Creating Consistent Scene Graphs Using a Probabilistic Grammar

Tianqiang Liu[1] Siddhartha Chaudhuri[1,2] Vladimir G. Kim[3]

Qi-Xing Huang[3,4] Niloy J. Mitra[5] Thomas Funkhouser[1]

1 Princeton University

2 Cornell University

3 Stanford University

4 TTIC

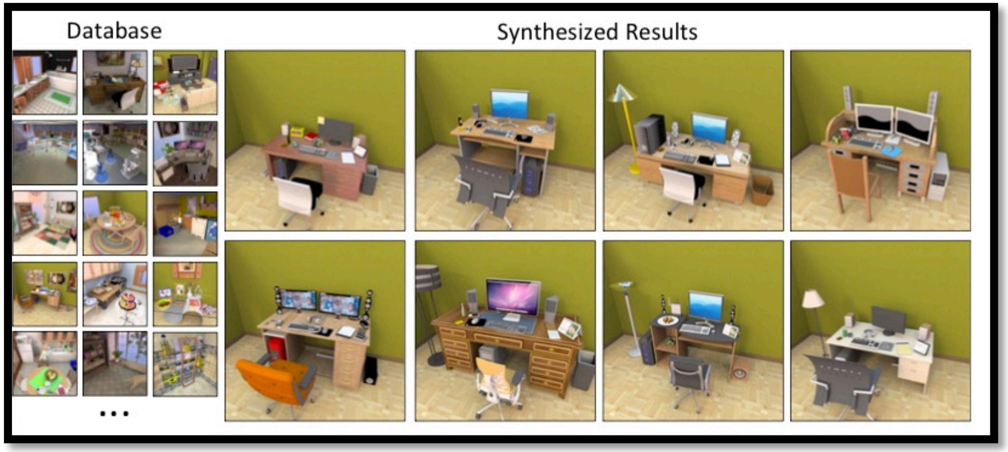
5 UCL

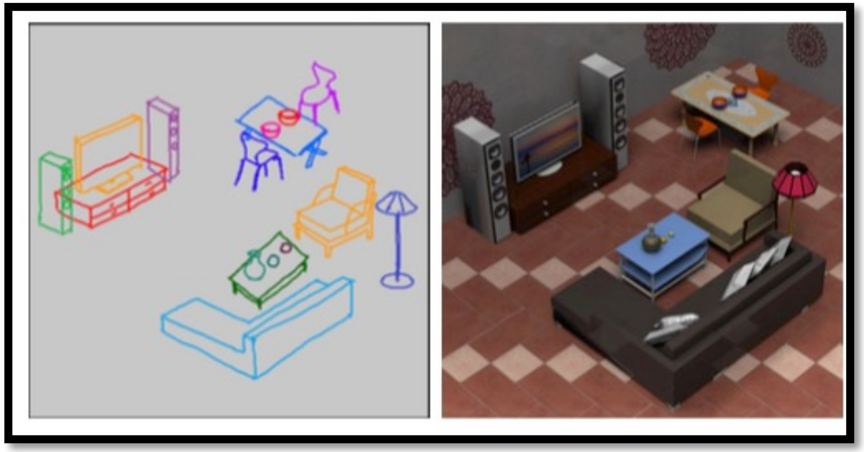
Motivation

Growing number of 3D scenes online.

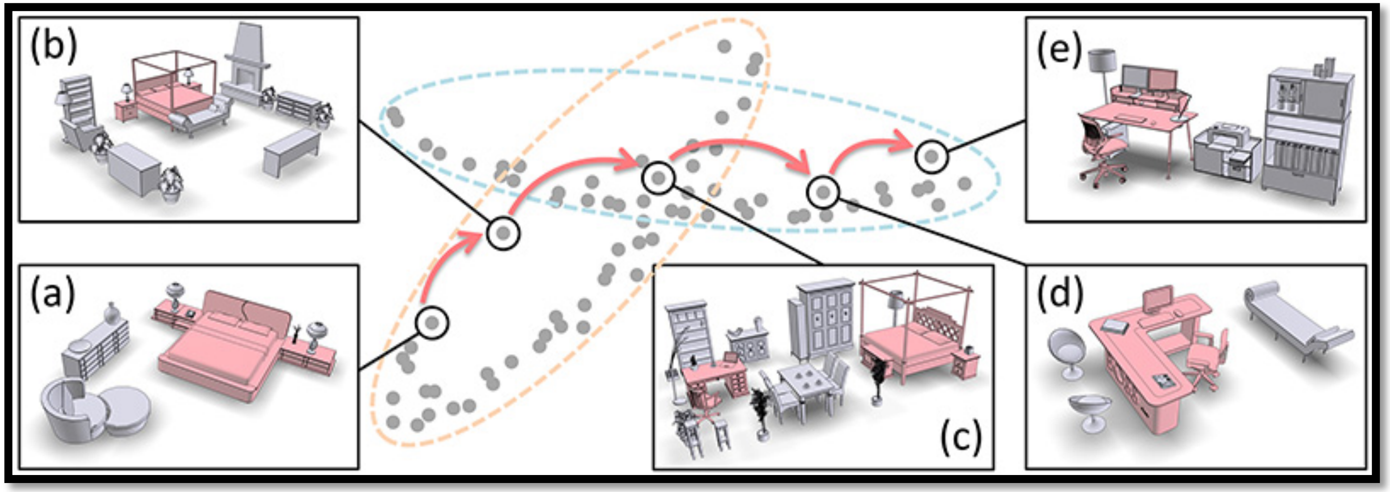
Google 3D warehouse

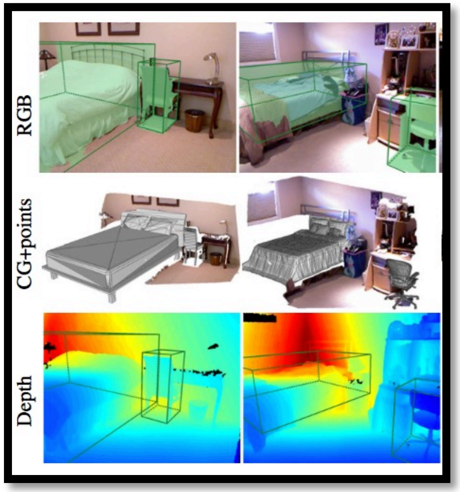
Motivation





Synthesis [Fisher et al 2012, Xu et al 2013]

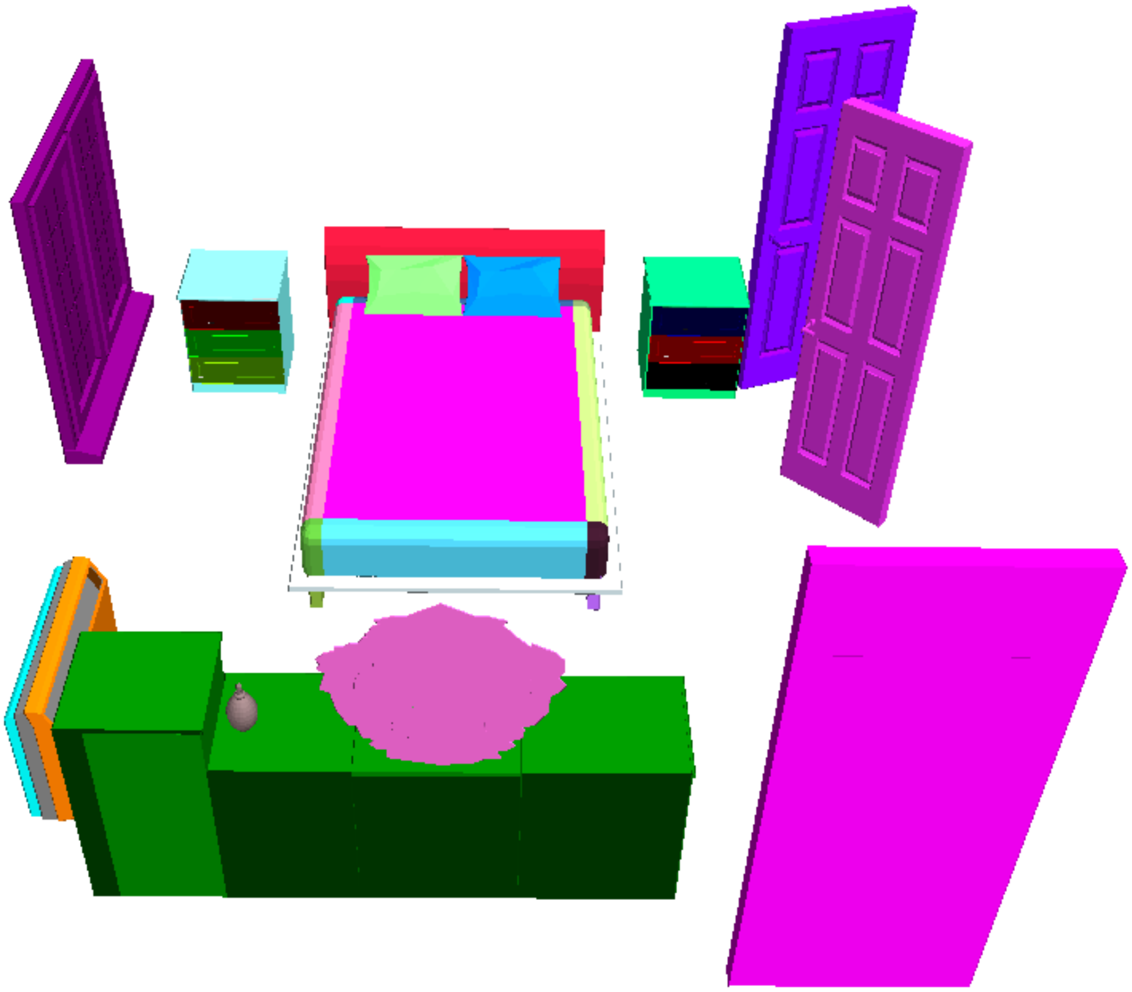




Understanding [Xu et al 2014, Song et al 2014]

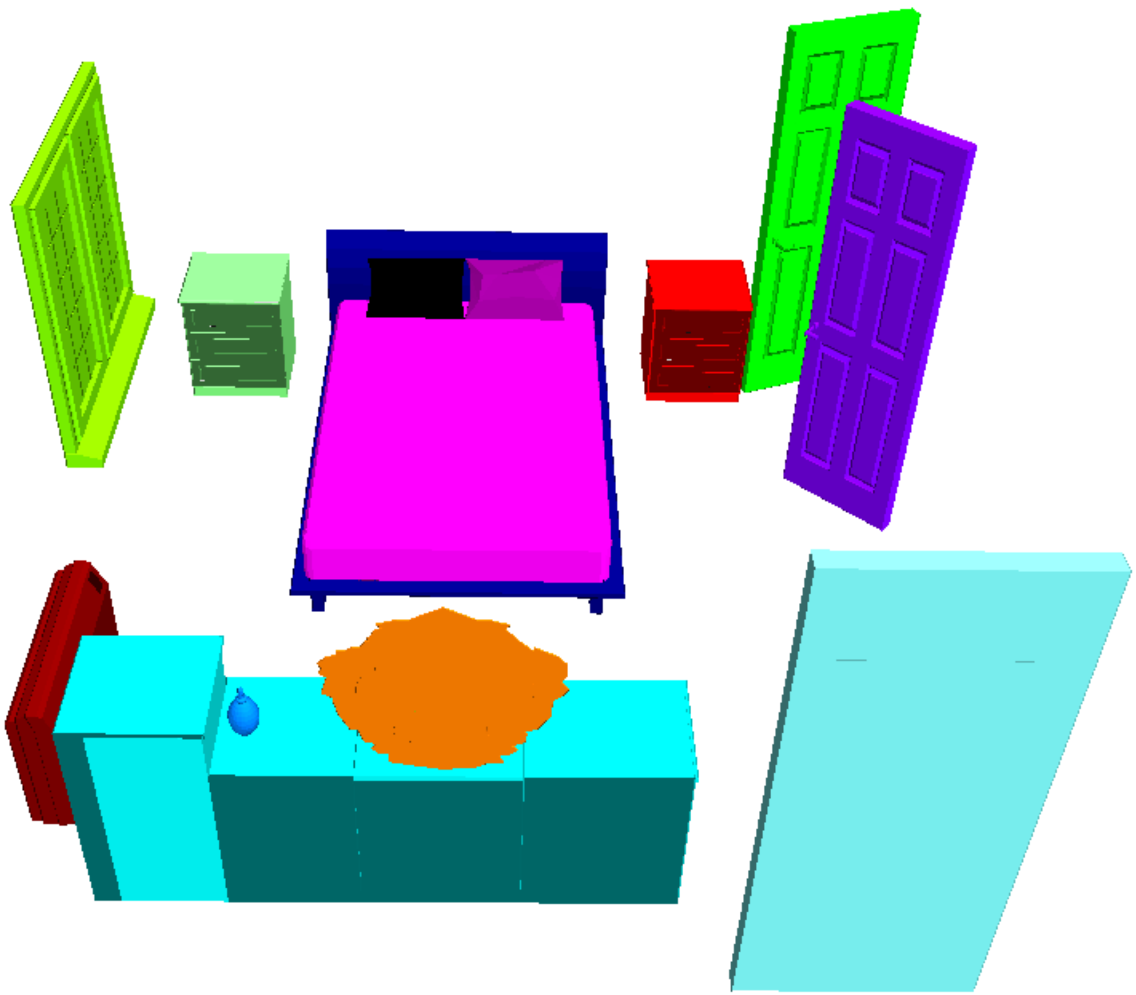
Goal

Input: A scene from Trimble 3D Warehouse



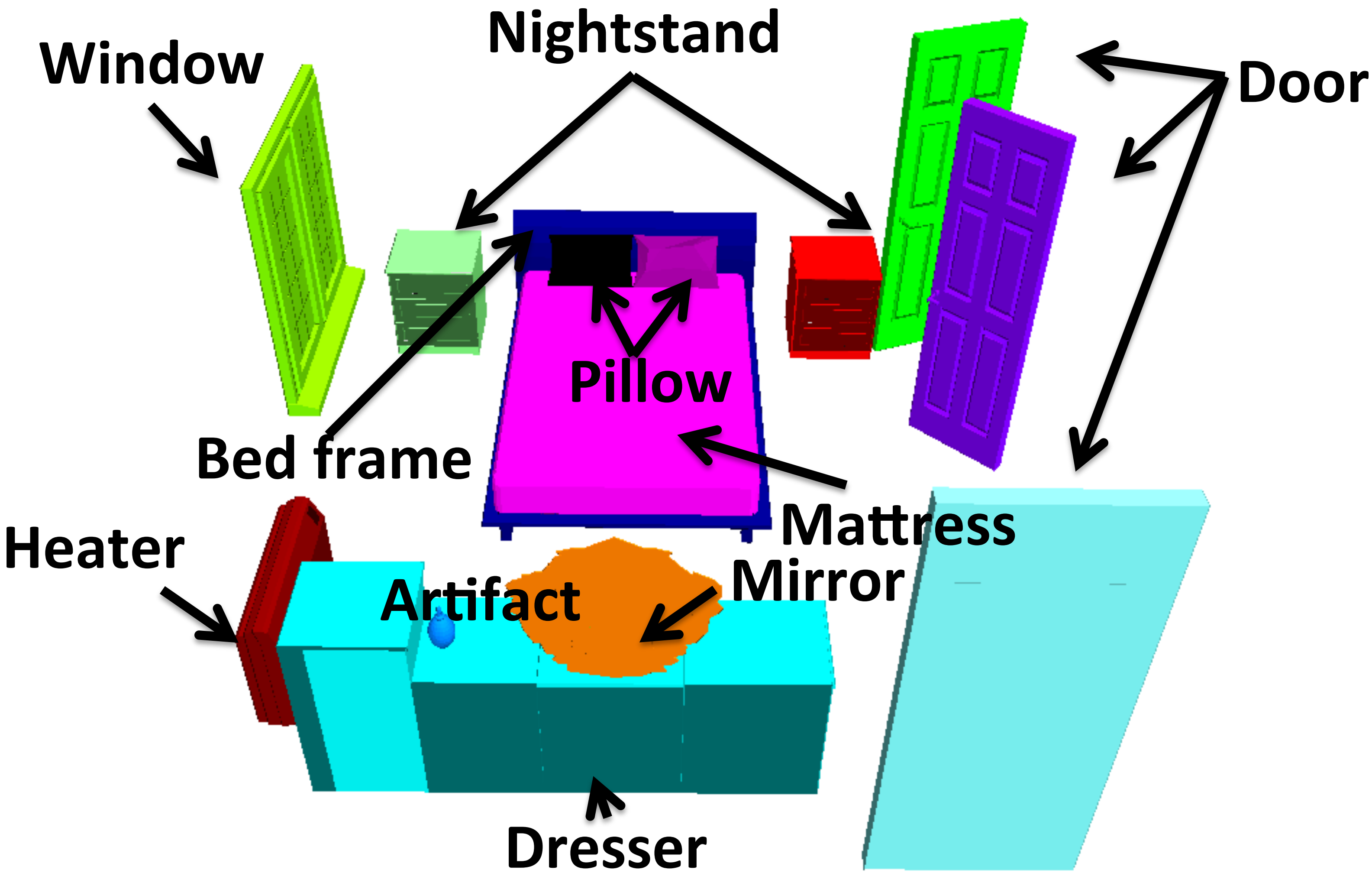
Goal

Output 1: Semantic segmentations



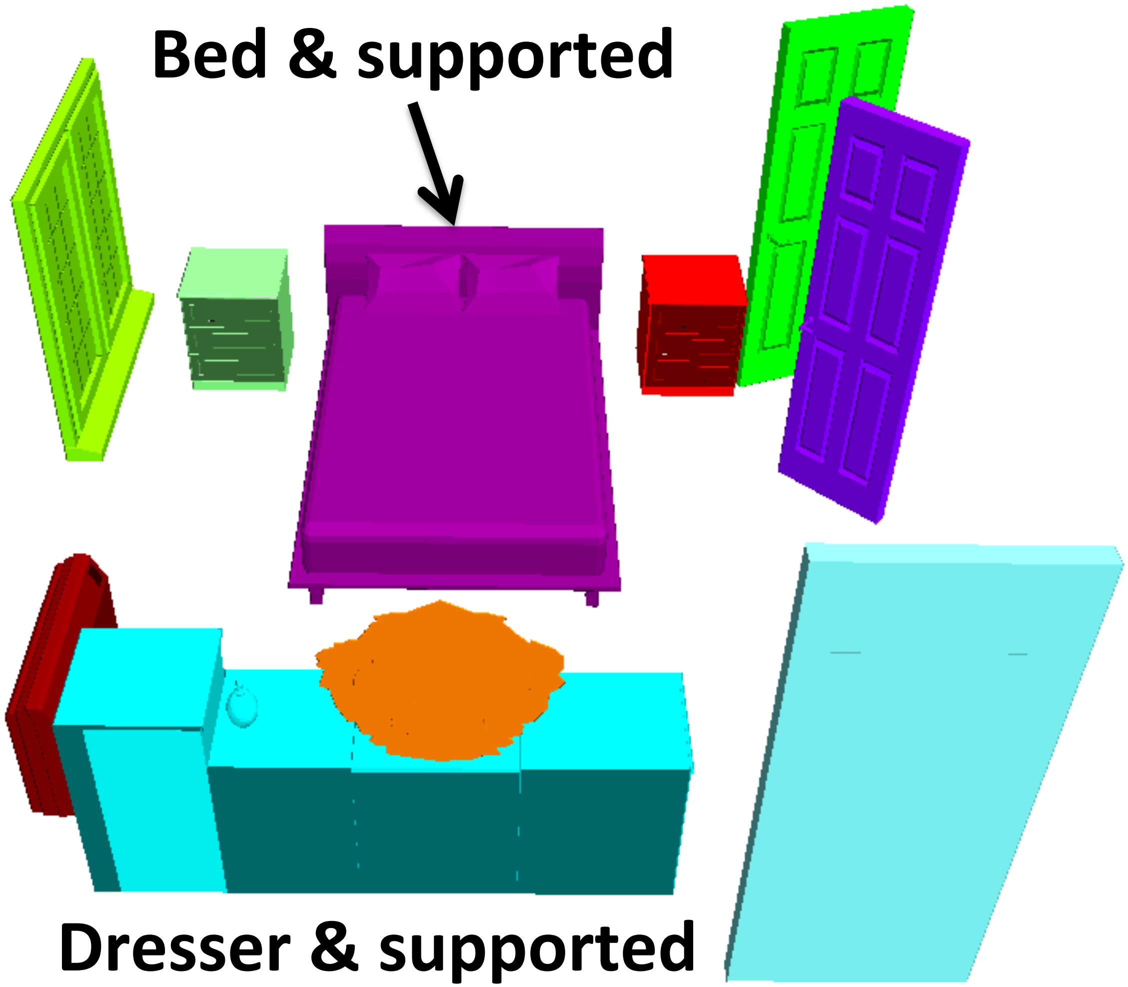
Goal

Output 2: Category labels.



