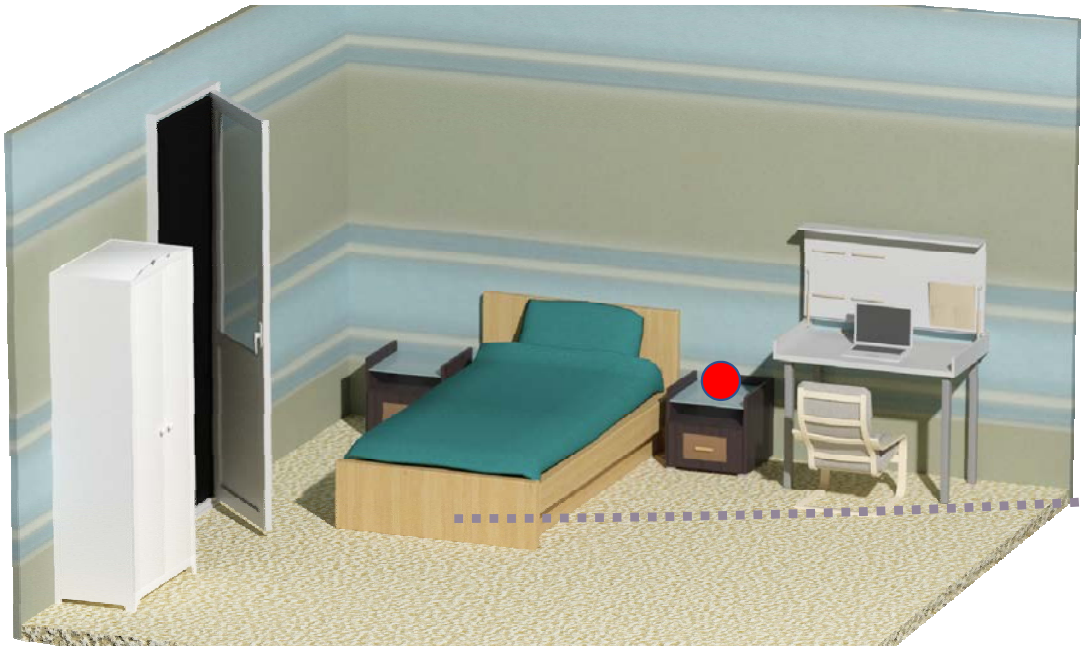
# Scene graph construction

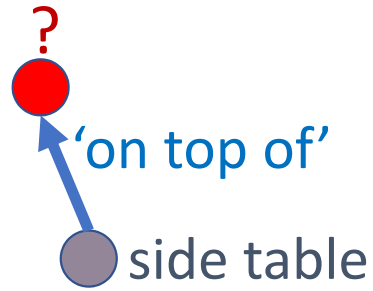
# Nodes represent existing objects + queries

# Edges represent relationships

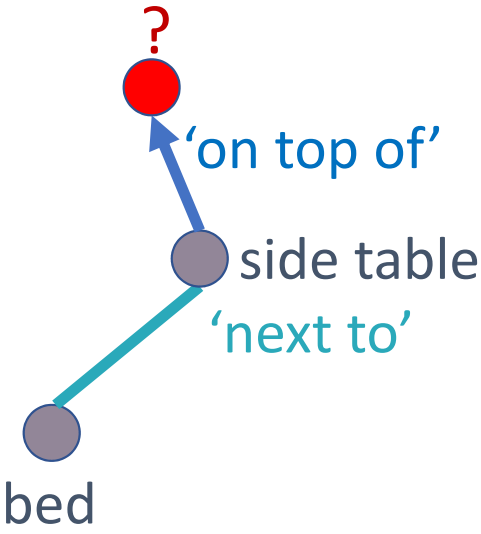


# Bed: “I am also here – I may affect decisions, too”

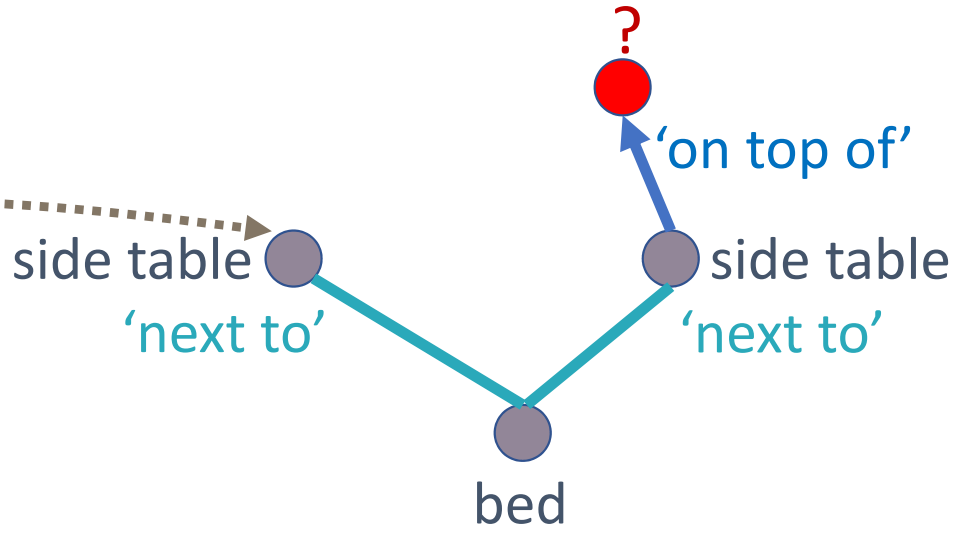




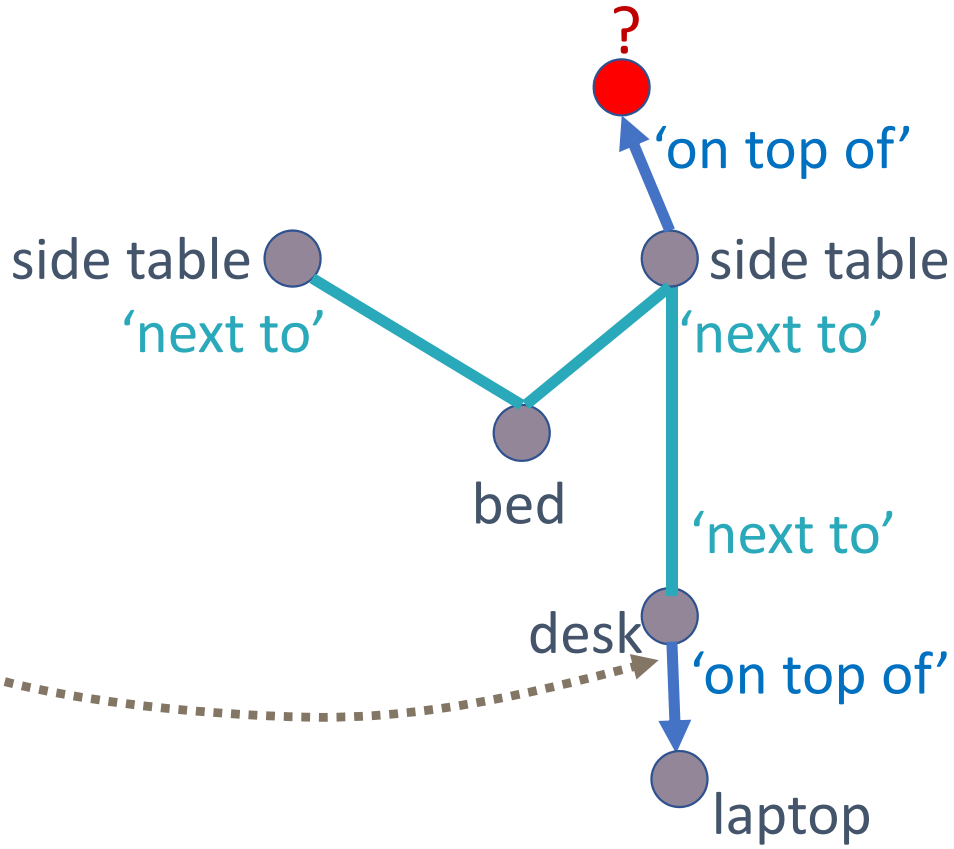

# More nodes / edges



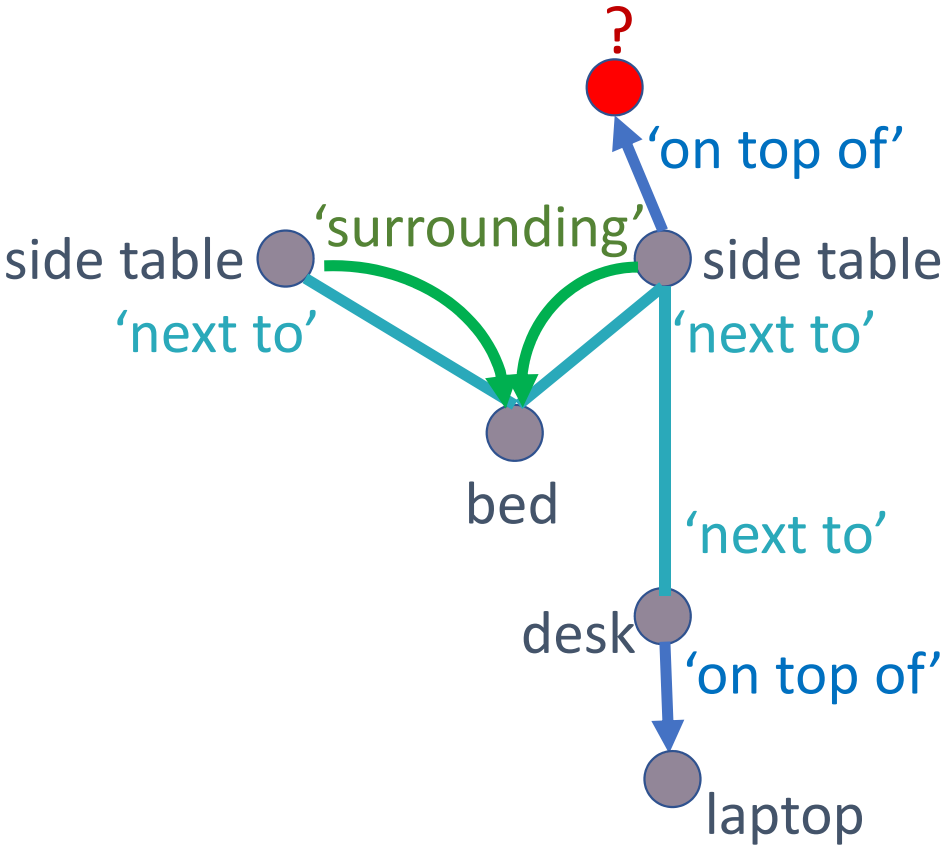
# More nodes / edges



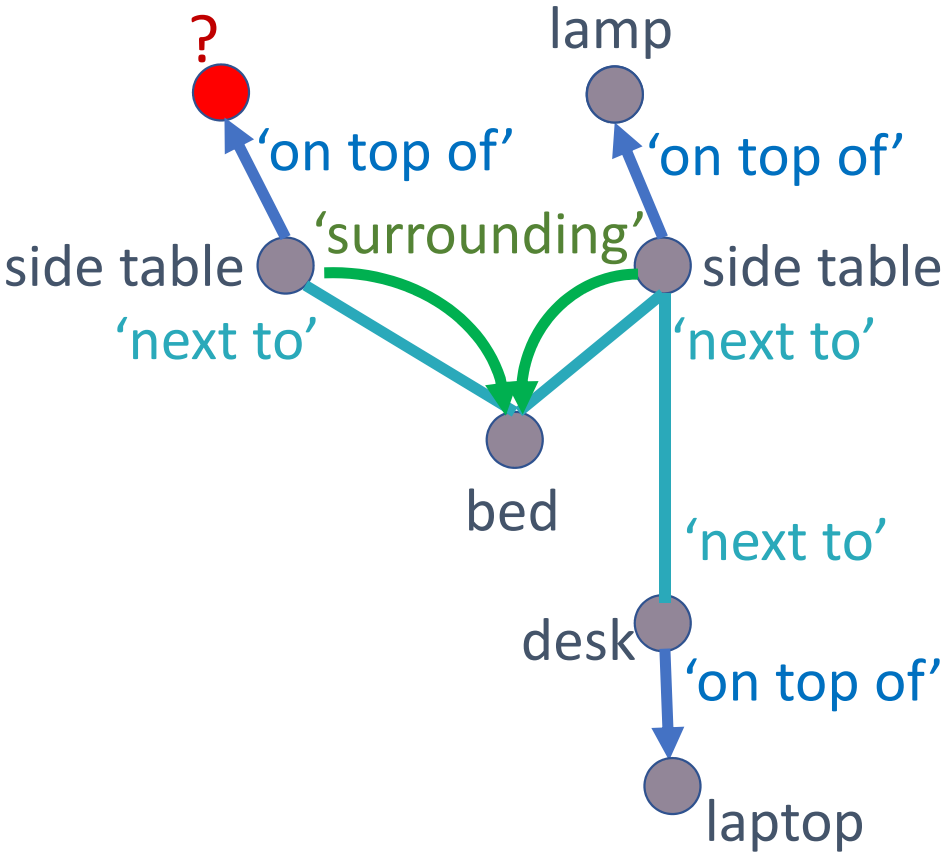
# Many more nodes / edges



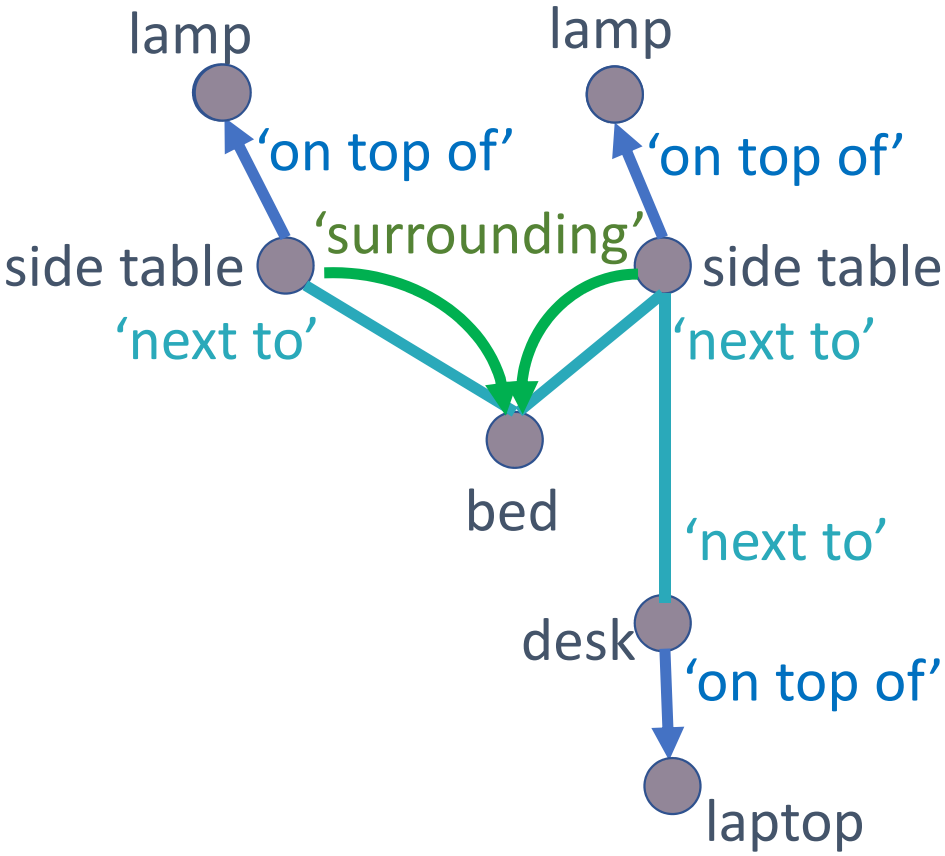
# Multiple edges for each pair of objects



