UNIVERSITY OF CALIFORNIA

Los Angeles

Human-like Holistic 3D Scene Understanding

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Statistics

by

Siyuan Huang

2021

ABSTRACT OF THE DISSERTATION

Human-like Holistic 3D Scene Understanding

by

Siyuan Huang
Doctor of Philosophy in Statistics
University of California, Los Angeles, 2021
Professor Song-Chun Zhu, Chair

Building an intelligent machine with human-like perception, interaction, learning, and reasoning remains a significant and challenging problem. Despite the recent remarkable progress in artificial intelligence, especially the deep learning techniques, we are still far from reaching this goal. Human intelligence exhibits unique advantages in learning to solve multiple tasks from limited data, acquiring skills and knowledge from interactions, learning efficiently with stages, and generalizing concepts to novel domains and environments. Merely combining individual algorithms without a human-centric architecture is hopeless for achieving such comprehensive capabilities.

In this dissertation, we study the human-like holistic understanding in 3D scenes, which is the most related scenario to the real world. The core idea is to imitate the human's capability in perception, interaction, learning, and reasoning for solving holistic tasks. We first propose a framework for human-centric 3D scene parsing, reconstruction, and synthesis, focusing on integrating imagined humans into the perception system for interpreting the underlying human activities and intentions beyond the pixels. Then we describe several works on human-centric interaction understanding, including the human-object interactions and human-human interactions. Finally, we imitate the human-like learning and reasoning abilities by studying how to learn concepts with curriculum, design efficient closed-loop neural-grammar-symbolic learning algorithm, and build a concept learning framework that achieves systematic generalization.

The dissertation of Siyuan Huang is approved.

Ying Nian Wu

Demetri Terzopoulos

Hongjing Lu

Song-Chun Zhu, Committee Chair

University of California, Los Angeles

2021

To my parents and Yiling.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I give my first and deepest thanks to my advisor Song-Chun Zhu, for introducing me to the field of 3D scene understanding, showing me the most visionary pictures of AI, and encouraging me to tackle the most challenging topics. I have been deeply touched by his passion for AI research, dedication to the lifetime career goal, and resistance regardless of any challenges. I treasure all the insightful ideas and constructive suggestions he shares.

I would always appreciate Prof. Ying Nian Wu, who is the kindest person I have ever met. He showed me what a pure researcher is and how to be a scientist who continually works on the front line. His pursuit for simplicity and elegance in the mathematical model encourages me to develop simple but effective models all the time. He would always be the one to help and encourage us. I will remember the many afternoons in his office, where we had profound conversations over broad research topics and life.

My committee members, Prof. Hongjing Lu and Prof. Demetri Terzopoulos, also gave me tremendous help along the path. They show me the fantastic world of human cognition and computer graphics, foster my multidisciplinary knowledge and efforts.

Among all the senior students I collaborated with, I should appreciate Yixin Zhu and Siyuan Qi the most. Yixin helped me address tons of research and non-research problems in my early years, and Siyuan showed me how to be rigorous over formulations and arguments.

I studied computer vision with Jiwen Lu and Jie Zhou at Tsinghua University during my undergraduate study, thank them for providing the chances to get into the field. I also appreciate Anil K. Jain for hosting my summer internship at MSU in 2015. Thank Yuanlu Xu and Tony Tung for hosting the internship at Facebook. We explored an exciting area of human affordance learning with Minh Vo, Chengcheng Tang, and Petr Kadleček. I also thank Viorica Patraucean and Simon Osindero for providing tremendous support for my internship at DeepMind. These internships are done remotely due to the COVID-19, but I still feel connected with their kindness and warmth.

It is my great pleasure to work with talented junior students and colleagues. Many of

them have shaped my research directions, pushed me forward, and contributed significantly to the works described in this dissertation. I thank Yixin Chen for being a pioneer for human-object interaction topics, Qing Li for exploring the fantastic field of neural-symbolic learning and concept generalization, Baoxiong Jia for his extraordinary efforts in action and video understanding, and Yining Hong for her solid work in math word problems and vision-language understanding.

I thank all other co-authors of my papers, including Ruiqi Gao, Lifeng Fan, Yaxuan Zhu, Feng Gao, Chi Zhang, Mark Edmonds, Daniel Ciao, Pan Lu, Xiaodan Liang, Ran Gong, Tao Gao, Hangxin Liu, Josh Tenenbaum, Jianwen Xie, Yufan Ren, Ping Wei, Yichen Xie, Tao Yuan, Wenguan Wang, Xinyu Tang, Yinxue Xiao, Chenfanfu Jiang, Jenny Lin, Xinwen Guo, Lap-Fai Yu, Baohua Chen, Lei Deng, Yuanqi Duan. My dissertation would not be possible without their great contributions.

I enjoyed my five years as a member of VCLA. People here give me the feeling of home. I thank Arjun R. Akula, Eric Fischer, Xiaofeng Gao, Muzhi Han, Ziyuan Jiao, Tengyu Liu, Xiaojian Ma, Jonathan Mitchell, Erik Nijkamp, Liang Qiu, Shuwen Qiu, Yuxin Qiu, Feng Shi, Shu Wang, Sirui Xie, Xu Xie, Yifei Xu, Luyao Yuan, Victor Zhang, Zeyu Zhang, Tianyang Zhao, Yizhou Zhao, Zilong Zheng. I learned a lot from them.

I appreciate the generous help of researchers from various institutes. I thank Chuhang Zou, Hao Su, Alexander Schwing, Shuran Song, Jiajun Wu, and Leo Guibas for their suggestions and efforts to organize the 3D scene understanding workshop. I also thank Derek Hoiem, David Forsyth, Sanja Fidler, and Philip Torr for acknowledging my work and supports during the job search.

I appreciate all the joy that my friends gave me through basketball, games, food, parties, and travels. These unforgettable moments helped me go through hard times and would encourage me along with my future adventures.

Finally, I thank my parents for their selfless love and the best support in every stage of my life. I thank my girlfriend, Yiling Chen, for her caring and tolerance to me.

VITA

2016-2021	Graduate Research Assistant, Department of Statistics, UCLA
2020-2021	Dissertation Year Fellowship, UCLA
2020	Research Scientist Intern, DeepMind
2020	Research Intern, Facebook Reality Lab
2012-2016	B.E. in Automation, Tsinghua University
2015	Research Intern, PRIP Lab, Michigan State University

PUBLICATIONS

(* indicates equal contribution.)

Learning Neural Representation for Camera Pose by View Synthesis. Y. Zhu, R. Gao, **S. Huang**, S.-C. Zhu, Y. N. Wu. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2021

Learning by Fixing: Solving Math Word Problems with Weak Supervision. Y. Hong, Q. Li, D. Ciao, **S. Huang**, S.-C. Zhu. Association for the Advancement of Artificial Intelligence (AAAI), 2021

SMART: A Situation Model for Algebra Story Problems via Attributed Grammar. Y. Hong, Q. Li, R. Gong, D. Ciao, **S. Huang**, S.-C. Zhu. Association for the Advancement of Artificial Intelligence (AAAI), 2021

Closed Loop Neural-Symbolic Learning via Integrating Neural Perception, Grammar Parsing, and Symbolic Reasoning. Q. Li, **S. Huang**, Y. Hong, Y. Chen, Y. N. Wu, S.-C. Zhu. International Conference on Machine Learning (ICML), 2020

A Competence-aware Curriculum for Visual Concepts Learning from Natural Supervision. Q. Li, **S. Huang**, Y. Hong, S.-C. Zhu. European Conference on Computer Vision (ECCV), 2020

LEMMA: A Multiview Dataset for Learning Multi-agent Multi-task Activities. B. Jia, Y. Chen, **S. Huang**, Y. Zhu, S.-C. Zhu. European Conference on Computer Vision (ECCV), 2020

Dark, Beyond Deep: A Paradigm Shift to Cognitive AI with Humanlike Common Sense. Y. Zhu, T. Gao, L. Fan, **S. Huang**, M. Edmonds, H. Liu, F. Gao, C. Zhang, S. Qi, Y. N. Wu, J. B. Tenenbaum, S.-C. Zhu. Engineering, Special Issue on Artificial Intelligence, 2020

A Generalized Earley Parser for Human Activity Parsing and Prediction. S. Qi, B. Jia, **S. Huang**, P. Wei, S.-C. Zhu. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2020

3D Object Detection from a Single RGB Image via Perspective Points. **S. Huang**, Y. Chen, T. Yuan, S. Qi, Y. Zhu, S.-C. Zhu. Advances in Neural Information Processing Systems (NeurIPS), 2019

Holistic++ Scene Understanding: Single-view 3D Holistic Scene Parsing and Human Pose Estimation with Human-Object Interaction and Physical Commonsense. Y. Chen*, **S. Huang***, T. Yuan, S. Qi, Y. Zhu, S.-C. Zhu. IEEE International Conference on Computer Vision (ICCV), 2019

Understanding Human Gaze Communication by Spatio-Temporal Graph Reasoning. L. Fan*, W. Wang*, **S. Huang**, S. Qi, X. Tang, S.-C. Zhu. IEEE International Conference on Computer Vision (ICCV), 2019

Cooperative Holistic Scene Understanding: Unifying 3D Object, Layout, and Camera Pose Estimation. **S. Huang**, S. Qi, Y. Xiao, Y. Zhu, Y. N. Wu, S.-C. Zhu. Advances in Neural Information Processing Systems (NeurIPS), 2018

Holistic 3D Scene Parsing and Reconstruction from a Single RGB Image. **S. Huang**, S. Qi, Y. Zhu, Y. Xiao, Y. Xu, S.-C. Zhu. European Conference on Computer Vision (ECCV), 2018

Configurable 3D Scene Synthesis and 2D Image Rendering with Per-Pixel Ground Truth using Stochastic Grammars. C. Jiang*, S. Qi*, Y. Zhu*, **S. Huang***, J. Lin, X. Guo, L.-F.Yu, D. Terzopoulos, S.-C. Zhu. Internatianal Journal of Computer Vision (IJCV), 2018

Human-centric Indoor Scene Synthesis using Stochastic Grammar. S. Qi, Y. Zhu, **S. Huang**, C. Jiang, S.-C. Zhu. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018

Predicting Human Activities Using Stochastic Grammar. S. Qi, **S. Huang**, P. Wei, S.-C. Zhu. IEEE International Conference on Computer Vision (ICCV), 2017

CHAPTER 1

Introduction

Humans have superior capabilities in understanding and interacting with environments. Specifically, humans can: (1) learn to recognize the objects and events, infer and understand the invisible causes with an efficient process from a limited amount of data; (2) learn the skills and knowledge from passive and active interactions with the surroundings; (3) learn efficiently with stages and curriculum; (4) generalize the concept and adapt to novel domains and environments. How should we build a machine or robot with similar abilities in perception, interaction, learning, and reasoning?

Over the last decade, significant progress has been made in recognition, classification, and reconstruction with the rapid advancement of deep learning, fueled by hardware accelerations and the availability of massive sets of labeled data. However, we are still far away from solving computer vision or real machine intelligence. The inference and reasoning abilities of current computer vision systems are narrow and highly specialized, require large sets of labeled training data designed for particular tasks, and lack a general understanding of commonsense knowledge for cognitive reasoning, an efficient method to learn skills and concepts incrementally from the interactions, and a generalization ability to novel scenarios.

To tackle this challenge and fill in the gap between modern artificial intelligence (AI) and human intelligence, we focus on studying the **human-like holistic 3D scene understanding** problem. This fundamental problem requires machines to mimic humans' capabilities for solving holistic (comprehensive) tasks in 3D environments. It bridges and combines frontier research in **computer vision, computer graphics, machine learning, robotics, and cognitive science** by requiring various algorithms and modules for **perception, interaction, learning, and reasoning**, as shown in Figure 1.1.

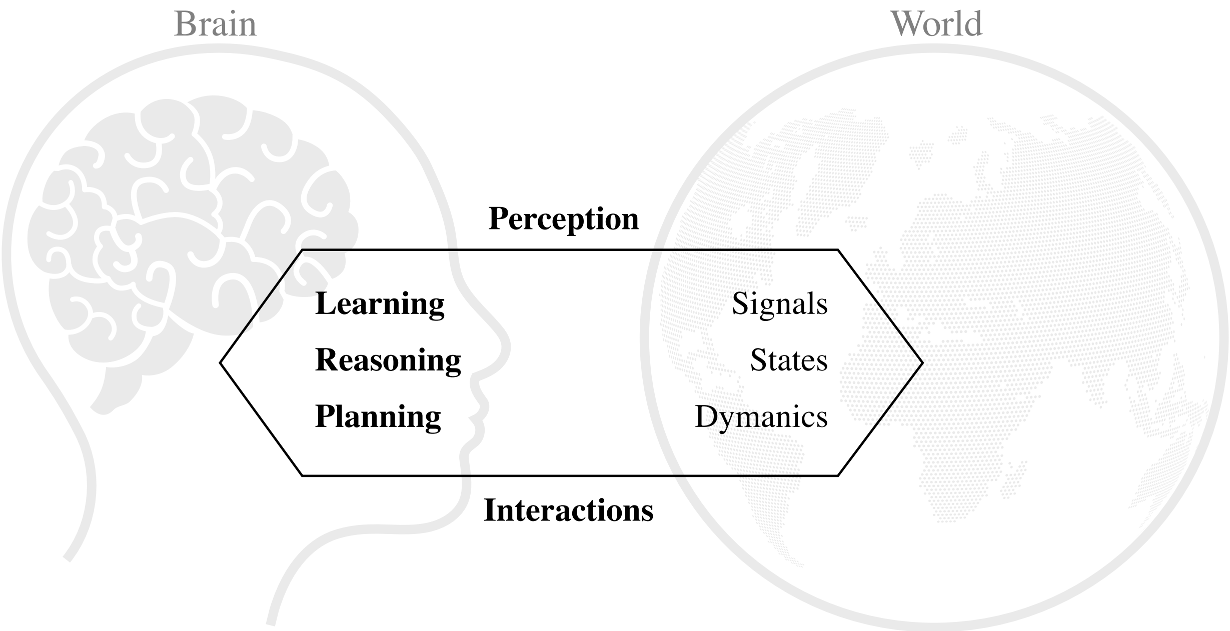


Figure 1.1: Illustration of how humans interact with the world. To build a machine with human-like holistic understanding of the 3D scenes, it is necessary to develop algorithms and systems for perception, interactions, learning, reasoning, and planning.

Specifically, we describe the problems, challenges, and our proposed framework for solving holistic tasks in the following sections. We leave the planning part for future research.

1.1 Perception: Human-like 3D Scene Understanding

Most previous 3D scene understanding approaches only focus on interpreting the visible pixels and entities in the images. However, they lack an in-depth understanding of the underlying invisible causes for the images, *i.e.*, the human activities, human intentions, and hidden physical and social commonsense. To achieve such capabilities, we propose human-centric algorithms for 3D scene parsing and reconstruction [HQZ18, HQX18, HCY19]. In [HQZ18], we jointly parse a single RGB image and reconstruct a holistic 3D configuration composed by a set of CAD models through analysis-by-synthesis. Besides, as shown in Figure 1.2, we incorporate imagined humans into the computational framework and reason by maximizing the marginal posterior probability over the visible objects and layouts, creating

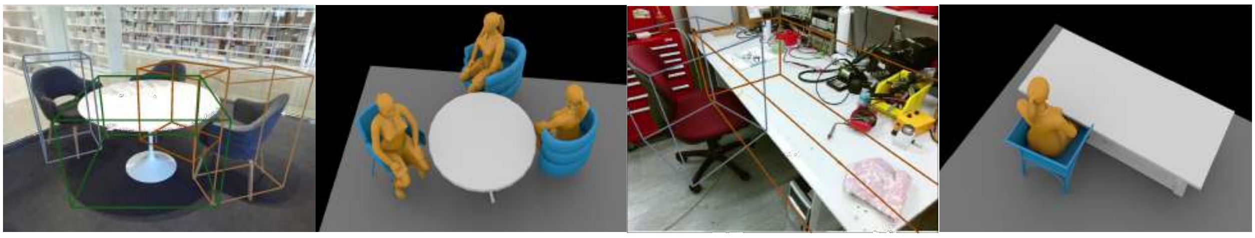


Figure 1.2: Human-centric 3D scene parsing and reconstruction. By incorporating imagined humans into the task-oriented computational framework, the perception system can generate interacting humans and infer the affordance of the reconstructed 3D scenes. (from [HQZ18])

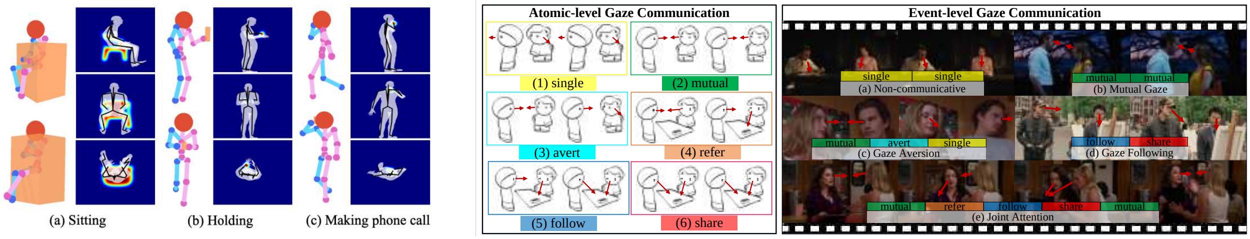
a task-oriented model that generates diverse human-like solutions. This approach serves as a slow-thinking reasoning process for holistic 3D scene understanding based on short-run Markov chain Monte Carlo (MCMC).