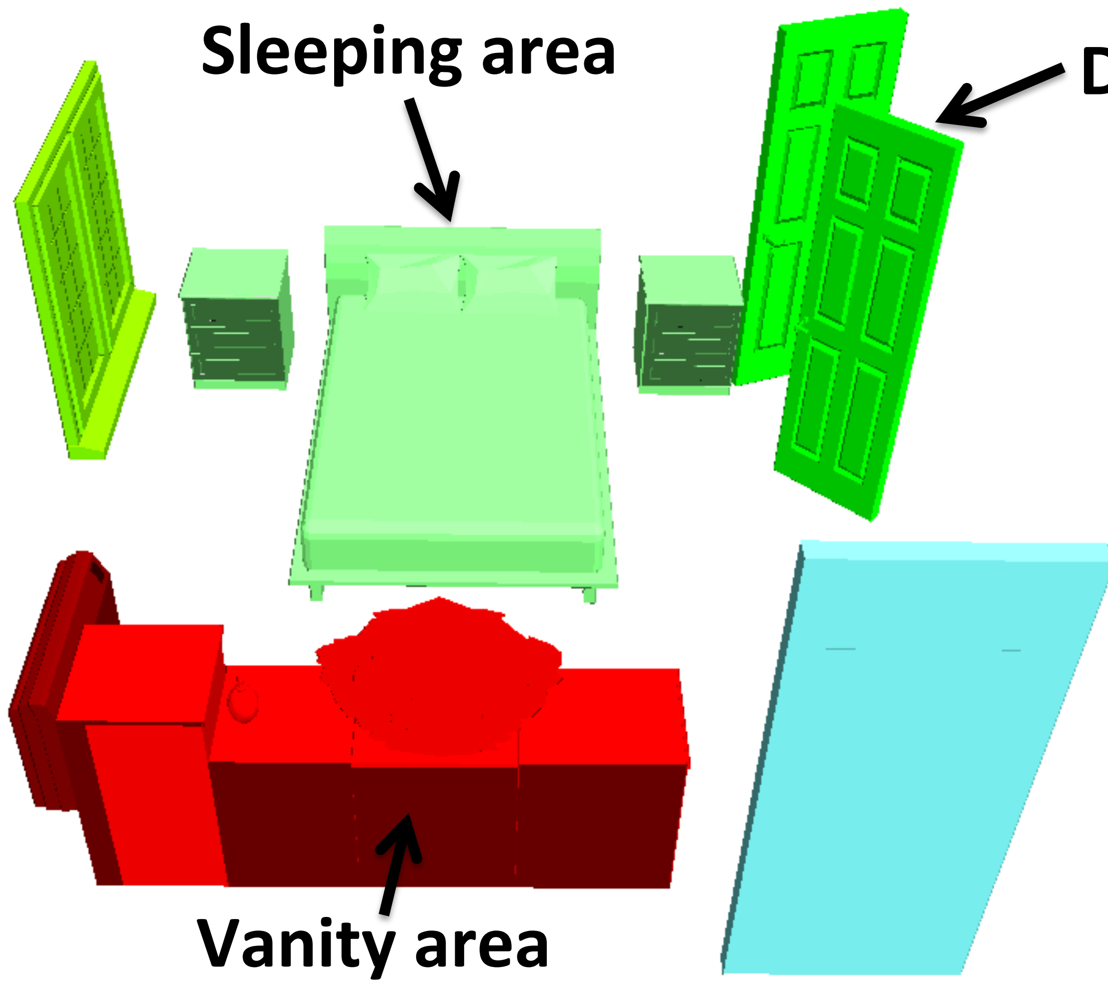
Goal

Output 2: Category labels at different levels.



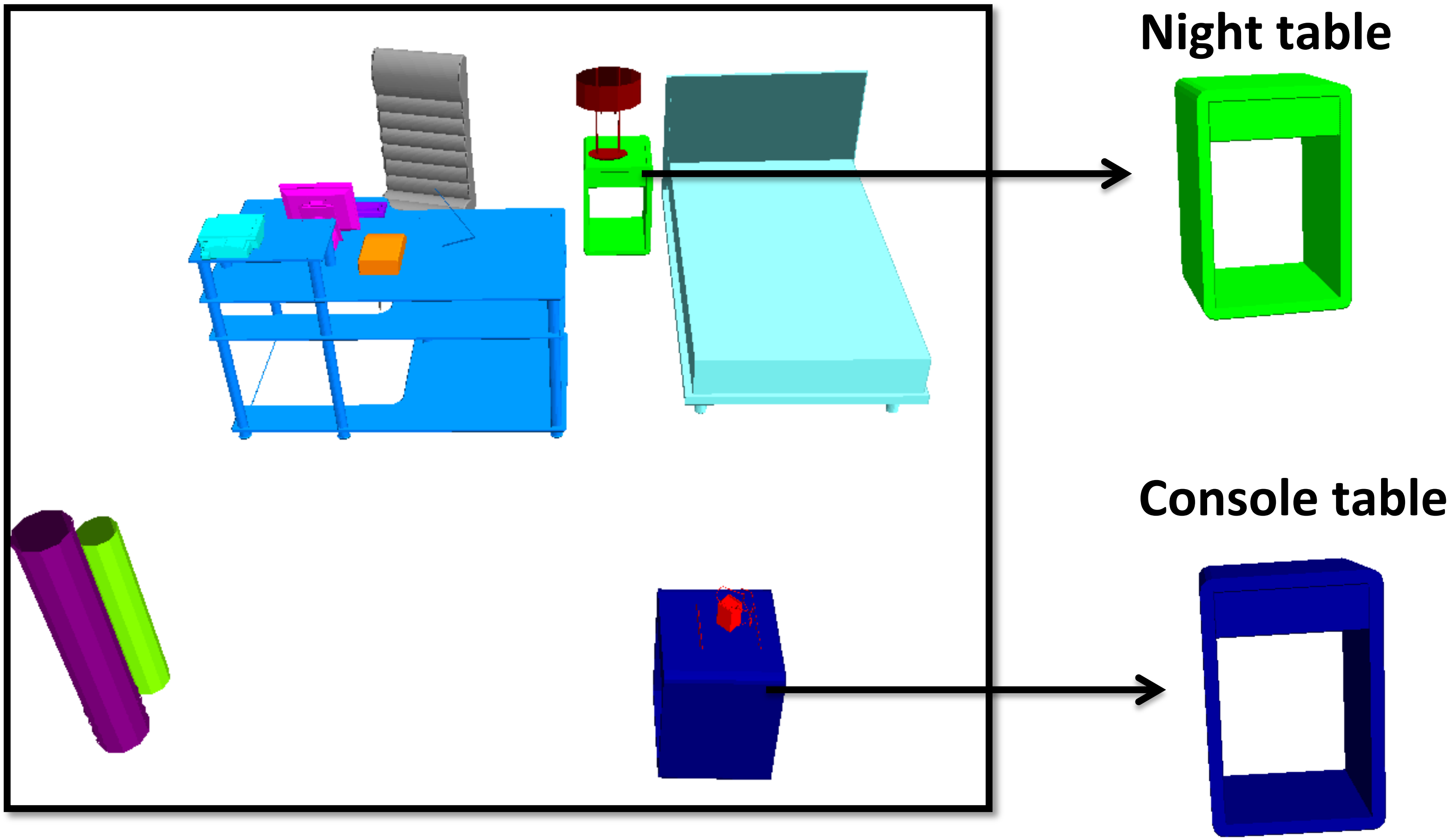
Goal

Output 2: Category labels at different levels.



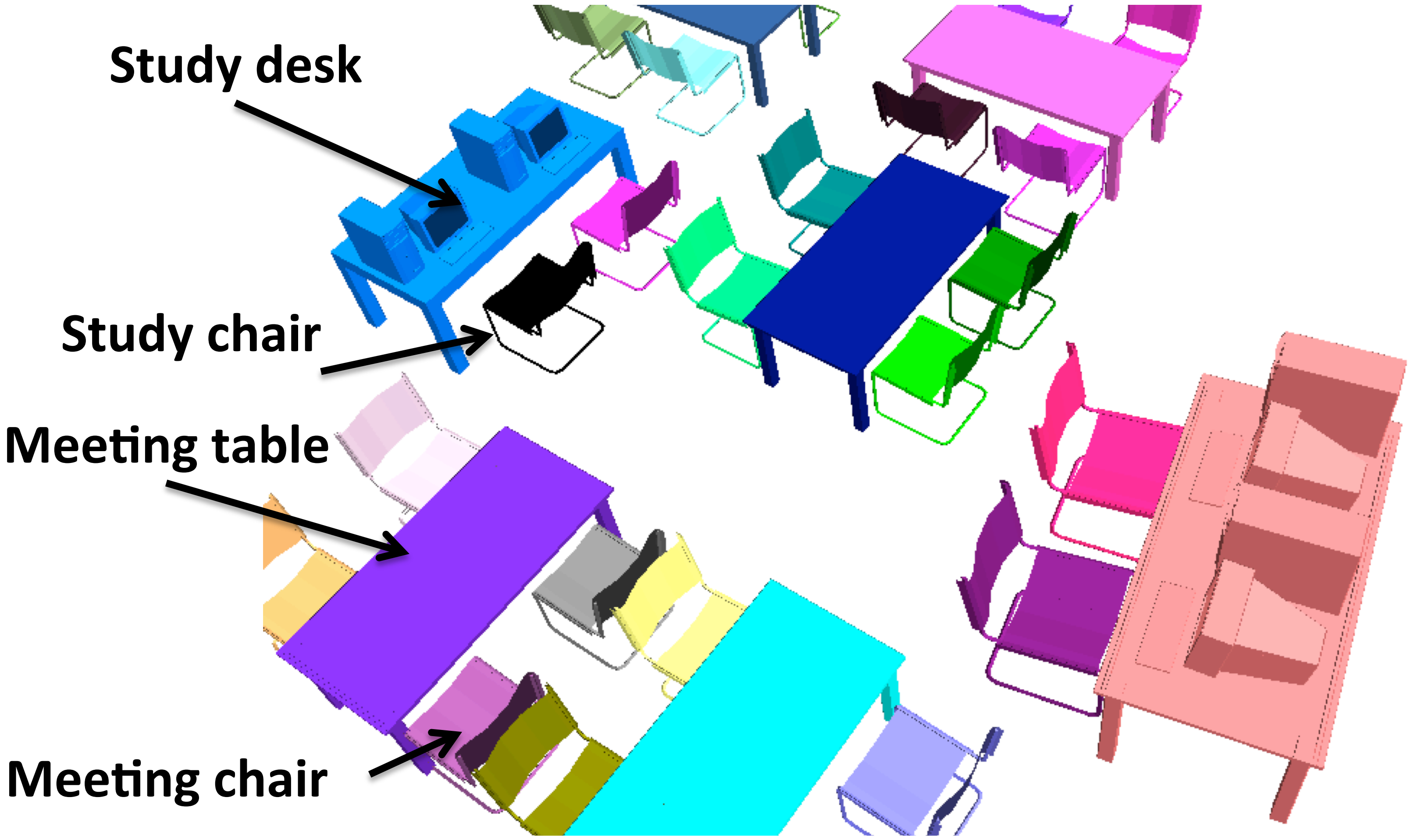
Challenges

Shape is not distinctive.



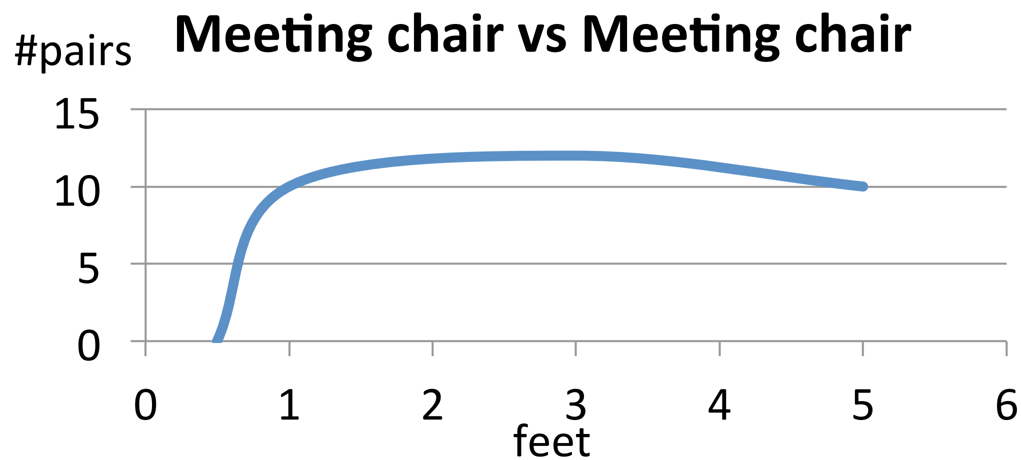
Challenges

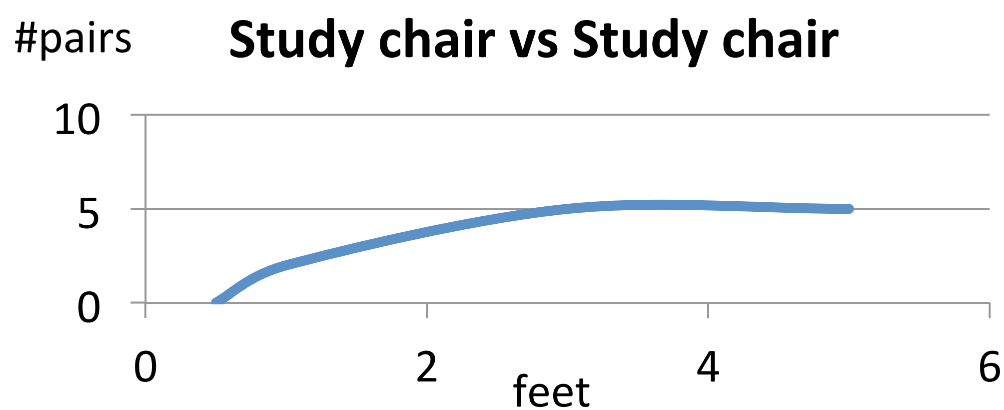
Contextual information



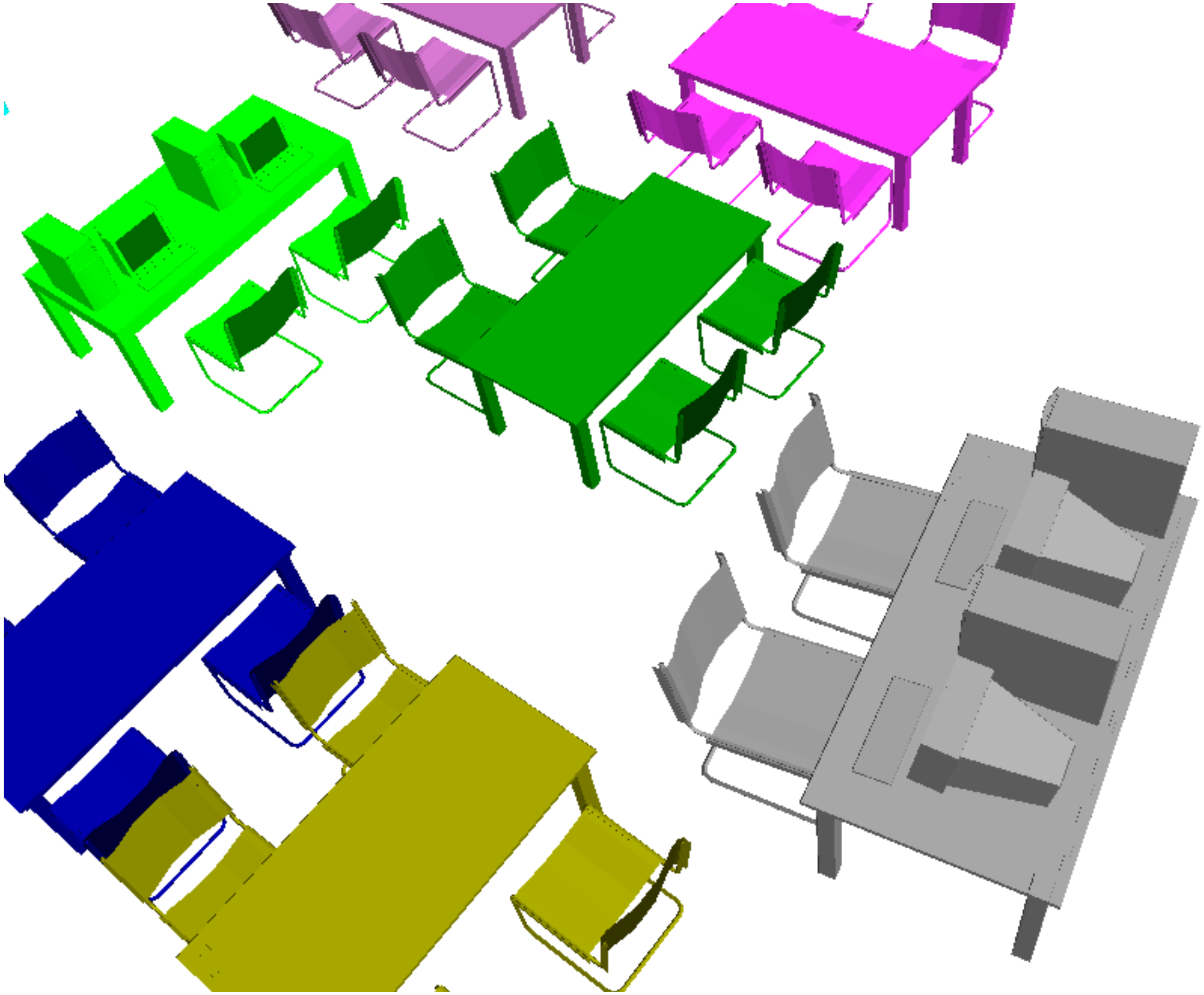
Challenges

All-pair contextual information is not distinctive.