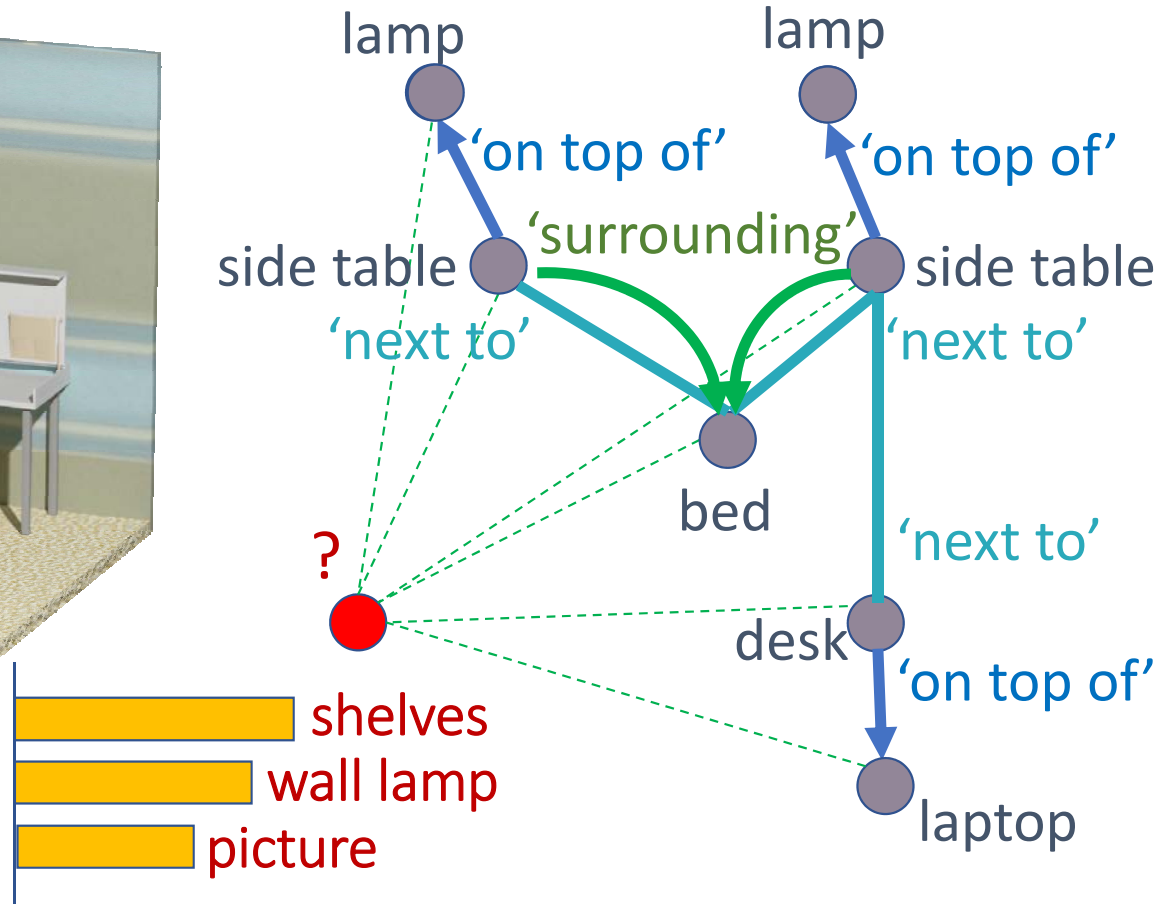
# More queries



# More queries

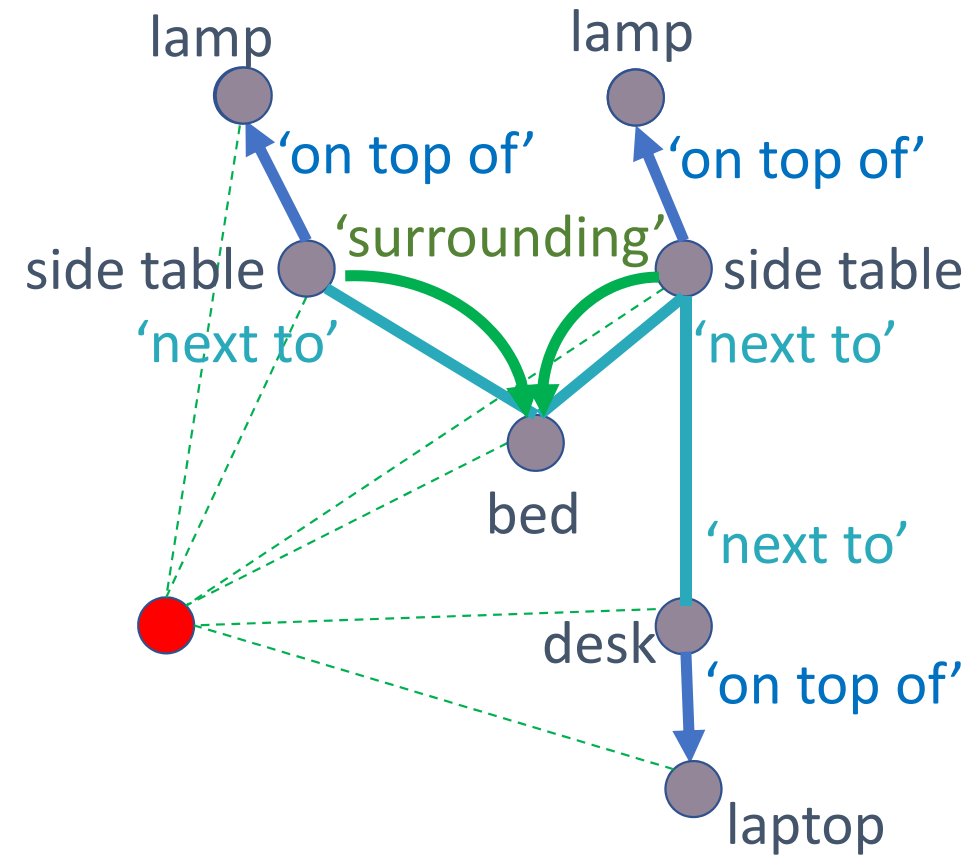


# Denser graphs yielded better predictions



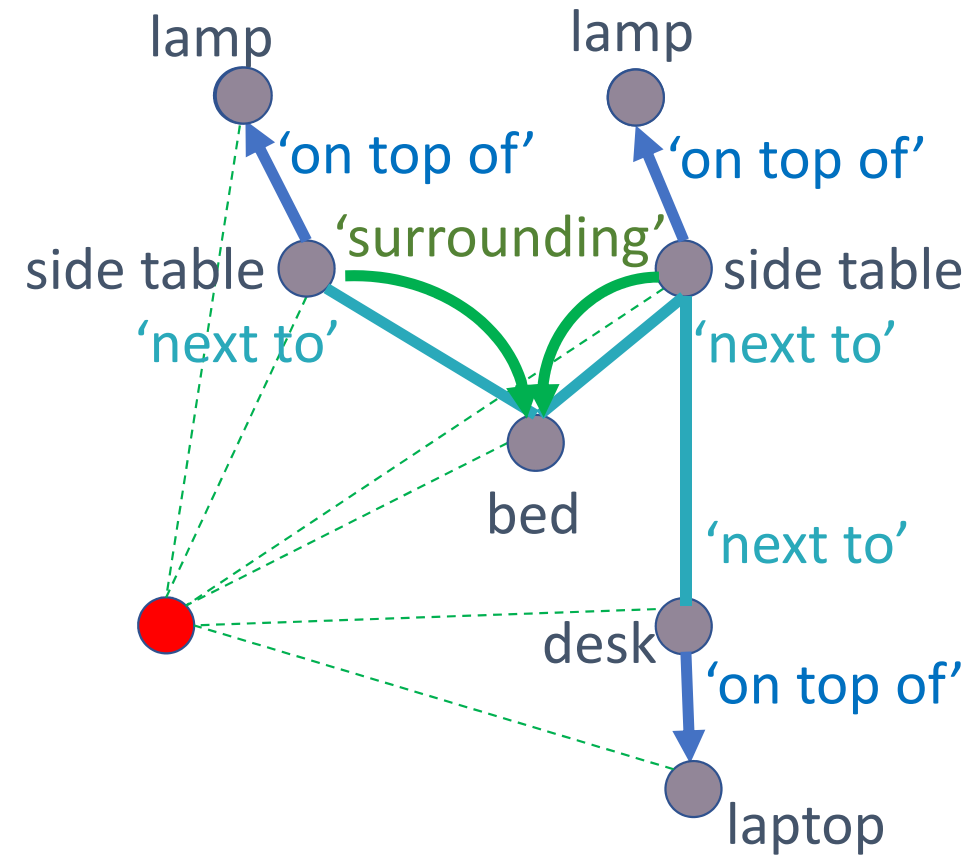
# Edge types

- **“On top of”**
- **“Under”**



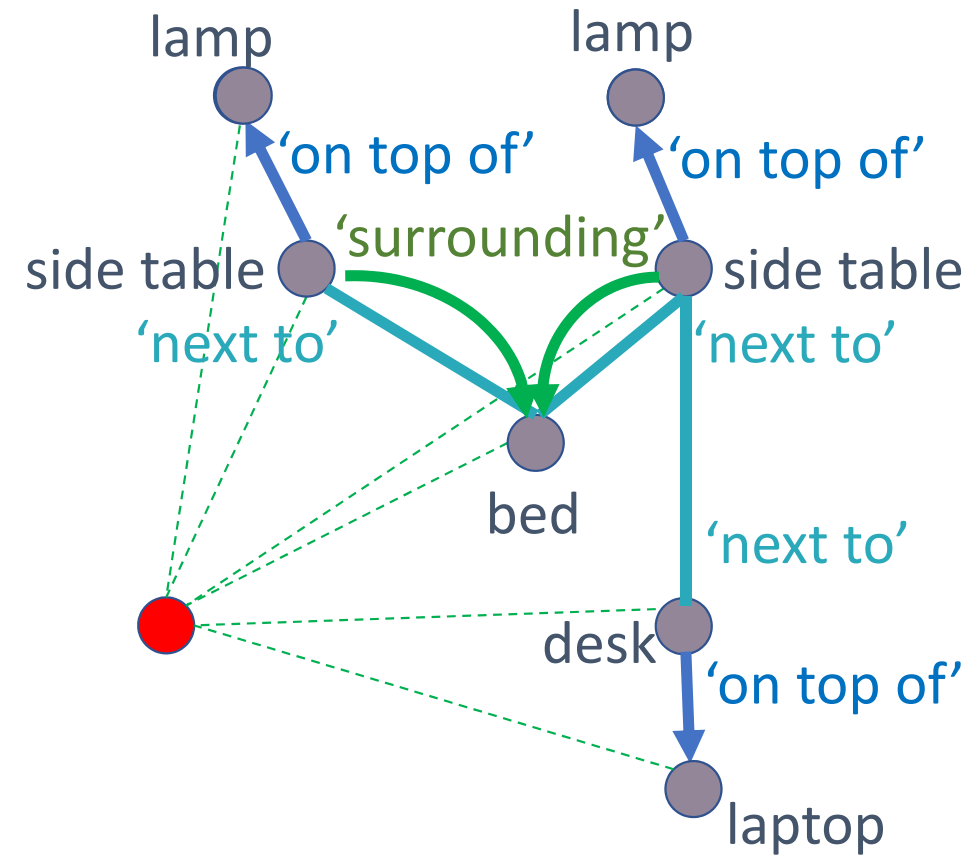
# Edge types

- **“On top of”**
- **“Under”**
- **“Surrounded-by”**
- **“Surrounding”**



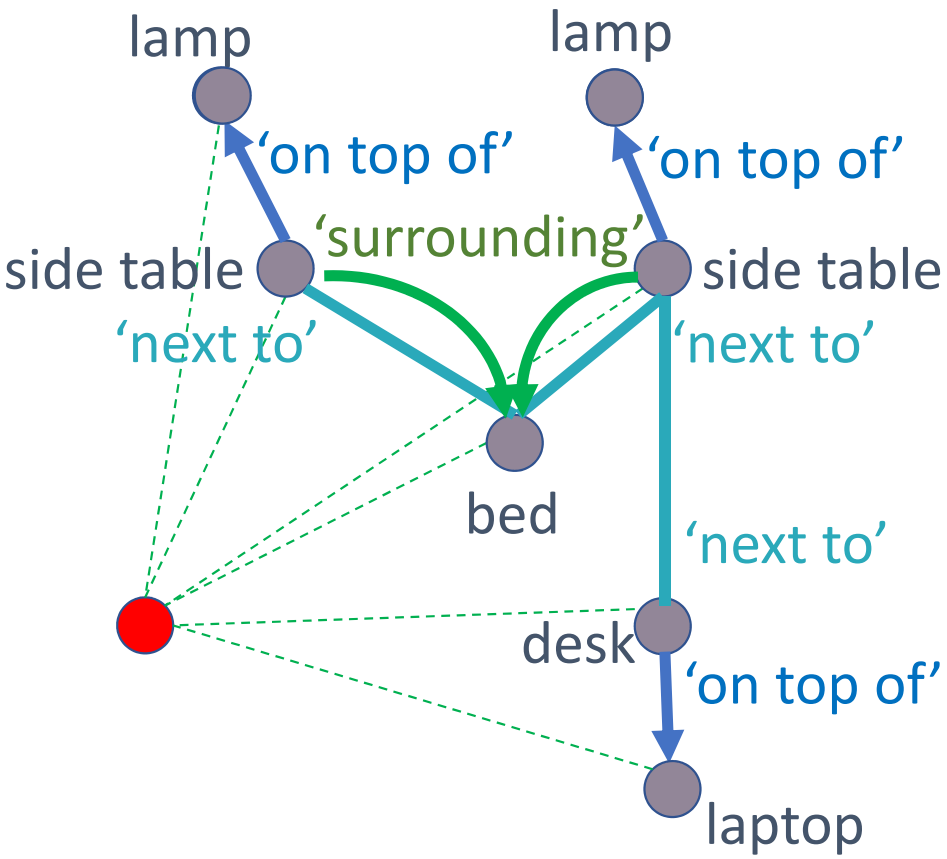
# Edge types

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

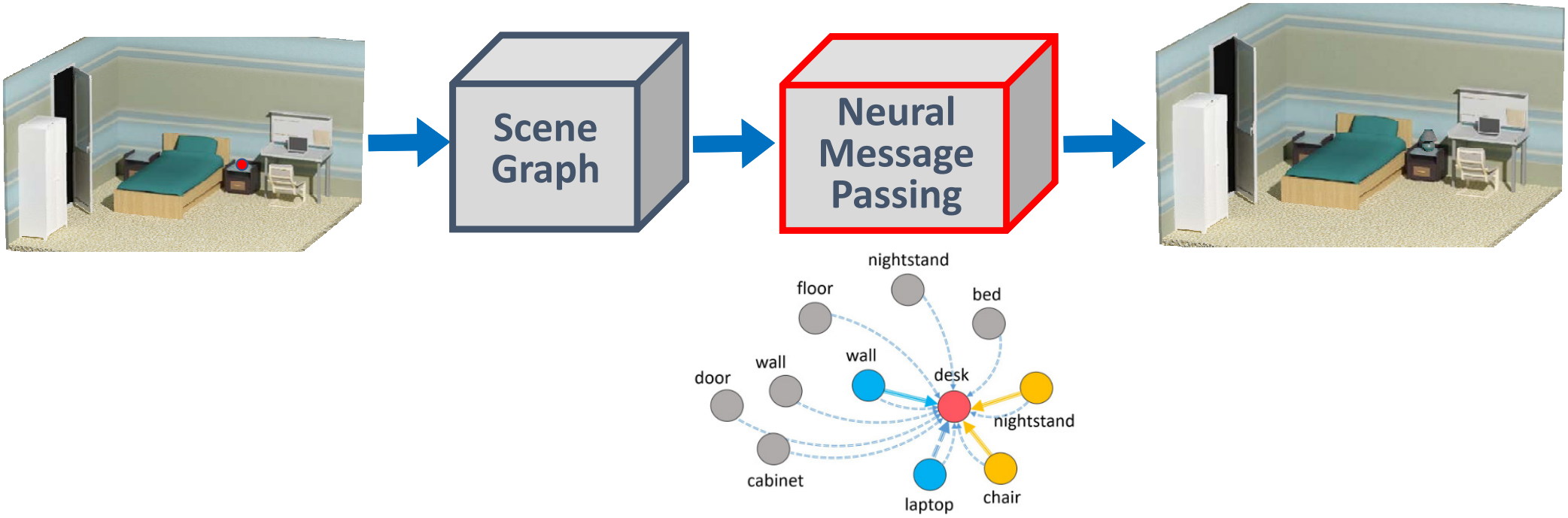


# Edge types

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



# Pipeline

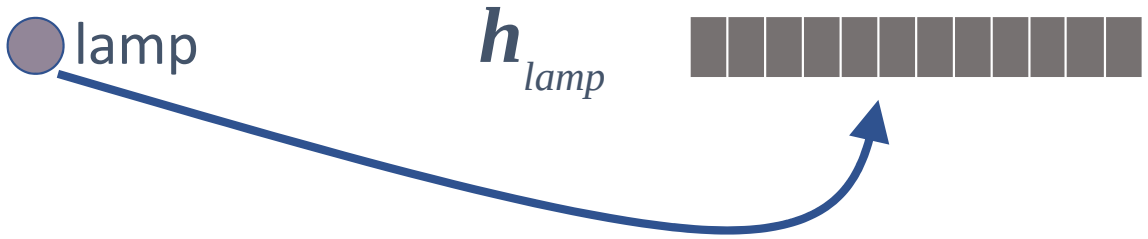