Moreover, we train multiple branches of deep networks that solve holistic tasks (*i.e.*, 3D object detection, 3D room layout estimation, and camera pose estimation) cooperatively in [HQX18, HCY19]. They serve as the fast-thinking initialization for the holistic 3D scene understanding. Combined with [HQZ18], we build up a complete computational perception framework for human-like holistic 3D scene understanding.

We further design human-centric 3D synthesis approaches [JQZ18, QZH18]. They optimize the room arrangements by considering the imagined humans and their potential activities as relational contexts. With a graphics rendering engine, they sample and synthesize 3D room layouts and 2D images to obtain large-scale realistic 2D/3D image data with the perfect per-pixel ground truth.

1.2 Interaction: Human-like 3D Interaction Understanding

Humans exhibit extraordinary abilities in developing skills and knowledge from the interaction with surrounding environments, making the human-like understanding of 3D interaction important. An intelligent machine is expected to understand the structures and essence of human-object interactions and human-human interactions for interpreting task-oriented activities, intentions, and social relations. To study the 3D human-object interactions (HOIs),



(a) Human-object interactions.

(b) Human-human interactions.

Figure 1.3: To interpret the human activities and social relations, we learn the human-object interactions (a) and human-human interactions (b) from images and videos.

we propose a joint parsing algorithm for understanding the 3D human-object interaction with physical commonsense [CHY19]. To understand how humans communicate and collaborate, we propose a spatio-temporal graph reasoning approach for understanding human gaze interaction [FWH19] and a dataset for learning multi-task multi-agent activities [JCH20].

1.3 Learning and Reasoning: Human-like Representation and Concept Learning

Opposite to learning narrow tasks with a massive amount of labeled data, humans are excel at (1) learning the representation with less supervision; (2) learning efficiently and incrementally with stages; and (3) generalizing concepts into novel domains and environments. In order to imitate such fascinating capabilities, we look at the triangulation strategy among observations from humans, property of data, and principles of designing algorithms, as shown in Figure 1.3. Drawing inspiration from the brain, we design algorithms that can efficiently learn from data and generalize to novel scenes.

Specifically, we propose a competence-aware curriculum for visual concept learning [LHH20b]; a closed-loop learning method for efficient neural-symbolic reasoning [LHH20a]; and an arithmetic approach for studying the systematic generalization of perception, syntax, and semantics [LHH21]. Some of these learning and reasoning approaches are studied in clean and straightforward settings but are promising for generalizing to real-world applications.

The dissertation is mainly structured by the aforementioned three parts of human-like

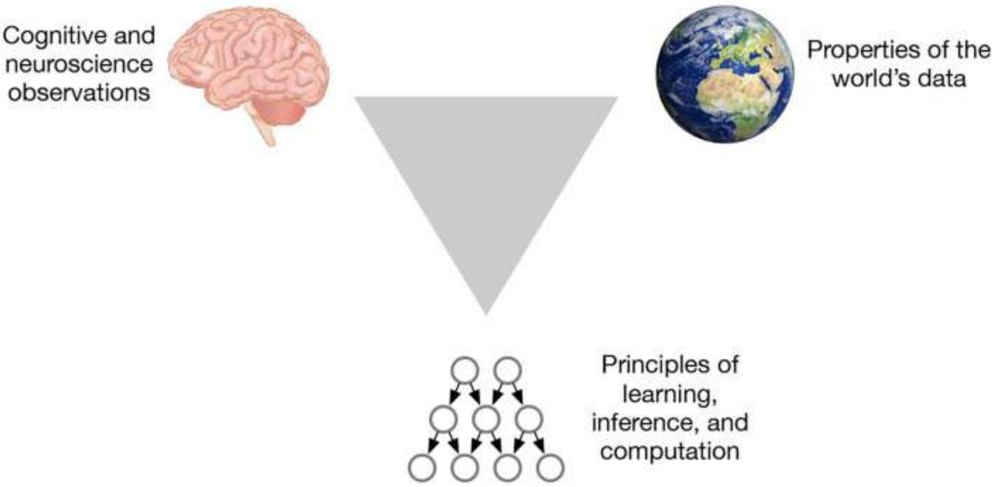


Figure 1.4: The triangulation strategy for designing the human-like learning and reasoning system. It extracts principles from the brain and looks at three aspects at the same time. (from [GLG20])

understanding. We summarize our methods and propose promising directions for future research in these areas in the last chapter.

Part I

Perception: Human-like 3D Scene Understanding

CHAPTER 2

Human-centric Task-oriented 3D Scene Parsing and Reconstruction

In this chapter, we introduce a complete computational framework for **human-centric task-oriented 3D scene parsing and reconstruction.** This framework jointly parse a single RGB image and reconstruct a holistic 3D configuration composed by a set of CAD models. The computation model consists of two parts: (1) a bottom-up initial module (Section 2.2 and Section 2.3) for proposing the objects, layout, and camera parameter as an initial parse graph; (2) a joint inference module (Section 2.1) that seeks optimal configuration using Markov chain Monte Carlo (MCMC), which efficiently traverses through the non-differentiable solution space, jointly optimizing object localization, 3D layout, and hidden human context.

2.1 Holistic 3D Scene Parsing and Reconstruction with Imagined Human

2.1.1 Introduction

In this section, We propose a computational framework to parse and reconstruct the 3D configuration of an indoor scene from a single RGB image using a stochastic grammar model. We introduce a Holistic Scene Grammar (HSG) to represent the 3D scene structure, which characterizes a joint distribution over the functional and geometric space of indoor scenes. The proposed HSG captures three essential but often latent dimensions of the indoor scenes: i) latent human context, describing the affordance and the functionality of a room arrangement, ii) geometric constraints over the scene configurations, and iii) physical constraints that guarantee physically plausible parsing and reconstruction. We solve this parsing and reconstruction problem in an analysis-by-synthesis fashion, seeking to minimize the differences between the input image and the rendered image generated by our 3D representation, over the space of depth, surface normal, and object segmentation map. The optimal configuration (*i.e.*, parse graph) is inferred using Markov chain Monte Carlo (MCMC), which efficiently traverses through the non-differentiable solution space, jointly optimizing object localization, 3D layout, and hidden human context. Experimental results demonstrate that the proposed algorithm improves the generalization ability and significantly outperforms prior methods on 3D layout estimation, 3D object detection, and holistic scene understanding.

The complexity and richness of human vision are not only reflected by the ability to recognize visible objects, but also to reason about the latent actionable information [Soa13], including inferring latent human context as the functionality of a scene [QZH18, JKS13], reconstructing 3D hierarchical geometric structure [GEH10, LZZ14], and complying with the physical constraints that guarantee the physically plausible scene configurations [ZZJ14]. Such rich understandings of an indoor scene are the essence for building an intelligent computational system, which transcends the prevailing appearance- and geometry-based recognition tasks to account also for the deeper reasoning of observed images or patterns.

One promising direction is *analysis-by-synthesis* [YK06] or "vision as inverse graphics" [Gre76, LB14]. In this paradigm, computer vision is treated as an inverse problem as opposed to computer graphics, of which the goal is to reverse-engineer hidden factors occurred in the physical process that produces observed images.

In this work, we embrace the concept of vision as inverse graphics, and propose a 3D indoor scene parsing and reconstruction algorithm that simultaneously reconstructs the functional hierarchy and the 3D geometric structure of an indoor scene from a RGB image. Figure 2.1 schematically illustrates the analysis-by-synthesis inference process. The joint inference algorithm takes proposals from various vision modules and infers the 3D structure by comparing various projections (*i.e.*, depth, normal, and segmentation) rendered from the recovered 3D structure with the ones directly estimated from an input image.

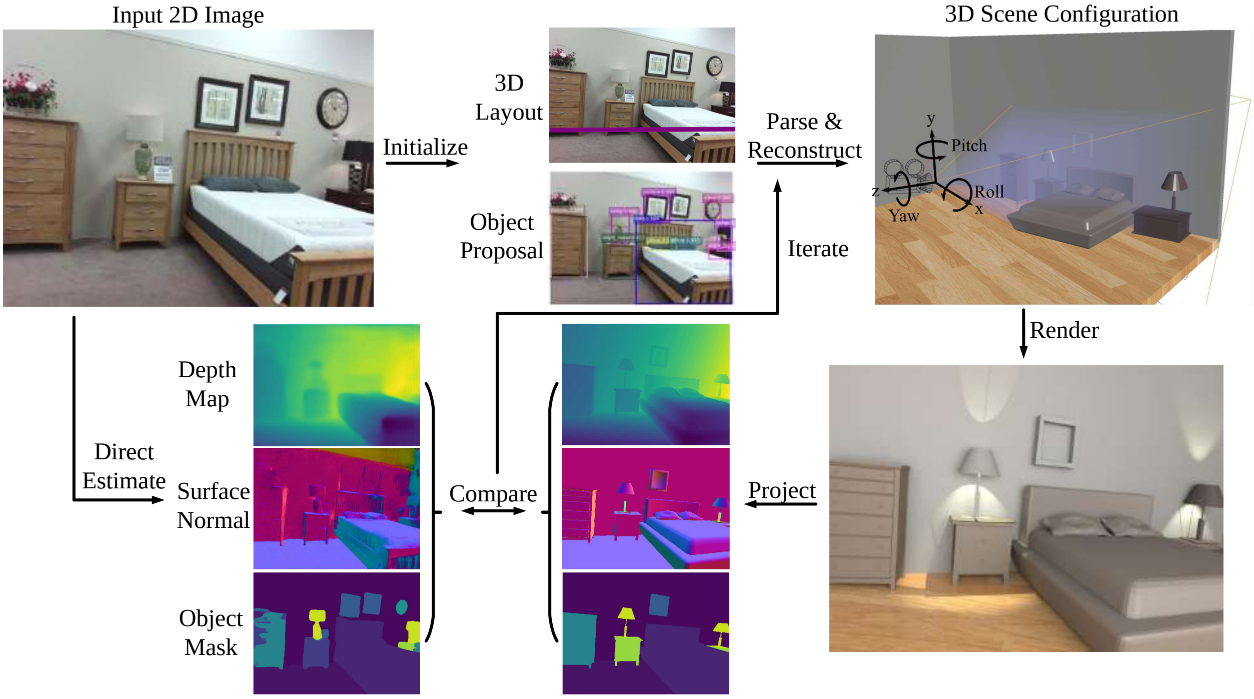


Figure 2.1: Illustration of the proposed 3D indoor scene parsing and reconstruction in an analysis-by synthesis fashion. A 3D representation is initialized by individual vision modules (*e.g.*, object detection, 2D layout estimation). A joint inference algorithm compares the differences between the rendered normal, depth, and segmentation map with the ones estimated directly from the input RGB image, and adjust the 3D structure iteratively.

Specifically, we introduce a Holistic Scene Grammar (HSG) to represent the hierarchical structure of a scene. As illustrated in Figure 2.2, our HSG decomposes a scene into hidden groups in the *functional space* (*i.e.*, hierarchical structure including activity groups) and object instances in the *geometric space* (*i.e.*, CAD models). For the functional space, in contrast to the conventional method that only models the object-object relations, we propose a novel method to model human-object relations by imagining latent human in activity groups to further help explain and parse the observed image. For the geometric space, the geometric attributes (*e.g.*, size, position, orientation) of individual objects are taken into considerations, as well as the geometric relations (*e.g.*, supporting relation) among them. In addition, physical constraints (*e.g.*, collision among the objects, violations of the layout) are incorporated to generate a physically plausible 3D parsing and reconstruction of the observed image.

Here, an indoor scene is represented by a parse graph (**pg**) of a grammar, which consists of a hierarchical structure and a Markov random field (MRF) over terminal nodes that captures the rich contextual relations between objects and room layout (*i.e.*, the room configuration of walls, floors, and ceilings).

A maximum a posteriori probability (MAP) estimate is designed to find the optimal solution that parses and reconstructs the observed image. The likelihood measures the similarity between the observed image and the rendered images projected from the inferred **pg** onto the 2D image space. Thus, the **pg** can be iteratively refined by sampling an MCMC with simulated annealing based on posterior probability. We evaluate our method on a large-scale RGB-D dataset by comparing the 3D reconstruction results with the ground-truth.

2.1.1.1 Related Work

Scene Parsing: Existing scene parsing approaches fall into two streams. i) Discriminative approaches [DHS15, ZJR15, NHH15, CPK17, LSD15, LMS17, ZSQ17] classify each pixel to a semantic label. Although prior work has achieved high accuracy in labeling the pixels, these methods lack a general representation of visual vocabulary and a principle approach to exploring the semantic structure of a general scene. ii) Generative approaches [ZZ11, ZZ13, CCP13, LFU13, GH13, ZST14, ZSY17a, ZLH17] can distill scene structure, making it closer to human-interpretable structure of a scene, enabling potential applications in robotics, VQA, *etc.*. In this work, we combine those two streams in an analysis-by-synthesis framework to infer the hidden factors that generate the image.

Single Image Scene Reconstruction: Previous approaches [HEH05, HZ05, SCN06] of indoor scene reconstruction using a single RGB image can be categorized into three streams. i) 2D or 3D room layout prediction by extracting geometric features and ranking the 3D cuboids proposals [HHF09, LHK09, ML15, DFC16, RLC16, ISS17a, LBM17, ZLY17]. ii) By representing objects via geometric primitives or CAD models, previous approaches [SNS13, AME14, LKT14, SX14, TM15, BRG16, SX16, WXL16, DL17] utilize 3D object recognition or pose estimation to align object proposals to a RGB or depth image. iii) Joint estimation of the room layout and 3D objects with contexts [ZZ13, CCP13, ZST14, SLX15, ISS17a, ZLH17, ZSY17a, SYZ17a]. In particular, Izadinia *et al.*[ISS17a] show promising results in inferring the layout and objects without the contextual relations and physical constraints. In contrast, our method models the hierarchical scene structure, hidden human context and physical constraints, providing a semantic representation. Furthermore, our method presents a joint inference algorithm using MCMC, which in theory can achieve a global optimal.

Scene Grammar: Scene grammar models have been used to infer the 3D structure and functionality from a RGB image [ZZ11, ZZ13, JKS13, JS14]. Our HSG differs from [ZZ11, ZZ13] in two major aspects: i) Our model represents the 3D objects with CAD models rather than geometric primitives, modeling detail contextual relations (*e.g.*, supporting relation), which provides better realization of parsing and reconstruction. ii) We infer latent human and activity groups in the HSG, which helps the explanation and parsing. Compared to [JKS13, JS14], we model and parse the 3D structure of objects and layouts from a single RGB image, rather than labeling the point-clouds using RGB-D images.

2.1.1.2 Contributions

This work makes four major contributions:

1. We integrate geometry and physics to interpret and reconstruct indoor scenes with CAD models. We jointly optimize 3D room layouts and object configurations, largely improving the performance of scene parsing and reconstruction on SUN RGB-D dataset [SLX15].

2. We incorporate hidden human context (*i.e.*, functionality) into our grammar, enabling to imagine latent human pose in each activity group by grouping and sampling. In this way, we can optimize the joint distribution of both visible and invisible [XTZ13] components of the scene.