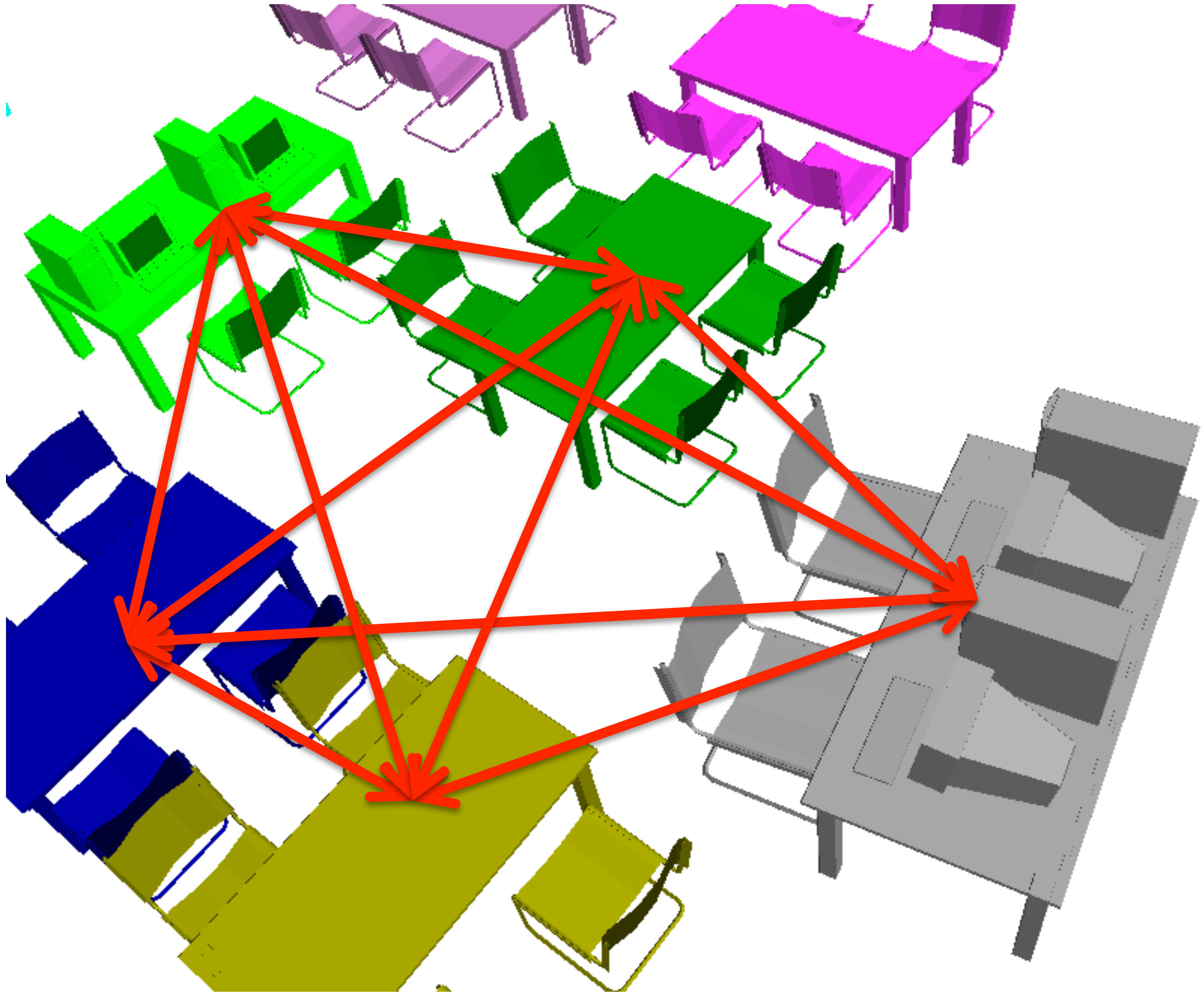




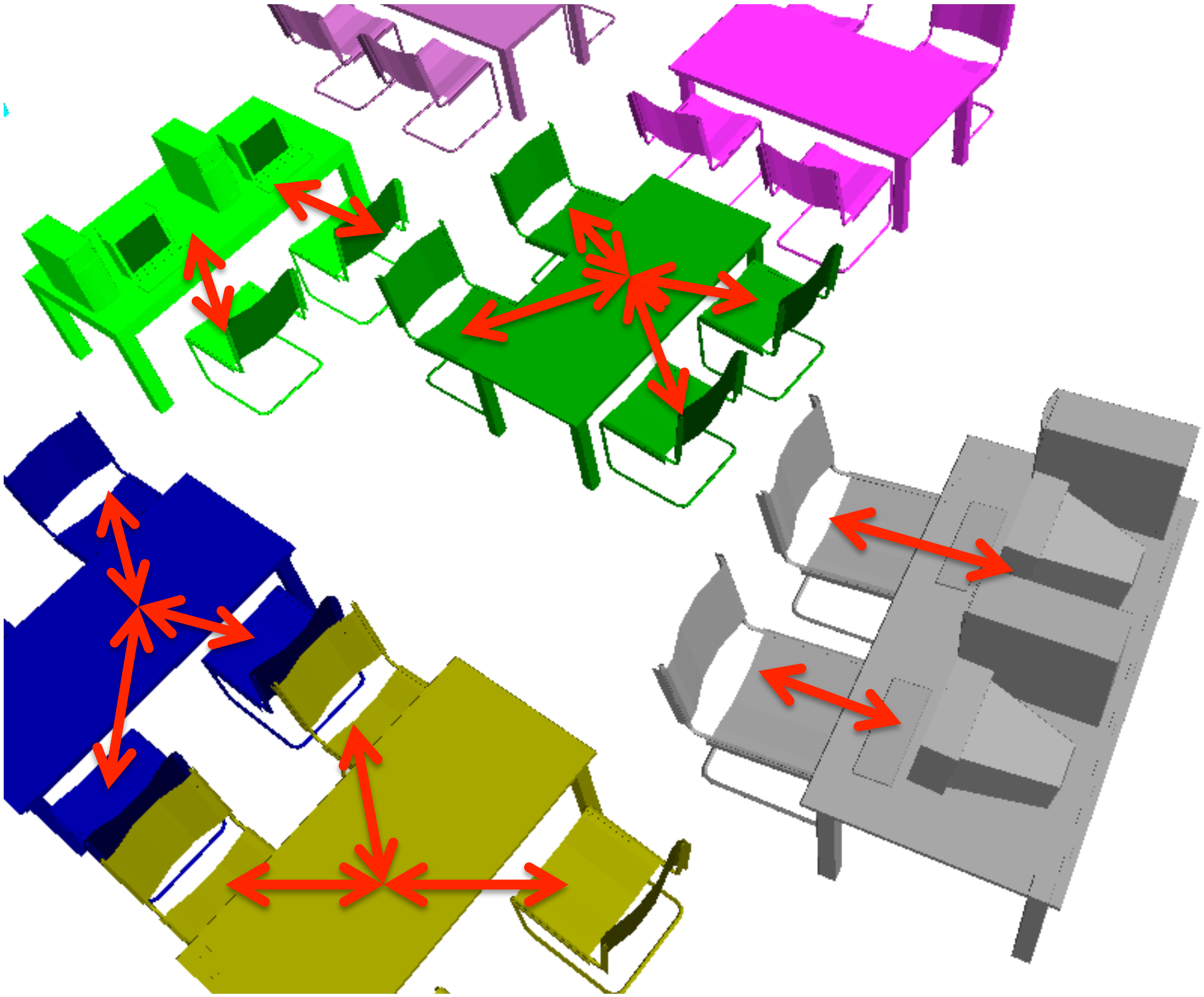
Challenges



Challenges

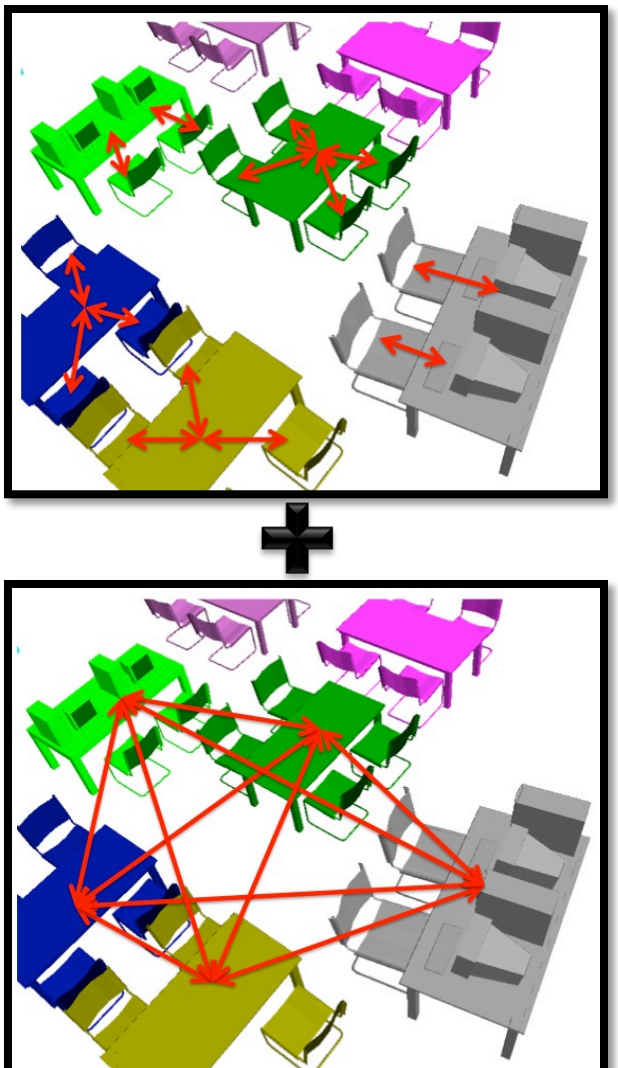


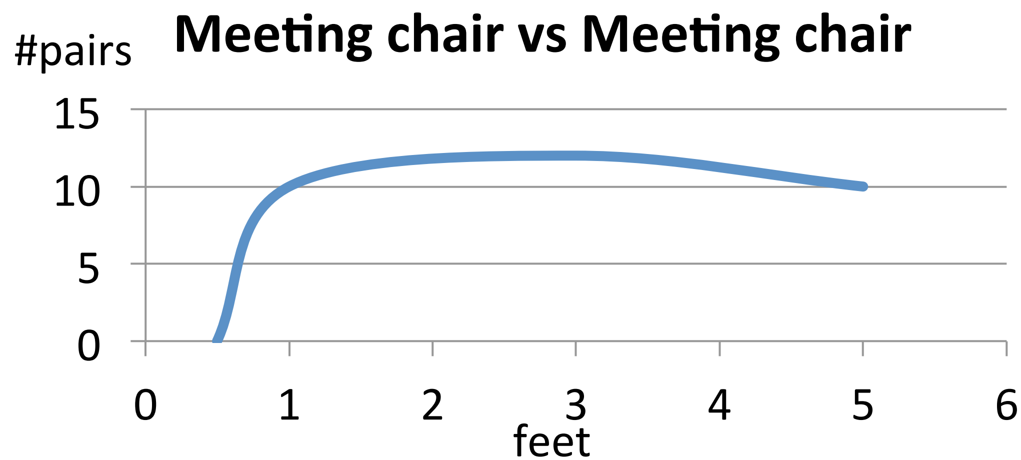
Challenges

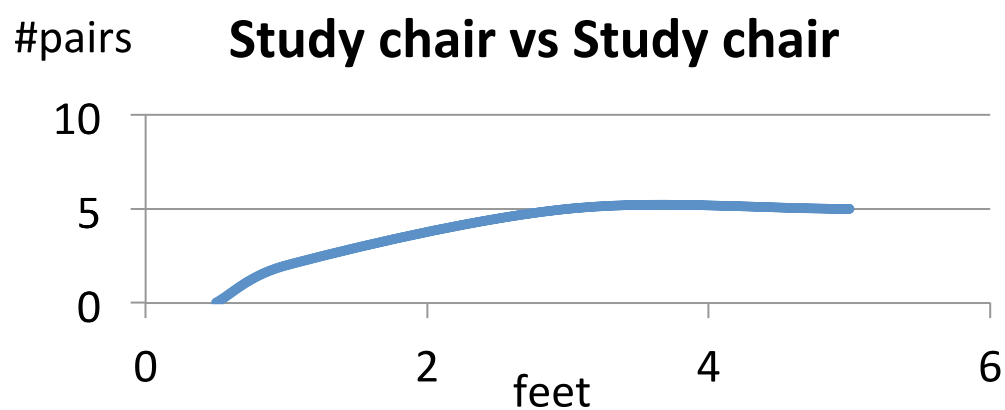


Challenges

All-pair contextual information is not distinctive.