**Inspired by [Gilmer et al. 2017]**

# Node representation

- lamp

# Node representation



# Raw node features

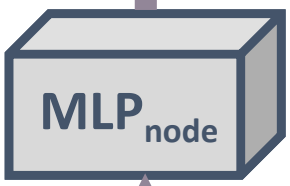
lamp

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

# Initial node representation

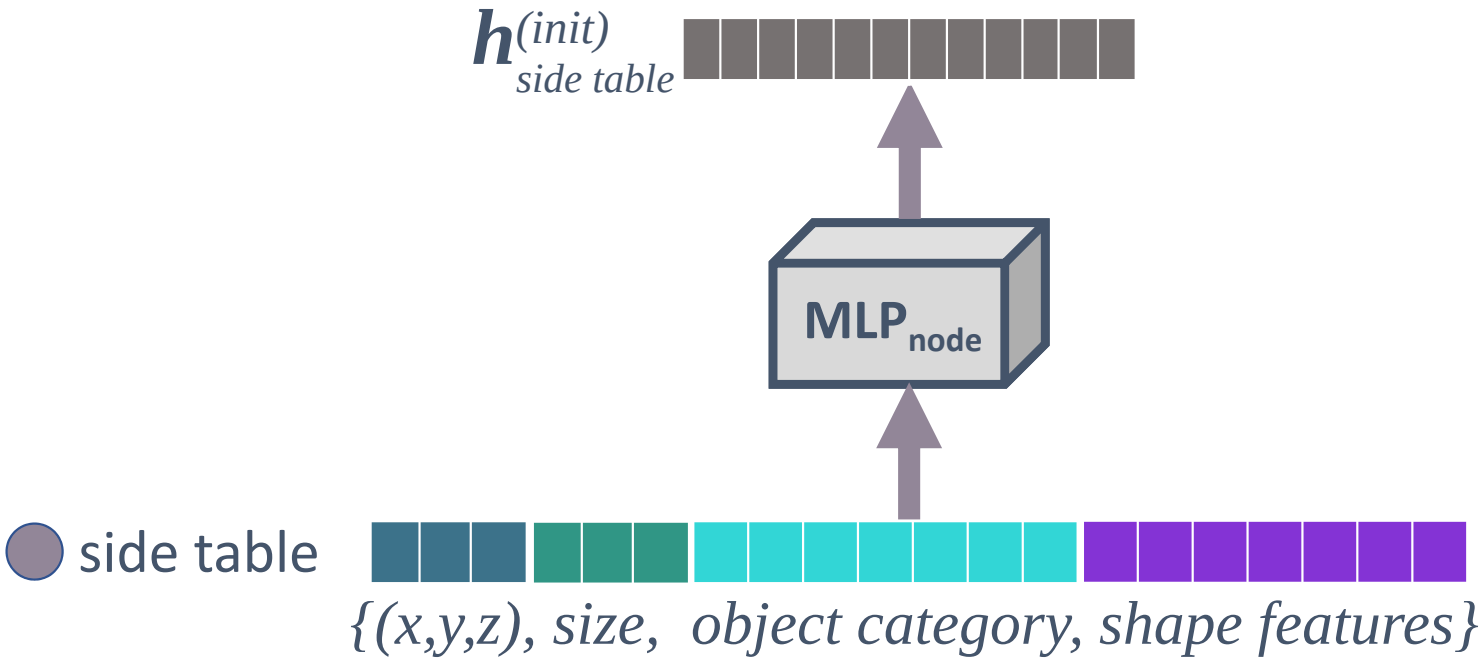
lamp

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

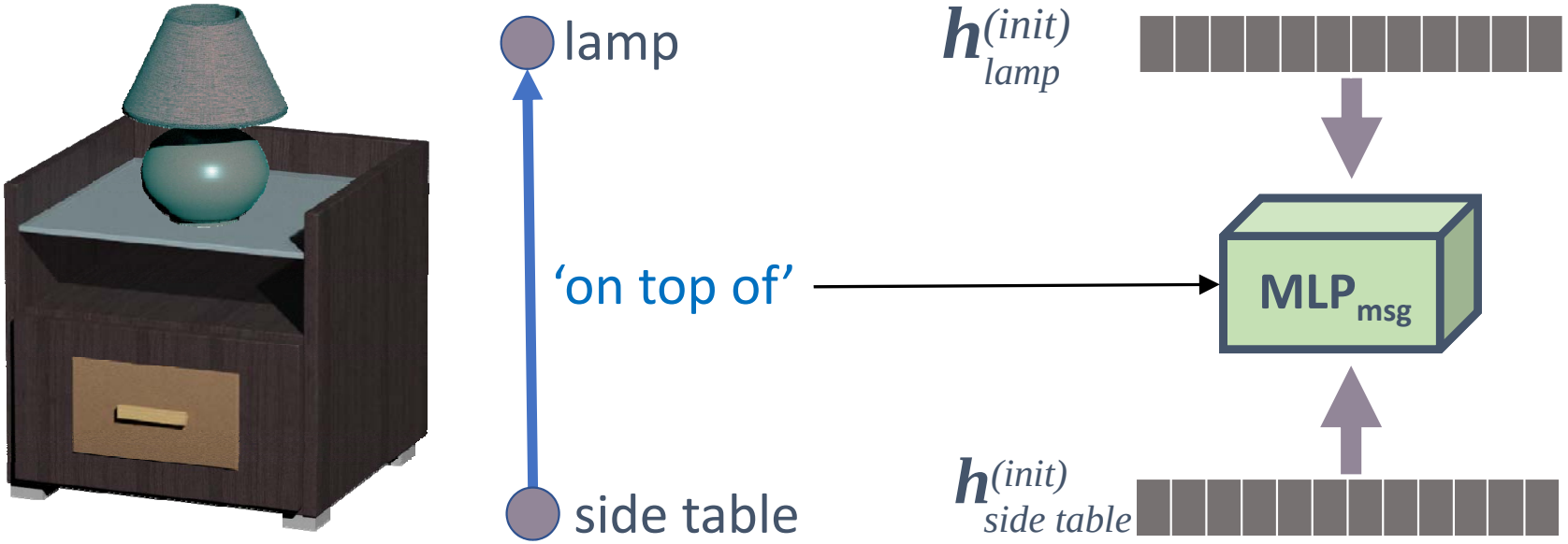



*{(x,y,z), size, object category, shape features}*

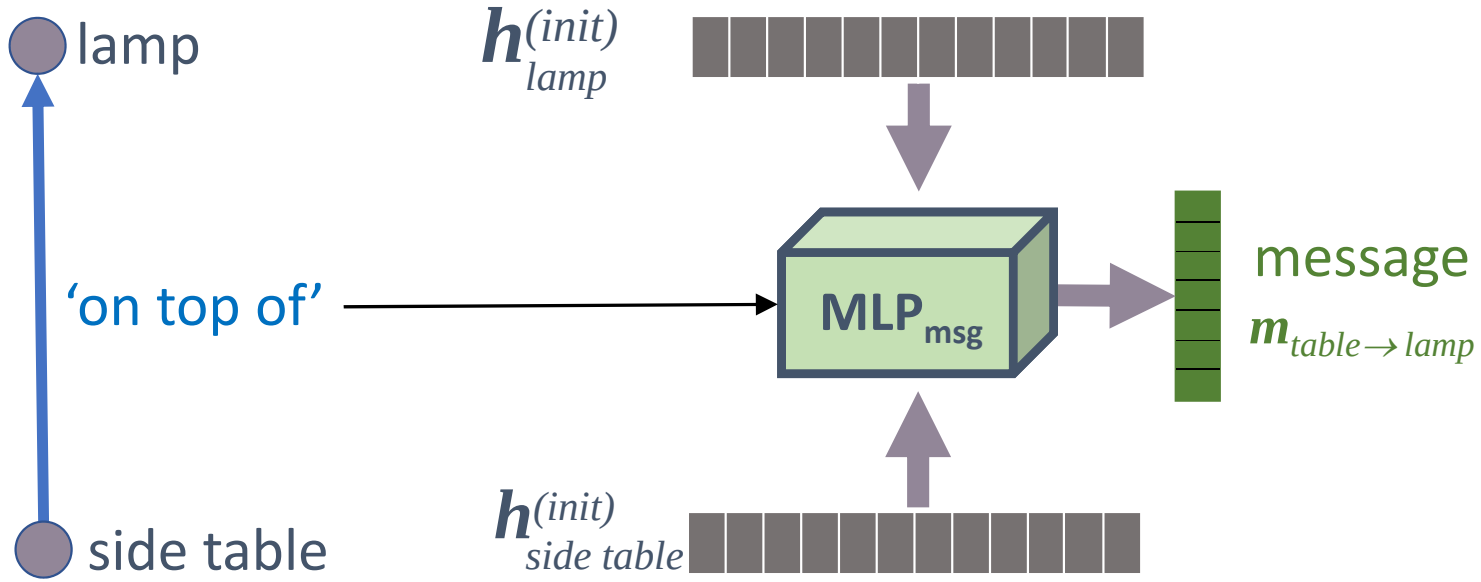
# Initial node representation



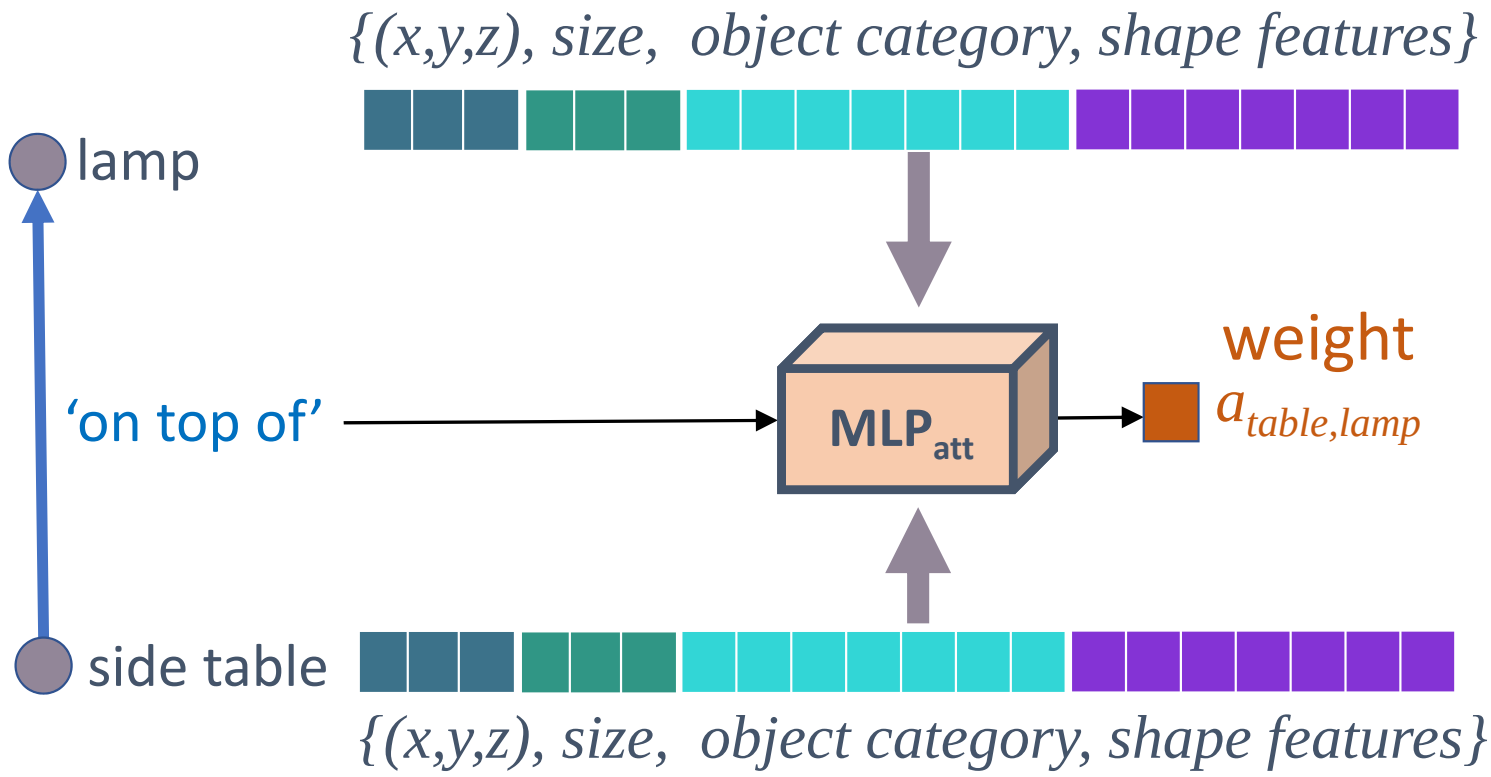
# Messages between objects



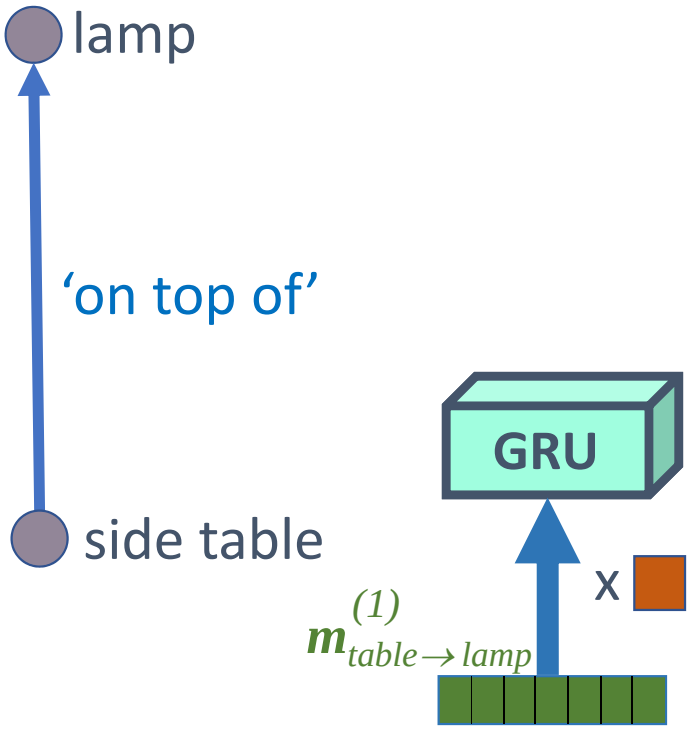
# Messages between objects



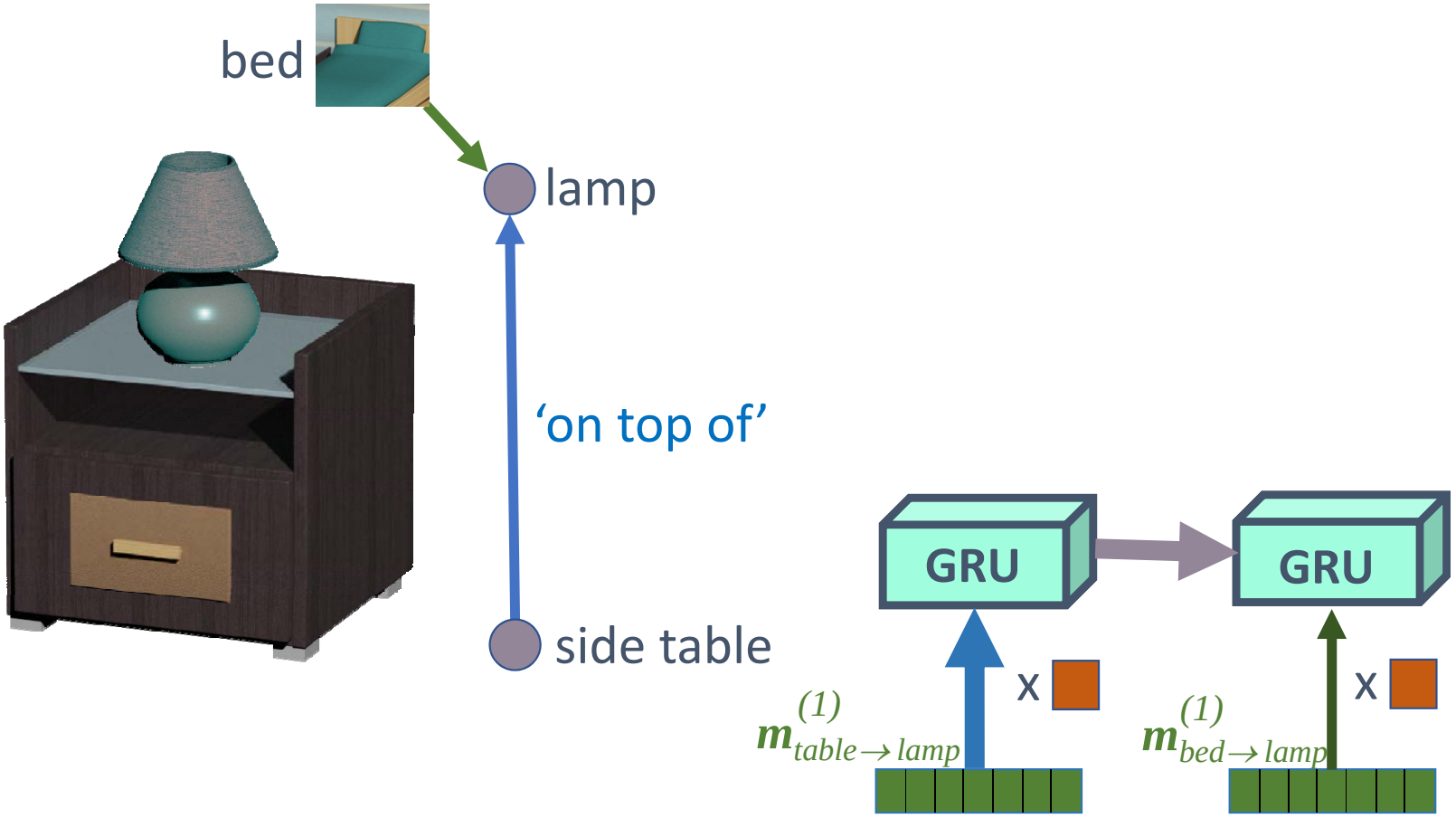