3. We propose a complete computational framework to combine generative model (*i.e.*, a stochastic grammar), discriminative models (*i.e.*, direct estimations of depth, normal, and segmentation maps) and graphics engines (*i.e.*, rendered images) in scene parsing and

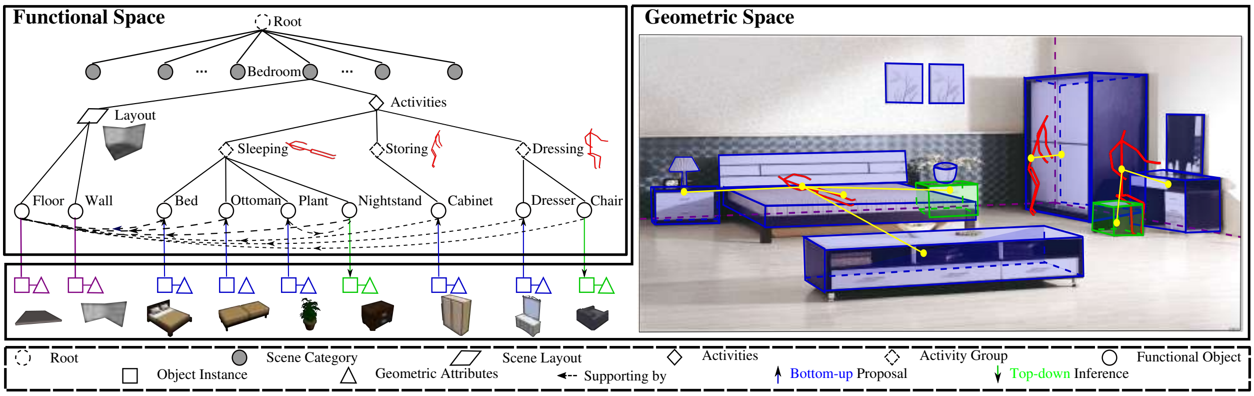


Figure 2.2: An indoor scene represented by a parse graph ($\boldsymbol{pg}$) of the HSG that spans across the functional space and the geometric space. The functional space characterizes the hierarchical structure and the geometric space encodes the spatial entities with contextual relations.

reconstruction. To the best of our knowledge, we are the first to use the inferred depth, surface normal and object segmentation map to aid parsing and reconstructing monocular scenes (room layout and multiple objects).

4. We model the supporting relations among objects, eliminating the widely adopted assumption that all objects must stand on the ground. Such flexibility provides better parsing and reconstruction of the real-world scenes with complex object relations.

2.1.2 Holistic Scene Grammar

We represent the hierarchical structure of indoor scenes by a Holistic Scene Grammar (HSG). An HSG consists of a latent hierarchical structure in the functional space $\mathbb{F}$ and terminal object entities in the geometric space $\mathbb{G}$. The intuition is that for human environments, the object arrangement in the geometric space can be viewed as a projection from the functional space (*i.e.*, human activities). The functional space as a probabilistic context free grammar (PCFG) captures the hierarchy of the functional groups, and the geometric space captures the spatial contexts among objects by defining an MRF on the terminal nodes. The two spaces together form a stochastic context-sensitive grammar (SCSG). The HSG starts from a root scene node and ends with a set of terminal nodes. An indoor scene is represented by a parse graph **pg** as illustrated in Figure 2.2.

Definition: The stochastic context-sensitive grammar HSG is defined as a 5-tuple $\langle S, V, R, E, P\rangle$. S denotes the root node of the indoor scene. V is the vertex set that includes both non-terminal nodes $V_f \in \mathbb{F}$ and terminal nodes $V_g \in \mathbb{G}$. R denotes the production rule, and E the contextual relations among the terminal nodes, which are represented by the horizontal links in the **pg**. P is the probability model defined on the **pg**.

Functional Space: The non-terminal nodes $V_f = \{V_f^c, V_f^a, V_f^o, V_f^l\} \in \mathbb{F}$ consist of the scene category nodes V_f^c, activity group nodes V_f^a, objects nodes V_f^o, and layout nodes V_f^l.

Geometric Space: The terminal nodes $V_g = \{V_g^o, V_g^l\} \in \mathbb{G}$ are the CAD models of object entities and room layouts. Each object $v \in V_g^o$ is represented as a CAD model, and the object appearance is parameterized by its 3D size, location, and orientation. The room layout $v \in V_g^l$ is represented as a cuboid which is further decomposed into five planar surfaces of the room (left wall, right wall, middle wall, floor, and ceiling with respect to the camera coordinate).

The following production rules R are defined for HSG:

Production Rule	Semantic Meaning	Instances
$r1: \ S \to V_f^c$	scene → category 1 | category 2 | ...	scene → office| kitchen
$r2: \ V_f^c \to V_f^a \cdot V_f^l$	category → activity groups · layout	office → (walking, reading) · layout
$r3: \ V_f^a \to V_f^o$	activity group → functional objects	sitting → (desk, chair)

where · denotes the deterministic decomposition, | alternative explanations, and () combination. Contextual relations E capture relations among objects, including their relative positions, relative orientations, grouping relations, and supporting relations. The objects could be supported by either other objects or the room layout; *e.g.*, a lamp could be supported by a night stand or the floor.

Finally, a scene configuration is represented by a **pg**, whose terminals are room layouts and objects with their attributes and relations. As shown in Figure 2.2, a **pg** can be decomposed as $\mathbf{pg} = (pg_f, pg_g)$, where pg_f and pg_g denote the functional part and geometric part of the **pg**, respectively. $E \in pg_g$ denotes the contextual relations in the terminal layer.

2.1.3 Probabilistic Formulation

The objective of the holistic scene parsing is to find an optimal **pg** that represents all the contents and relations observed in the scene. Given an input RGB image I, the optimal **pg** could be derived by an MAP estimator,

$$\begin{aligned} p(\mathbf{pg}|I) &\propto p(\mathbf{pg}) \cdot p(I|\mathbf{pg}) && (2.1) \\ &\propto p(pg_f) \cdot p(pg_g|pg_f) \cdot p(I|pg_g) && (2.2) \\ &= \frac{1}{Z} \exp\left\{-\mathcal{E}(pg_f) - \mathcal{E}(pg_g|pg_f) - \mathcal{E}(I|pg_g)\right\}, && (2.3) \end{aligned}$$

where the prior probability $p(\mathbf{pg})$ is decomposed into $p(pg_f)p(pg_g|pg_f)$, and $p(I|\mathbf{pg}) = p(I|pg_g)$ since the image space is independent of the functional space given the geometric space. We model the joint distribution with a Gibbs distribution; $\mathcal{E}(pg_f)$, $\mathcal{E}(pg_g|pg_f)$ and $\mathcal{E}(I|pg_g)$ are the corresponding energy terms.

Functional Prior $\mathcal{E}(pg_f)$ characterizes the prior of the functional aspect in a **pg**, which models the hierarchical structure and production rules in the functional space. For production rules of alternative explanations | and combination (), each rule selects child nodes and the probability of the selections is modeled with a multinomial distribution. The production rule · is deterministically expanded with probability 1. Given a set of production rules R, the energy could be written as:

$$\mathcal{E}(pg_f) = \sum\nolimits_{r_i \in R} - \log p(r_i). \tag{2.4}$$

Geometric Prior $\mathcal{E}(pg_g|pg_f)$ characterizes the prior of the geometric aspect in a **pg**. Besides modeling the size, position and orientation distribution of each object, we also consider two types of contextual relations $E = \{E_s, E_a\}$ among the objects: i) relations E_s between supported objects and their supporting objects (*e.g.*, monitor and desk); ii) relations E_a between imagined human and objects in an activity group (*e.g.*, relation between imagined human and the chair in an activity group of reading).

We define different potential functions for each type of contextual relations, constructing an MRF in the geometric space including four terms:

$$\mathcal{E}(pg_g|pg_f) = \mathcal{E}_{sc}(pg_g|pg_f) + \mathcal{E}_{spt}(pg_g|pg_f) + \mathcal{E}_{grp}(pg_g|pg_f) + \mathcal{E}_{phy}(pg_g). \tag{2.5}$$

• *Size Consistency* $\mathcal{E}_{sc}$ constrains the size of an object. We model the distribution of object scale using a non-parametric way, *i.e.*, kernel density estimation (KDE),

$$\mathcal{E}_{sc}(pg_g|pg_f) = \sum\nolimits_{v_i \in V_g^o} -\log p\left(s_i|V_f^o\right), \tag{2.6}$$

where s_i denotes the size of object v_i. Empirically, we find that KDE fits better than a parametric estimation (*e.g.*, multivariate normal), and it is easier to sample from.

• *Supporting Constraint* $\mathcal{E}_{spt}$ characterizes the contextual relations between supported objects and supporting objects (including floors, walls and ceilings). We model the distribution with their relative heights and overlapping areas:

$$\mathcal{E}_{spt}(pg_g|pg_f) = \sum\nolimits_{(v_i,v_j)\in E_s} \mathcal{K}_o(v_i, v_j) + \mathcal{K}_h(v_i, v_j) - \lambda_s \log\, p\left(v_i, v_j \mid V_f^l, V_f^o\right), \tag{2.7}$$

where $\mathcal{K}_o(v_i, v_j) = 1 - area(v_i \cup v_j)/area(v_i)$ defines the overlapping ratio in xy-plane, and $\mathcal{K}_h(v_i, v_j)$ defines the relative height between the lower surface of v_i and the upper surface of v_j. $\mathcal{K}_o(\cdot)$ and $\mathcal{K}_h(\cdot)$ is 0 if supporting object is floor and wall, respectively. $p(v_i, v_j|V_f^l, V_f^o)$ is the prior frequency of the supporting relation modeled by multinoulli distributions. λ_s is a balancing constant.

• *Human-Centric Grouping Constraint* $\mathcal{E}_{grp}$. For each activity group, we imagine the invisible and latent human poses to help parse and understand the scene. The intuition is that the indoor scenes are designed to serve human daily activities, thus the indoor images should be jointly interpreted by the observed entities and the unobservable human activities. This is known as the *Dark Matter* [XTZ13] in computer vision that drives the visible components in the scene. Prior methods on scene parsing often merely model the object-object relations. In this work, we go beyond passive observations to model the latent human-object relations, thereby proposing a human-centric grouping relationship and a joint inference algorithm over the visible scene and invisible latent human context. Specifically, for each activity group $v \in V_f^a$, we define correspondent imagined human with a six tuple $< y, \mu, t, r, s, \tilde{\mu} >$, where y is the activity type, $\mu \in \mathbb{R}^{25\times3}$ is the mean pose of activity type y, t denotes the translation, r denotes the rotation, s denotes the scale, and $\tilde{\mu}$ is the imagined position to place a person: $\tilde{\mu} = \mu \cdot r \cdot s + t$. The energy among the imagined human and objects is defined as:

$$\begin{aligned}\mathcal{E}_{grp}(pg_g|pg_f) &= \textstyle\sum_{v_i \in V_f^a} \mathcal{E}_{grp}(\tilde{\mu}_i|v_i) \\ &= \textstyle\sum_{v_i \in V_f^a} \sum_{v_j \in ch(v_i)} \mathcal{D}_d(\tilde{\mu}_i, \nu_j; \bar{d}) + \mathcal{D}_h(\tilde{\mu}_i, \nu_j; \bar{h}) + \mathcal{D}_o(\tilde{\mu}_i, \nu_j; \bar{o}),\end{aligned} \tag{2.8}$$

where $ch(v_i)$ denotes the set of child nodes of v_i, ν_j denotes the 3D position of v_j. $\mathcal{D}_d(\cdot)$, $\mathcal{D}_h(\cdot)$ and $\mathcal{D}_o(\cdot)$ denote geometric distances, heights and orientation differences, respectively, calculated by the center of the imagined human pose to the object center subtracted by their mean (*i.e.*, $\bar{d}$, $\bar{h}$ and $\bar{o}$). Figure 2.3 shows some examples of the imagined human.

- *Physical Constraints:* Additionally, in order to avoid violating physical laws during parsing, we define the physical constraints $\mathcal{E}_{phy}(pg_g)$ to penalize physical violations. Exceeding the room cuboid or overlapping among the objects are defined as violations. This term is formulated as:

$$\mathcal{E}_{phy}(pg_g) = \textstyle\sum_{v_i \in V_g^o} (\sum_{v_j \in V_g^o \backslash v_i} \mathcal{O}_o(v_i, v_j) + \sum_{v_j \in V_g^l} \mathcal{O}_l(v_i, v_j)), \tag{2.9}$$

where $\mathcal{O}_o(\cdot)$ denotes the overlapping area between objects, and $\mathcal{O}_l(\cdot)$ denotes the area of objects exceeding the layout.

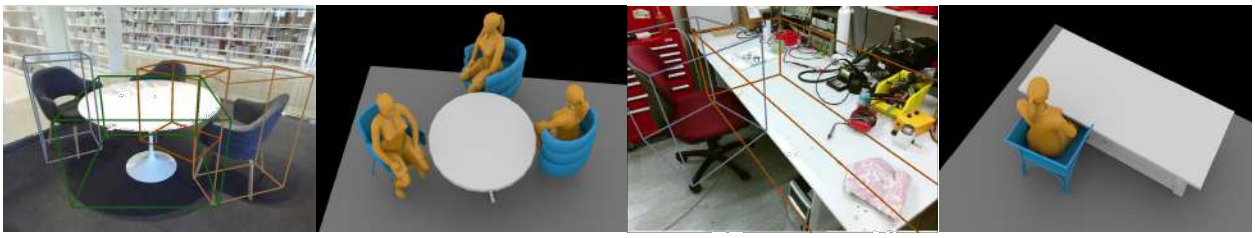


Figure 2.3: Illustration of imagined human in scene parsing. We learn the distribution of the human-object relation and utilize it to sample human poses.

Likelihood $\mathcal{E}(I|pg_g)$ characterizes the similarity between the observed image and the rendered image generated by the parsing results. Since there is still a difference between the two images due to various lighting conditions, textures, and material properties, we solve the problem in an *analysis-by-synthesis* fashion. By combining generative models and discriminative models, this approach tries to reverse-engineer the hidden factors that generate the observed image.

Specifically, we first use discriminative methods to project the original image I to various feature spaces. In this work, we directly estimate three intermediate images including the depth map $\Phi_d(I)$, surface normal map $\Phi_n(I)$ and object segmentation map $\Phi_m(I)$, as the feature representation of the observed image I.

Meanwhile, a **pg** inferred by our method represents the 3D structure of the observed image. Thus, we can use the inferred **pg** to reconstruct image I', and recover the corresponding depth map $\Phi_d(I')$, surface normal map $\Phi_n(I')$, and object segmentation map $\Phi_m(I')$ through a forward graphics rendering.

Finally, we compute the likelihood term by comparing these rendered results from the generative model with the directly estimated results calculated by the discriminative models. Specifically, the likelihood is computed by pixel-wise differences between the two sets of maps,

$$\mathcal{E}(I|pg_g) = \mathcal{D}_p(\Phi_d(I), \Phi_d(I')) + \mathcal{D}_p(\Phi_n(I), \Phi_n(I')) + \mathcal{D}_p(\Phi_m(I), \Phi_m(I')), \tag{2.10}$$

where function $\mathcal{D}_p(\cdot)$ indicates the summation of pixel-wise Euclidean distances between the two maps.

2.1.4 Learning

The learning process contains two major steps: i) collecting the statistics of scene categories, object categories, object sizes and supporting relations from SUN RGB-D dataset [SLX15]; ii) collecting the statistics of grouping occurrences and the geometric relations between objects and human from Watch-n-Patch [WZS15].

Using SUN RGB-D, we model the prior of scene types, object categories and support relations by multinoulli distributions. For example, a lamp is supported by the floor with a probability of 0.4 and by a desk with a probability of 0.2. The branching probability is simply counting the frequency of each alternative choice. The distribution of the object sizes is learned via non-parametric KDE.

The human-centric grouping occurrence and human-object interactions in 3D space are learned from the Watch-n-Patch. This dataset collects the RGB-D videos of human activities in offices and kitchens. Since some activities are irrelevant with objects, we learn the activities of 'reading', 'play-computer', 'take-item' and 'put-down-item' in all the office videos. For each activity, we first extract key frames from each sequence with group activity labels. Then we compute the occurrence frequency of the objects around human within a distance threshold, and model the prior of object category using a multinomial distribution. The geometric relations between the objects and humans are similarly learned by fitting normal distributions of relative distance, height, and orientation between each joint of a human pose and the object center.

2.1.5 Inference

Given a single RGB image as the input, the goal in the inference phrase is to find the optimal **pg** that best explains the hidden factors that generate the observed image while recovering the 3D scene structure.

The inference process includes three major steps.

- *Room geometry estimation:* estimate the room geometry by predicting the 2D room layout and the camera parameter, and projecting the estimated 2D layout to 3D. Details are provided in Section 2.1.5.1.

- *Objects initialization:* detect objects and retrieve CAD models correspondingly with the most similar appearance, then roughly estimate their 3D poses, positions, sizes, and initialize the support relations. See Section 2.1.5.2.