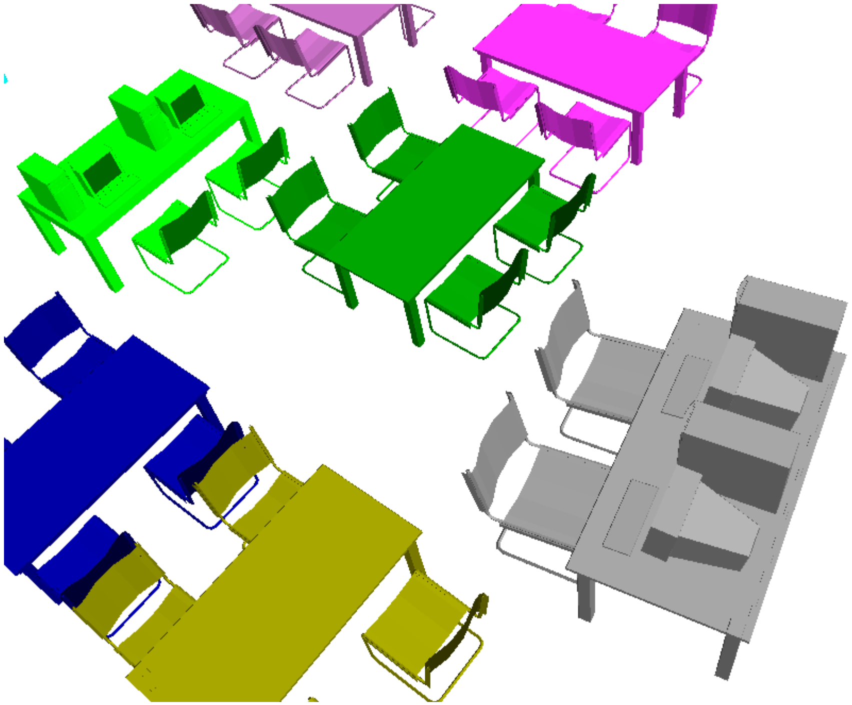




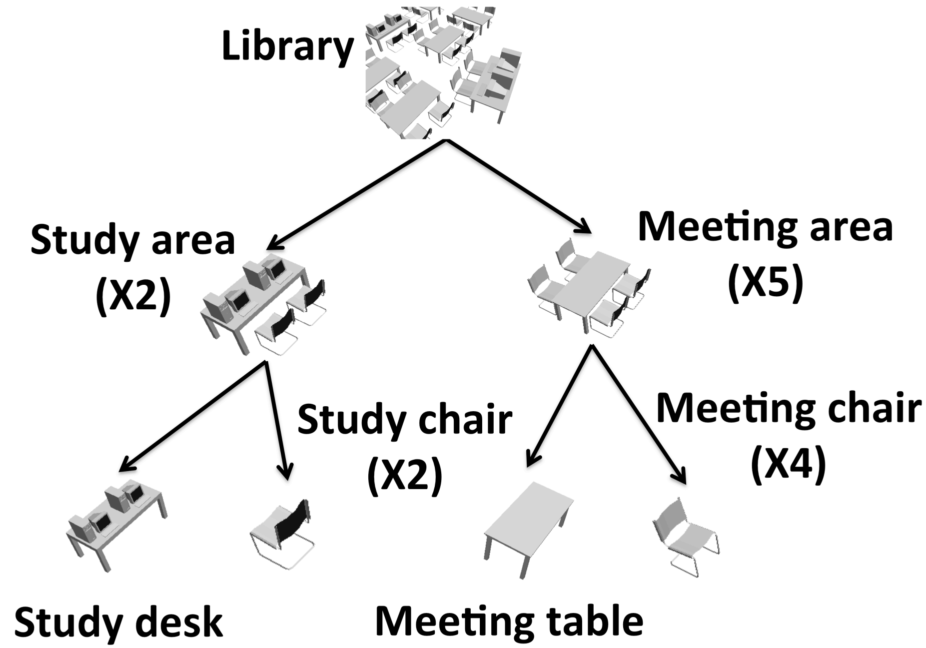


Key Idea

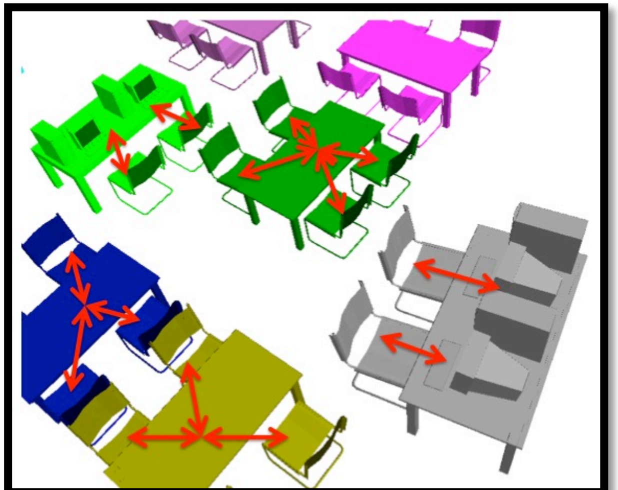
Semantic groups

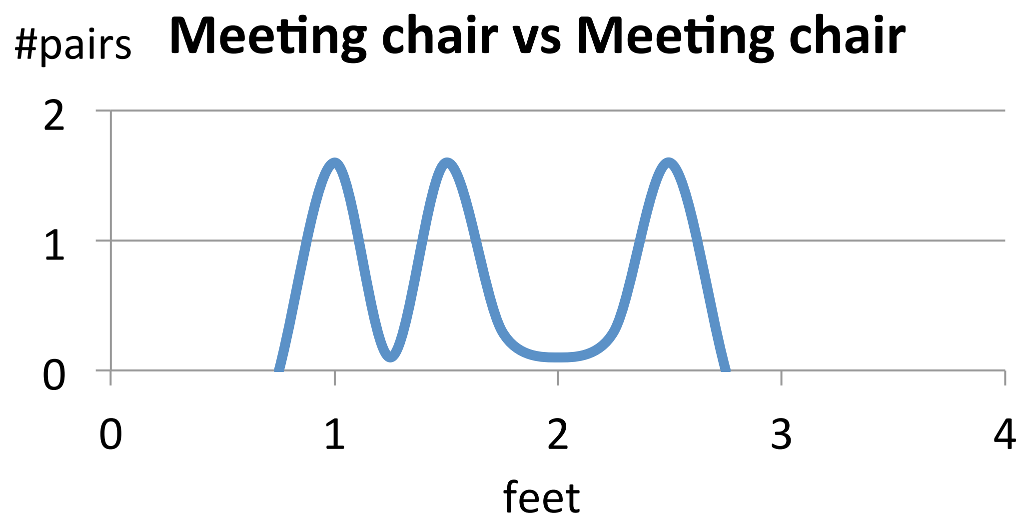


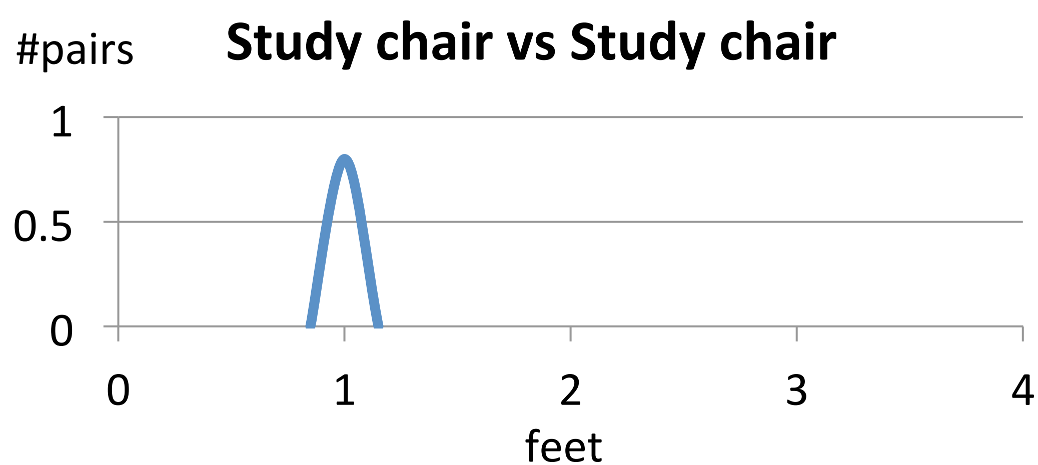
Semantic hierarchy



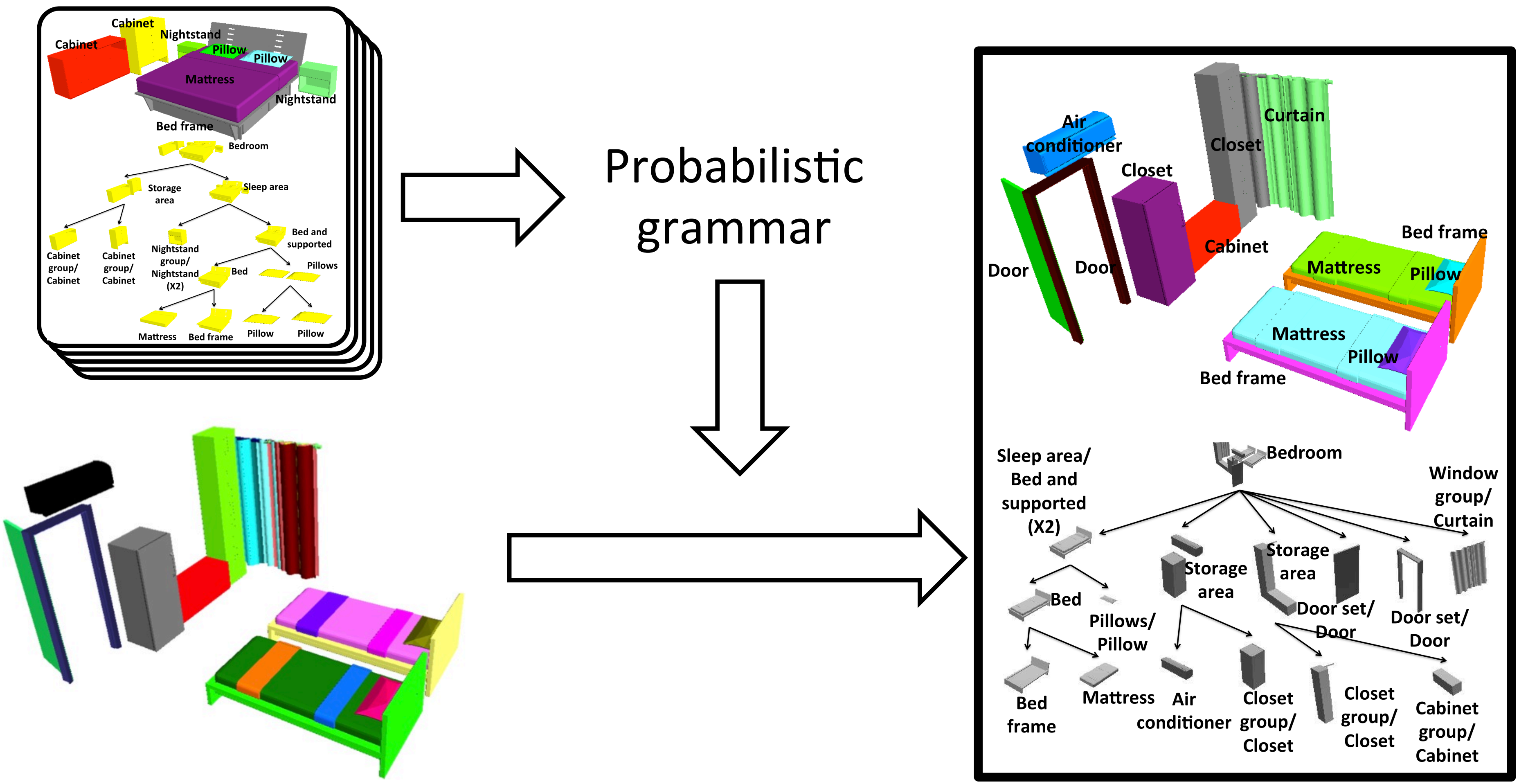
Key Idea




#pairs
Meeting chair vs Meeting chair
2
1
0
0
1
2
3
4
feet


#pairs
Study chair vs Study chair
1
0.5
0
0
1
2
3
4
feet

Pipeline

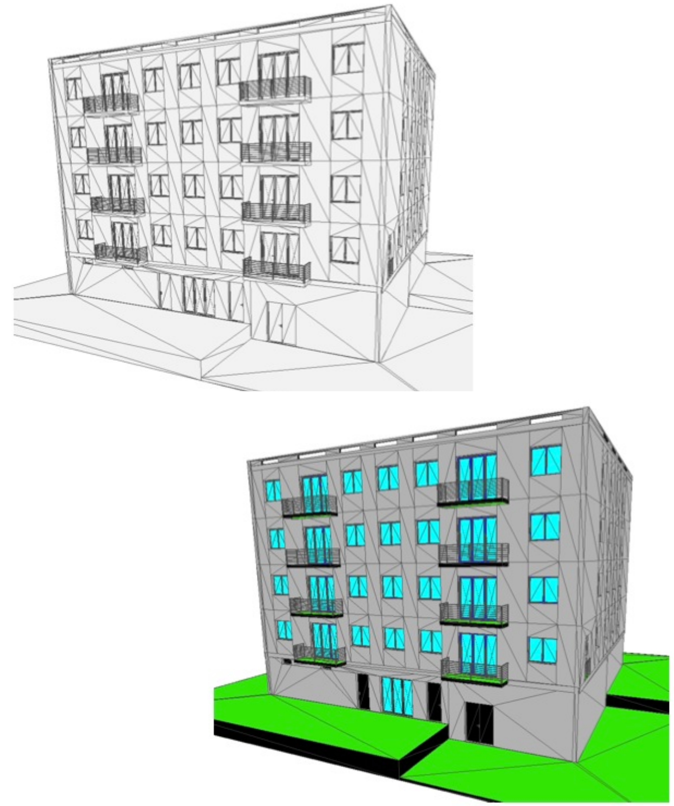

Cabinet
Cabinet
Nightstand
Pillow
Pillow
Mattress
Nightstand
Bed frame
Bedroom
Storage area
Sleep area
Cabinet group/ Cabinet
Cabinet group/ Cabinet
Nightstand group/ Nightstand (X2)
Bed and supported
Bed
Pillows
Mattress
Bed frame
Pillow
Pillow
Probabilistic grammar
Air conditioner
Closet
Curtain
Closet
Door
Door
Cabinet
Bed frame
Mattress
Pillow
Mattress
Pillow
Bed frame
Sleep area/ Bed and supported (X2)
Bedroom
Window group/ Curtain
Storage area
Storage area
Bed
Pillows/ Pillow
Door set/ Door
Door set/ Door
Bed frame
Mattress
Air conditioner
Closet group/ Closet
Closet group/ Closet
Cabinet group/ Cabinet

Related Work

Van Kaick et al. 2013

Related Work

Van Kaick et al. 2013



Boulch et al. 2013

Overview

→ Grammar Structure

Learning a Probabilistic Grammar

Scene Parsing

Results

Probabilistic Grammar

Labels

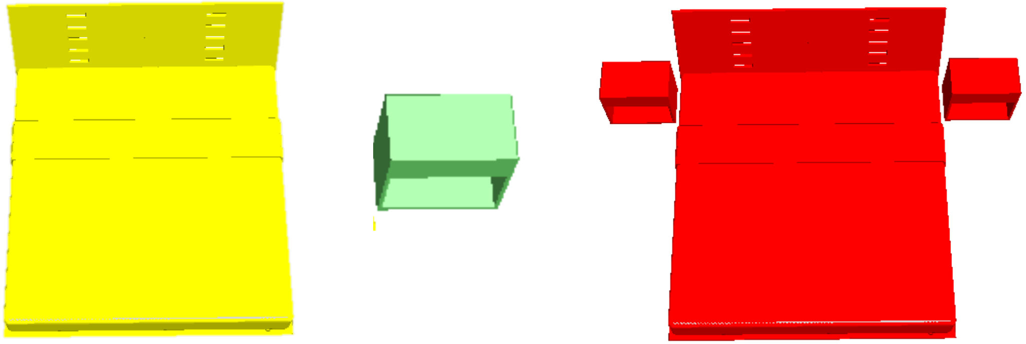
Rules

Probabilities

Labels

Examples:

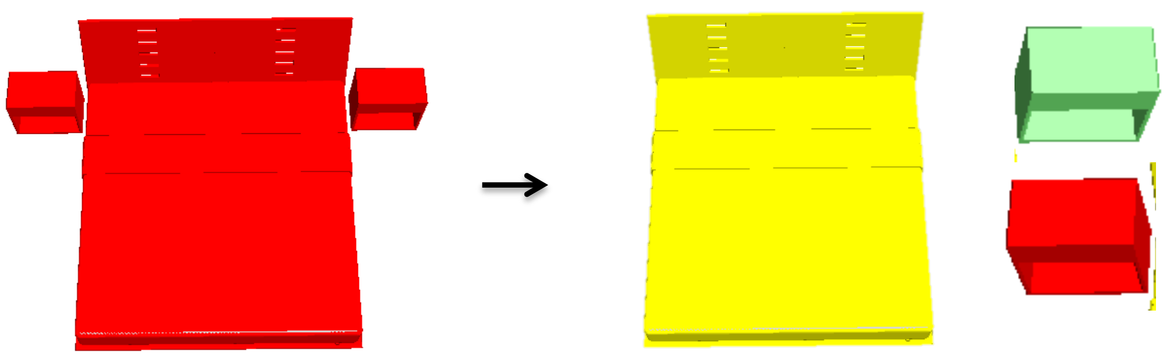
bed, night table, sleeping area



Rules

Example:

sleeping area → bed, night table



Probabilities

Derivation probabilities

Cardinality probabilities

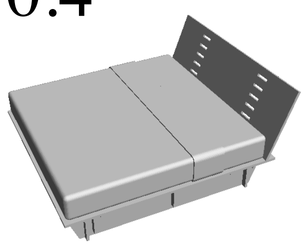
Geometry probabilities

Spatial probabilities

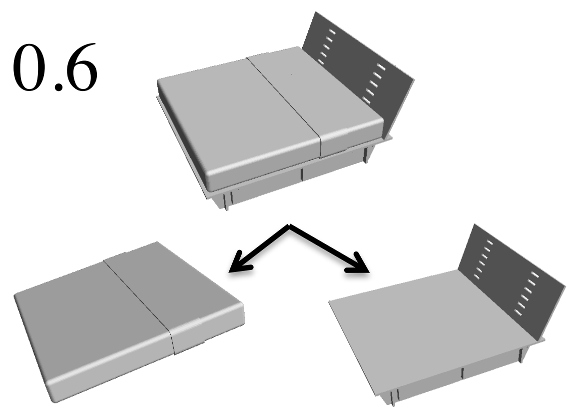
Derivation probability P_{nt}

bed $\xrightarrow{0.6}$ bed frame, mattress

$P = 0.4$

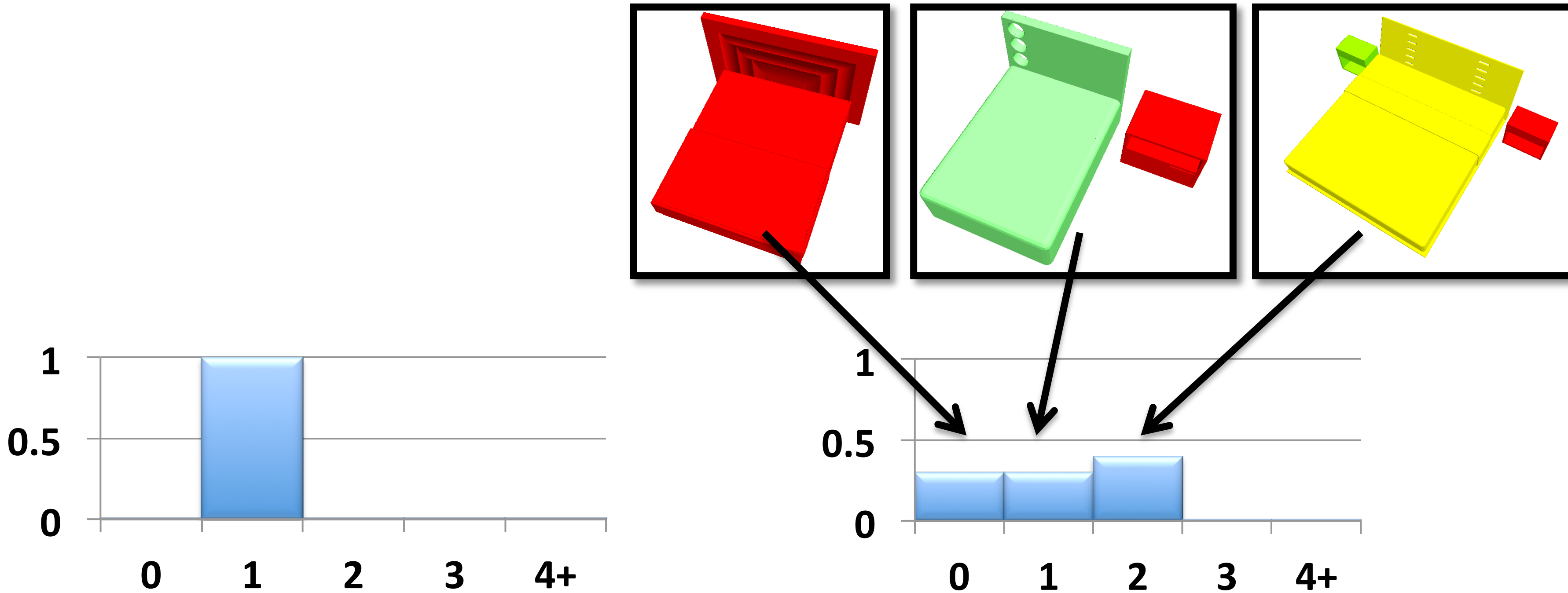


$P = 0.6$



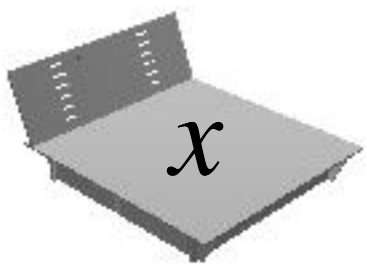
Cardinality probability P_{card}

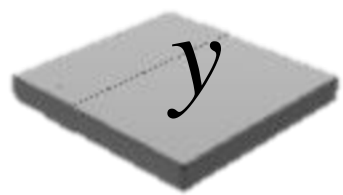
sleeping area → bed, night table



$P_{card}(bed \mid sleepingarea)$ $P_{card}(nighttable \mid sleepingarea)$

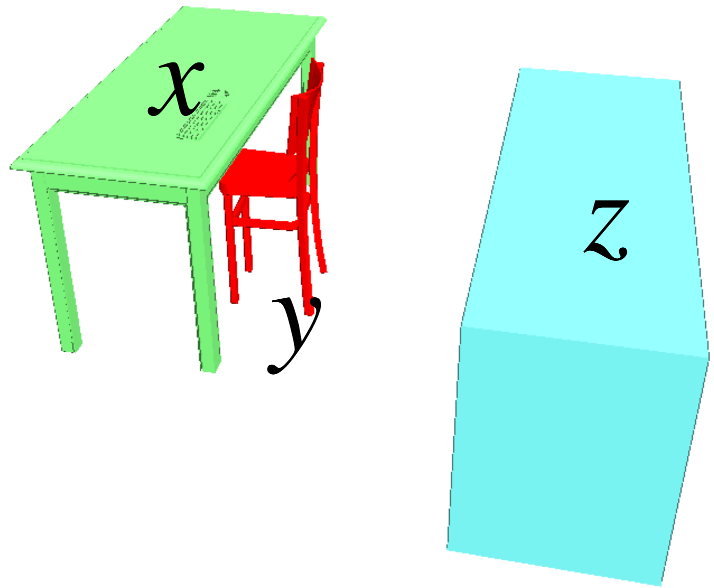
Geometry probability P_g





$$P_g(x \mid bedframe) > P_g(y \mid bedframe)$$

Spatial probability P_s



$$P_s(x, y \mid desk, chair, studyarea) > P_s(z, y \mid desk, chair, studyarea)$$