# Edge weights



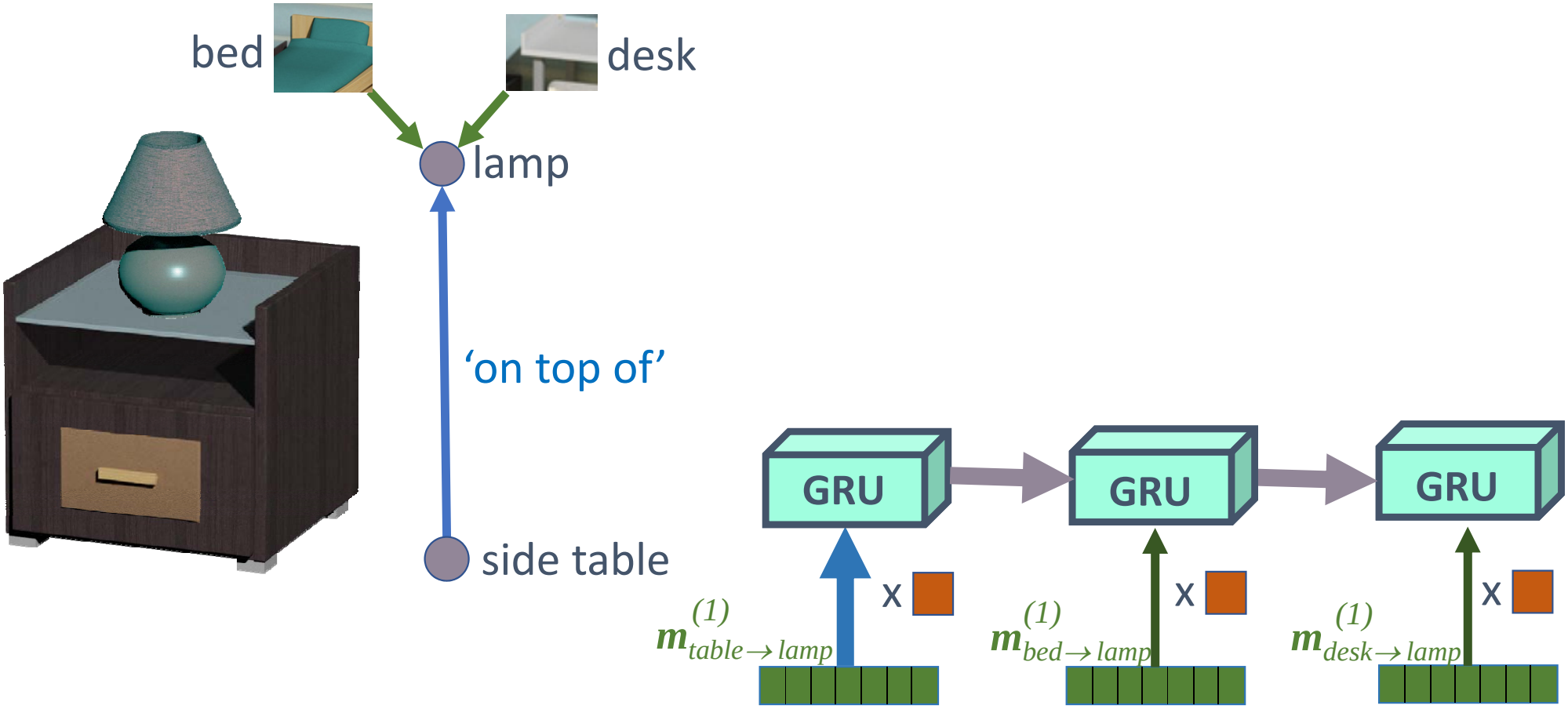
# Node update



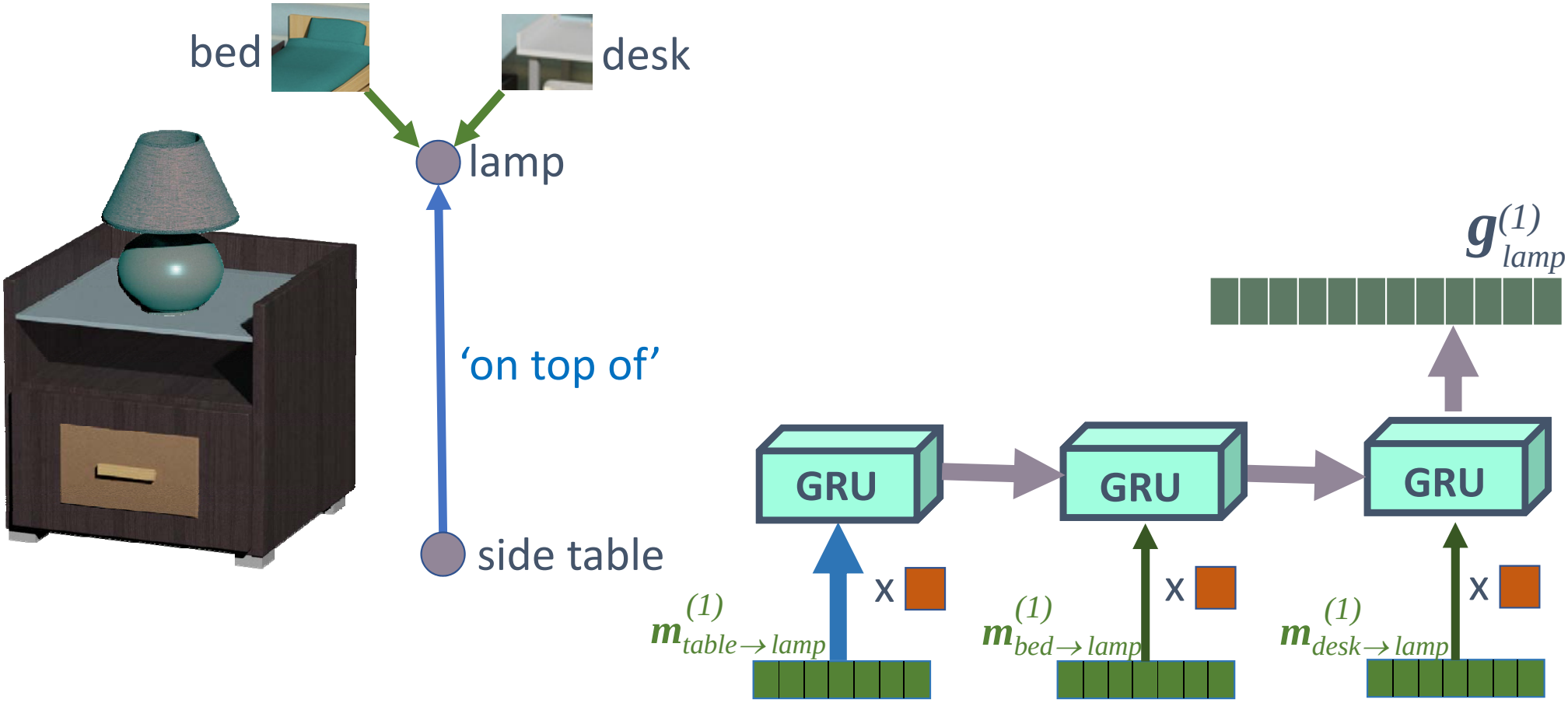
# Node update


bed
lamp
'on top of'
side table
GRU
GRU
x
x
$m^{(1)}_{table \rightarrow lamp}$
$m^{(1)}_{bed \rightarrow lamp}$

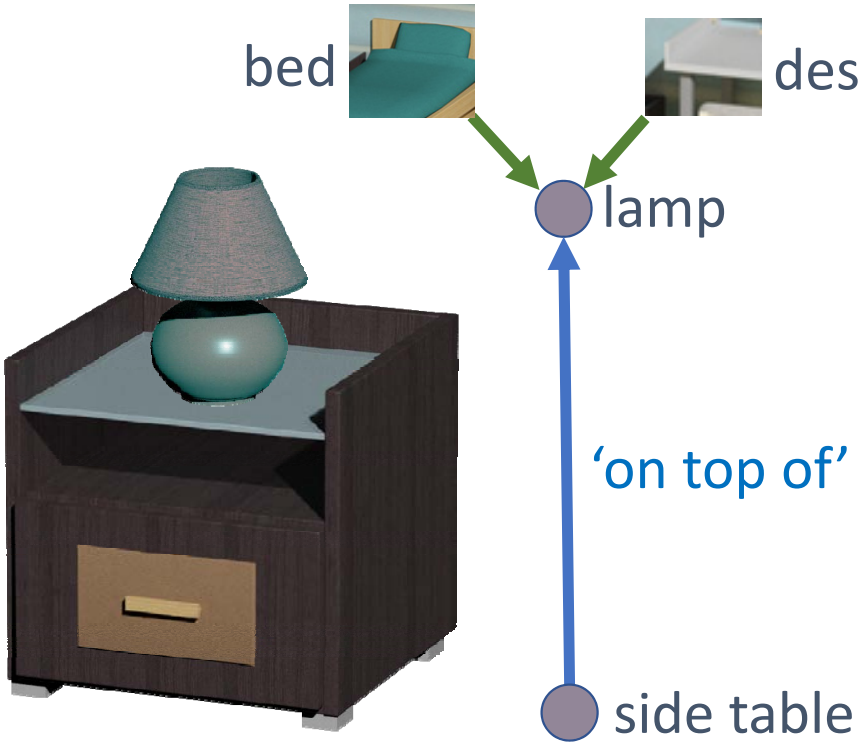
# Node update

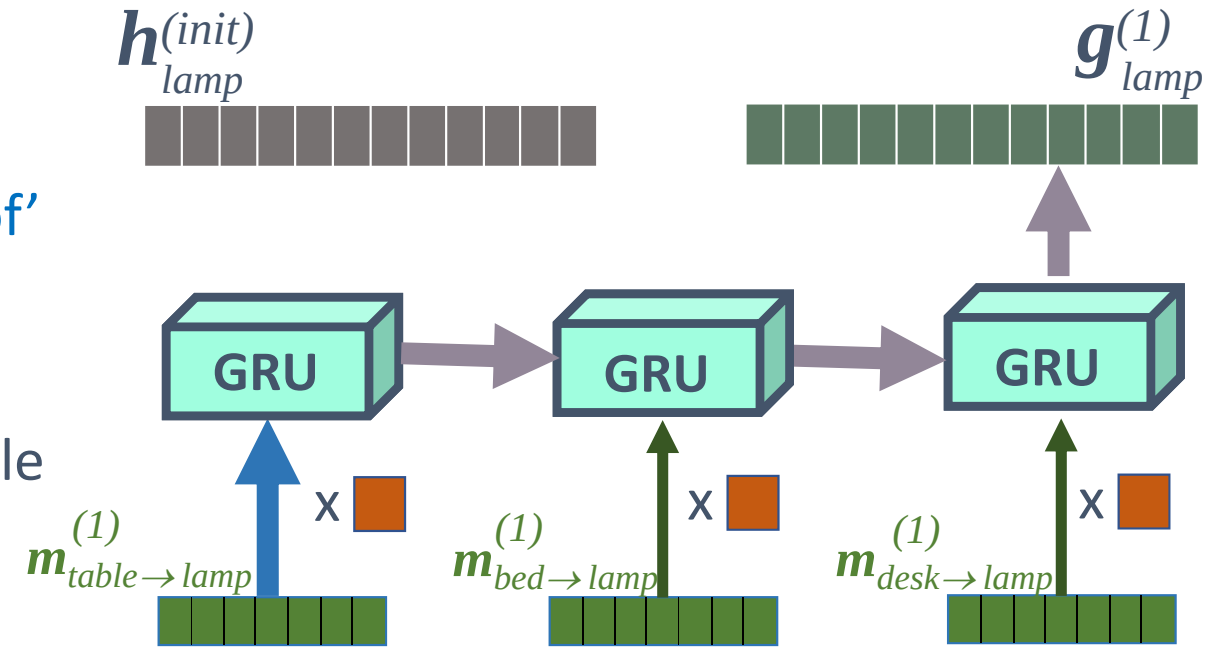

bed
desk
lamp
'on top of'
side table
GRU
GRU
GRU
x
x
x
$m^{(1)}_{table \to lamp}$
$m^{(1)}_{bed \to lamp}$
$m^{(1)}_{desk \to lamp}$

# Node update

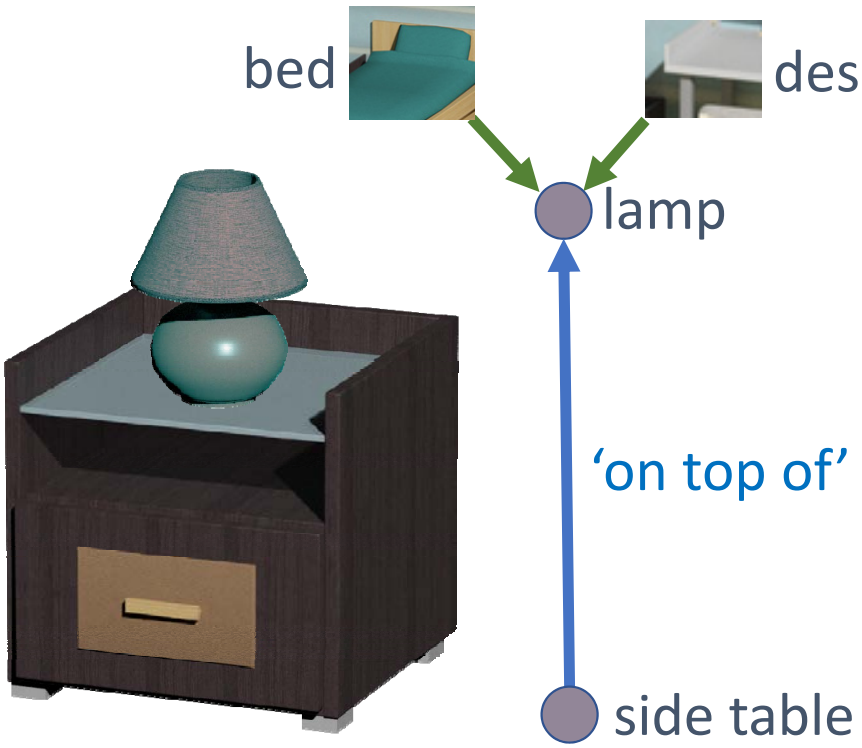

bed
desk
lamp
'on top of'
side table
GRU
GRU
GRU
x
x
x
$m^{(1)}_{table \to lamp}$
$m^{(1)}_{bed \to lamp}$
$m^{(1)}_{desk \to lamp}$
$g^{(1)}_{lamp}$