- *Joint inference:* optimize the objects, layout and hidden human context in the 3D scene in an analysis-by-synthesis fashion by maximizing the posterior probability of the **pg**. Details are provided in Section 2.1.5.3.

2.1.5.1 Room Geometry Estimation

Although recent approaches [ISS17a, LBM17, ZLY17] are capable of generating a relatively robust prediction of the 2D layout using CNN features, 3D layout estimations are still inaccurate due to its sensitivity to noises of camera parameter estimation. To address the inconsistency between the 2D layout estimation and camera parameter estimation, we design a deep neural network to estimate the 2D layout, and use the layout heatmap to estimate the camera parameter.

2D Layout Estimation: Similar to [LBM17], we represent the 2D layout with its room layout type and keypoint positions. It optimizes the cost function that incorporates the Euclidean loss for layout heatmap regression and the cross-entropy loss for room type estimation. Instead of adopting the SegNet [BKC17] as a basic network module, we use the "stacked hourglass" network [NYD16] as our basic network architecture. It addresses the keypoint estimation problem very well and achieves the state-of-the-art performance in 2D layout estimation. It addresses the keypoint estimation problem very well and achieves the state-of-the-art performance in 2D layout estimation.

Camera Parameter: Traditional geometry-based method [HHF09] computes the camera parameter by estimating the vanishing points from the observed image, which is sensitive to local noises and thus unstable in many indoor scenes. Inspired by [WXL16], we propose a learning-based method that uses the keypoints heatmaps to predict the camera parameters, *i.e.*, focal length, the pitch, yaw and roll of the camera. Since ϕ is incorporated into the evaluation of room layout, we estimate the remaining three variables by stacking four FC layers (1024-128-16-3) on the keypoint heatmaps. Similarly, we estimate the scene category F_c by stacking three FC layers (512-16-1) on the keypoint heatmaps.

3D Layout Initialization: Using the estimated 2D layout and camera parameters, we construct a 3D room as a cuboid by projecting the four corners of the 2D layout to 3D. We assume the camera is 1.2 meters high, and the ceiling is 3.0 meters high. For the convenience of stochastic inference, we translate and rotate the room layout so that one of the visible room corners is at the origin of the world coordinate system. In our method, camera parameter estimation and 2D layout estimation share the same low-level features, which could largely avoid the inaccuracy that local noise brings to the camera parameter, thus improving the performance of 3D layout estimation.

2.1.5.2 Objects Initialization

We fine-tune the Soft-NMS [BSC17] to detect 2D bounding boxes as our 2D object proposals. To initialize the 3D objects, we retrieve the most similar CAD models and initialize their 3D poses, sizes, and positions.

Model Retrieval: We consider all the models in the ShapeNetSem repository [CFG15, SCH15] and render each model from $16 \times 3 = 48$ viewpoints consisting of uniformly sampled 16 azimuth and 3 elevation angles. We extract 7×7 features from the ROI-pooling layer of the fine-tuned Soft-NMS of images in the detected bounding boxes and candidate rendered images. By ranking the cosine distance between each detected object feature and rendered image feature in the same object category, we obtain the top-10 CAD models with corresponding poses.

Geometric Attributes Estimation: The geometric attributes of an object are represented by a 9D vector of 3D pose, position, and size, where 3D poses are initialized from the retrieval procedure. It is hard to recover the original 3D position and size if only given a 2D object bounding box since i) a 2D point can be projected from an infinite number of 3D points, and ii) the bounding box corners are not usually on the object. Prior work roughly project 2D points to 3D and recover the 3D position and size by assuming that all the objects are on the floor. Such approach shows limitations in complex scenarios.

Without making the above assumption, we estimate the depth of each object by computing the average depth value of the pixels that are in both the detection bounding box and the segmentation map. Then we compute its 3D position using the depth value. This is more robust since per-pixel depth estimation error is within a small range even in cluttered scenes. To avoid the alignment problem of 2D bounding boxes, we initialize the object size by sampling object sizes from a learned distribution and choose the one with the largest probability.

Supporting Relation Estimation: For each object $v_i \in V_f^o$, we determine the supporting object by choosing the object or layout v_j^* node with minimal supporting energy:

$$v_j^* = \arg\min_{v_j} \mathcal{K}_o(v_i, v_j) + \mathcal{K}_h(v_i, v_j) - \lambda_s \log p(v_i, v_j | V_f^l, V_f^o), \quad v_j \in (V_f^l, V_f^o). \tag{2.11}$$

2.1.5.3 Joint Inference

Given an image I, we first estimate the room geometry, object attributes and relations as described in the above two subsections. The goal of joint inference is to (1) optimize the objects and layout; (2) group objects, assign activity label and imagine human pose in each activity group; and (3) optimize the objects, layout and human pose iteratively.

In each step, we use distinct MCMC processes. Specifically, to travel through non-differentiable solution spaces, we design Markov Chain dynamics $\{q_1^o, q_2^o, q_3^o\}$ for objects, $\{q_1^l, q_2^l\}$ for layout, and $\{q_1^h, q_2^h, q_3^h\}$ for human pose. Specifically,

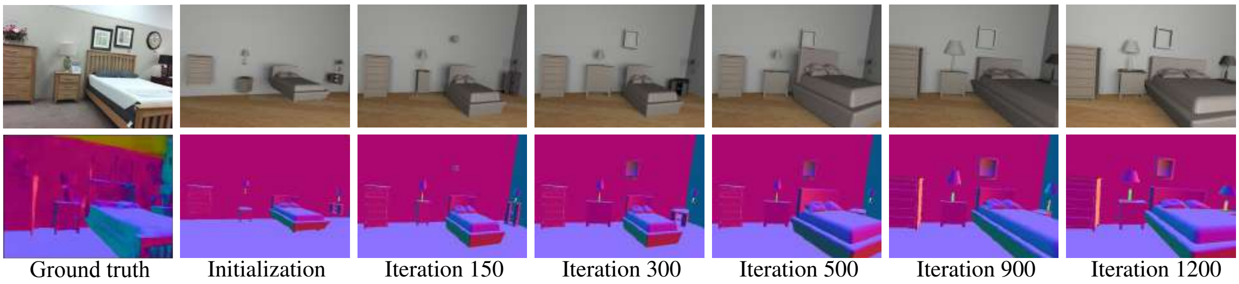


Figure 2.4: The process of joint inference of objects and layout by MCMC with simulated annealing. The first row contains rendered RGB images (for visualization), the second row contains rendered surface normal maps. During the joint inference, objects and layout are optimized iteratively.

- *Object Dynamics:* Dynamics q_1^o adjusts the position of a random object, which translates the object center in one of the three coordinate directions. Instead of translating the object center and changing the object size directly, Dynamics q_2^o translates one of the six faces of the cuboid to generate a smoother diffusion. Dynamics q_3^o proposes rotation of the object with a specified angle. Each dynamic can diffuse in two directions, *e.g.*, each object can translate in direction of '$+x$' and '$-x$', or rotate in direction of clockwise and counterclockwise. By computing the local gradient of $P(\mathbf{pg}|I)$, the dynamics propose to move following the direction of the gradient with a proposal probability of 0.8, or the inverse direction of the gradient with proposal probability of 0.2.

- *Layout Dynamics:* Dynamics q_1^l translates the faces of the layout, which also optimizes the predefined camera height while translating the floor. Dynamics q_2^l proposes to rotate the layout.

- *Human pose Dynamics* q_1^h, q_2^h and q_3^h are designed to translate, rotate and scale the human pose, respectively.

Given a current $\mathbf{pg}$, each dynamic will propose a new $\mathbf{pg}'$ according to a proposal probability $p(\mathbf{pg}'|\mathbf{pg}, I)$. The proposal is accepted according to an acceptance probability $\alpha(\mathbf{pg} \rightarrow \mathbf{pg}')$ defined by the Metropolis-Hasting algorithm [Has70]:

$$\alpha(\mathbf{pg} \rightarrow \mathbf{pg}') = \min(1, \frac{p(\mathbf{pg}|\mathbf{pg}', I)p(\mathbf{pg}'|I)}{p(\mathbf{pg}'|\mathbf{pg}, I)p(\mathbf{pg}|I)}). \tag{2.12}$$

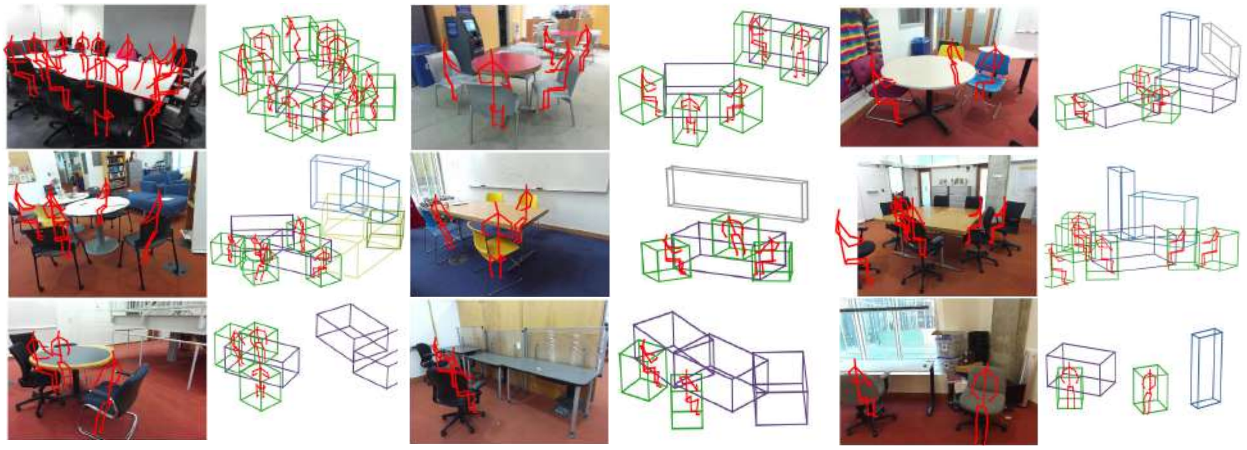


Figure 2.5: Sampled human poses in various indoor scenes. Objects in multiple activity groups have multiple poses. We visualize the pose with the highest likelihood.

The above three inference steps are summarized in Algorithm 1. Figure 2.4 shows the process of step (1).

In step (2), we design an algorithm to group objects and assign activity labels. For each type of activity, there is a major object category which has the highest occurrence frequency (*i.e.*, chair in activity 'reading'). Intuitively, the correspondence between objects and activities should be n-to-n but not n-to-one, which means each object can belong to several activity groups. In order to find out all possible activity groups, for each type of activity, we define an activity group around each major object and incorporate nearby objects (within a distance threshold) with prior larger than 0. For each activity group $v_i \in V_f^a$, the pose of the imagined human is estimated by maximizing the likelihood $p(v_i|\tilde{\mu}_i)$, which is equivalent to minimize the grouping energy $\mathcal{E}_{grp}(\tilde{\mu}_i|v_i)$ defined in Equation (2.8),

$$y_i^*, m_i^*, t_i^*, r_i^*, s_i^* = \underset{y_i, m_i, t_i, r_i, s_i}{\arg\min}\ \mathcal{E}_{grp}(\tilde{\mu}_i|v_i), \tag{2.13}$$

Figure 2.5 shows the results of sampled human poses in various indoor scenes.

Algorithm 1 Joint inference algorithm.

```
Given Image $I$, initialized parse graph $\mathbf{pg}_{\text{init}}$
procedure STEP1($V_g^o, V_g^l$)                                    ▷ Inference without hidden human context
    for different temperatures do                ▷ Different temperatures are adopted in simulated annealing
        for $\gamma_1$ iterations do
            randomly choose layout, apply layout dynamics to optimize layout $V_g^l$
        for each object $v_i \in V_g^o$ do
            for $\gamma_2$ iterations do
                randomly apply object dynamics to optimize object $v_i$
procedure STEP2($V_f^a, \{\tilde{\mu}\}$)                                 ▷ Inference of hidden human context
    group objects and assign activity labels (see last paragraph in Section 2.1.5.3)
    for each activity group $v_i \in V_f^a$ do
        repeat
            randomly apply human pose dynamics to optimize $\tilde{\mu}_i$
        until $\mathcal{E}(\tilde{\mu}_i|v_i)$ converges             ▷ Maximizing grouping energy in Equation (2.13)
procedure STEP3($V_g^o, V_g^l, \{\tilde{\mu}\}$)                        ▷ Iterative inference of whole parse graph
    for different temperatures do
        for $\gamma_3$ iterations do
            randomly choose layout, objects or human pose
            apply random dynamics to minimize $P(\mathbf{pg}|I)$
Return $\mathbf{pg}_{\text{optimized}}$
```

2.1.6 Experiments

We use the SUN RGB-D dataset [SLX15] to evaluate our approach on 3D scene parsing and reconstruction. It has 47 scene categories with high-quality 3D bounding box annotations for most of the 3D objects, as well as 3D room corners for most of the scenes. It also provides benchmarks for various 3D scene understanding tasks. The dataset has 5050 testing images and 10,355 images in total. Although it provides RGB-D data, we only use the RGB images as the input for training and testing. The point cloud is further generated using ground-truth depth images. Figure 2.6 shows some qualitative parsing results (top 20%).

We evaluate our method on three major tasks: i) 3D layout estimation, ii) 3D object detection, and iii) holistic scene understanding with all the 5050 testing images of SUN RGB-D across all scene categories. The capability of generalization to all the scene categories is difficult for most of the conventional methods due to the inaccuracy of camera parameter estimation and severe sensitivity to the occlusions in cluttered scenes. In this work, we alleviate it by using the proposed learning-based camera parameter estimation and a novel method to initialize the geometric attributes. In addition, we also achieve the state-of-the-art results in 2D layout estimation on LSUN dataset [ZYS15] and Hedau dataset [HHF09].

Implementation Details: For 2D object detection, we fine-tune the detector (Soft-NMS [BSC17]) on SUN RGB-D with 30 object categories. Since [ZYS15] and [HHF09] have no ground-truth of the camera parameter, we train the layout estimation module using [ZYS15] as the initial model, followed by using the feature of the heatmap to further train camera parameter and scene category on SUN RGB-D. During the initialization and joint inference process, we use the depth estimation model as described in [LRB16], surface normal estimation in [ZSY17a], and semantic segmentation in [LMS17]. These models are trained on the training set of the SUN RGB-D or NYU v2 dataset [SHK12] (included in the SUN RGB-D). Here, we further incorporate human context inference on the subset of offices and skip it on other scenes. During joint inference, we fix the scene category, object categories and support relations to reduce the computational complexity. We used OpenGL [SG09] to render the depth, surface normal and segmentation map. Rendering each map takes about 1 second. On average, our joint inference process takes about one hour for each image using a single CPU core.

Evaluation of 3D Layout Estimation: The 3D room layout is optimized using the proposed joint inference. We compare the estimation by our method (with and without joint inference) with 3DGP [CCP13]. Following the evaluation protocal defined in [SLX15], we calculated the average intersection-over-union (IoU) between the free space from the ground truth and the free space estimated by our method. Table 2.1 shows our method outperforms 3DGP by a large margin. We also improve the performance by 8.2% after jointly inferring the

Table 2.1: Quantitative comparisons of 3D scene parsing and reconstruction on SUN RGB-D dataset.

Method	# of image	3D Layout Estimation	Holistic Scene Understanding			
		IoU	P_g	R_g	R_r	IoU
3DGP [CCP13]	5050	19.2	2.1	0.7	0.6	13.9
Ours (init.)	5050	46.7	25.9	15.5	12.2	36.6
Ours (joint.)	5050	**54.9**	**37.7**	**23.0**	**18.3**	**40.7**
3DGP [CCP13]	749	33.4	5.3	2.7	2.1	34.2
IM2CAD [ISS17a]	484	62.6	-	-	-	49.0
Ours (init.)	749	64.2	29.7	17.3	14.4	47.1
Ours (joint.)	749	**66.4**	**40.5**	**26.8**	**21.7**	**52.1**

Table 2.2: Comparisons of 3D object detection on SUN RGB-D dataset.

Method	bed	chair	sofa	table	desk	toilet	fridge	sink	bathtub	bookshelf	counter	door	dresser	lamp	tv	mAP
[CCP13]	5.62	2.31	3.24	1.23	-	-	-	-	-	-	-	-	-	-	-	-
Ours (init.)	45.55	5.91	23.64	4.20	2.50	1.91	14.00	2.12	0.55	2.16	0.34	0.01	5.69	1.12	0.62	7.35
Ours (joint.)	**58.29**	**13.56**	**28.37**	**12.12**	**4.79**	**16.50**	**15.18**	**2.18**	**2.84**	**7.04**	**1.6**	**1.56**	**13.71**	**2.41**	**1.04**	**12.07**

objects and layout, demonstrating the usefulness of integrating the joint inference process.