Overview

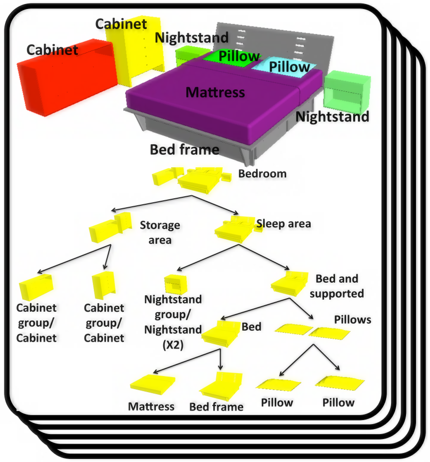
Grammar Structure

→ Learning a Probabilistic Grammar

Scene Parsing

Results

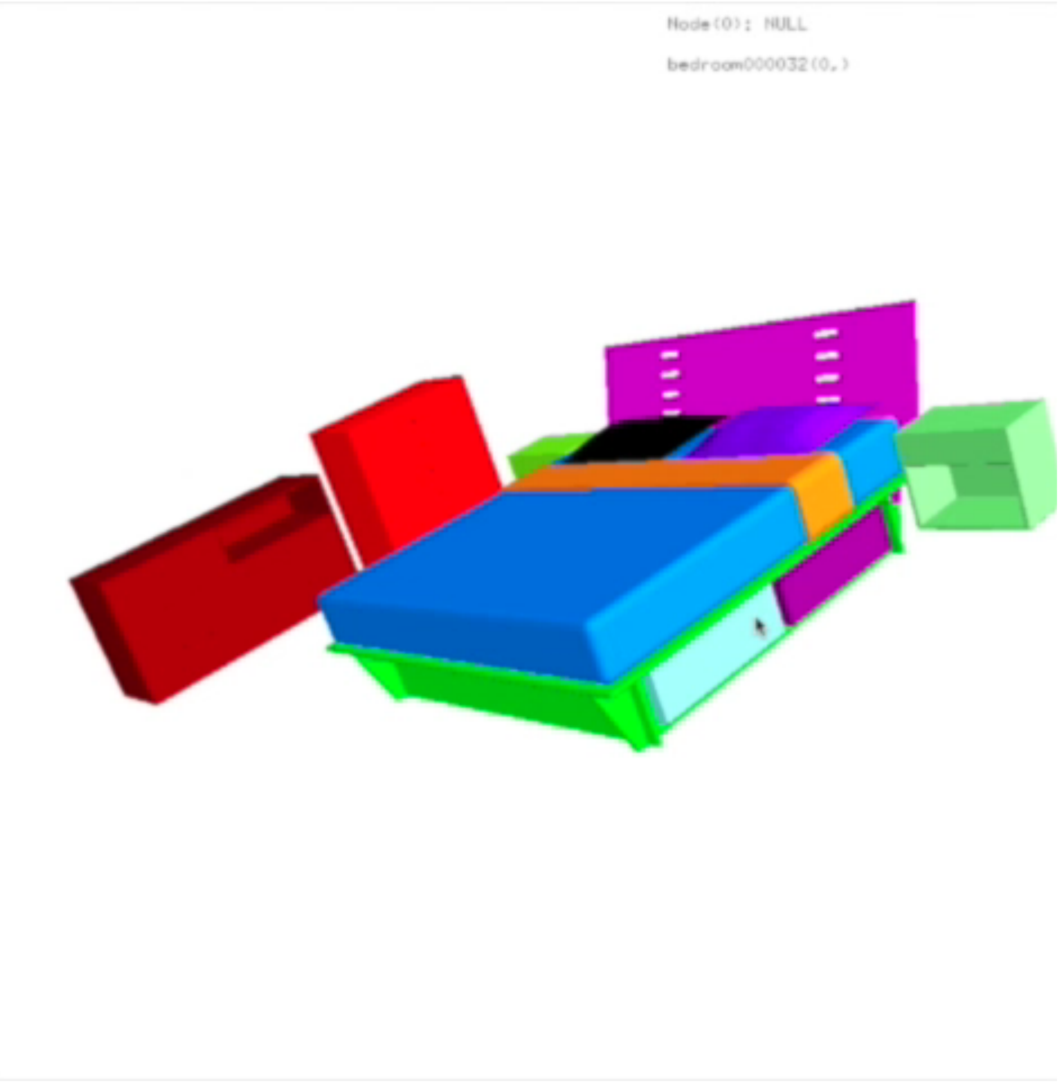
Pipeline



Probabilistic grammar

Learning a Probabilistic Grammar

Identify objects



Speed X 5

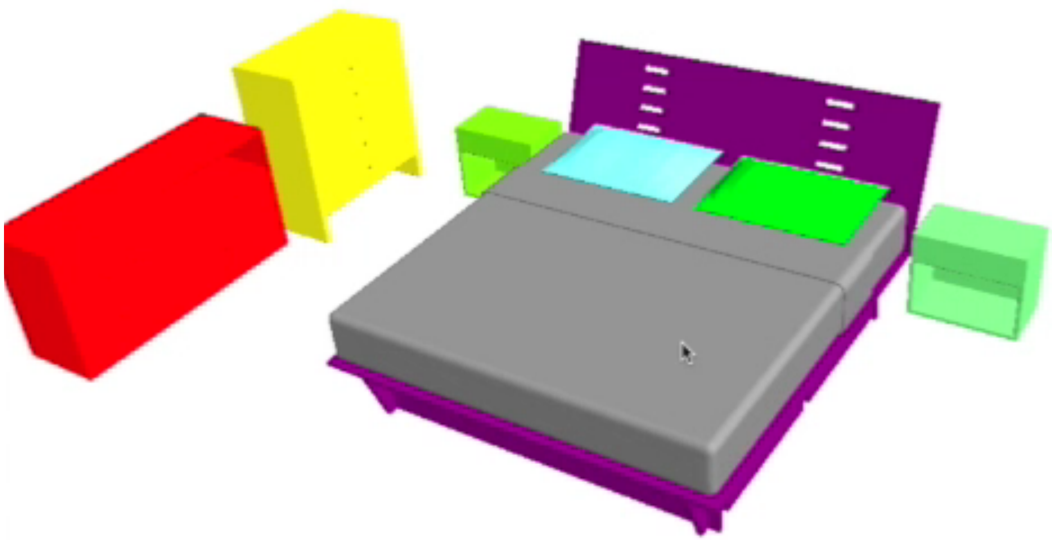
Learning a Probabilistic Grammar

Label objects

Speed X 5

Learning a Probabilistic Grammar

Group objects



Speed X 5

Learning a Probabilistic Grammar

Grammar generation

→ Labels all unique labels

Rules

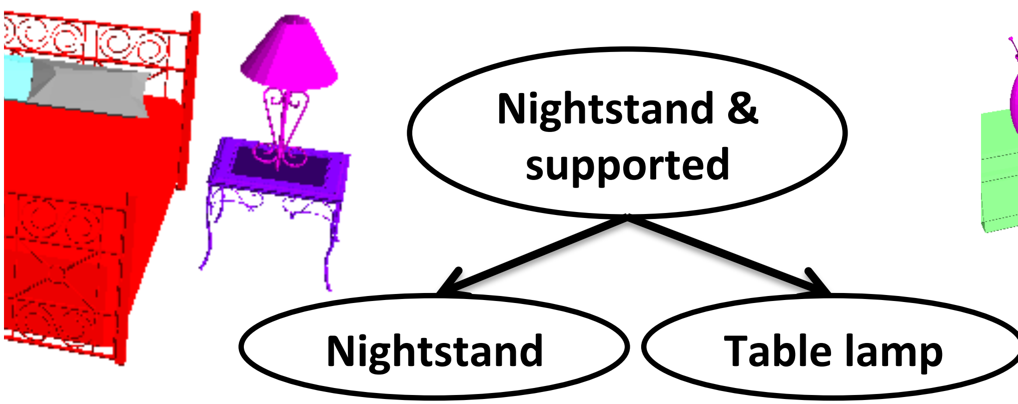
Probabilities

Learning a Probabilistic Grammar

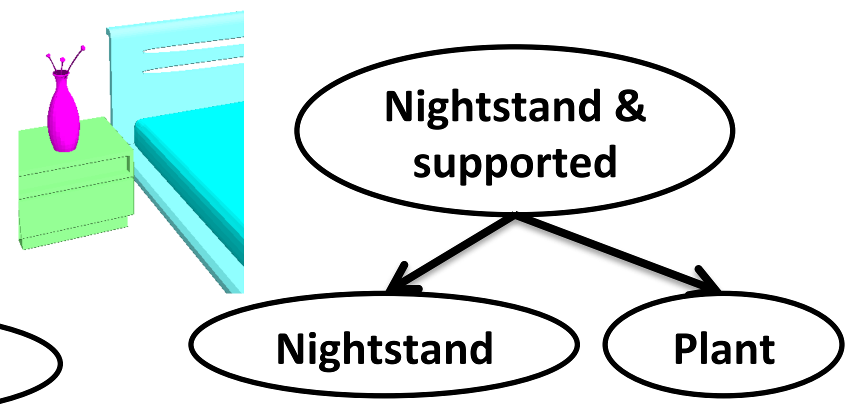
Grammar generation

- Labels
- → Rules — concatenating all children for each label
- Probabilities

Training example 1:



Training example 2:



Learning a Probabilistic Grammar

Grammar generation

- Labels
- Rules
- → Probabilities

P_{nt}, P_{card} : learning from occurrence statistics

P_g : estimating Gaussian kernels

P_s : kernel density estimation

Overview

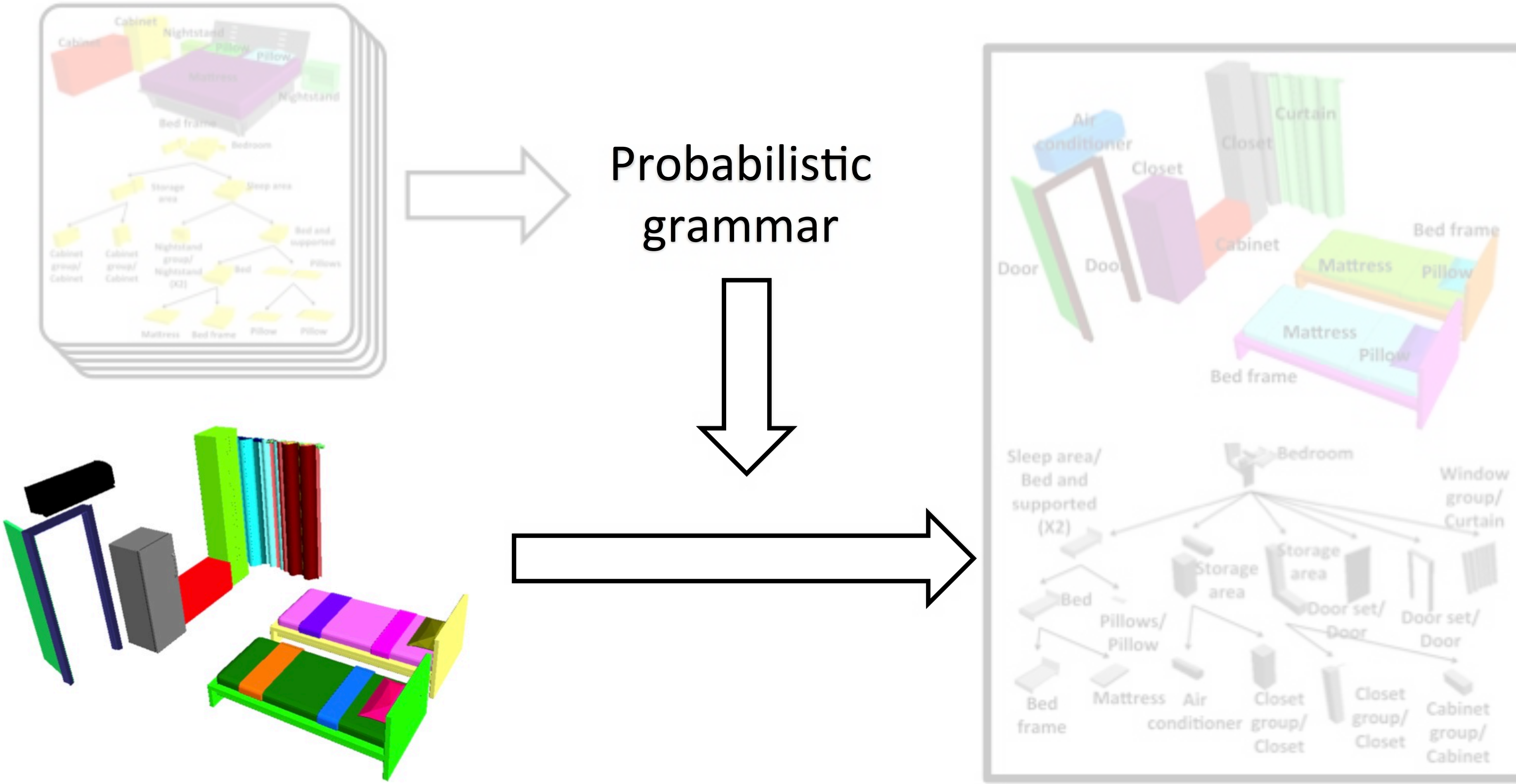
Grammar Structure

Learning a Probabilistic Grammar

→ Scene Parsing

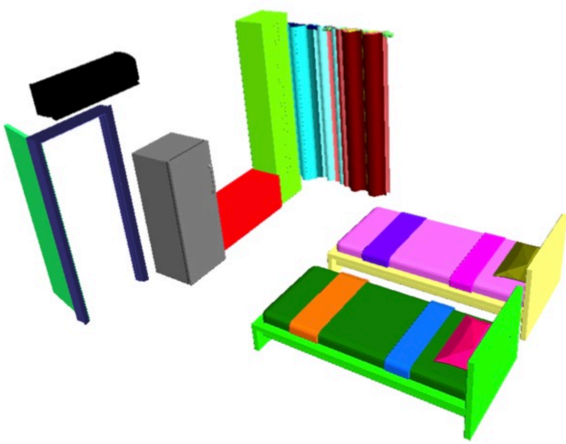
Results

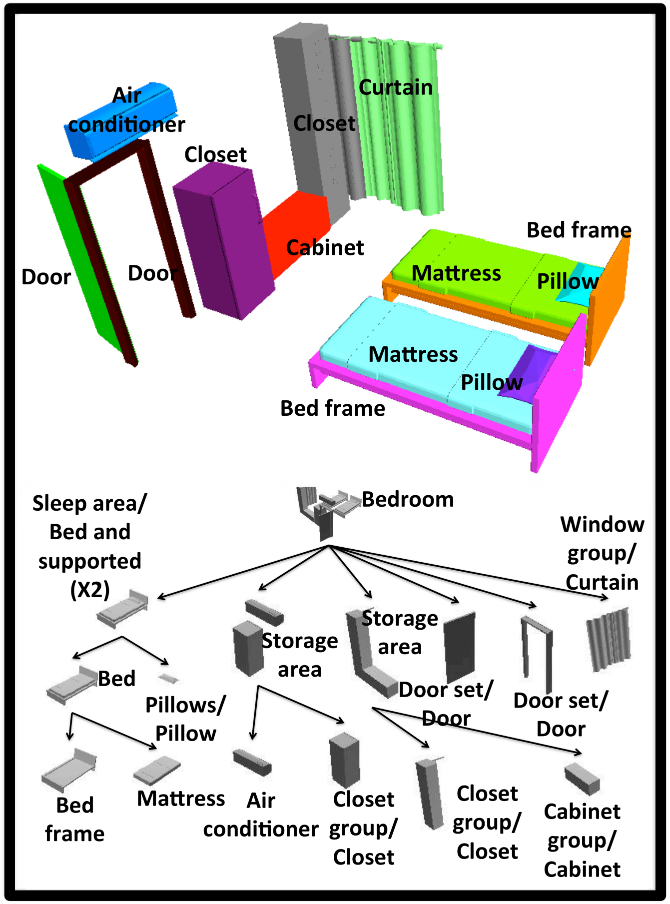
Pipeline


Probabilistic
grammar
Air conditioner
Closet
Curtain
Door
Cabinet
Bed frame
Mattress
Pillow
Bedroom
Sleep area/
Bed and
supported
(X2)
Window
group/
Curtain
Storage
area
Bed
Pillows/
Pillow
Door set/
Door
Bed
frame
Mattress
Air
conditioner
Closet
group/
Closet
Cabinet
group/
Cabinet

Pipeline

Probabilistic grammar





Scene parsing

Objective function

$$H^* = \text{argmax}_H \, P(H \mid S, G)$$

- H is the unknown hierarchy
- S is the input scene
- G is the probabilistic grammar

Scene parsing

After applying Bayes' rule

$$H^* = \text{argmax}_H P(H \mid G) P(S \mid H, G)$$

Scene parsing

After applying Bayes' rule

$$H^* = \text{argmax}_H P(H \mid G) P(S \mid H, G)$$

Prior of hierarchy $$P(H \mid G) = \prod_{x \in H} P_{prod}(x)^{T(x)}$$

Scene parsing

After applying Bayes' rule

$$H^* = \text{argmax}_H \, P(H \mid G) P(S \mid H, G)$$

Prior of hierarchy $$P(H \mid G) = \prod_{x \in H} P_{prod}(x)^{T(x)}$$

$P_{prod}(x)$: probability of a single derivation

formulated using P_{nt}, P_{card}

Scene parsing

After applying Bayes' rule

$$H^* = \text{argmax}_H \, P(H \mid G) P(S \mid H, G)$$

Prior of hierarchy $$P(H \mid G) = \prod_{x \in H} P_{prod}(x)^{T(x)}$$

$T(x)$ compensates for decreasing probability as H has more internal nodes.