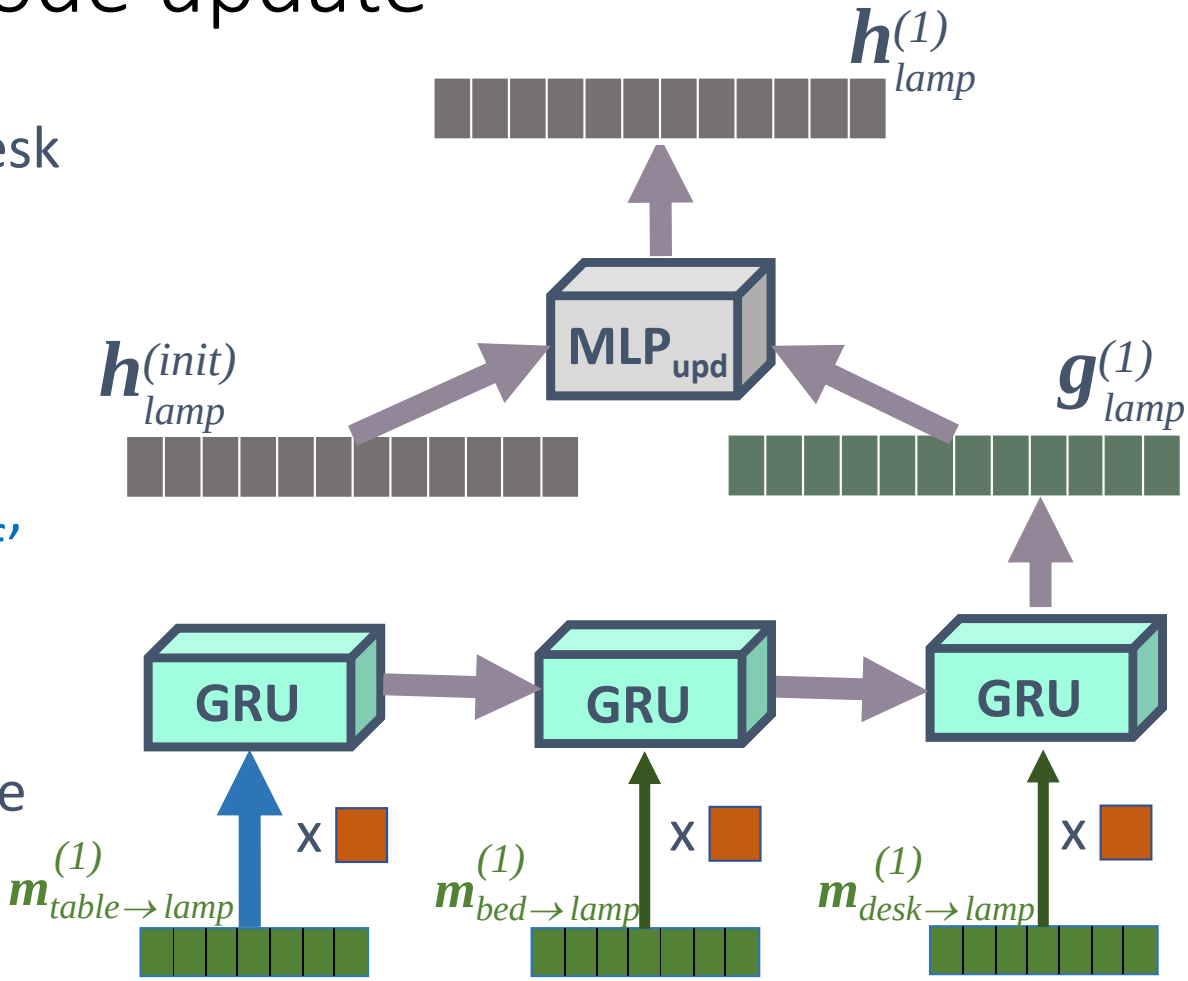
# Node update


bed
desk
lamp
'on top of'
side table

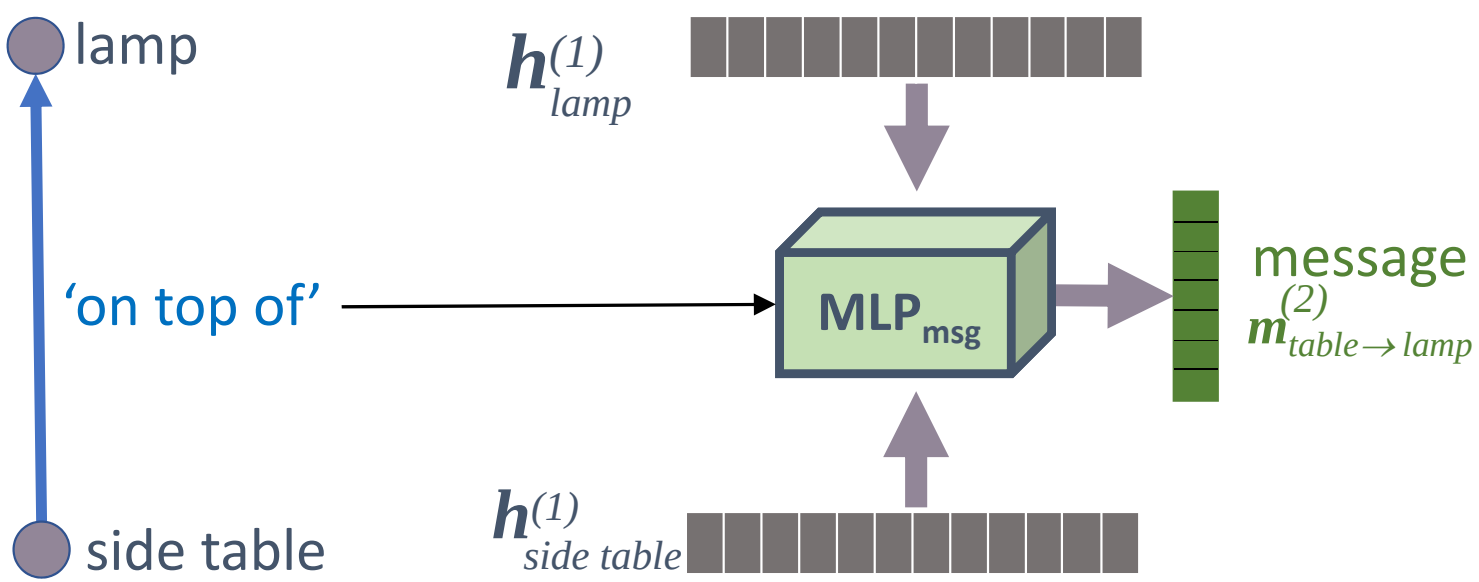

$h^{(init)}_{lamp}$
$g^{(1)}_{lamp}$
GRU
GRU
GRU
x
x
x
$m^{(1)}_{table \to lamp}$
$m^{(1)}_{bed \to lamp}$
$m^{(1)}_{desk \to lamp}$

# Node update

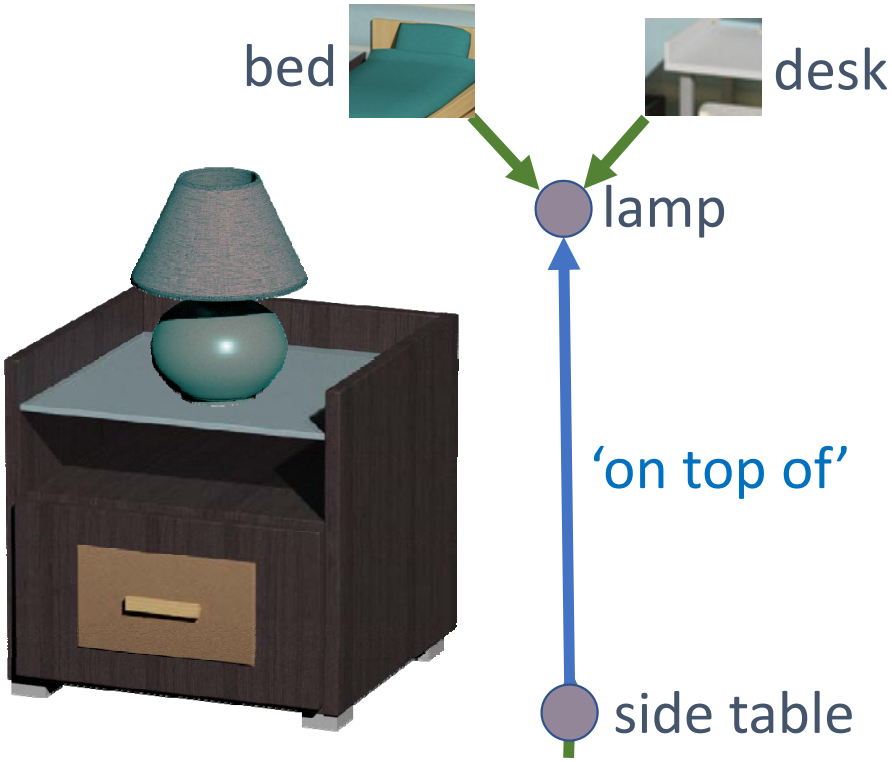

bed
desk
lamp
'on top of'
side table

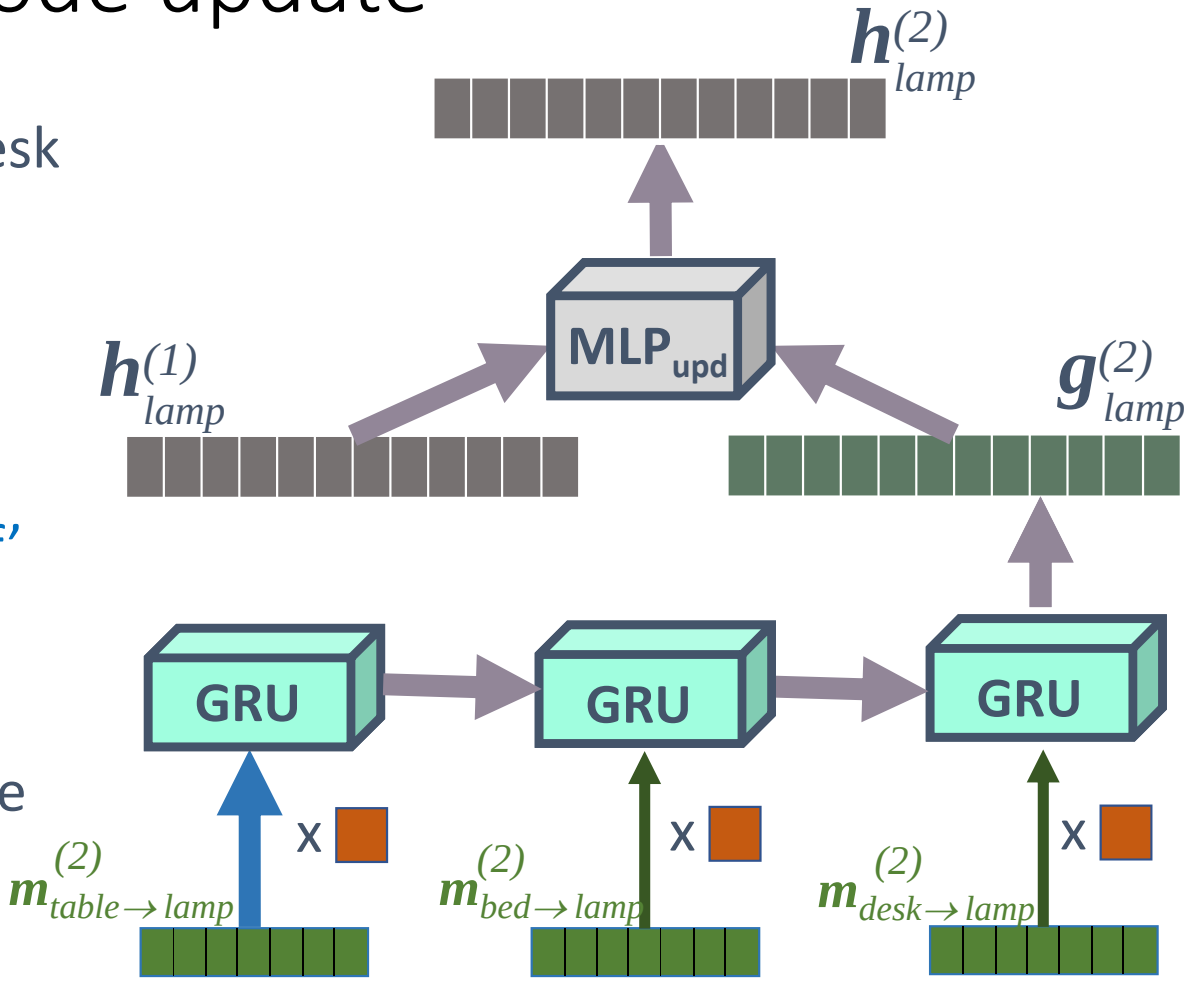

$h^{(1)}_{lamp}$
$h^{(init)}_{lamp}$
$MLP_{upd}$
$g^{(1)}_{lamp}$
GRU
GRU
GRU
x
x
x
$m^{(1)}_{table \to lamp}$
$m^{(1)}_{bed \to lamp}$
$m^{(1)}_{desk \to lamp}$

# Messages between objects

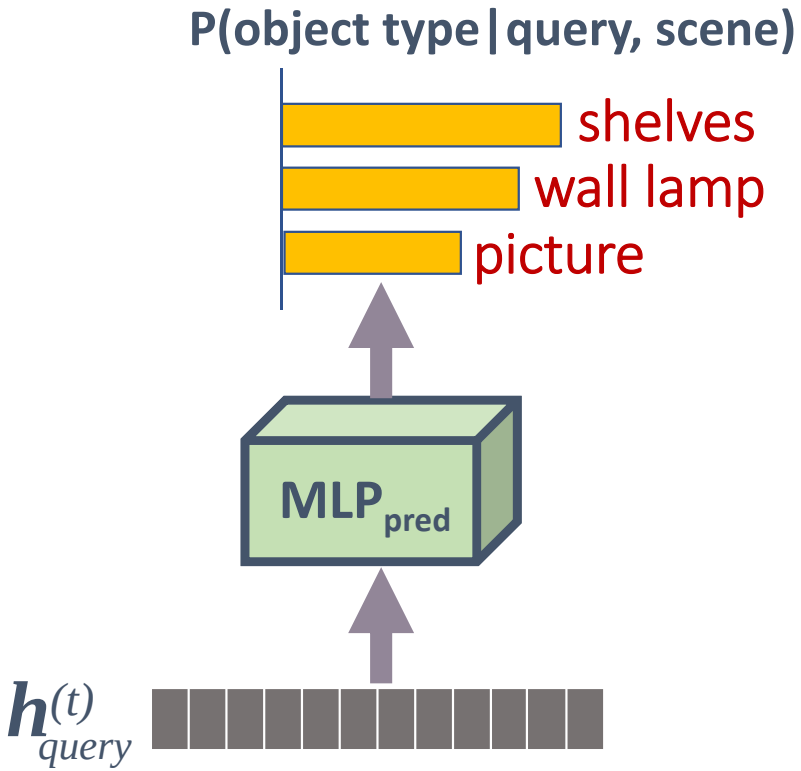


# Node update


bed
desk
lamp
'on top of'
side table

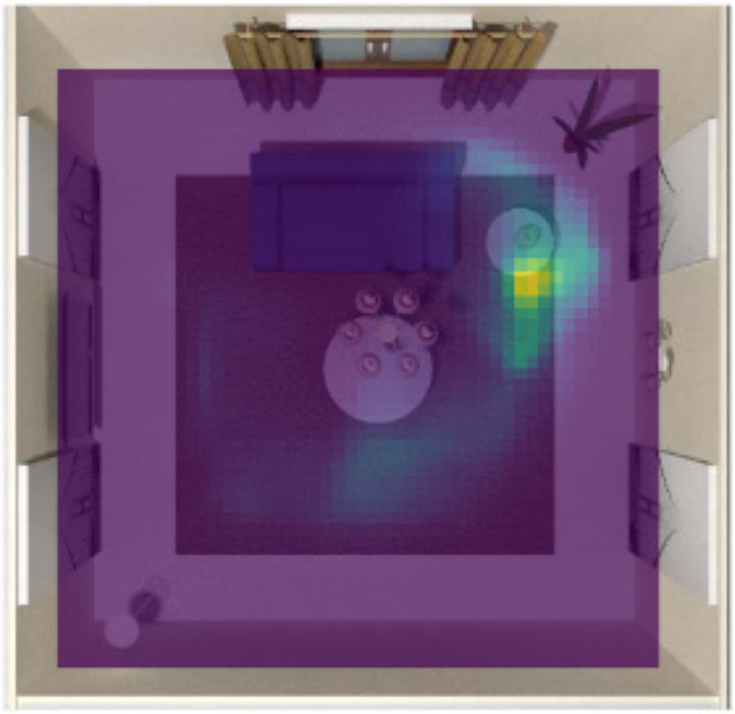

$h^{(2)}_{lamp}$
$MLP_{upd}$
$h^{(1)}_{lamp}$
$g^{(2)}_{lamp}$
GRU
GRU
GRU
x
x
x
$m^{(2)}_{table \to lamp}$
$m^{(2)}_{bed \to lamp}$
$m^{(2)}_{desk \to lamp}$