Since IM2CAD [ISS17a] manually selected 484 images (from 794 living rooms and bedrooms) from the dataset without releasing the image list, we compare our method with IM2CAD on the entire set of living rooms and bedrooms. Table 2.1 shows our method surpasses IM2CAD, especially after incorporating the joint inference process.

Evaluation of 3D Object Detection: We evaluate our 3D object detection results using the metrics defined in [SLX15]. We compute the mean average precision (mAP) using the 3D IoU between the predicted and ground truth 3D bounding boxes. In the absence of depth, we adjust threshold IoU from 0.25 (evaluation setting with depth as the input) to 0.15 and report our results in Table 2.2. Only 15 out of 30 object categories are presented here due to the limited size. The results indicate our method not only exceeds the detection score by a significant margin but also makes it possible to evaluate the entire object categories. Note that although IM2CAD also evaluates the detection, they use the metric related to a specified distance threshold. Here, we also compare with IM2CAD on the subset with this

Table 2.3: Ablative analysis of our method on SUN RGB-D dataset. We evaluate on holistic scene understanding under different settings. We denote support relation as C_1, physical constraint as C_2 and human imagination as C_3. Similarly, we denote the setting of only optimizing the layout during inference as S_4, only optimizing the objects during inference as S_5.

Setting	w/o C_1	w/o C_2	w/o C_3	w/o (C_1, C_2, C_3)	S_4	S_5	All
IoU	42.3	41.3	43.8	38.4	39.4	36.3	**44.7**
P_g	29.3	23.5	32.1	19.4	14.9	28.4	**34.4**
R_g	17.4	15.6	20.4	12.4	11.2	19.7	**24.1**
R_r	14.1	10.5	16.5	8.7	8.6	13.3	**19.2**

special metric rather than IoU threshold. We are able to obtain an mAP of 80.2%, higher than an mAP of 74.6% reported in the IM2CAD.

Evaluation of Holistic Scene Understanding: We estimate the whole 3D scene including objects and room layout. Using the metrics proposed in [SLX15], we evaluate the geometric precision P_g, geometric recall R_g, semantic recall R_r with the IoU threshold set to 0.15. We also evaluate the IoU between the free space (3D voxels inside the room polygon but outside any object bounding box) of ground truth and estimation. Table 2.1 shows that we improve the previous approaches in a large scale. Moreover, we improve the initialization result by 12.2% on geometric precision, 7.5% on geometric recall, 6.1% on semantic recall and 4.1% on free space estimation. The improvement of total scene understanding directly reflects that our joint inference process could largely improve the performance of each task. Using the same setting with 3D layout estimation, we compare with IM2CAD [ISS17a] and improve the free space IoU by 3.1%.

Ablative Analysis: The proposed HSG incorporates several key components including supporting relations, physics constraints and latent human contextual relations. To analyze how each component would influence the final results, as well as how much the joint inference process would benefit each task, we conduct the ablative analysis on holistic scene understanding under different settings, through turning on and off certain components or skipping certain steps during joint inference. The experiments are tested on the subset of offices where we incorporate the latent human context. Table 2.3 summarizes the results. Among all the energy terms we incorporate, physical constraints influence the performance the most, which demonstrates the importance of the physical common sense during inference. It also reflects the efficiency of joint inference since the performances decline by a large margin without the iterative joint inference.

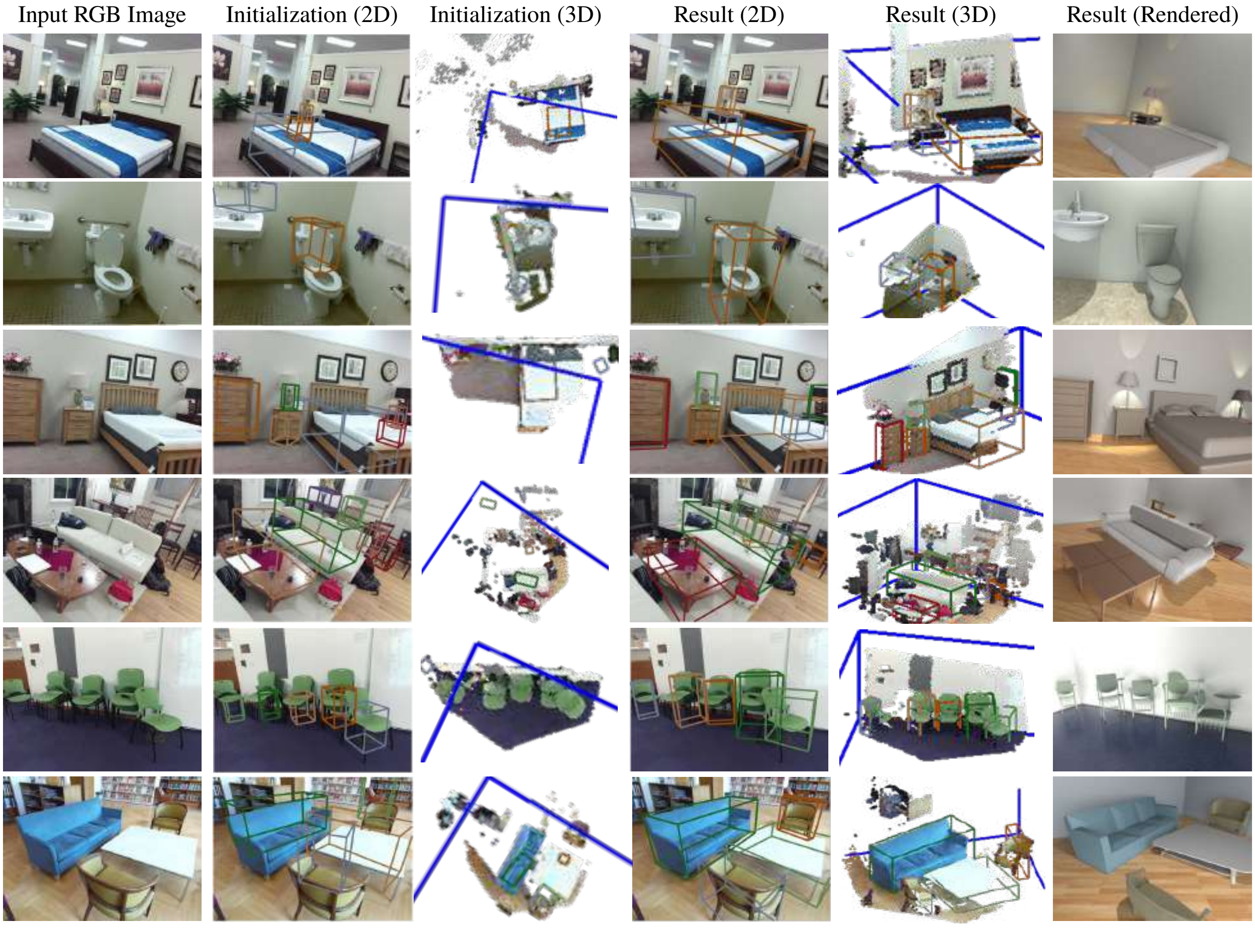


Figure 2.6: Qualitative results of our method on SUN RGB-D dataset. The joint inference significantly improves the performance over individual modules.

2.1.7 Conclusion

We present an analysis-by-synthesis framework to recover the 3D structure of an indoor scene from a single RGB image using a stochastic grammar model integrated with latent human context, geometry and physics. We demonstrate the effectiveness of our algorithm in three perspectives: i) the joint inference algorithm significantly improves results in various tasks, ii) our method outperforms other methods in 3D layout estimation, 3D object detection, and holistic scene understanding, and iii) ablative analysis shows each of module plays an important role in the whole framework. In general, we believe this will be a step towards a unifying framework for the holistic 3D scene understanding.

2.1.8 Appendix: Additional Results

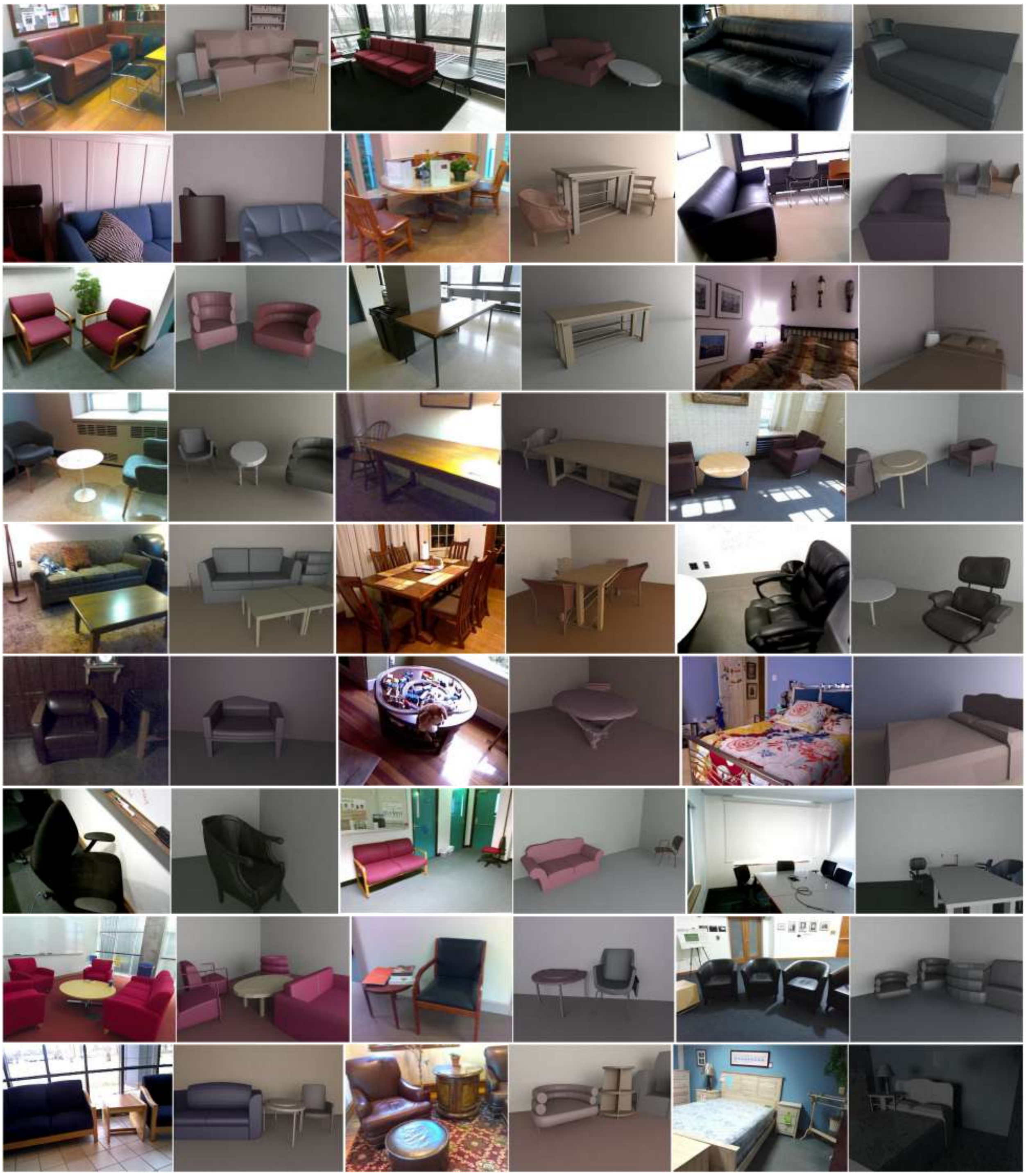


Figure 2.7: Additional qualitative parsing results.

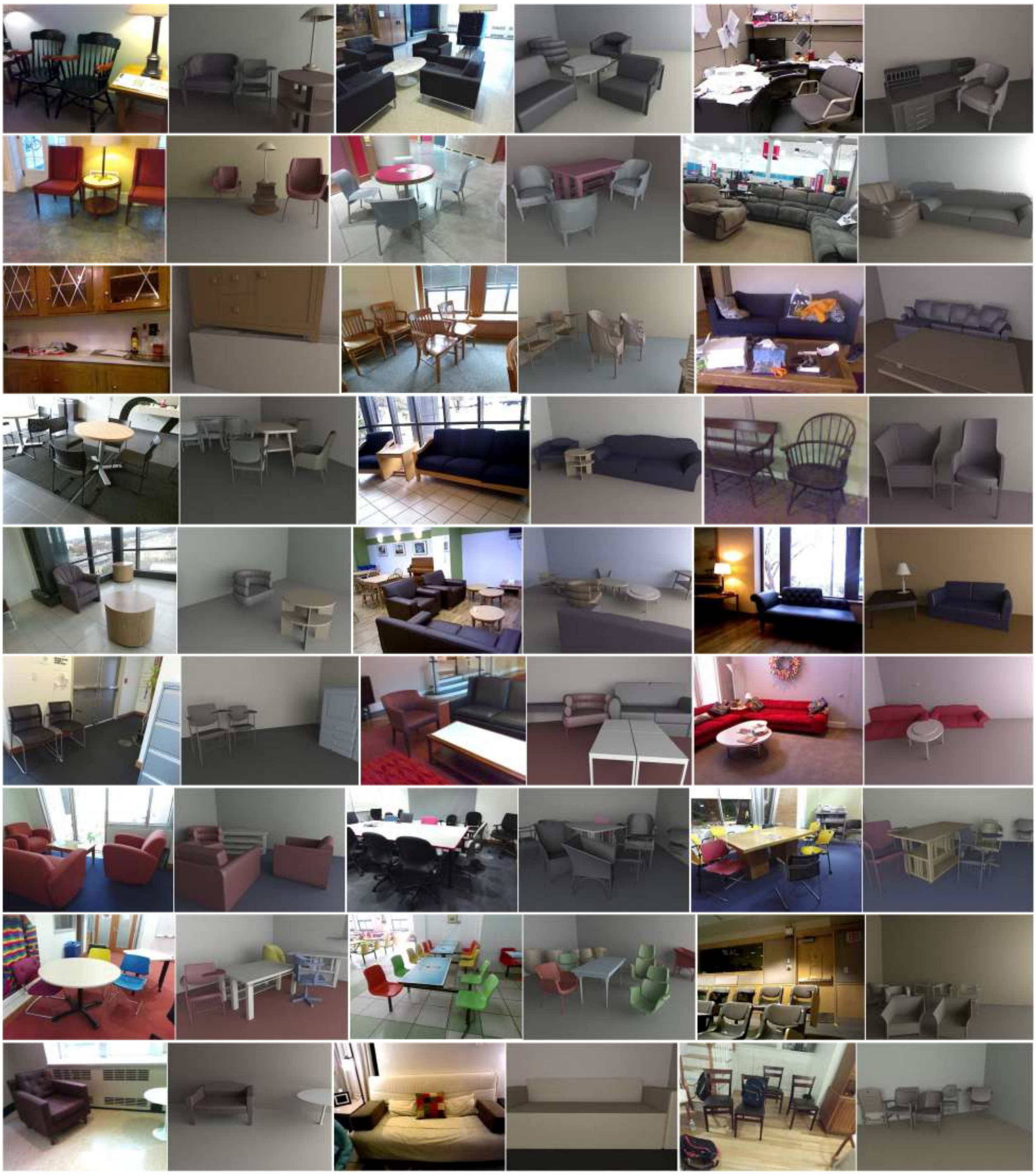


Figure 2.8: Additional qualitative parsing results.