Scene parsing

After applying Bayes' rule

$$H^* = \operatorname{argmax}_H P(H \mid G) P(S \mid H, G)$$

Likelihood of scene

$$P(S \mid H, G) = \prod_{x \in H} P_g(x)^{T(x)} P_s^*(x)^{T(x)}$$

Scene parsing

After applying Bayes' rule

$$H^* = \text{argmax}_H P(H \mid G) P(S \mid H, G)$$

Likelihood of scene

$$P(S \mid H, G) = \prod_{x \in H} P_g(x)^{T(x)} P_s^*(x)^{T(x)}$$

$P_g(x)$: geometry probability

Scene parsing

After applying Bayes' rule

$$H^* = \text{argmax}_H P(H \mid G) P(S \mid H, G)$$

Likelihood of scene

$$P(S \mid H, G) = \prod_{x \in H} P_g(x)^{T(x)} P_s^*(x)^{T(x)}$$

$P_s^*(x)$: sum of all pairwise spatial probabilities $P_s(x)$

Scene parsing

We work in the negative logarithm space

$$E(H) = \log P(H \mid G) P(S \mid H, G)$$

$$= -\sum_{x \in H} T(x) \log\left(P_{prod}(x) P_g(x) P_s^*(x)\right)$$

Scene parsing

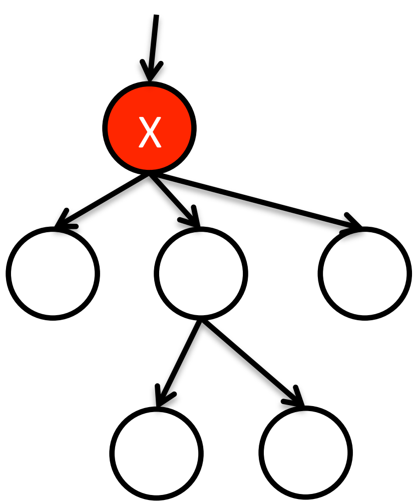
Rewrite the objective function recursively

$$E(H) = \overline{E}(R)$$

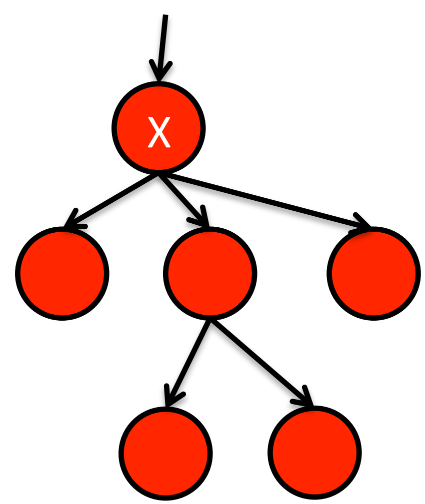
$$\overline{E}(x) = E(x) + \sum_{y \in x.children} \overline{E}(y)$$

where R is the root of H, $\overline{E}$ is the energy of a sub-tree.

$E(x)$



$\overline{E}(x)$



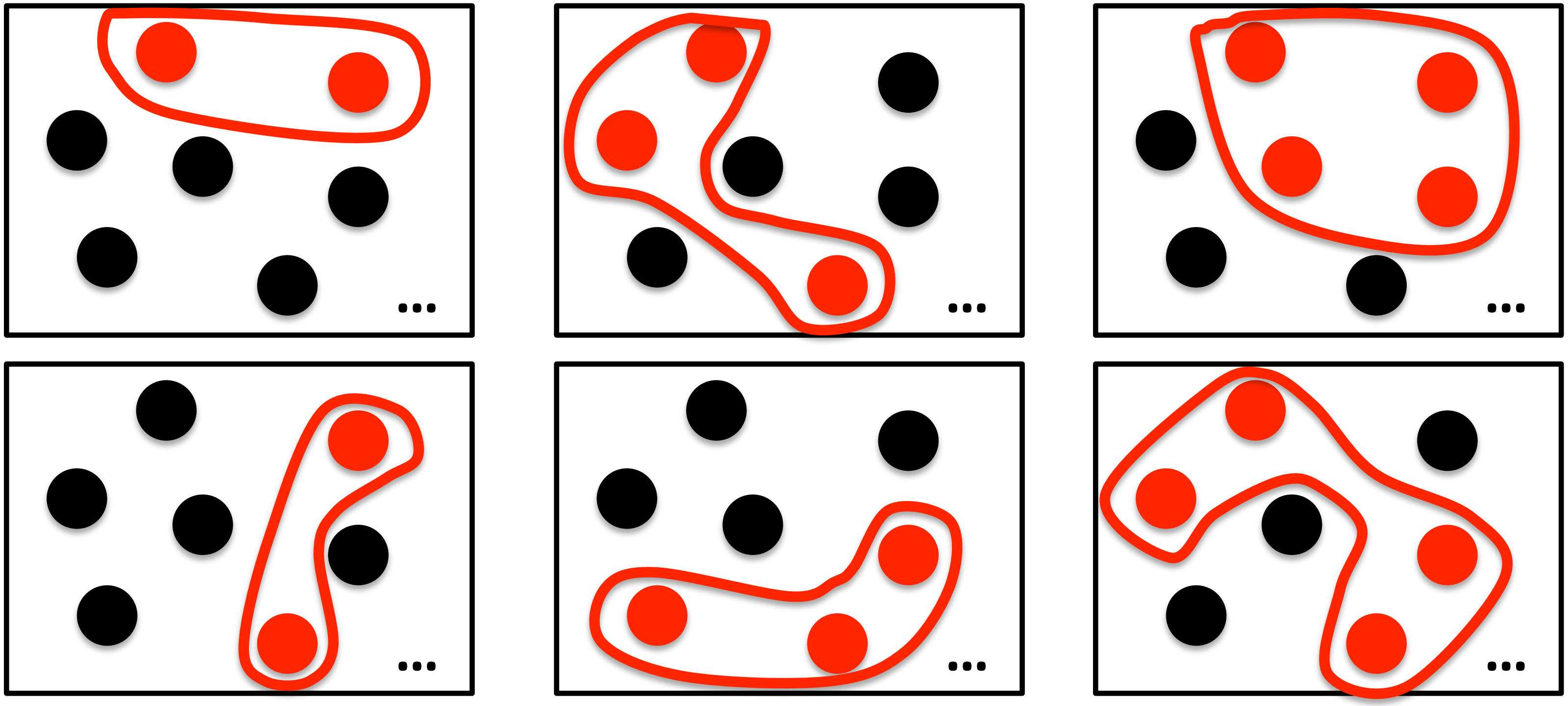
Scene parsing

The search space is prohibitively large …

- Problem 1: #possible groups is exponential.
- Problem 2: #label assignments is exponential.

Scene parsing

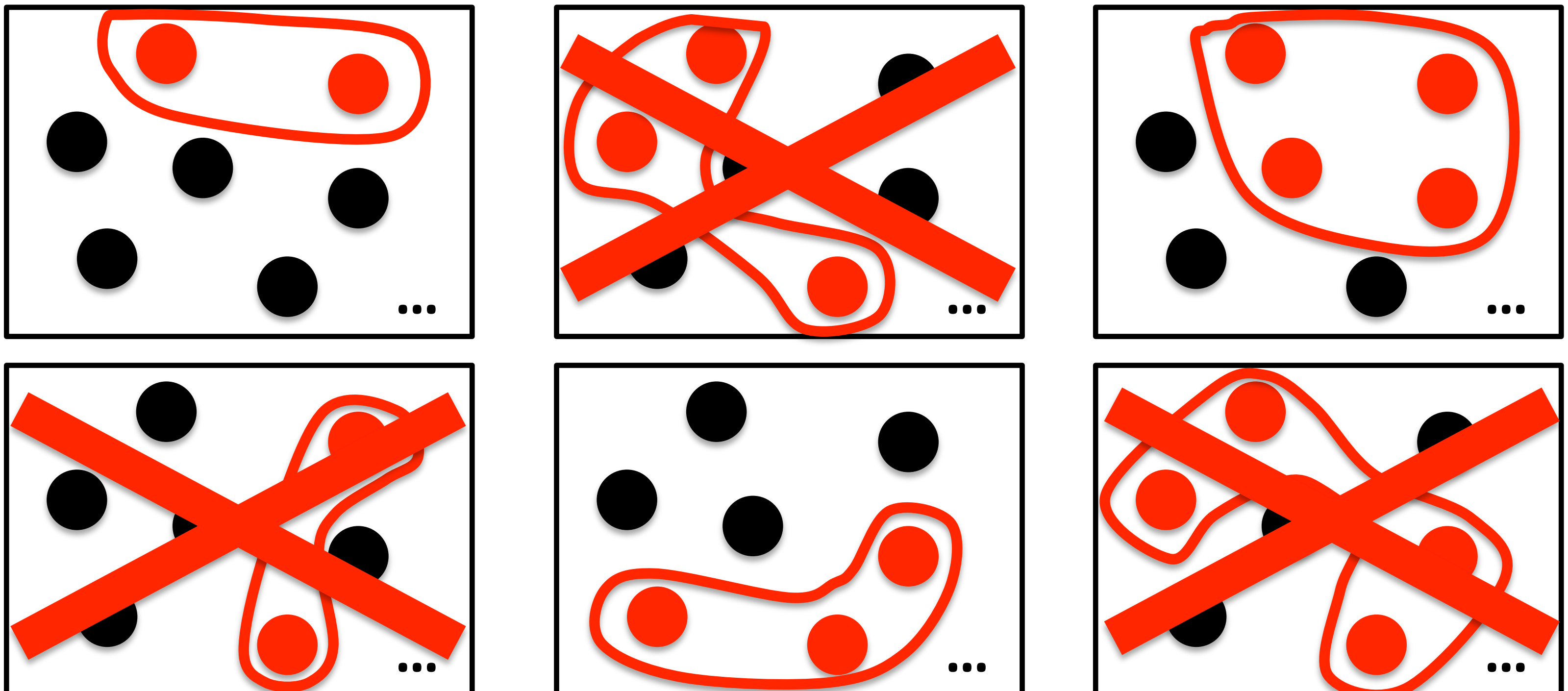
Problem 1: #possible groups is exponential.



Scene parsing

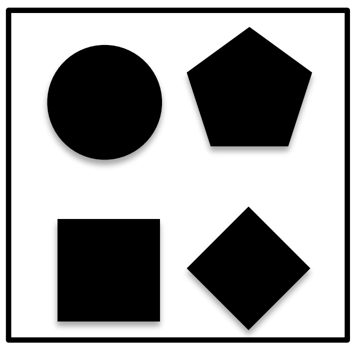
Problem 1: #possible groups is exponential.

Solution: proposing candidate groups.

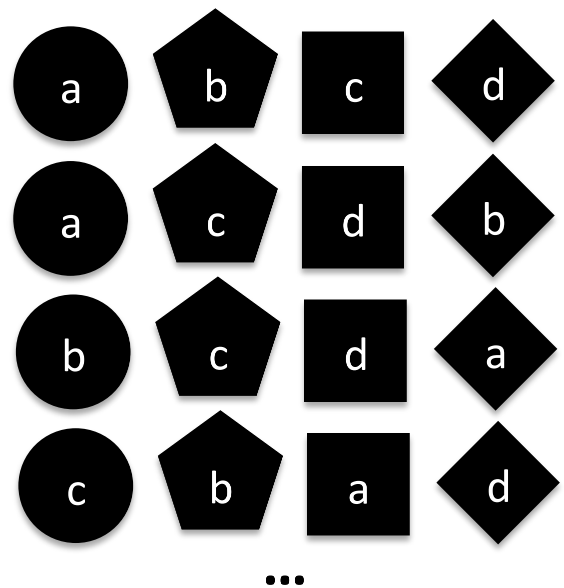


Scene parsing

Problem 2: #label assignments is exponential.



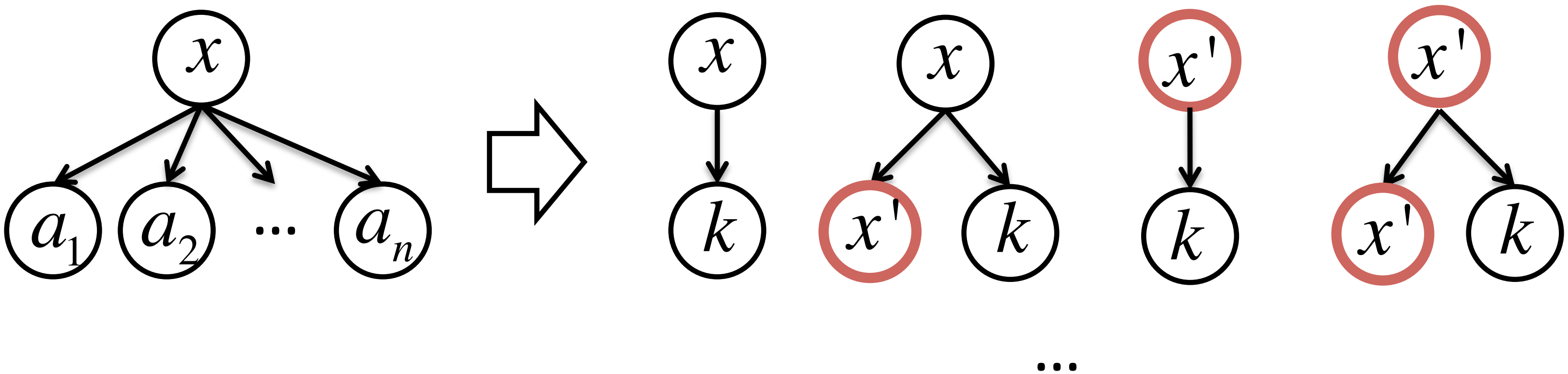
Rule: $r \rightarrow a, b, c, d$



Scene parsing

Problem 2: #label assignments is exponential.

Solution: bounding #RHS by grammar binarization

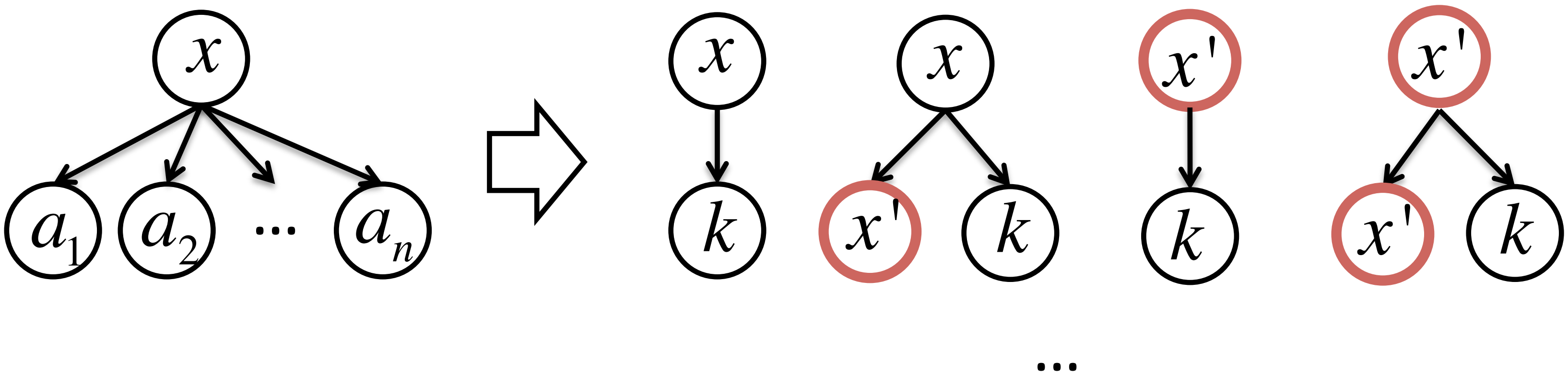


where x' is partial label of x, $k \in \{a_1, a_2, ..., a_n\}$

Scene parsing

Problem 2: #label assignments is exponential.

Solution: bounding #RHS by grammar binarization



where x' is partial label of x, $k \in \{a_1, a_2, \ldots, a_n\}$

Now #rules and #assignments are both polynomial.

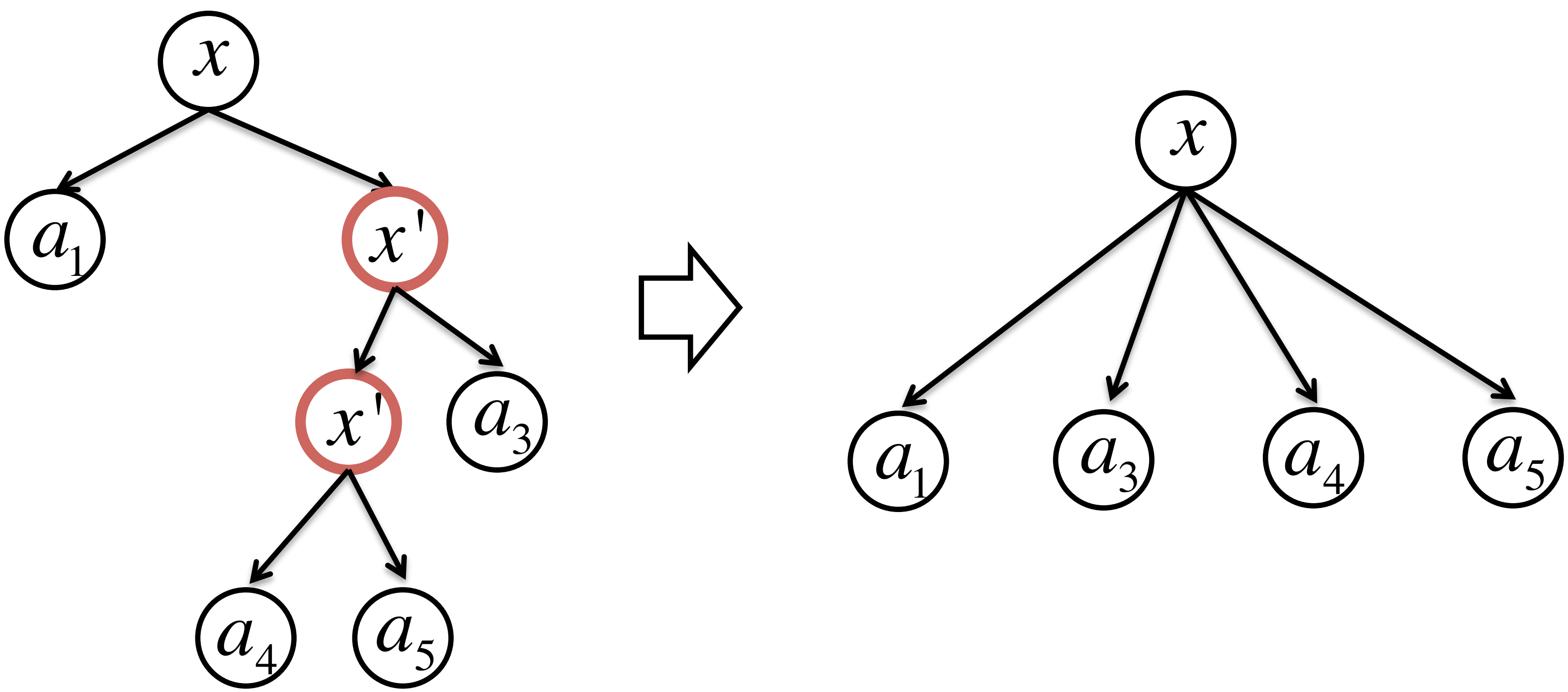
The problem can be solved by dynamic programming.

Scene parsing

Problem 2: #label assignments is exponential.