# Decoding query node representations



# Iterative synthesis

**Incomplete Scene**



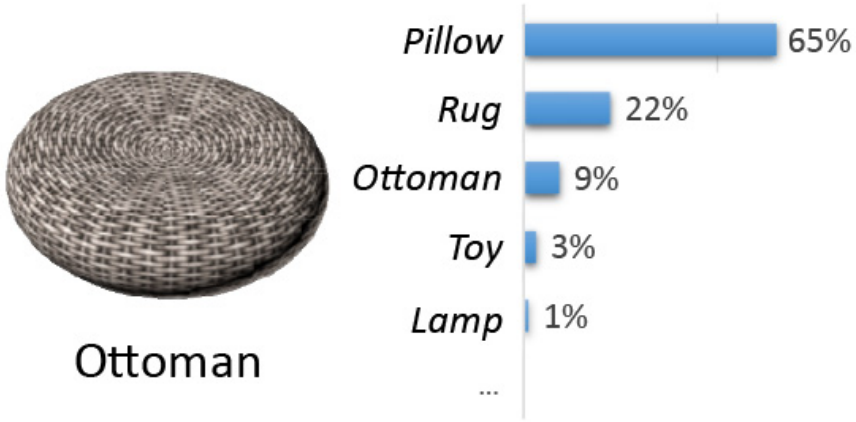

**Evaluate output probability in a dense grid of locations e.g., chair**

# Iterative synthesis

**Incomplete Scene**

**Augmented Scene**

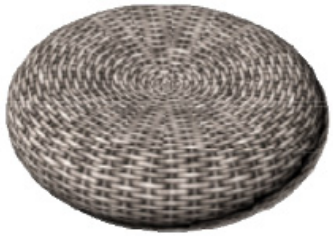
# Application: context-driven object recognition



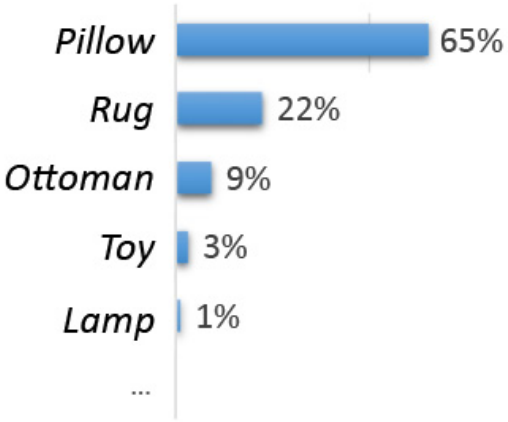

**P(object type| shape )**

(multi-view CNN, Su et al. 2015])

# Application: context-driven object recognition

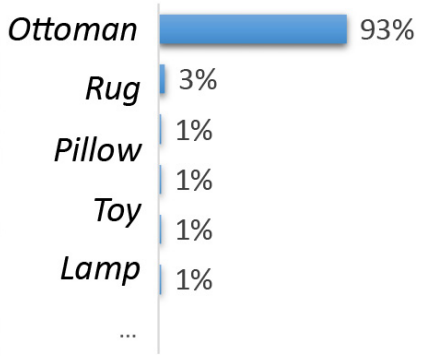


Ottoman




**P(object type| shape )**

(multi-view CNN, Su et al. 2015])




**P(object type| query, scene)** x

**P(object type| shape )**

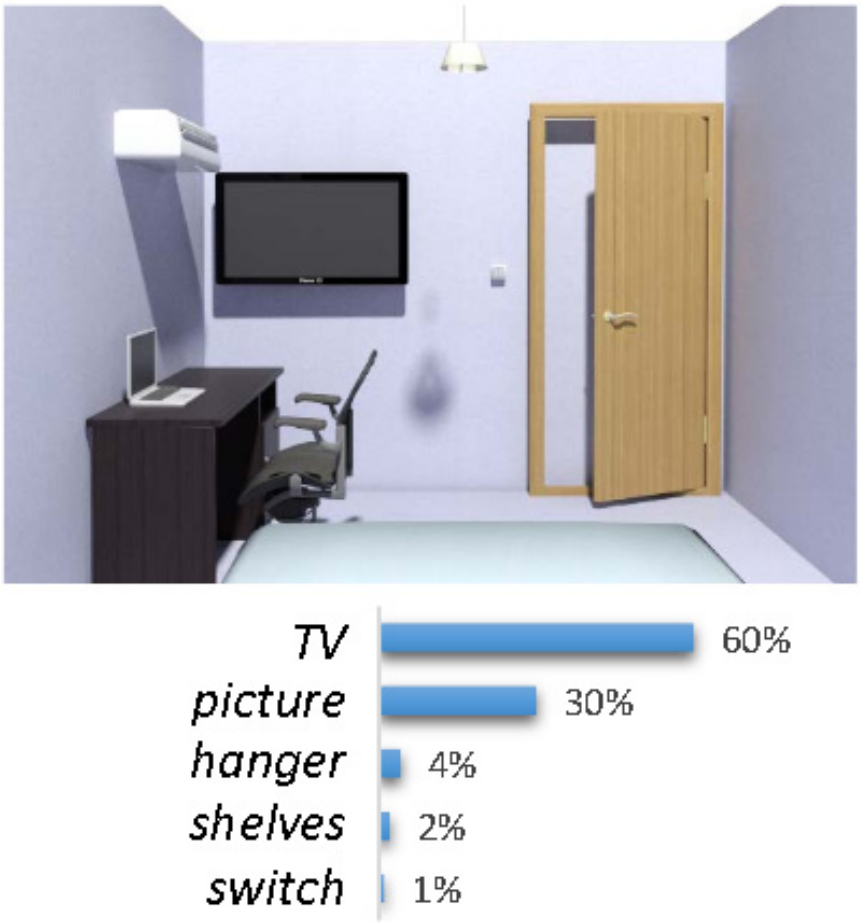
# Training

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

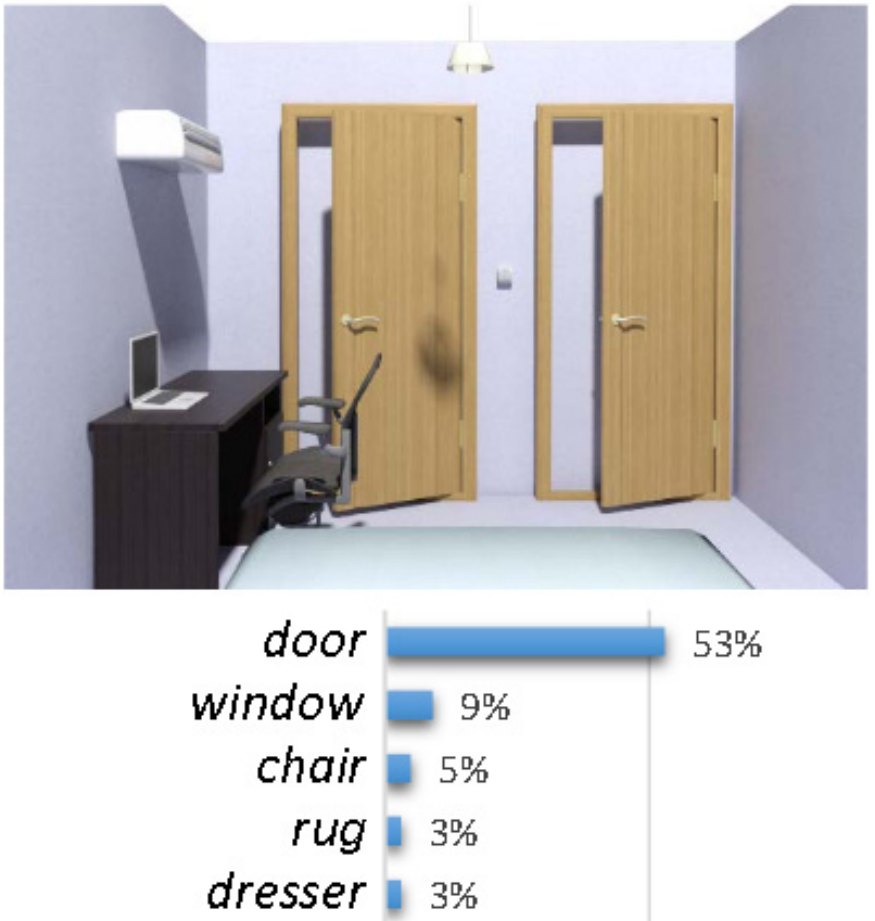
We remove random objects, then train our network to predict them

**[Song et al. 16]**

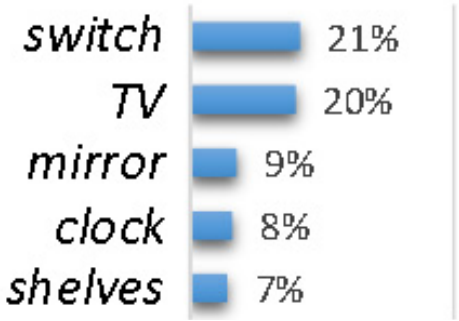
# SceneGraphNet Results

# 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?

| Method | Correct object type |
|---|---|
| “GRAINS” * | 44.2% |
| | |
| | |

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

# Numerical Evaluation

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

| Method | Correct object type |
|---|---|
| “GRAINS” * | 44.2% |
| Wang et al.** | 50.9% |
| | |

*** 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?

| Method | Correct object type |
|---|---|
| “GRAINS” * | 44.2% |
| Wang et al.** | 50.9% |
| **SceneGraphNet** | **67.3%** |

*** 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?

| Method | Correct object within top-5 predictions |
|---|---|
| “GRAINS” * | 73.6% |
| Wang et al.** | 79.3% |
| **SceneGraphNet** | **89.5%** |

*** 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

| Method | Classification accuracy |
|---|---|
| MVCNN * | 59.2% |
| MVCNN + **SceneGraphNet** | 72.2% |