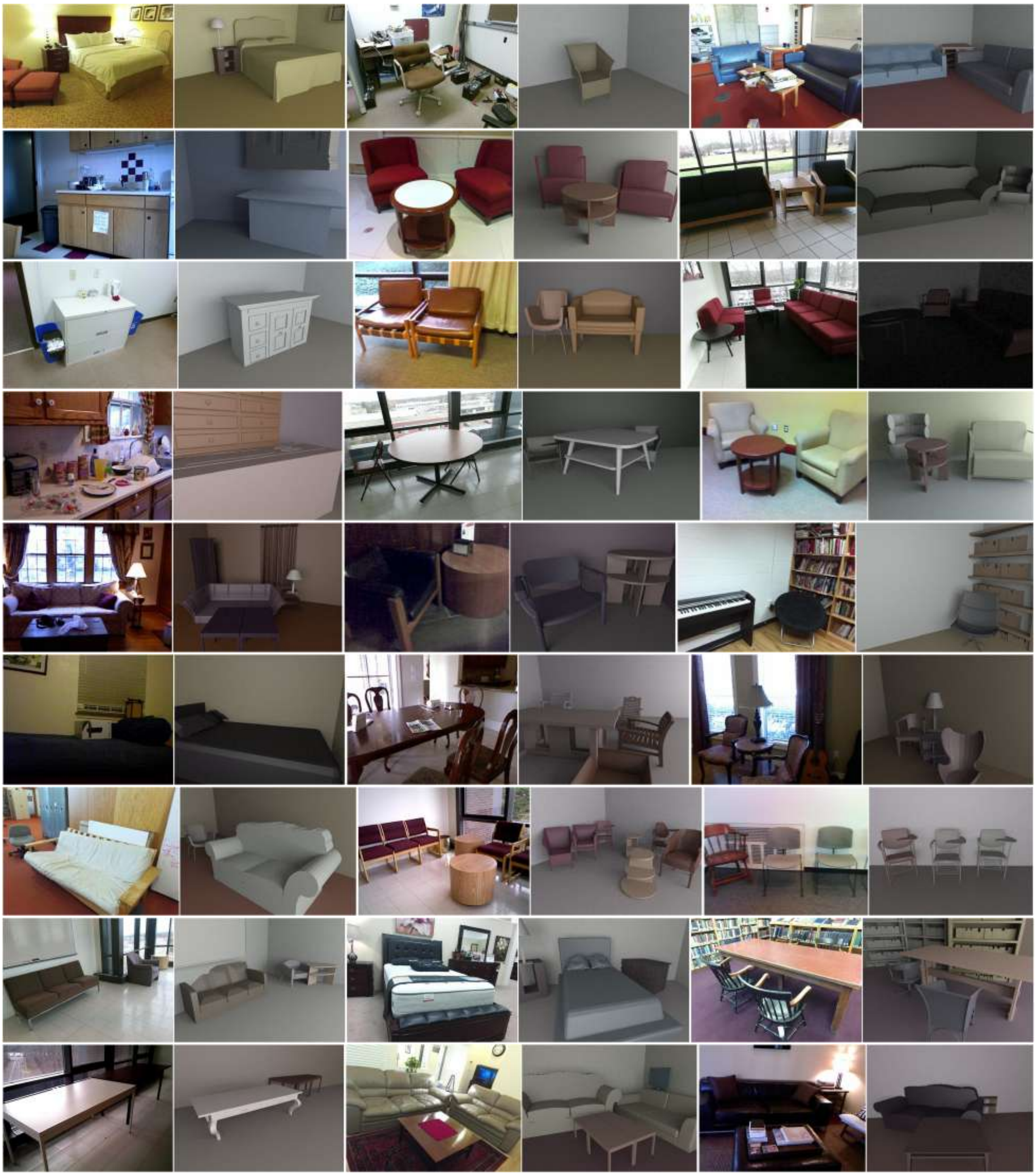


Figure 2.9: Additional qualitative parsing results.

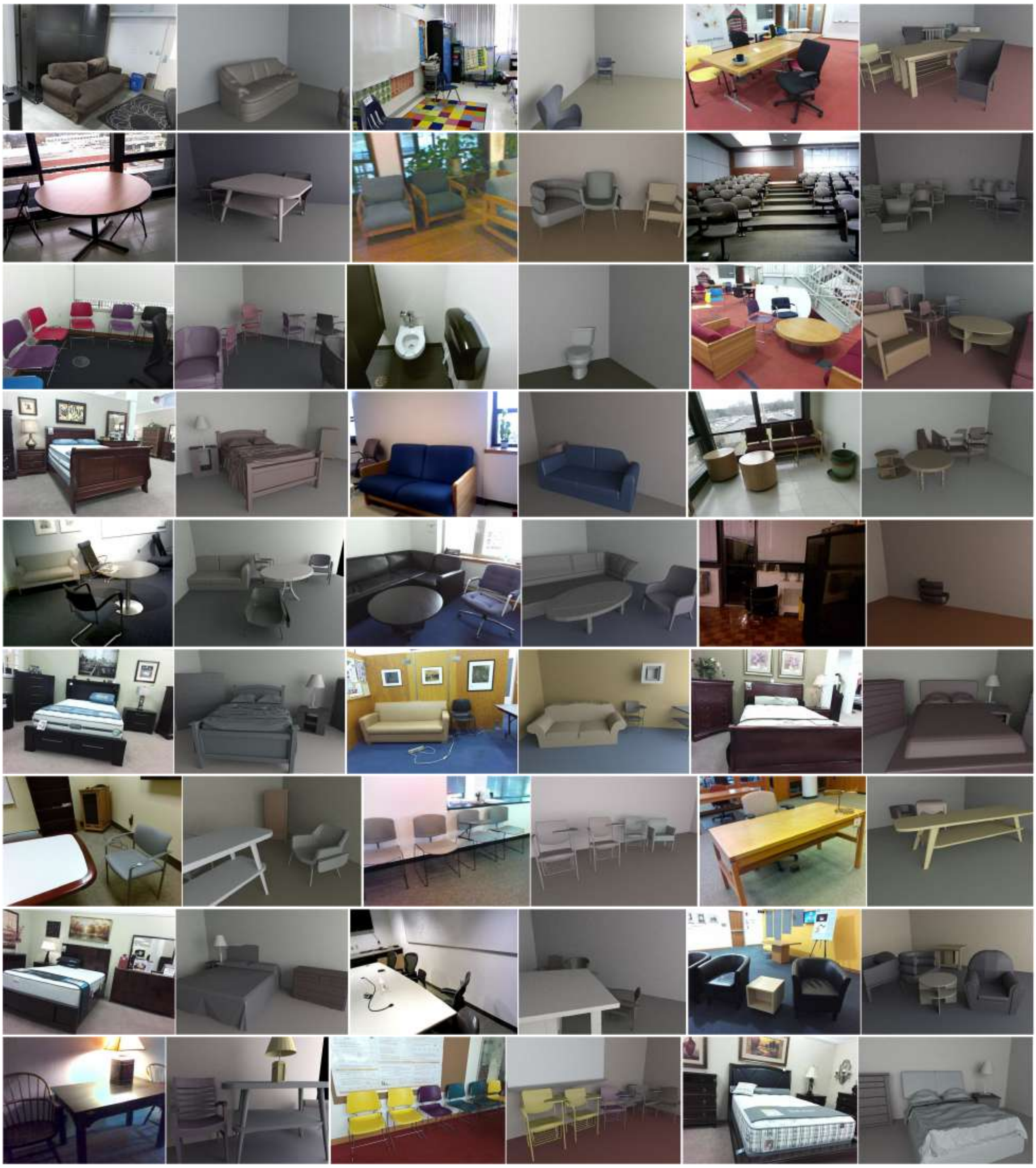


Figure 2.10: Additional qualitative parsing results.

2.2 Unifying 3D Object, Layout, and Camera Pose Estimation

Holistic 3D indoor scene understanding refers to jointly recovering the i) object bounding boxes, ii) room layout, and iii) camera pose, all in 3D. The existing methods either are ineffective or only tackle the problem partially.

In this section, we propose an end-to-end model that *simultaneously* solves all three tasks in *real-time* given only a single RGB image. The essence of the proposed method is to improve the prediction by i) *parametrizing* the targets (*e.g.*, 3D boxes) instead of directly estimating the targets, and ii) *cooperative training* across different modules in contrast to training these modules individually. Specifically, we parametrize the 3D object bounding boxes by the predictions from several modules, *i.e.*, 3D camera pose and object attributes. The proposed method provides two major advantages: i) The parametrization helps maintain the consistency between the 2D image and the 3D world, thus largely reducing the prediction variances in 3D coordinates. ii) Constraints can be imposed on the parametrization to train different modules simultaneously. We call these constraints ”cooperative losses” as they enable the joint training and inference. We employ three cooperative losses for 3D bounding boxes, 2D projections, and physical constraints to estimate a *geometrically consistent* and *physically plausible* 3D scene. Experiments on the SUN RGB-D dataset shows that the proposed method significantly outperforms prior approaches on 3D object detection, 3D layout estimation, 3D camera pose estimation, and holistic scene understanding.

2.2.1 Introduction

Holistic 3D scene understanding from a single RGB image is a fundamental yet challenging computer vision problem, while humans are capable of performing such tasks effortlessly within 200 ms [Pot75, Pot76, SO94, TFM96]. The primary difficulty of the holistic 3D scene understanding lies in the vast, but ambiguous 3D information attempted to recover from a single RGB image. Such estimation includes three essential tasks:

- The estimation of the 3D camera pose that captures the image. This component helps

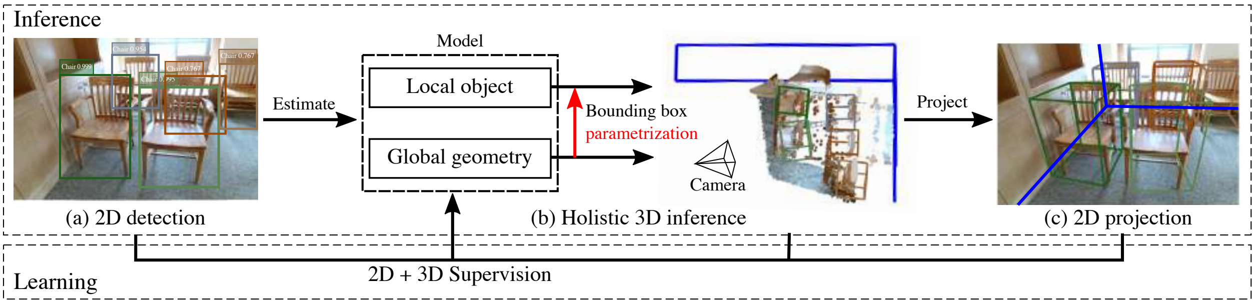


Figure 2.11: Overview of the proposed framework for cooperative holistic scene understanding. (a) We first detect 2D objects and generate their bounding boxes, given a single RGB image as the input, from which (b) we can estimate 3D object bounding boxes, 3D room layout, and 3D camera pose. The blue bounding box is the estimated 3D room layout. (c) We project 3D objects to the image plane with the learned camera pose, forcing the projection from the 3D estimation to be consistent with 2D estimation.

to maintain the *consistency* between the 2D image and the 3D world.

- The estimation of the 3D room layout. Combining with the estimated 3D camera pose, it recovers a *global* geometry.
- The estimation of the 3D bounding boxes for each object in the scene, recovering the *local* details.

Most current methods either are inefficient or only tackle the problem partially. Specifically,

- Traditional methods [GHK10, ZZ11, ZZ13, CCP13, SFP13, ZST14, ISS17a, HQZ18] apply sampling or optimization methods to infer the geometry and semantics of indoor scenes. However, those methods are computationally expensive; it usually takes a long time to converge and could be easily trapped in an unsatisfactory local minimum, especially for cluttered indoor environments. Thus both stability and scalability become issues.
- Recently, researchers attempt to tackle this problem using deep learning. The most straightforward way is to directly predict the desired targets (*e.g.*, 3D room layouts or 3D bounding boxes) by training the individual modules separately with isolated

losses for each module. Thereby, the prior work [MAF17, LBM17, KMT17, KLR18, ZCS18, LYC18] only focuses on the individual tasks or learn these tasks separately rather than jointly inferring all three tasks, or only considers the inherent relations without explicitly modeling the connections among them [TGF18].

- Another stream of approach takes both an RGB-D image and the camera pose as the input [LFU13, SX14, SX16, SYZ17a, DL17, ZLH17, QLW18, LG17, ZBK17], which provides sufficient geometric information from the depth images, thereby relying less on the consistency among different modules.

In this work, we aim to address the missing piece in the literature: to recover a *geometrically consistent* and *physically plausible* 3D scene and jointly solve all three tasks in an *efficient* and *cooperative* way, only from a single RGB image. Specifically, we tackle three important problems:

1. *2D-3D consistency* A good solution to the aforementioned three tasks should maintain a high consistency between the 2D image plane and the 3D world coordinate. How should we design a method to achieve such consistency?

2. *Cooperation* Psychological studies have shown that our biologic perception system is extremely good at rapid scene understanding [SO94], particularly utilizing the fusion of different visual cues [LMJ95, Jac02]. Such findings support the necessities of cooperatively solving all the holistic scene tasks together. Can we devise an algorithm such that it can *cooperatively* solve these tasks, making different modules reinforce each other?

3. *Physically Plausible* As humans, we excel in inferring the physical attributes and dynamics [KHL17]. Such a deep understanding of the physical environment is imperative, especially for an interactive agent (*e.g.*, a robot) to navigate the environment or collaborate with a human agent. How can the model estimate a 3D scene in a physically plausible fashion, or at least have some sense of physics?

To address these issues, we propose a novel parametrization of the 3D bounding box as well as a set of cooperative losses. Specifically, we parametrize the 3D boxes by the predicted camera pose and object attributes from individual modules. Hence, we can construct the 3D boxes starting from the 2D box centers to maintain a 2D-3D consistency, rather than predicting 3D coordinates directly or assuming the camera pose is given, which loses the 2D-3D consistency.

Cooperative losses are further imposed on the parametrization in addition to the direct losses to enable the joint training of all the individual modules. Specifically, we employ three cooperative losses on the parametrization to constrain the 3D bounding boxes, projected 2D bounding boxes, and physical plausibility, respectively:

- The 3D bounding box loss encourages accurate 3D estimation.
- The differentiable 2D projection loss measures the consistency between 3D and 2D bounding boxes, which permits our networks to learn the 3D structures with only 2D annotations (*i.e.*, no 3D annotations are required). In fact, we can directly supervise the learning process with 2D objects annotations using the common sense of the object sizes.
- The physical plausibility loss penalizes the intersection between the reconstructed 3D object boxes and the 3D room layout, which prompts the networks to yield a physically plausible estimation.

Figure 2.11 shows the proposed framework for cooperative holistic scene understanding. Our method starts with the detection of 2D object bounding boxes from a single RGB image. Two branches of convolutional neural networks are employed to learn the 3D scene from both the image and 2D boxes: i) The *global geometry network* (GGN) learns the global geometry of the scene, predicting both the 3D room layout and the camera pose. ii) The *local object network* (LON) learns the object attributes, estimating the object pose, size, distance between the 3D box center and camera center, and the 2D offset from the 2D box center to the projected 3D box center on the image plane. The details are discussed in Section 2.2.2.

By combining the camera pose from the GGN and object attributes from the LON, we can parametrize 3D bounding boxes, which grants jointly learning of both GGN and LON with 2D and 3D supervisions.

Another benefit of the proposed parametrization is improving the training stability by reducing the variance of the 3D boxes prediction, due to that i) the estimated 2D offset has relatively low variance, and ii) we adopt a hybrid of classification and regression method to estimate the variables of large variances, inspired by [RHG15, MAF17, QLW18].

We evaluate our method on SUN RGB-D Dataset [SLX15]. The proposed method outperforms previous methods on four tasks, including 3D layout estimation, 3D object detection, 3D camera pose estimation, and holistic scene understanding. Our experiments demonstrate that a cooperative method performing holistic scene understanding tasks can significantly outperform existing methods tackling each task in isolation, further indicating the necessity of joint training.

Our contributions are four-fold. i) We formulate an end-to-end model for 3D holistic scene understanding tasks. The essence of the proposed model is to cooperatively estimate 3D room layout, 3D camera pose, and 3D object bounding boxes. ii) We propose a novel parametrization of the 3D bounding boxes and integrate physical constraint, enabling the cooperative training of these tasks. iii) We bridge the gap between the 2D image plane and the 3D world by introducing a differentiable objective function between the 2D and 3D bounding boxes. iv) Our method significantly outperforms the state-of-the-art methods and runs in real-time.

2.2.2 Method

In this section, we describe the parametrization of the 3D bounding boxes and the neural networks designed for the 3D holistic scene understanding. The proposed model consists of two networks, shown in Figure 2.12: a *global geometric network* (GGN) that estimates the 3D room layout and camera pose, and a *local object network* (LON) that infers the attributes of each object. Based on these two networks, we further formulate differentiable

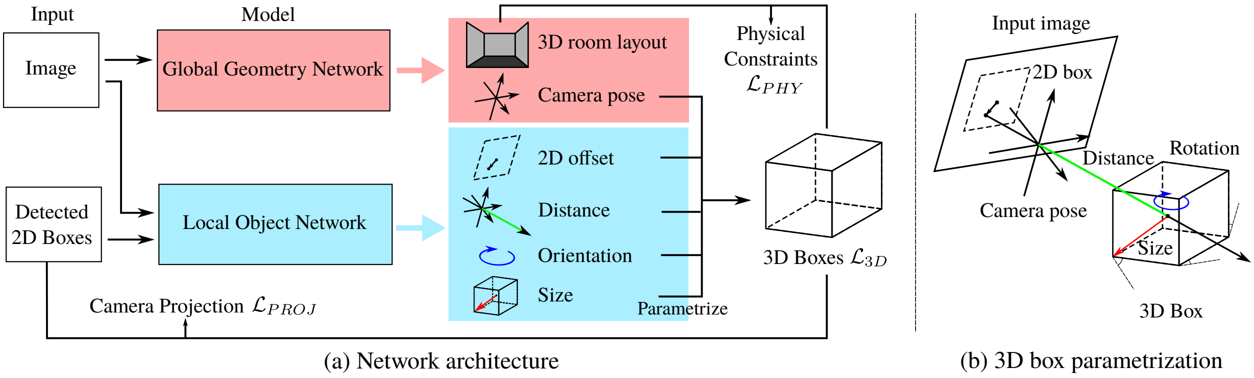


Figure 2.12: Illustration of (a) network architecture and (b) parametrization of 3D object bounding box.

loss functions to train the two networks cooperatively.