Solution: bounding #RHS by grammar binarization

Convert the result to a parse tree of the original grammar



Overview

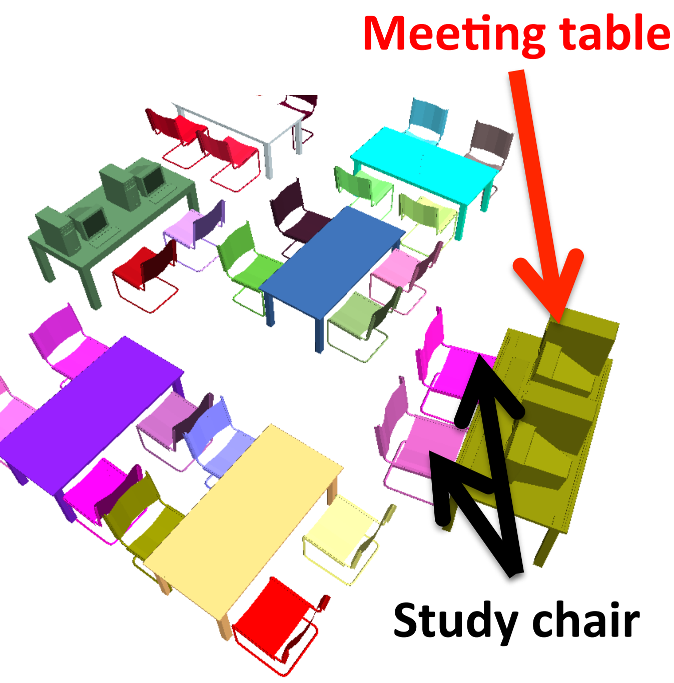
Grammar Structure

Learning a Probabilistic Grammar

Scene Parsing

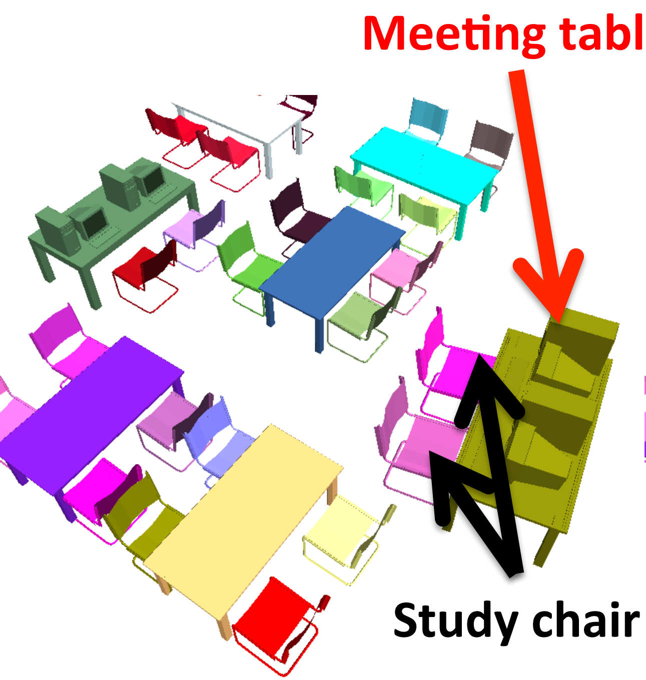
→ Results

Benefit of hierarchy

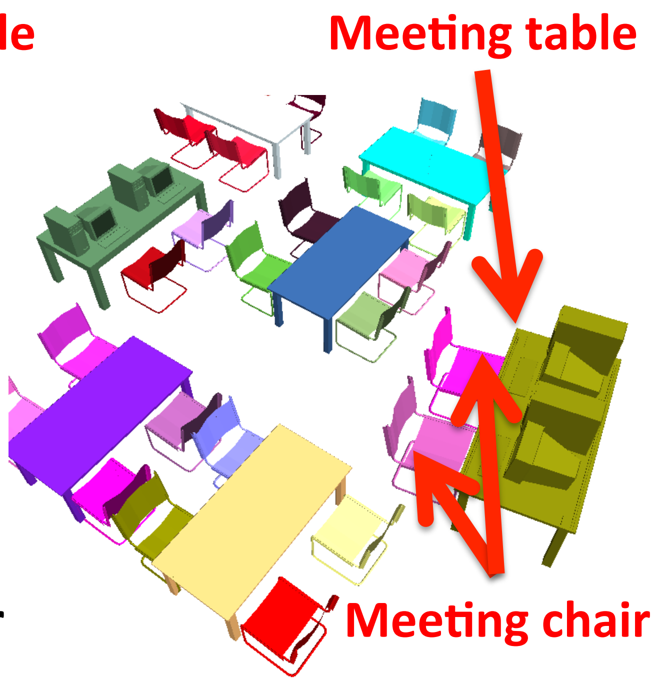


Shape only

Benefit of hierarchy

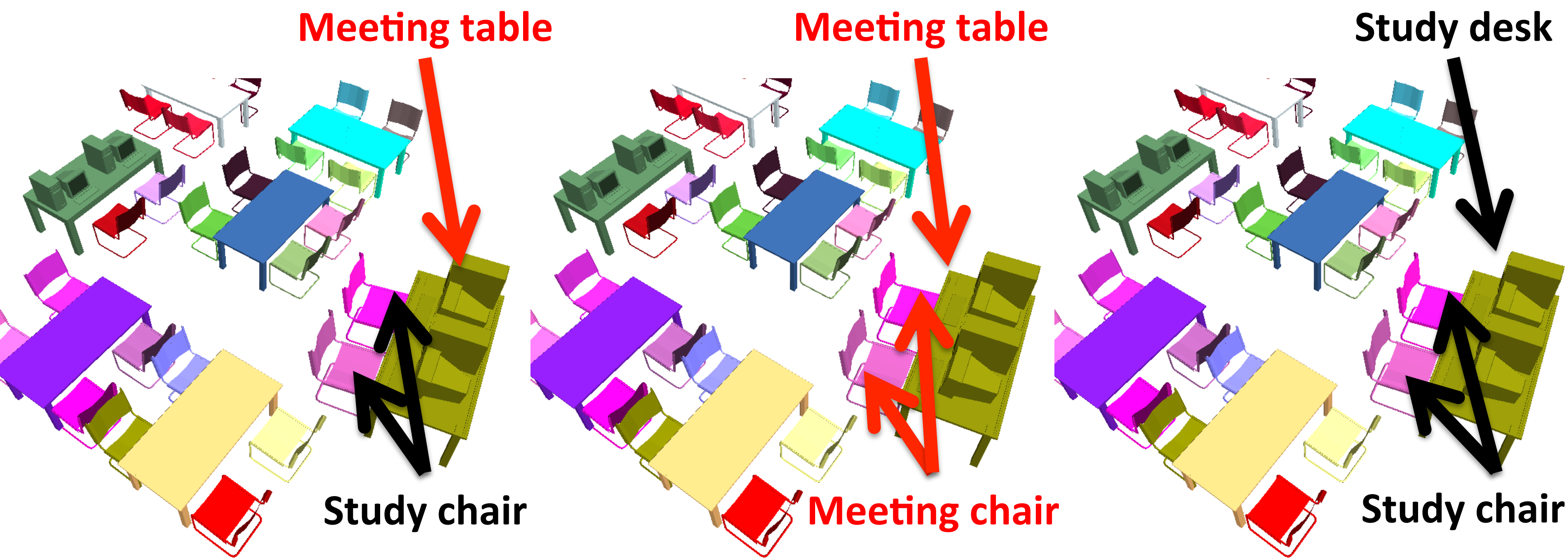


Shape only



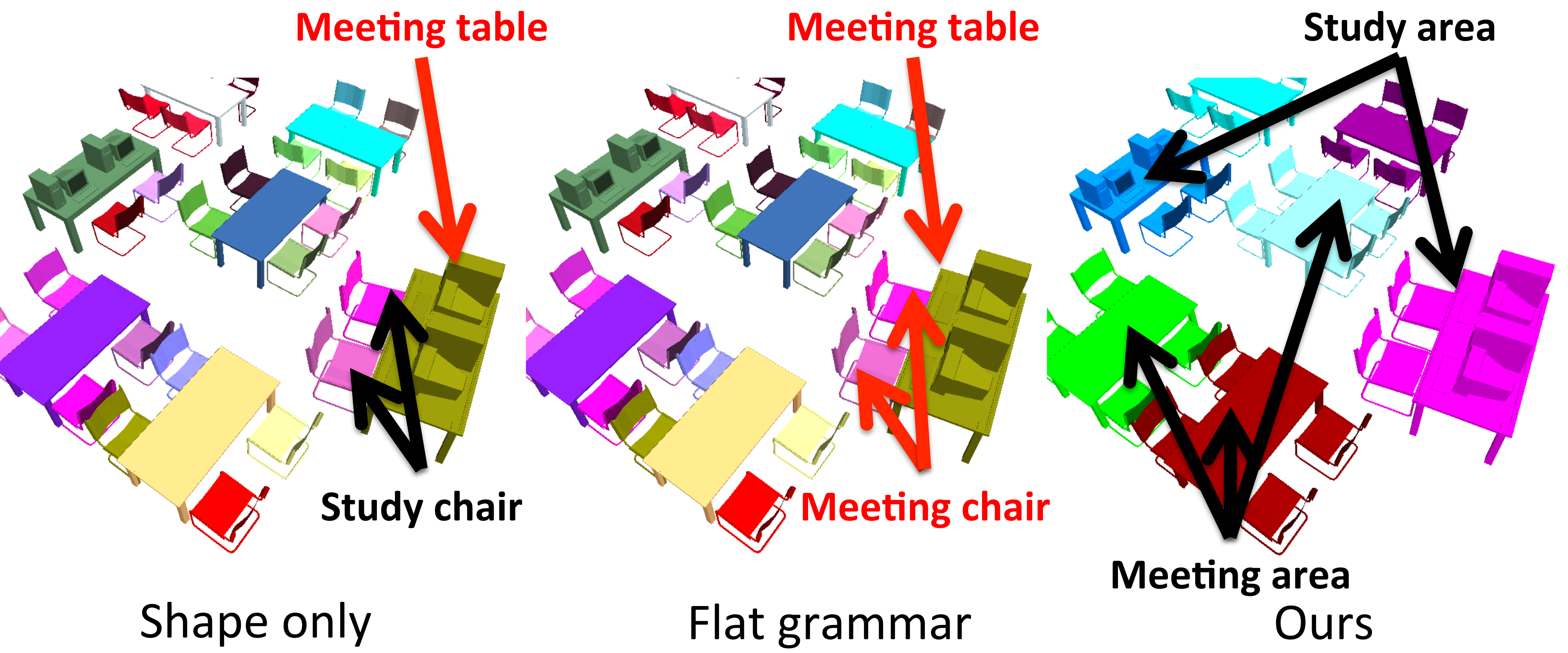
Flat grammar

Benefit of hierarchy

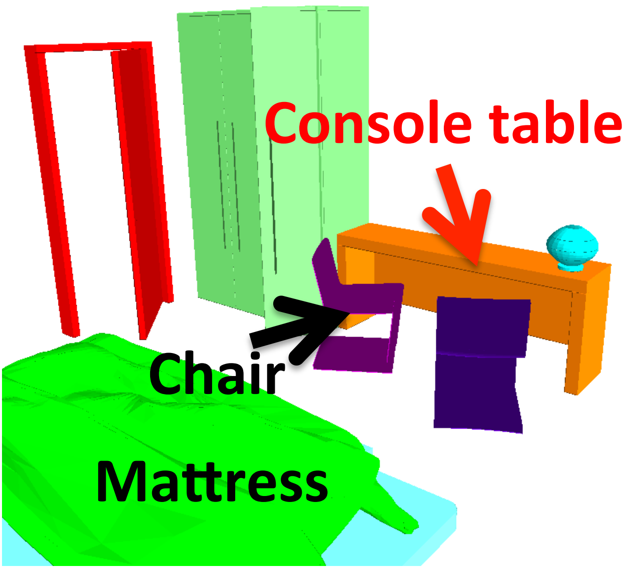


Shape only　　　Flat grammar　　　Ours

Benefit of hierarchy

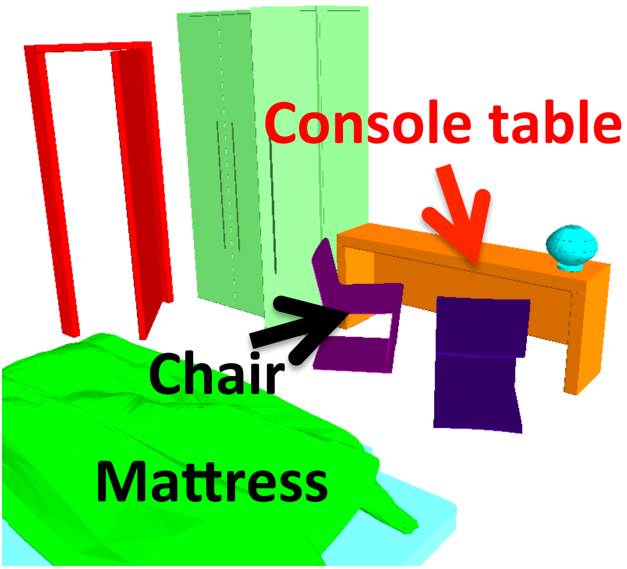


Benefit of hierarchy

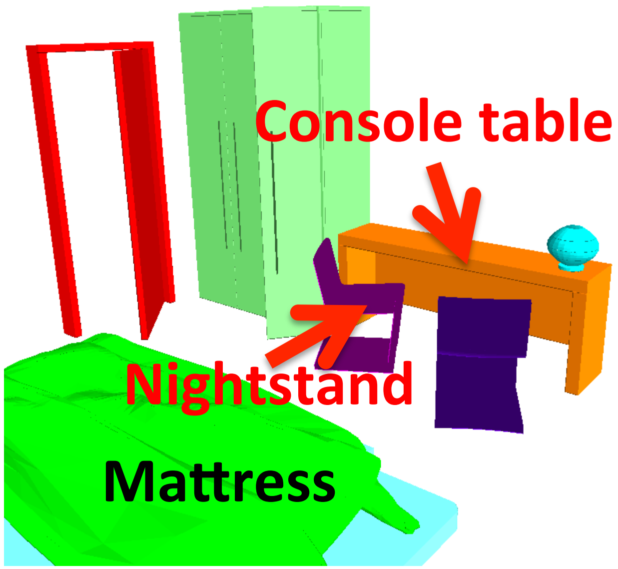


Shape only

Benefit of hierarchy

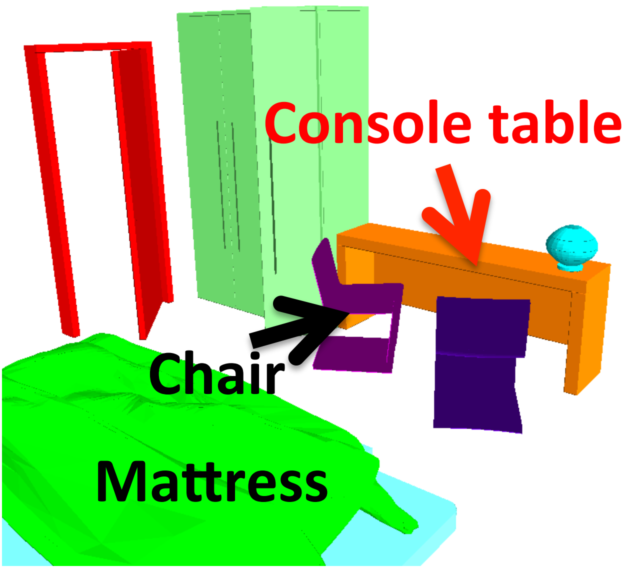


Shape only

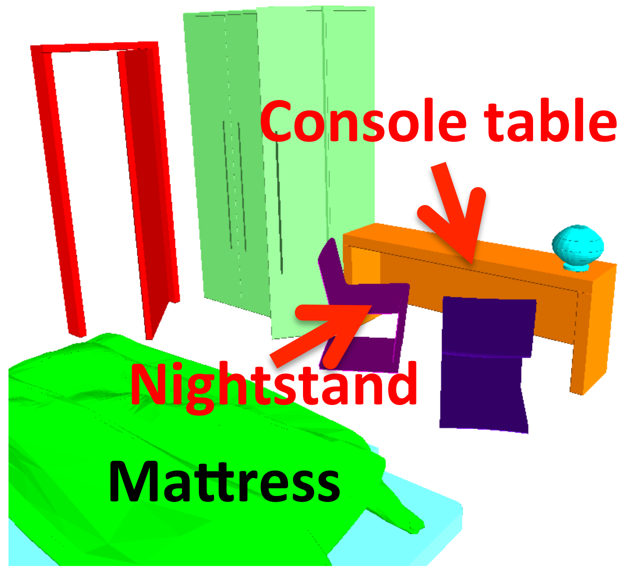


Flat grammar

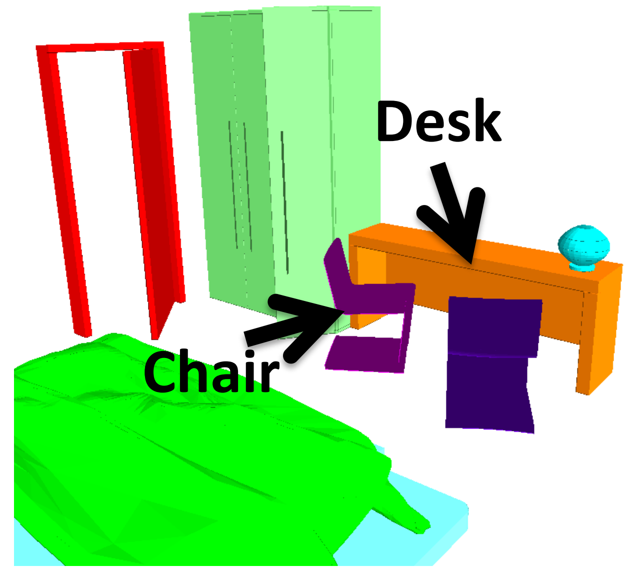
Benefit of hierarchy



Shape only

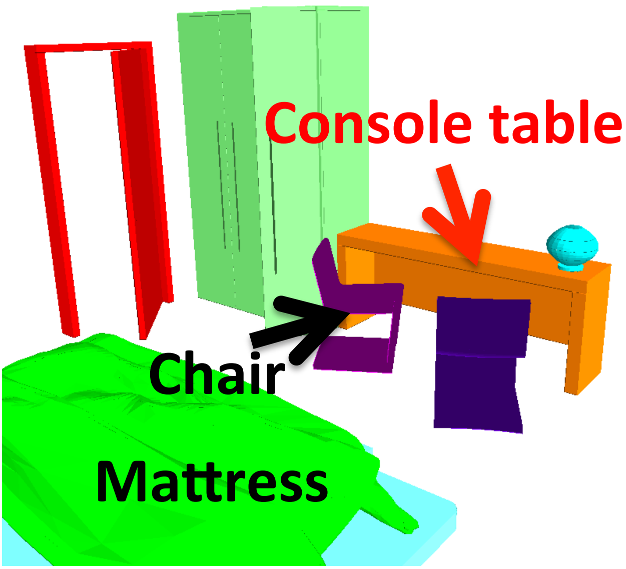


Flat grammar

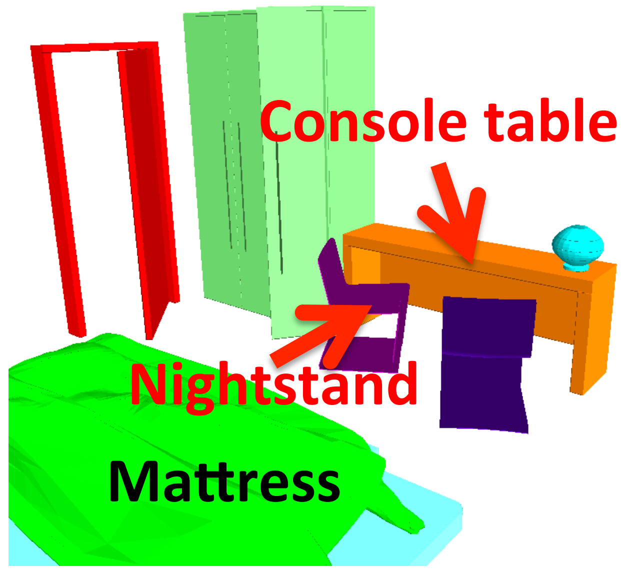


Ours

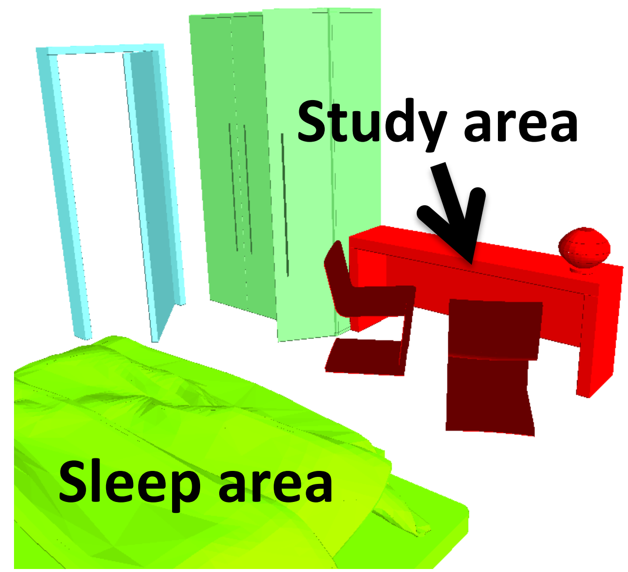
Benefit of hierarchy



Shape only

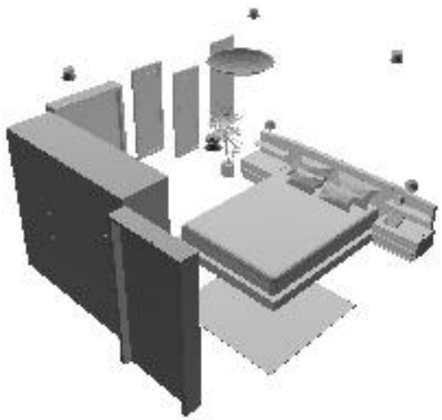


Flat grammar

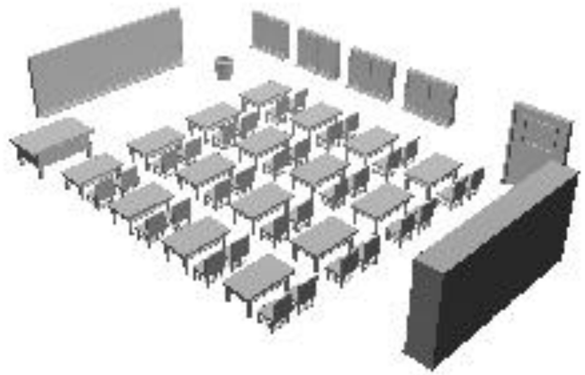