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

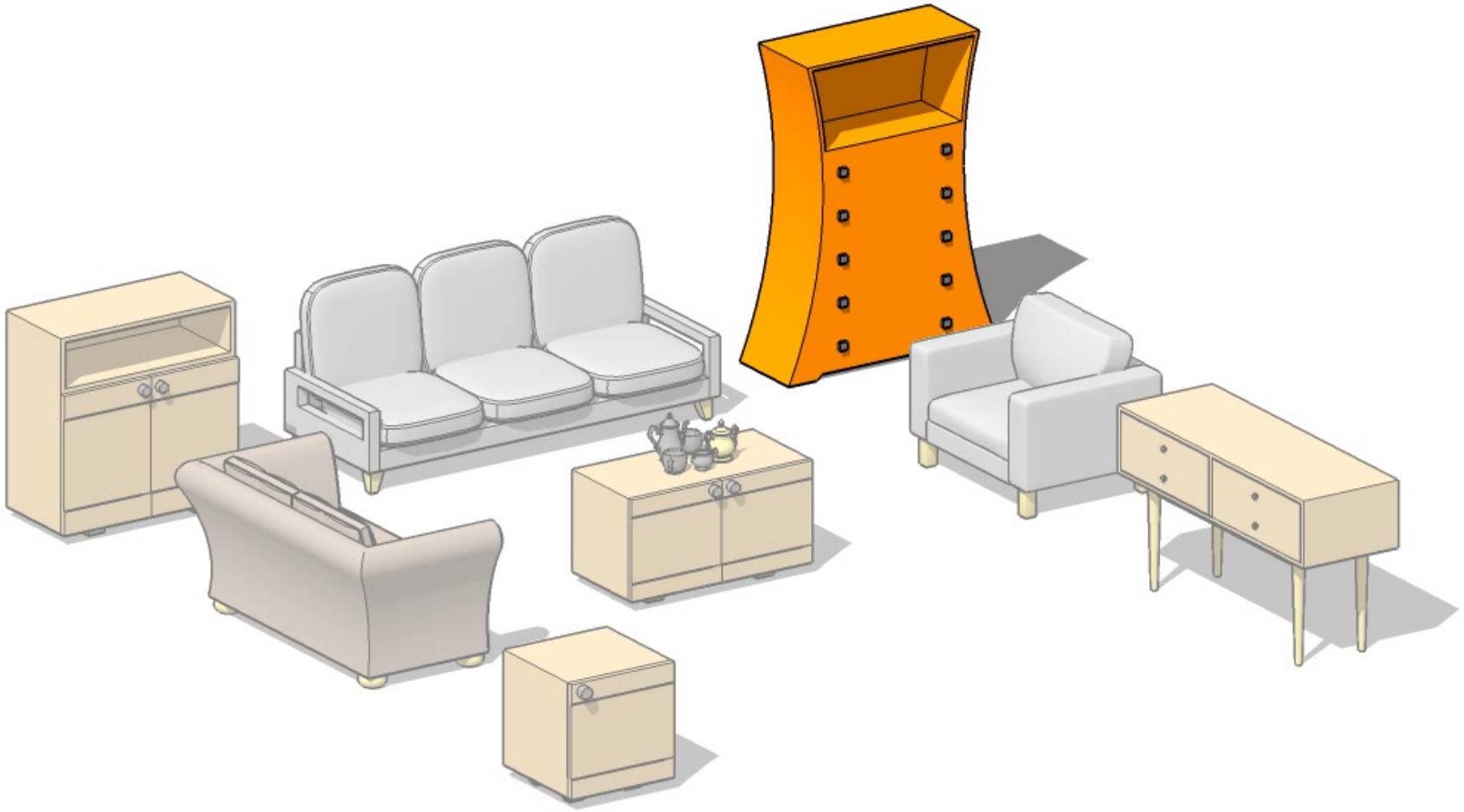
# Summary

- 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

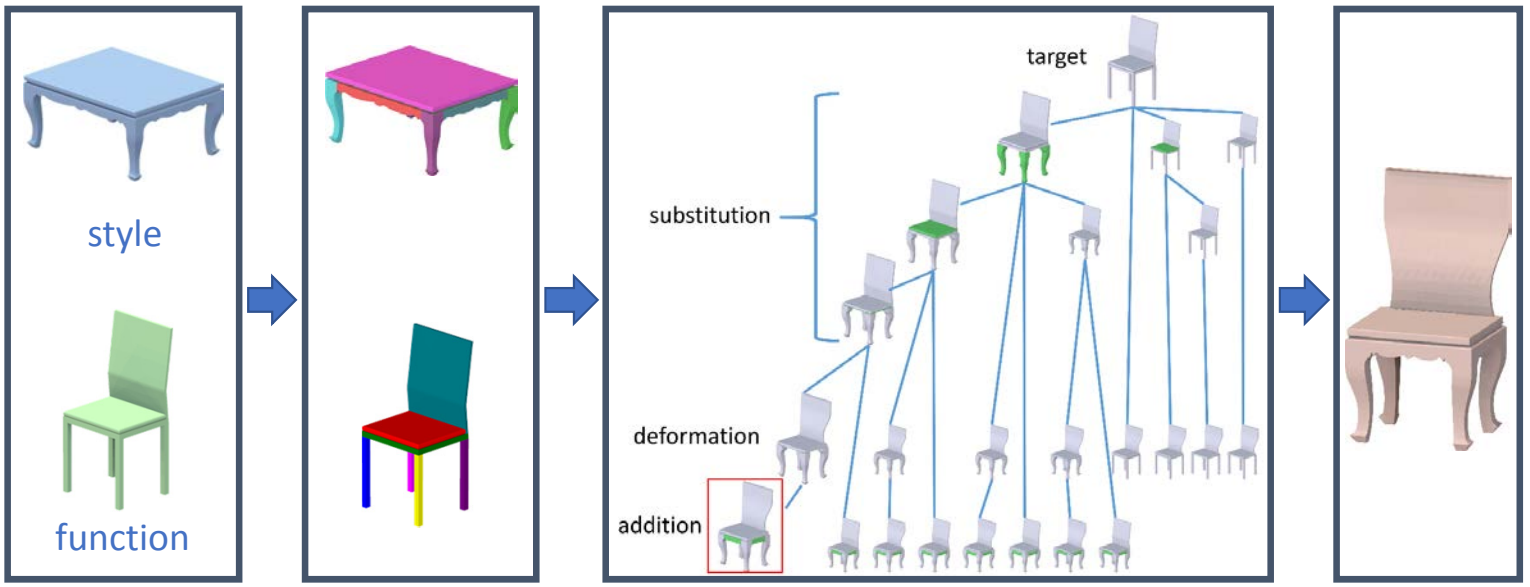
# Key idea

Search for **compatible** element operations that**:**

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

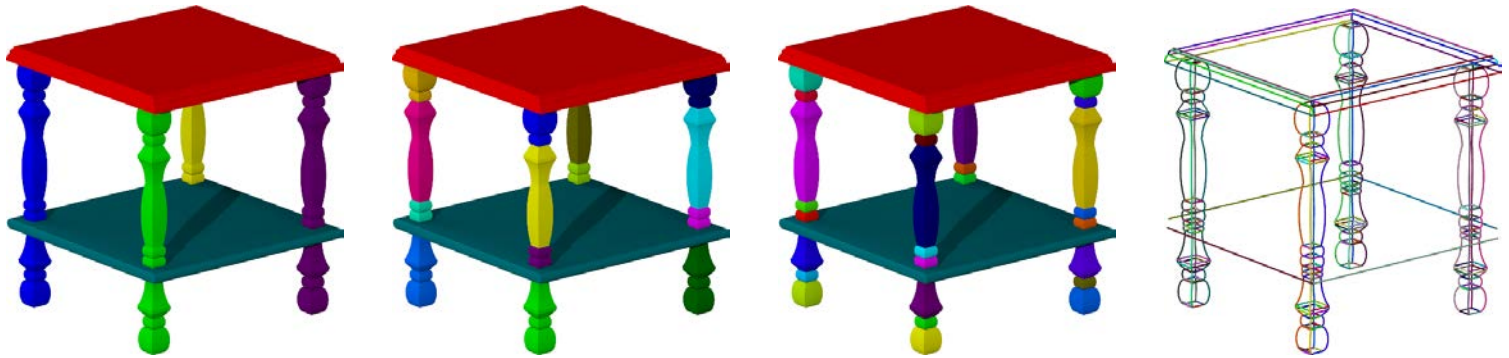
- **Increase style similarity to exemplar**
  => **style transfer**

# Framework overview

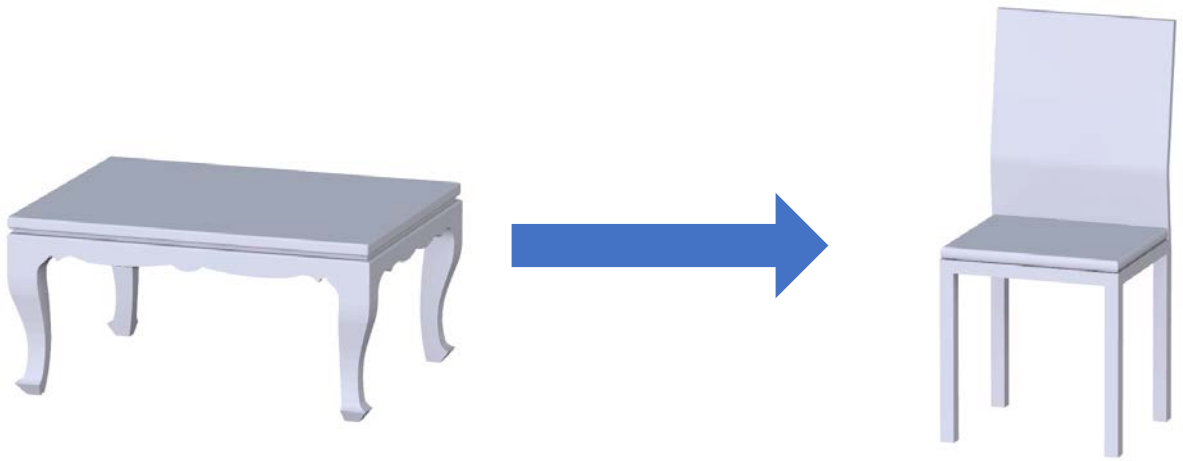



Input Segmentation Tabu Search Output

# Hierarchical segmentation and extracted curves

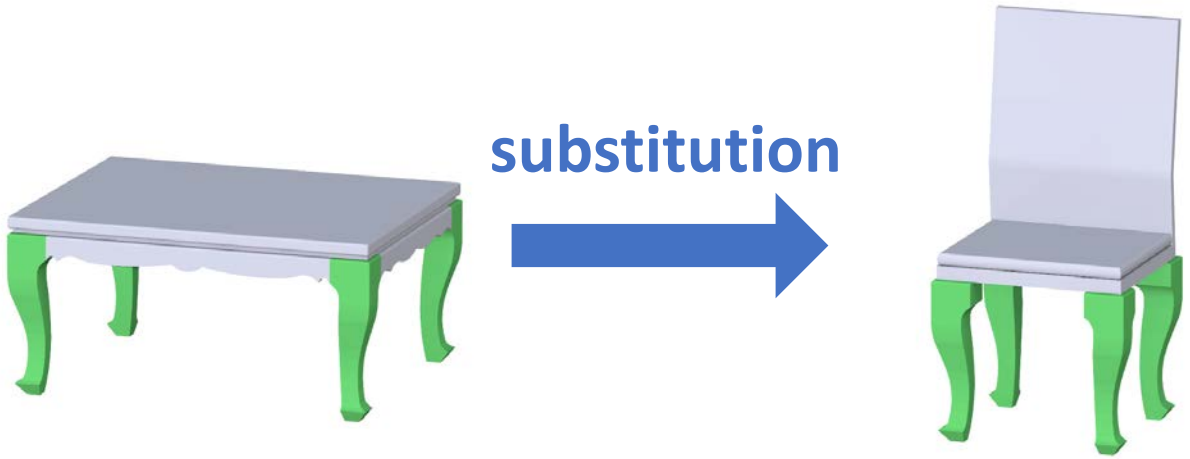


# Element-level operations



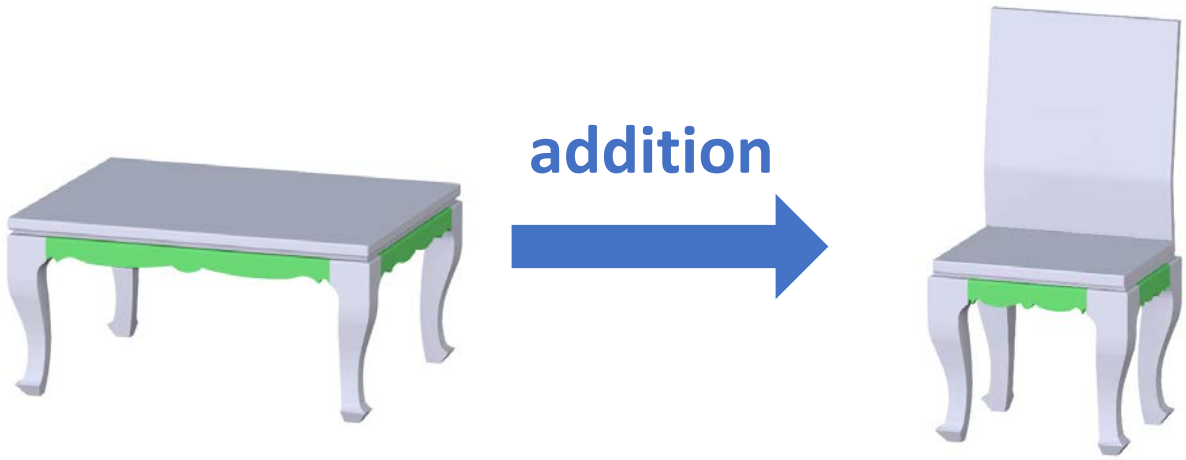
# Element-level operations

- part substitution



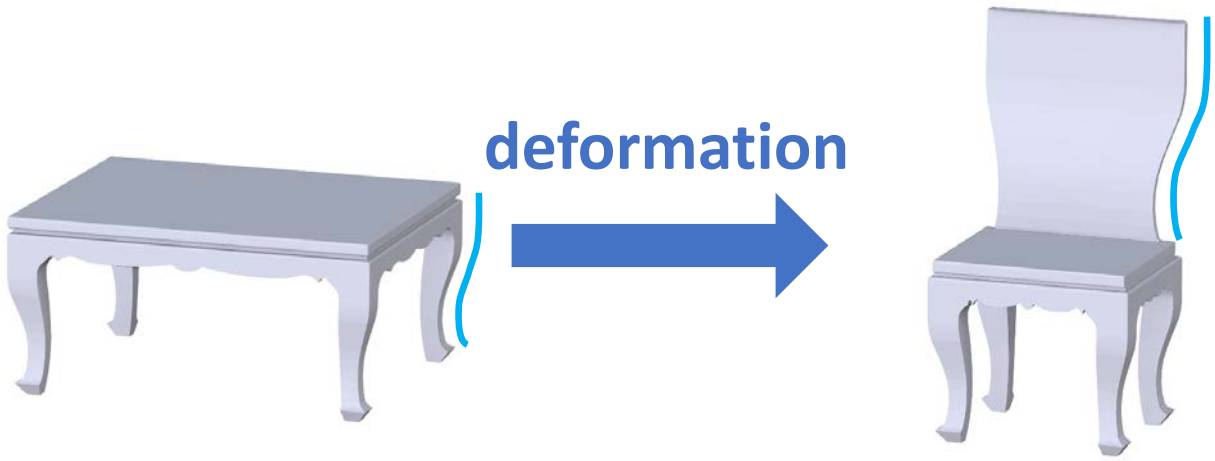
# Element-level operations

- part substitution
- part addition / removal



# Element-level operations

- part substitution
- part addition / removal
- curve-based deformation



# Unconstrained element edits

style

function

output

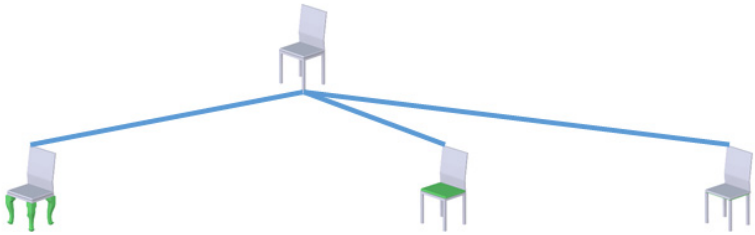
# Compatible element edits

style

function

output

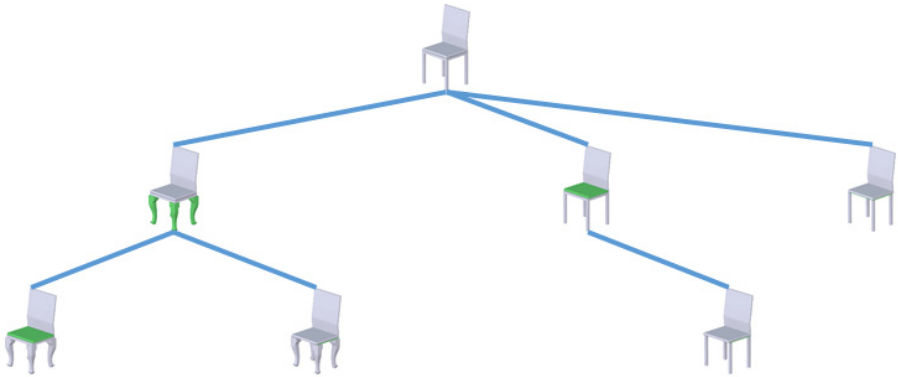
# Search for compatible element-level operations