2.2.2.1 Parametrization

3D Objects We use the 3D bounding box $X^W \in \mathbb{R}^{3\times 8}$ as the representation of the estimated 3D object in the world coordinate. The 3D bounding box is described by its 3D center $C^W \in \mathbb{R}^3$, size $S^W \in \mathbb{R}^3$, and orientation $R(\theta^W) \in \mathbb{R}^{3\times 3}$: $X^W = h(C^W, R(\theta^W), S)$, where θ is the heading angle along the up-axis, and $h(\cdot)$ is the function that composes the 3D bounding box.

Without any depth information, estimating 3D object center C^W directly from the 2D image may result in a large variance of the 3D bounding box estimation. To alleviate this issue and bridge the gap between 2D and 3D object bounding boxes, we parametrize the 3D center C^W by its corresponding 2D bounding box center $C^I \in \mathbb{R}^2$ on the image plane, distance D between the camera center and the 3D object center, the camera intrinsic parameter $K \in \mathbb{R}^{3\times 3}$, and the camera extrinsic parameters $R(\phi, \psi) \in \mathbb{R}^{3\times 3}$ and $T \in \mathbb{R}^3$, where ϕ and ψ are the camera rotation angles. As illustrated in Figure 2.12(b), since each 2D bounding box and its corresponding 3D bounding box are both manually annotated, there is always an offset $\delta^I \in \mathbb{R}^2$ between the 2D box center and the projection of 3D box center. Therefore, the 3D object center C^W can be computed as

$$C^W = T + DR(\phi, \psi)^{-1} \frac{K^{-1}\left[C^I + \delta^I, 1\right]^T}{\left\| K^{-1}\left[C^I + \delta^I, 1\right]^T \right\|}. \tag{2.14}$$

Since T becomes $\vec{0}$ when the data is captured from the first-person view, the above equation could be written as $C^W = p(C^I, \delta^I, D, \phi, \psi, K)$, where p is a differentiable projection function.

In this way, the parametrization of the 3D object bounding box unites the 3D object center C^W and 2D object center C^I, which helps maintain the 2D-3D consistency and reduces the variance of the 3D bounding box estimation. Moreover, it integrates both object attributes and camera pose, promoting the cooperative training of the two networks.

3D Room Layout Similar to 3D objects, we parametrize 3D room layout in the world coordinate as a 3D bounding box $X^L \in \mathbb{R}^{3\times 8}$, which is represented by its 3D center $C^L \in \mathbb{R}^3$, size $S^L \in \mathbb{R}^3$, and orientation $R(\theta^L) \in \mathbb{R}^{3\times 3}$, where θ^L is the rotation angle. In this work, we estimate the room layout center by predicting the offset from the pre-computed average layout center.

2.2.2.2 Direct Estimations

As shown in Figure 2.12(a), the *global geometry network* (GGN) takes a single RGB image as the input, and predicts both 3D room layout and 3D camera pose. Such design is driven by the fact that the estimations of both the 3D room layout and 3D camera pose rely on low-level global geometric features. Specifically, GGN estimates the center C^L, size S^L, and the heading angle θ^L of the 3D room layout, as well as the two rotation angles ϕ and ψ for predicting the camera pose.

Meanwhile, the *local object network* (LON) takes 2D image patches as the input. For each object, LON estimates object attributes including distance D, size S^W, heading angle θ^W, and the 2D offsets δ^I between the 2D box center and the projection of the 3D box center.

Direct estimations are supervised by two losses $\mathcal{L}_{\text{GGN}}$ and $\mathcal{L}_{\text{LON}}$. Specifically, $\mathcal{L}_{\text{GGN}}$ is defined as

$$\mathcal{L}_{\mathrm{GGN}} = \mathcal{L}_{\phi} + \mathcal{L}_{\psi} + \mathcal{L}_{C^L} + \mathcal{L}_{S^L} + \mathcal{L}_{\theta^L}, \tag{2.15}$$

and $\mathcal{L}_{\mathrm{LON}}$ is defined as

$$\mathcal{L}_{\mathrm{LON}} = \frac{1}{N} \sum_{j=1}^{N} (\mathcal{L}_{D_j} + \mathcal{L}_{\delta_j^I} + \mathcal{L}_{S_j^W} + \mathcal{L}_{\theta_j^W}), \tag{2.16}$$

where N is the number of objects in the scene. In practice, directly regressing objects' attributes (*e.g.*, heading angle) may result in a large error. Inspired by [RHG15, MAF17, QLW18], we adopt a hybrid method of classification and regression to predict the sizes and heading angles. Specifically, we pre-define several size templates or equally split the space into a set of angle bins. Our model first classifies size and heading angles to those pre-defined categories, and then predicts residual errors within each category. For example, in the case of the rotation angle ϕ, we define $\mathcal{L}_{\phi} = \mathcal{L}_{\phi-cls} + \mathcal{L}_{\phi-reg}$. Softmax is used for classification and smooth-L1 (Huber) loss is used for regression.

2.2.2.3 Cooperative Estimations

Psychological experiments have shown that human perception of the scene often relies on global information instead of local details, known as the gist of the scene [Oli05, OT06b]. Furthermore, prior studies have demonstrated that human perceptions on specific tasks involve the cooperation from multiple visual cues, *e.g.*, on depth perception [LMJ95, Jac02]. These crucial observations motivate the idea that the attributes and properties are naturally coupled and tightly bounded, thus should be estimated cooperatively, in which individual component would help to boost each other.

Using the parametrization described in Section 2.2.2.1, we hope to cooperatively optimize GGN and LON, simultaneously estimating 3D camera pose, 3D room layout, and 3D object bounding boxes, in the sense that the two networks enhance each other and cooperate to make the definitive estimation during the learning process. Specifically, we propose three cooperative losses which jointly provide supervisions and fuse 2D/3D information into a physically plausible estimation. Such cooperation improves the estimation accuracy of 3D bounding boxes, maintains the consistency between 2D and 3D, and generates a physically plausible scene. We further elaborate on these three aspects below.

3D Bounding Box Loss As neither GGN or LON is directly optimized for the accuracy of the final estimation of the 3D bounding box, learning directly through GGN and LON is evidently not sufficient, thus requiring additional regularization. Ideally, the estimation of the object attributes and camera pose should be cooperatively optimized, as both contribute to the estimation of the 3D bounding box. To achieve this goal, we propose the 3D bounding box loss with respect to its 8 corners

$$\mathcal{L}_{\text{3D}} = \frac{1}{N}\sum_{j=1}^{N} \left\| h(C_j^W, R(\theta_j), S_j) - X_j^{W*} \right\|_2^2, \tag{2.17}$$

where X^{W*} is the ground truth 3D bounding boxes in the world coordinate. [QLW18] proposes a similar regularization in which the parametrization of 3D bounding boxes is different.

2D Projection Loss In addition to the 3D parametrization of the 3D bounding boxes, we further impose an additional consistency as the 2D projection loss, which maintains the coherence between the 2D bounding boxes in the image plane and the 3D bounding boxes in the world coordinate. Specifically, we formulate the learning objective of the projection from 3D to 2D as

$$\mathcal{L}_{\text{PROJ}} = \frac{1}{N}\sum_{j=1}^{N} \left\| f(X_j^W, R, K) - X_j^{I*} \right\|_2^2, \tag{2.18}$$

where $f(\cdot)$ denotes a differentiable projection function which projects a 3D bounding box to a 2D bounding box, and $X_j^{I*} \in \mathbb{R}^{2\times 4}$ is the 2D object bounding box (either detected or the ground truth).

Physical Loss In the physical world, 3D objects and room layout should not intersect with each other. To produce a physically plausible 3D estimation of a scene, we integrate the physical loss that penalizes the physical violations between 3D objects and 3D room layout

$$\mathcal{L}_{\text{PHY}} = \frac{1}{N} \sum_{j=1}^{N} \left(\text{ReLU}(\text{Max}(X_j^W) - \text{Max}(X^L)) + \text{ReLU}(\text{Min}(X^L) - \text{Min}(X_j^W)) \right), \tag{2.19}$$

where ReLU is the activate function, Max$(\cdot)$ / Min$(\cdot)$ takes a 3D bounding box as the input and outputs the max/min value along three world axes. By adding the physical constraint loss, the proposed model connects the 3D environments and the 3D objects, resulting in a more natural estimation of both 3D objects and 3D room layout.

To summarize, the total loss can be written as

$$\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{GGN}} + \mathcal{L}_{\text{LON}} + \lambda_{\text{COOP}} \left(\mathcal{L}_{\text{3D}} + \mathcal{L}_{\text{PROJ}} + \mathcal{L}_{\text{PHY}} \right), \tag{2.20}$$

where λ_{COOP} is the trade-off parameter that balances the cooperative losses and the direct losses.

2.2.2.4 Implementation

Both the GGN and LON adopt ResNet-34 [HZR16] architecture as the encoder, which encodes a 256x256 RGB image into a 2048-D feature vector. As each of the networks consists of multiple output channels, for each channel with an L-dimensional output, we stack two fully connected layers (2048-1024, 1024-L) on top of the encoder to make the prediction.

We adopt a two-step training procedure. First, we fine-tune the 2D detector [DQX17, BSC17] with 30 most common object categories to generate 2D bounding boxes. The 2D and 3D bounding box are matched to ensure each 2D bounding box has a corresponding 3D bounding box.

Second, we train two 3D estimation networks. To obtain good initial networks, both GGN and LON are first trained individually using the synthetic data (SUNCG dataset [SYZ17a]) with photo-realistically rendered images [ZSY17a]. We then fix six blocks of the encoders of GGN and LON, respectively, and fine-tune the two networks jointly on SUN RGBD dataset

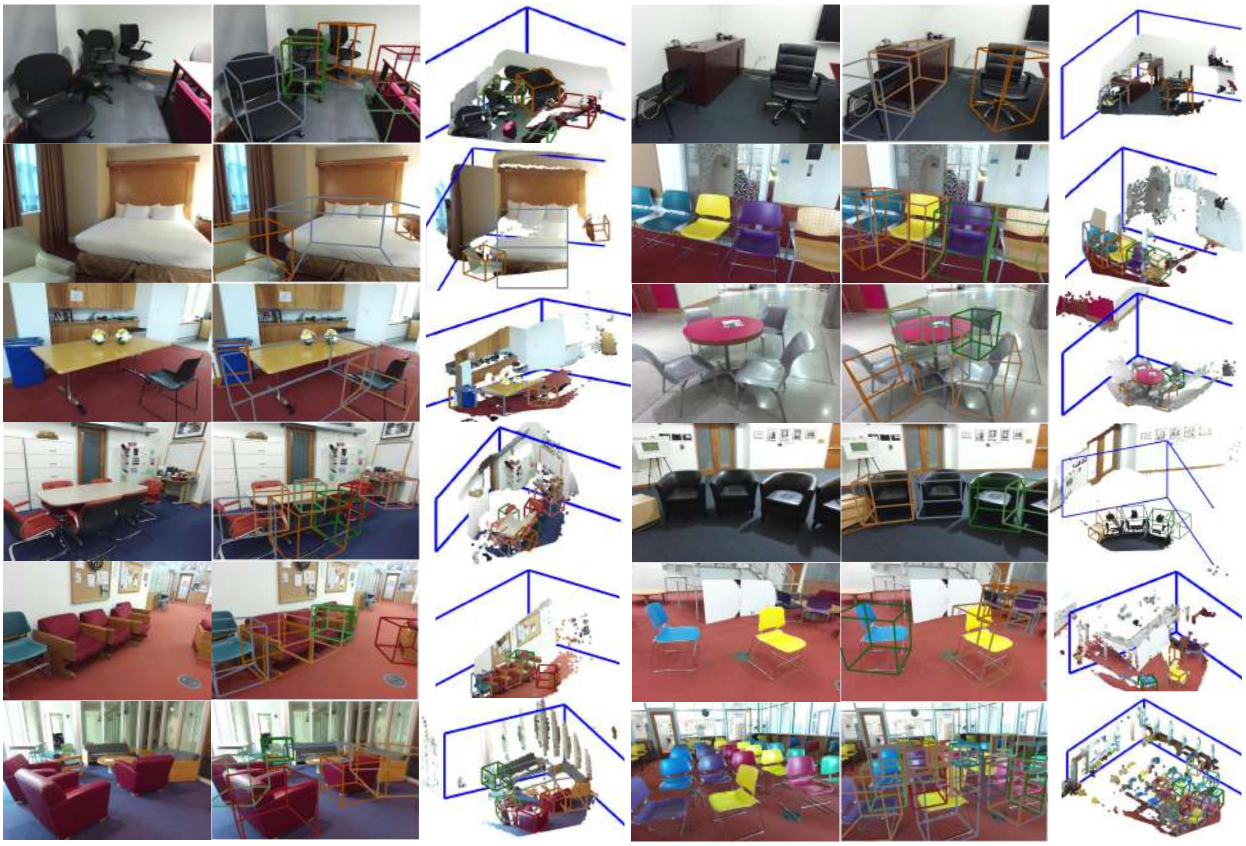


Figure 2.13: Qualitative results (top 50%). (Left) Original RGB images. (Middle) Results projected in 2D. (Right) Results in 3D. Note that the depth input is only used to visualize the 3D results.

[SLX15].

To avoid over-fitting, a data augmentation procedure is performed by randomly flipping the images or randomly shifting the 2D bounding boxes with corresponding labels during the cooperative training. We use Adam [KB14] for optimization with a batch size of 1 and a learning rate of 0.0001. In practice, we train the two networks cooperatively for ten epochs, which takes about 10 minutes for each epoch. We implement the proposed approach in PyTorch [PGC17].

2.2.3 Evaluation

We evaluate our model on SUN RGB-D dataset [SLX15], including 5050 test images and 10335 images in total. The SUN RGB-D dataset has 47 scene categories with high-quality 3D room layout, 3D camera pose, and 3D object bounding boxes annotations. It also provides benchmarks for various 3D scene understanding tasks. Here, we only use the RGB images as the input. Figure 2.13 shows some qualitative results. We discard the rooms with no detected 2D objects or invalid 3D room layout annotation, resulting in a total of 4783 training images and 4220 test images.

We evaluate our model on five tasks: i) 3D layout estimation, ii) 3D object detection, iii) 3D box estimation iv) 3D camera pose estimation, and v) holistic scene understanding, all with the test images across all scene categories. For each task, we compare our cooperatively trained model with the settings in which we train GGN and LON individually without the proposed parametrization of 3D object bounding box or cooperative losses. In the individual training setting, LON directly estimates the 3D object centers in the 3D world coordinate.

3D Layout Estimation Since SUN RGB-D dataset provides the ground truth of 3D layout with arbitrary numbers of polygon corners, we parametrize each 3D room layout as a 3D bounding box by taking the output of the Manhattan Box baseline from [SLX15] with eight layout corners, which serves as the ground truth. We compare the estimation of the proposed model with three previous methods—3DGP [CCP13], IM2CAD [ISS17a] and HoPR [HQZ18]. Following the evaluation protocol defined in [SLX15], we compute the average IoU between the free space of the ground truth and the free space estimated by the proposed method. Table 2.4 shows our model outperforms HoPR by 2.0%. The results further show that there is an additional 1.5% performance improvement compared with individual training, demonstrating the efficacy of our method. Note that IM2CAD [ISS17a] manually selected 484 images from 794 test images of living rooms and bedrooms. For fair comparisons, we evaluate our method on the entire set of living room and bedrooms, outperforming IM2CAD by 2.1%.

3D Object Detection We evaluate our 3D object detection results using the metrics defined in [SLX15]. Specifically, the MAP is computed using the 3D IoU between the predicted and the ground truth 3D bounding boxes. In the absence of depth, the threshold of

Table 2.4: Comparison of 3D room layout estimation and holistic scene understanding on SUN RGB-D.