Ours

Datasets

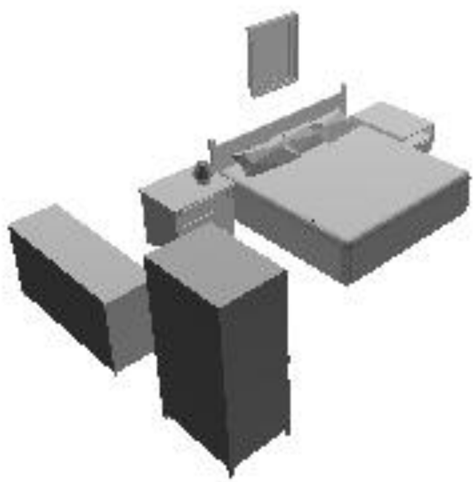


77 bedrooms

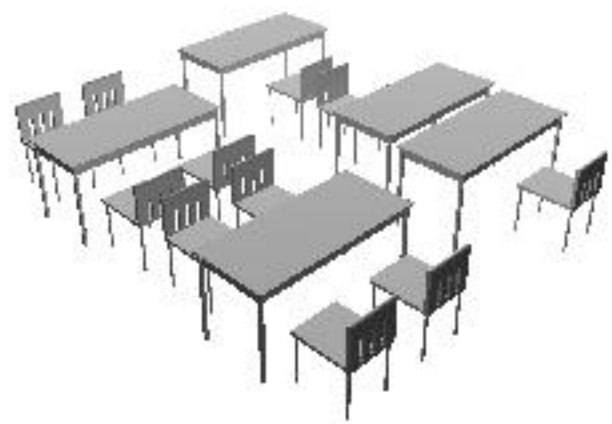


30 classrooms

8 libraries

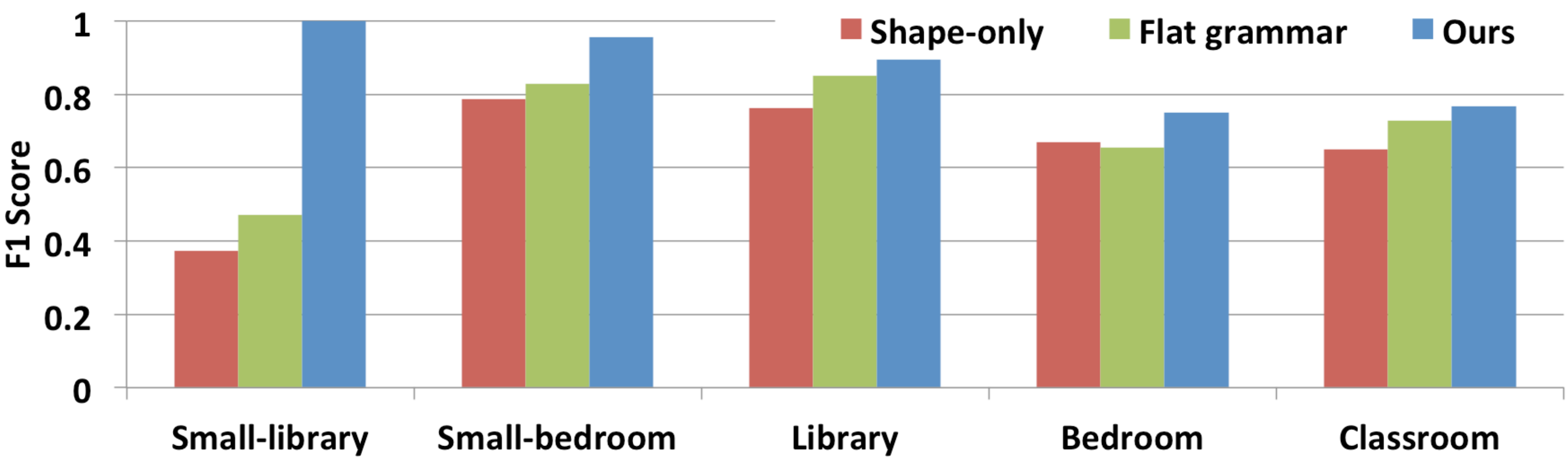


17 small bedrooms



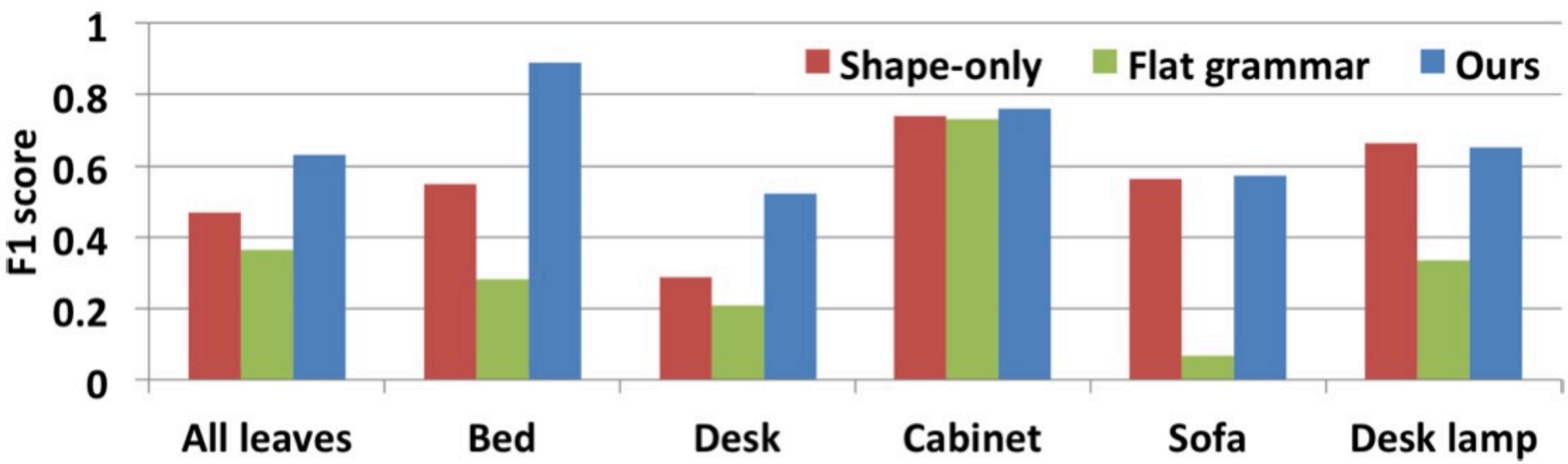
8 small libraries

Benefit of hierarchy



Object classification

Generalization of our method



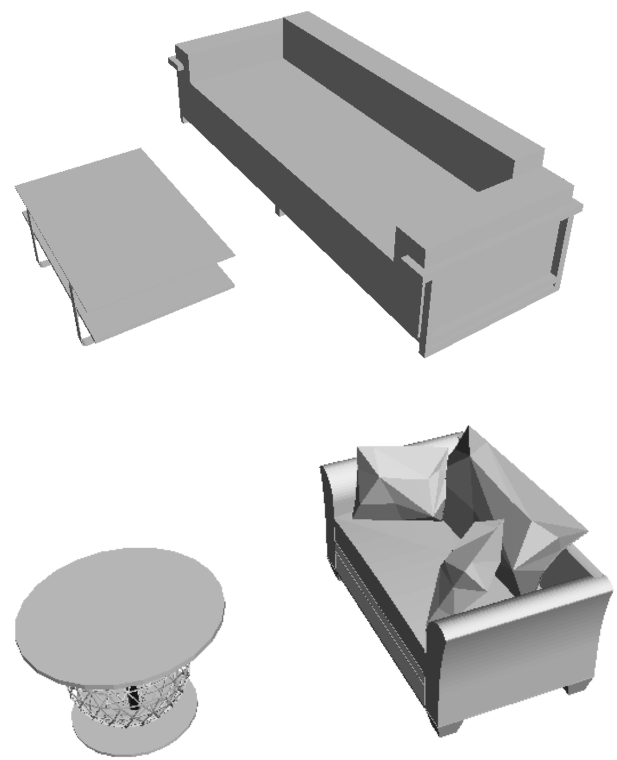
Parsing Sketch2Scene data set

Take-away message

- Modeling hierarchy improves scene understanding.

Limitation and Future Work

- Modeling correlation in probabilistic grammar.

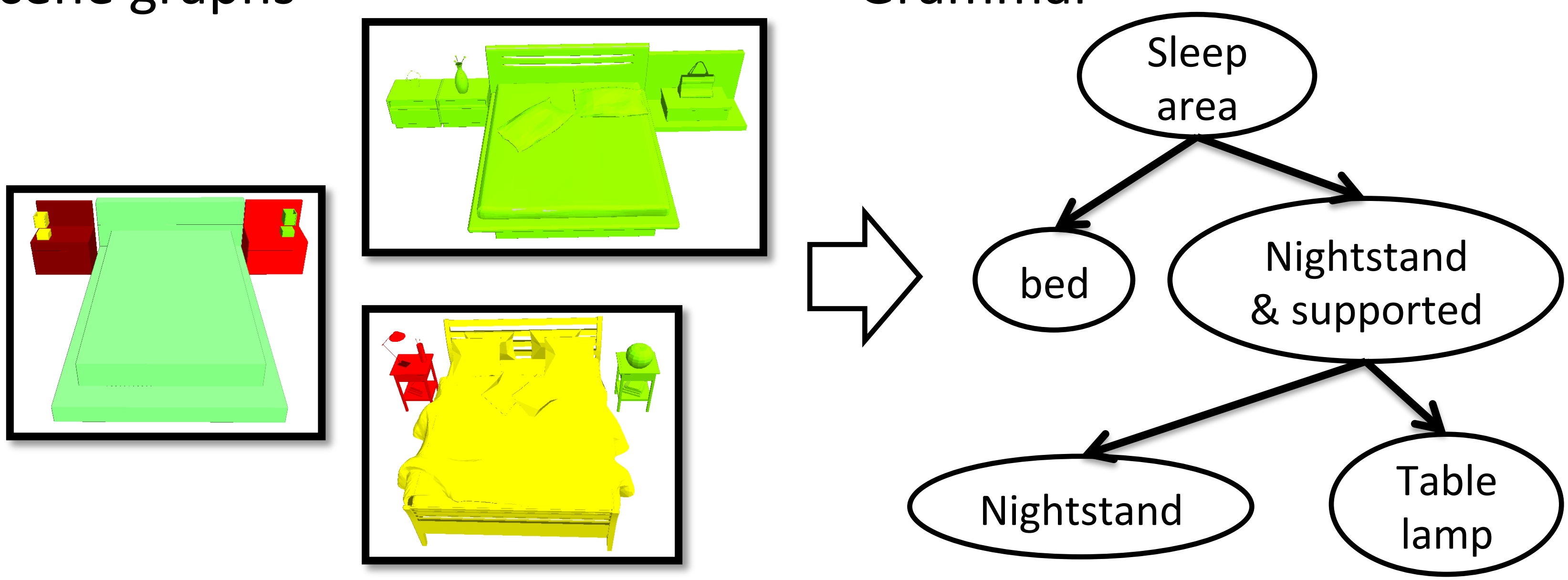


Limitation and Future Work

- Modeling correlation in probabilistic grammar.
- Grammar learning from noisy data.

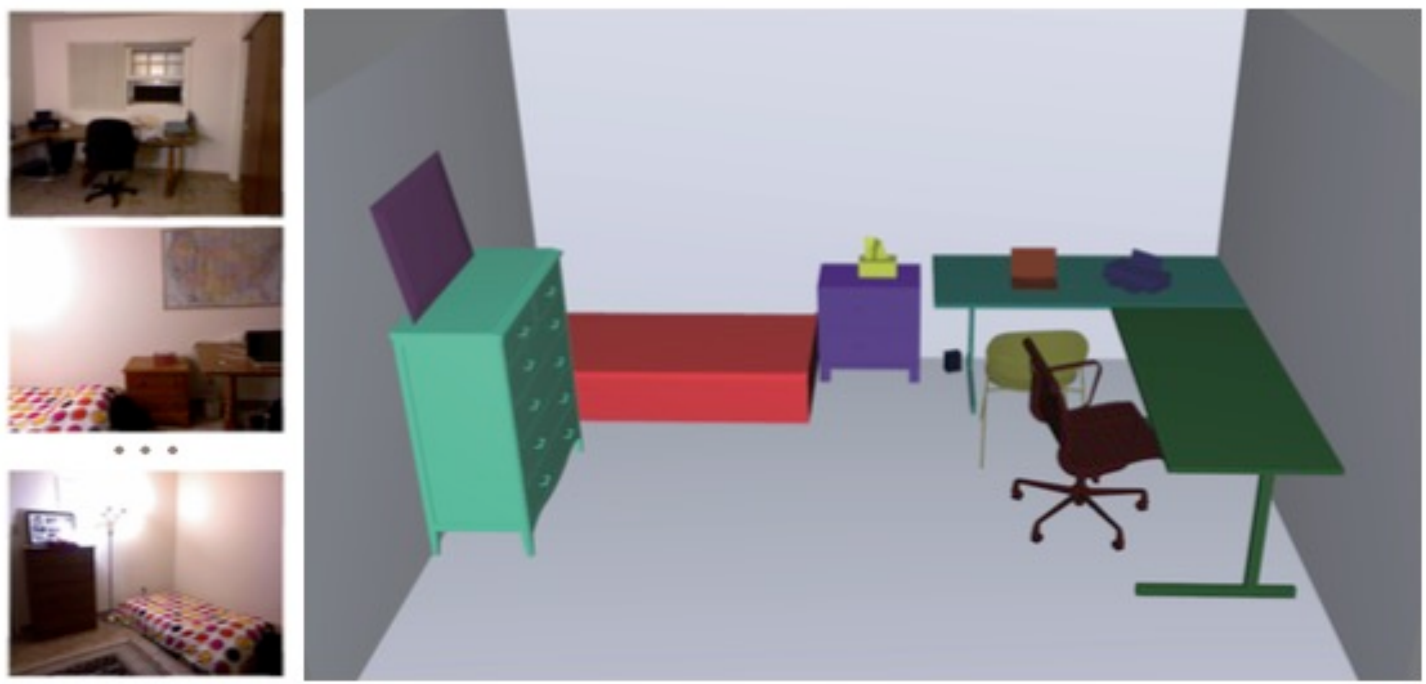
Input scene graphs

Grammar



Limitation and Future Work

- Modeling correlation in probabilistic grammar.
- Grammar learning from noisy data.
- Applications in other fields.



Modeling from RGB-D data [Chen et al. 2014]

Acknowledgement

Data

- Kun Xu

Discussion

- Christiane Fellbaum, Stephen DiVerdi

Funding

- NSF, ERC Starting Grant, Intel, Google, Adobe

Code and Data

http://www.cs.princeton.edu/~tianqian/projects/hierarchy/