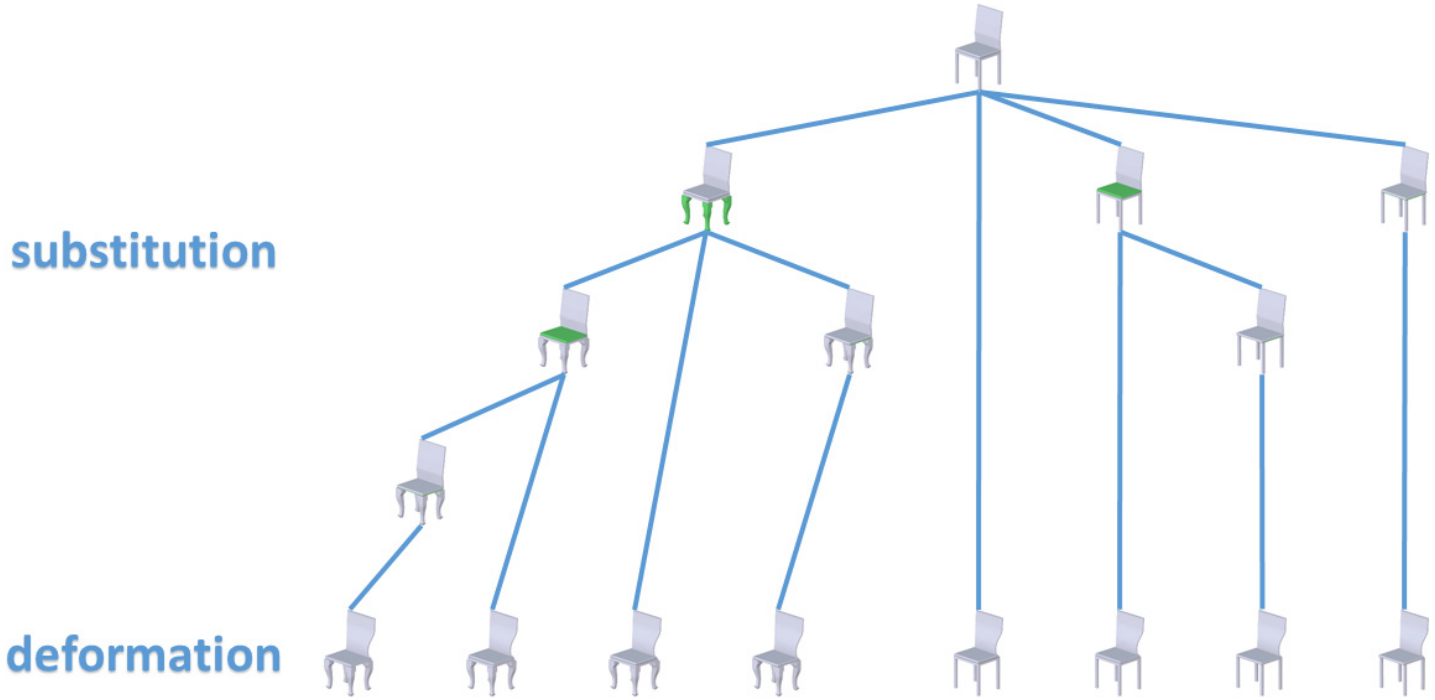
**substitution**

# Search for compatible element-level operations

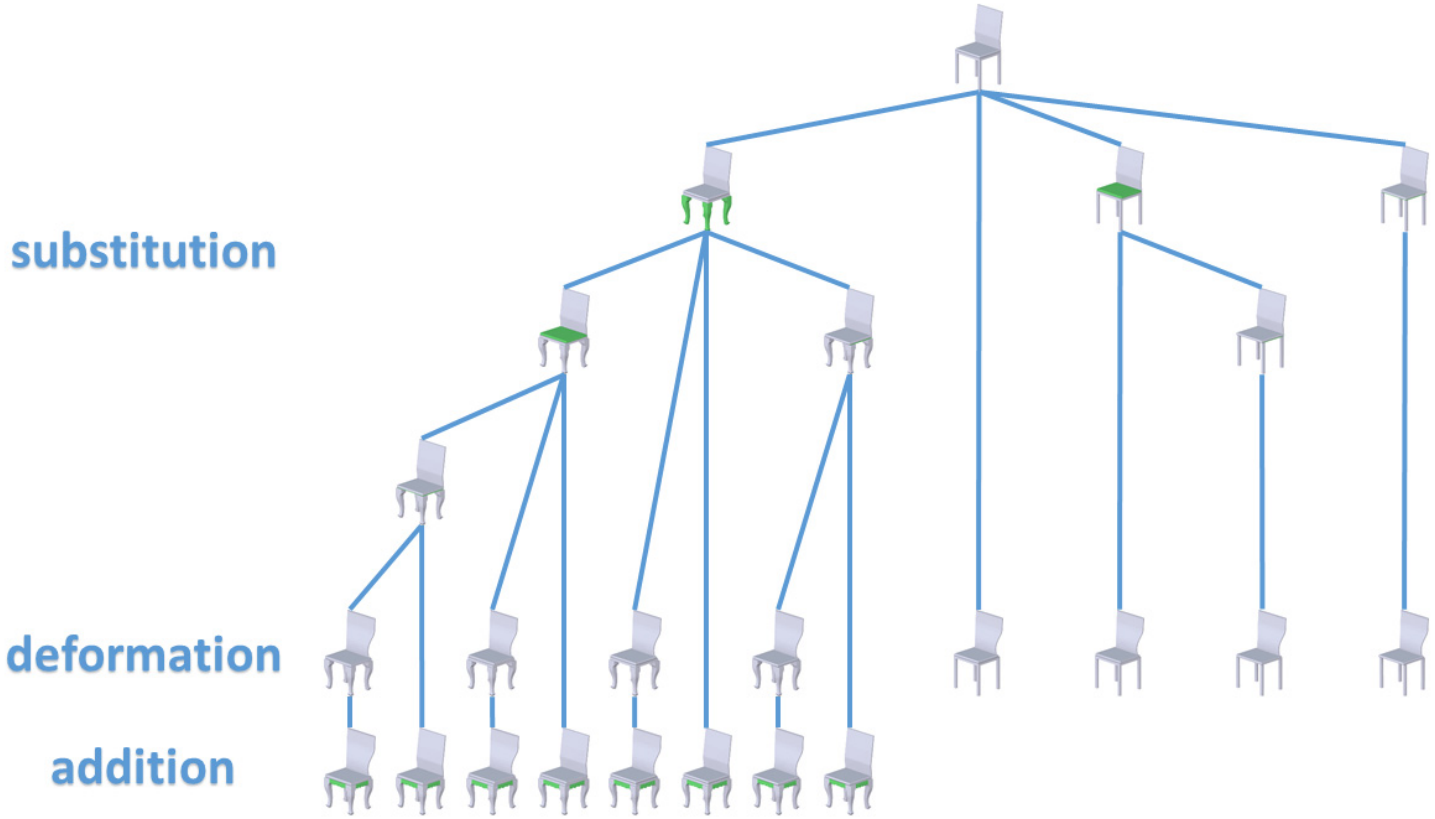
**substitution**



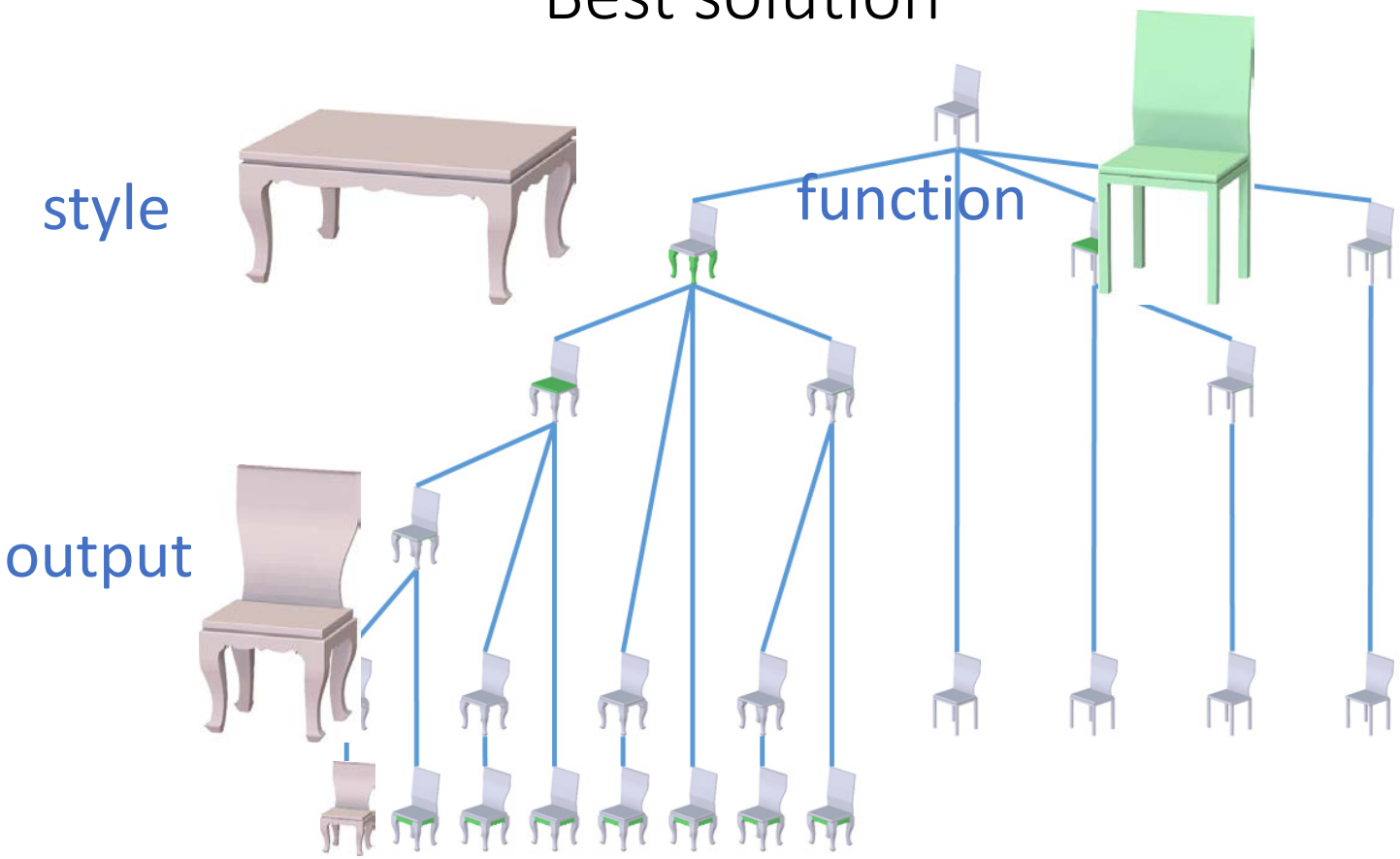
# Search for compatible element-level operations



# Search for compatible element-level operations



# Best solution



# 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**

$D_{func}$(  ,  ) = 1.32

$D_{func}$(  ,  ) = 0.57

# Measures

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

- **Functional compatibility measure**

$D_{func}$(  ) = 1.32

$D_{func}$(  ) = 0.57

# Measures

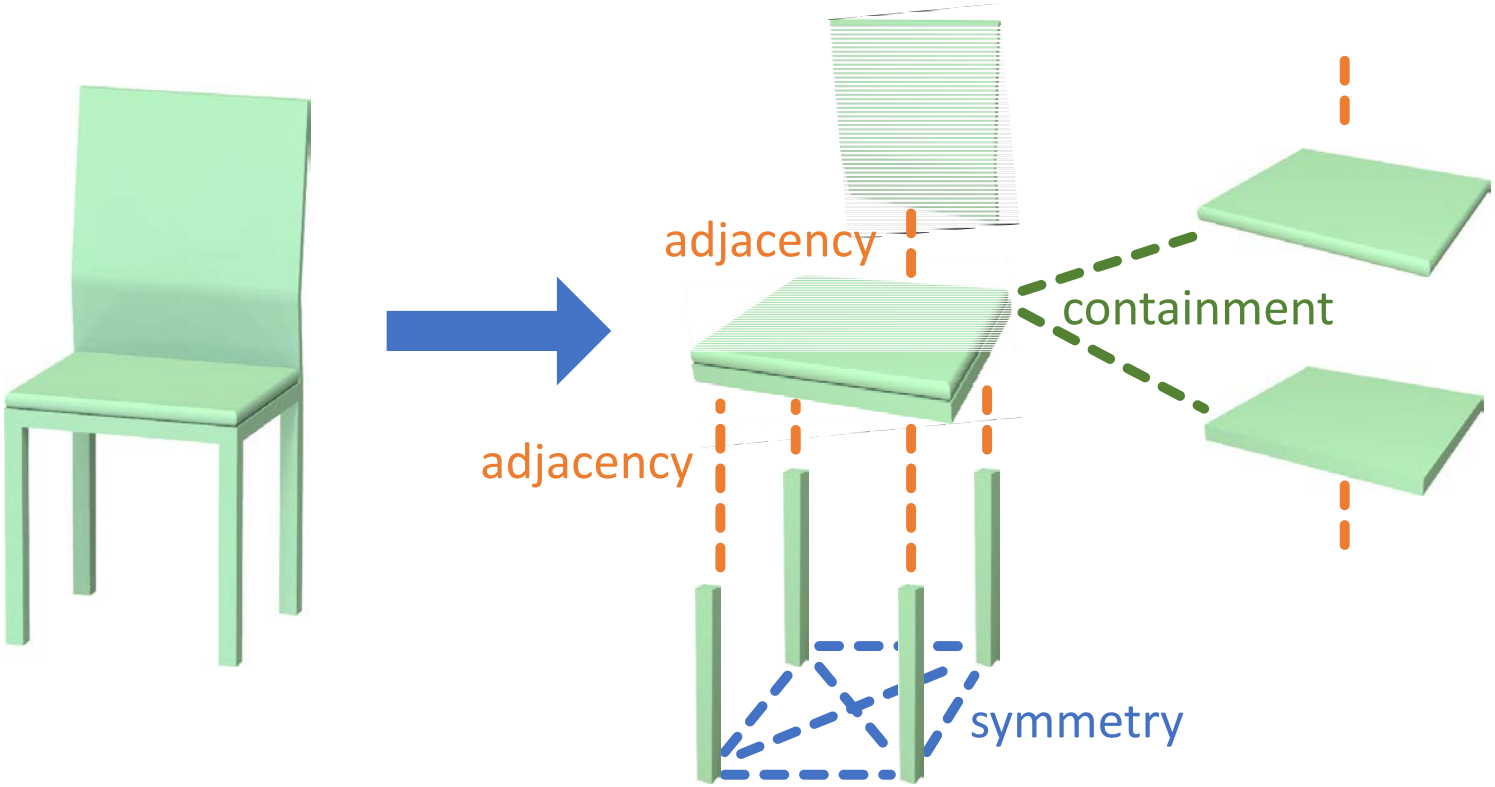
- Style similarity measure from:
  "Elements of Style: Learning Perceptual Shape Style Similarity",
  Lun, Kalogerakis, Sheffer, TOG 2015

- **Functional compatibility measure**

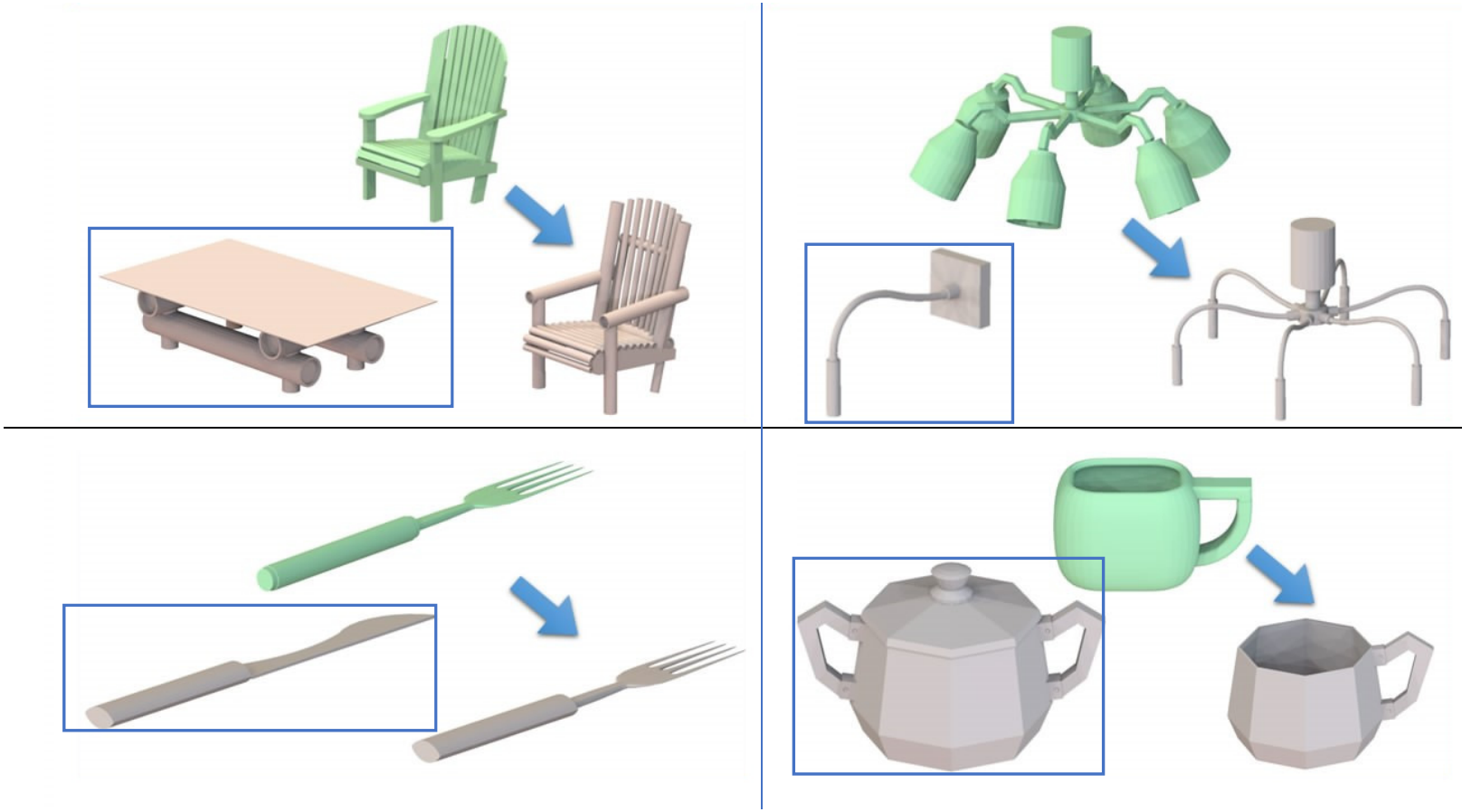
$D_{func}$(  ,  ) = 1.32

$D_{func}$(  ,  ) = 0.57

# Element relation graph



# Results



# User study: style similarity

target (function)

A

exemplar (style)

B our top-ranked output

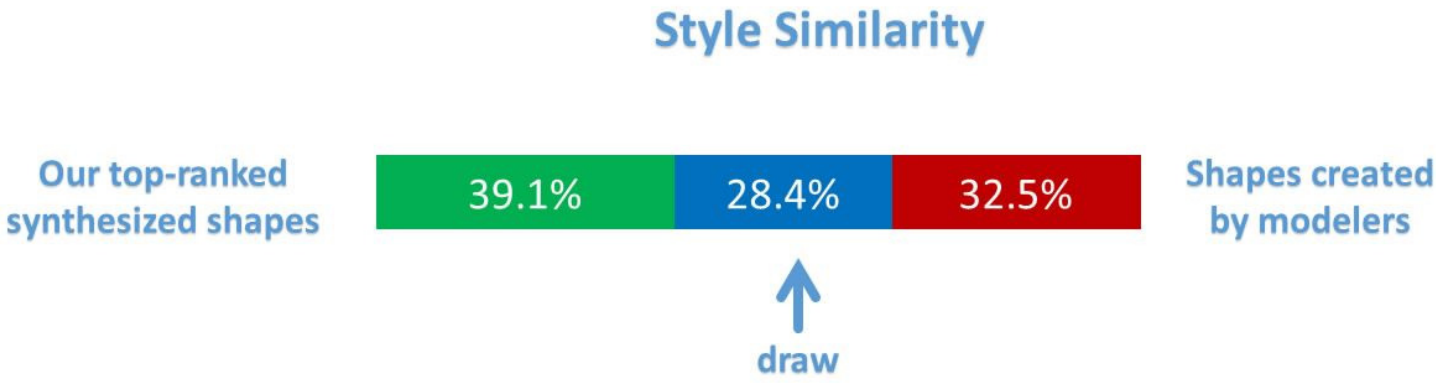
C created by modeler

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

NEXT

# User study: style similarity



# 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