Method	3D Layout Estimation	Holistic Scene Understanding			
	IoU	P_g	R_g	R_r	IoU
3DGP [CCP13]	19.2	2.1	0.7	0.6	13.9
HoPR [HQZ18]	54.9	37.7	23.0	18.3	40.7
Ours (individual)	55.4	36.8	22.4	20.1	39.6
Ours (cooperative)	**56.9**	**49.3**	**29.7**	**28.5**	**42.9**

Table 2.5: Comparisons of 3D object detection on SUN RGB-D.

Method	bed	chair	sofa	table	desk	toilet	bin	sink	shelf	lamp	mAP
[CCP13]	5.62	2.31	3.24	1.23	-	-	-	-	-	-	-
[HQZ18]	58.29	13.56	28.37	12.12	4.79	16.50	0.63	2.18	1.29	2.41	14.01
Ours (individual)	53.08	7.7	27.04	22.80	5.51	28.07	0.54	5.08	2.58	0.01	15.24
Ours (cooperative)	**63.58**	**17.12**	**41.22**	**26.21**	**9.55**	**58.55**	**10.19**	**5.34**	**3.01**	**1.75**	**23.65**

IoU is adjusted from 0.25 (evaluation setting with depth image input) to 0.15 to determine whether two bounding boxes are overlapped. The 3D object detection results are reported in Table 2.5. The results indicate our method outperforms HoPR by 9.64% on MAP and improves the individual training result by 8.41%. Compared with the model using individual training, the proposed cooperative model makes a significant improvement, especially on small objects such as bins and lamps. The accuracy of the estimation easily influences 3d detection of small objects; oftentimes, it is nearly impossible for prior approaches to detect. In contrast, benefiting from the parametrization method and 2D projection loss, the proposed cooperative model maintains the consistency between 3D and 2D, substantially reducing the estimation variance. Note that although IM2CAD also evaluates the 3D detection, they use a metric related to a specific distance threshold. For fair comparisons, we further conduct experiments on the subset of living rooms and bedrooms, using the same object categories with respect to this particular metric rather than an IoU threshold. We obtain an MAP of 78.8%, 4.2% higher than the results reported in IM2CAD.

Table 2.6: 3D box estimation results on SUN RGB-D.

	bed	chair	sofa	table	desk	toilet	bin	sink	shelf	lamp	mIoU
IoU (3D)	33.1	15.7	28.0	20.8	15.6	25.1	13.2	9.9	6.9	5.9	17.4
IoU (2D)	75.7	68.1	74.4	71.2	70.1	72.5	69.7	59.3	62.1	63.8	68.7

Table 2.7: Comparisons of 3D camera pose estimation on SUN RGB-D.

Method	Mean Absolute Error (degree)	
	yaw	roll
[HHF09]	3.45	33.85
[HQZ18]	3.12	7.60
Ours (individual)	2.48	4.56
Ours (cooperative)	**2.19**	**3.28**

3D Box Estimation The 3D object detection performance of our model is determined by both the 2D object detection and the 3D bounding box estimation. We first evaluate the accuracy of the 3D bounding box estimation, which reflects the ability to predict 3D boxes from 2D image patches. Instead of using MAP, 3D IoU is directly computed between the ground truth and the estimated 3D boxes for each object category. To evaluate the 2D-3D consistency, the estimated 3D boxes are projected back to 2D, and the 2D IoU is evaluated between the projected and detected 2D boxes. Results using the full model are reported in Table 2.6, which shows 3D estimation is still under satisfactory, despite the efforts to maintain a good 2D-3D consistency. The underlying reason for the gap between 3D and 2D performance is the increased estimation dimension. Another possible reason is due to the lack of context relations among objects.

Camera Pose Estimation We evaluate the camera pose by computing the mean absolute error of yaw and roll between the model estimation and ground truth. As shown in Table 2.7, comparing with the traditional geometry-based method [HHF09] and previous learning-based method [HQZ18], the proposed cooperative model gains a significant improvement. It also improves the individual training performance with 0.29 degree on yaw and 1.28 degree on roll.

Holistic Scene Understanding Per definition introduced in [SLX15], we further estimate the holistic 3D scene including 3D objects and 3D room layout on SUN RGB-D. Note that the holistic scene understanding task defined in [SLX15] misses 3D camera pose estimation compared to the definition in this work, as the results are evaluated in the world coordinate.

Using the metric proposed in [SLX15], we evaluate the geometric precision P_g, the geometric recall R_g, and the semantic recall R_r with the IoU threshold set to 0.15. We also evaluate the IoU between free space (3D voxels inside the room polygon but outside any object bounding box) of the ground truth and the estimation. Table 2.4 shows that we improve the previous approaches by a significant margin. Moreover, we further improve the individually trained results by 8.8% on geometric precision, 5.6% on geometric recall, 6.6% on semantic recall, and 3.7% on free space estimation. The performance gain of total scene understanding directly demonstrates that the effectiveness of the proposed parametrization method and cooperative learning process.

2.2.3.1 Discussion

In the experiment, the proposed method outperforms the state-of-the-art methods on four tasks. Moreover, our model runs at 2.5 fps (0.4s for 2D detection and 0.02s for 3D estimation) on a single Titan Xp GPU, while other models take significantly much more time; *e.g.*, [ISS17a] takes about 5 minutes to estimate one image. Here, we further analyze the effects of different components in the proposed cooperative model, hoping to shed some lights on how parametrization and cooperative training help the model using a set of ablative analysis.

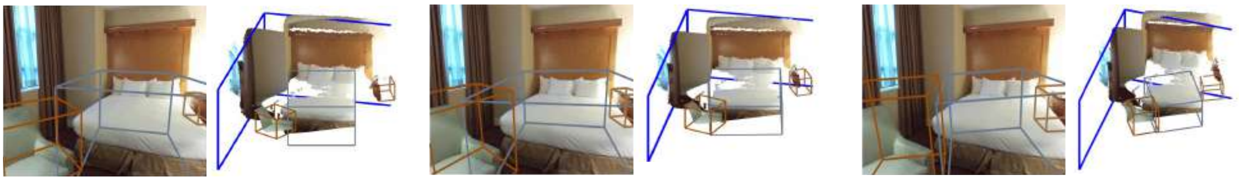
2.2.3.2 Ablative Analysis

We compare four variants of our model with the full model trained using $\mathcal{L}_{\text{SUM}}$:

1. The model trained without the supervision on 3D object bounding box corners (w/o

Table 2.8: The ablative analysis of the proposed cooperative model on SUN RGB-D. We evaluate holistic scene understanding, 3D mIoU and 2D mIoU of box estimation under different settings.

Setting	S_1	S_2	S_3	S_4	S_5	S_6	Full
IoU	42.8	42.0	41.7	35.9	40.2	43.0	**43.3**
P_g	41.8	**48.3**	47.2	28.1	36.3	45.4	46.5
R_g	25.3	**30.1**	27.5	17.1	22.1	29.7	28.0
R_r	23.8	**28.7**	26.4	15.6	20.6	27.1	26.7
3D mIoU	14.4	**18.2**	17.3	9.8	12.7	17.0	17.4
2D mIoU	65.2	60.7	68.5	64.3	65.3	67.7	**68.7**



(a) Full model (b) Model without 2D supervision (c) Model without 3D supervision

Figure 2.14: Comparison with two variants of our model.

$\mathcal{L}_{\text{3D}}$, S_1).

2. The model trained without the 2D supervision (w/o $\mathcal{L}_{\text{PROJ}}$, S_2).

3. The model trained without the penalty of physical constraint (w/o $\mathcal{L}_{\text{PHY}}$, S_3).

4. The model trained in an unsupervised fashion where we only use 2D supervision to estimate the 3D bounding boxes (w/o $\mathcal{L}_{\text{3D}} + \mathcal{L}_{\text{GGN}} + \mathcal{L}_{\text{LON}}$, S_4).

Additionally, we compare two variants of training settings: i) the model trained directly on SUN RGB-D without pre-train (S_5), and ii) the model trained with 2D bounding boxes projected from ground truth 3D bounding boxes (S_6). We conduct the ablative analysis over all the test images on the task of holistic scene understanding. We also compare the 3D mIoU and 2D mIoU of 3D box estimation. Table 2.8 summarizes the quantitative results.

Experiment $\mathbf{S_1}$ and $\mathbf{S_3}$ Without the supervision on 3D object bounding box corners or physical constraint, the performance of all the tasks decreases since it removes the cooperation between the two networks.

Experiment $\mathbf{S_2}$ The performance on the 3D detection is improved without the projection loss, while the 2D mIoU decreases by 8.0%. As shown in Figure 2.14(b), a possible reason is that the 2D-3D consistency $\mathcal{L}_{\text{PROJ}}$ may hurt the performance on 3D accuracy compared with directly using 3D supervision, while the 2D performance is largely improved thanks to the consistency.

Experiment $\mathbf{S_4}$ The training entirely in an unsupervised fashion for 3D bounding box estimation would fail since each 2D pixel could correspond to an infinite number of 3D points. Therefore, we integrate some common sense into the unsupervised training by restricting the size of the object close to the average size. As shown in Figure 2.14(c), we can still estimate the 3D bounding box without 3D supervision quite well, although the orientations are usually not accurate.

Experiment $\mathbf{S_5}$ and $\mathbf{S_6}$ S_5 demonstrates the efficiency of using a large amount of synthetic training data, and S_6 indicates that we can gain almost the same performance even if there are no 2D bounding box annotations.

2.2.4 Related Work

Single Image Scene Reconstruction Existing 3D scene reconstruction approaches fall into two streams. i) Generative approaches model the reconfigurable graph structures in generative probabilistic models [ZZ11, ZZ13, CCP13, LFU13, GH13, ZST14, ZLH17, HQZ18]. ii) Discriminative approaches [ISS17a, TGF18, SYZ17a] reconstruct the 3D scene using the representation of 3D bounding boxes or voxels through direct estimations. Generative approaches are better at modeling and inferring scenes with complex context, but they rely on sampling mechanisms and are always computational ineffective. Compared with prior discriminative approaches, our model focus on establishing cooperation among each scene module.

Gap between 2D and 3D It is intuitive to constrain the 3D estimation to be consistent with 2D images. Previous research on 3D shape completion and 3D object reconstruction explores this idea by imposing differentiable 2D-3D constraints between the shape and silhouettes [WXL16, REM16, YYY16, TM15, WWX17]. [MAF17] infers the 3D bounding boxes by matching the projected 2D corners in autonomous driving. In the proposed cooperative model, we introduce the parametrization of the 3D bounding box, together with a differentiable loss function to impose the consistency between 2D-3D bounding boxes for indoor scene understanding.

2.2.5 Conclusion

Using a single RGB image as the input, we propose an end-to-end model that recovers a 3D indoor scene in real-time, including the 3D room layout, camera pose, and object bounding boxes. A novel parametrization of 3D bounding boxes and a 2D projection loss are introduced to enforce the consistency between 2D and 3D. We also design differentiable cooperative losses which help to train two major modules cooperatively and efficiently. Our method shows significant improvements in various benchmarks while achieving high accuracy and efficiency.

2.3 3D Object Detection with Perspective Points

Detecting 3D objects from a single RGB image is intrinsically ambiguous, thus requiring appropriate prior knowledge and intermediate representations as constraints to reduce the uncertainties and improve the consistencies between the 2D image plane and the 3D world coordinate.

In this section, we address this challenge by proposing to adopt perspective points as a new intermediate representation for 3D object detection, defined as the 2D projections of local Manhattan 3D keypoints to locate an object; these perspective points satisfy geometric constraints imposed by the perspective projection. We further devise PerspectiveNet, an end-to-end trainable model that simultaneously detects the 2D bounding box, 2D perspective points, and 3D object bounding box for each object from a single RGB image. PerspectiveNet yields three unique advantages: (i) 3D object bounding boxes are estimated based on perspective points, bridging the gap between 2D and 3D bounding boxes *without* the need of category-specific 3D shape priors. (ii) It predicts the perspective points by a *template-based* method, and a perspective loss is formulated to maintain the perspective constraints. (iii) It maintains the consistency between the 2D perspective points and 3D bounding boxes via a *differentiable* projective function. Experiments on SUN RGB-D dataset show that the proposed method significantly outperforms existing RGB-based approaches for 3D object detection.

2.3.1 Introduction

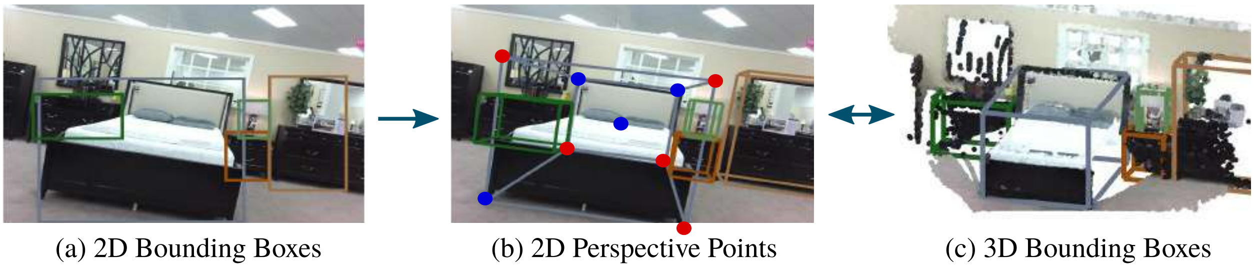
> If one hopes to achieve a full understanding of a system as complicated as a nervous system, . . . , or even a large computer program, then one must be prepared to contemplate different kinds of explanation at different levels of description that are linked, at least in principle, into a cohesive whole, even if linking the levels in complete details is impractical. — David Marr [Mar82], pp. 20–21

In a classic view of computer vision, David Marr [Mar82] conjectured that the perception of a 2D image is an *explicit* multi-phase information process, involving (i) an early vision system of perceiving textures [Jul62, ZWM98] and textons [Jul81, ZGW05] to form a primal sketch as a perceptually lossless conversion from the raw image [GZW03, GZW07], (ii) a mid-level vision system to construct 2.1D (multiple layers with partial occlusion) [NM90, WA93, WA94] and 2.5D [MN78] sketches, and (iii) a high-level vision system that recovers the full 3D [Bin71, Bro81, Kan81]. In particular, he highlighted the importance of different levels of organization and the internal representation [Bro85].

In parallel, the school of Gestalt Laws [Wer12, WEK12, WFG12, Koh20, Koh38, Wer23, Wer38, Kof13] and perceptual organization [Low12, Pen87] aims to resolve the 3D reconstruction problem from a single RGB image without forming the depth cues; but rather, they often use some sorts of priors—groupings and structural cues [Wal75, BT81] that are likely to be invariant over wide ranges of viewpoints [Low87], resulting in the birth of the SIFT feature [Low04]. Later, from a Bayesian perspective at a scene level, such priors, independent of any 3D scene structures, were found in the human-made scenes, known as the Manhattan World assumption [CY03]. Importantly, further studies found that such priors help to improve object detection [CY99].

In this work, inspired by these two classic schools in computer vision, we seek to test the following two hypotheses using modern computer vision methods: (i) Could an *intermediate representation* facilitate modern computer vision tasks? (ii) Is such an intermediate representation a better and more *invariant prior* compared to the priors obtained directly from specific tasks?

In particular, we tackle the challenging task of 3D object detection from a single RGB image. Despite the recent success in 2D scene understanding (*e.g.*, [RHG15, HGD17], there is still a significant performance gap for 3D computer vision tasks based on a single 2D image. Recent modern approaches directly regress the 3D bounding boxes [CKZ16, MAF17, HQX18] or reconstruct the 3D objects with specific 3D object priors [KLR18, HQZ18, YHZ18, HS19]. In contrast, we propose an end-to-end trainable framework, PerspectiveNet, that sequentially estimates the 2D bounding box, 2D perspective points, and 3D bounding box for each object with a local Manhattan assumption [XF14], in which the perspective points serve as the


(a) 2D Bounding Boxes

(b) 2D Perspective Points

(c) 3D Bounding Boxes

Figure 2.15: Traditional 3D object detection methods directly estimate (c) the 3D object bounding boxes from (a) the 2D bounding boxes, which suffer from the uncertainties between the 2D image plane and the 3D world. The proposed PerspectiveNet utilizes (b) the 2D perspective points as the intermediate representation to bridge the gap. The perspective points are the 2D perspective projection of the 3D bounding box corners, containing rich 3D information (*e.g.*, positions, orientations). The red dots indicate the perspective points of the bed that are challenging to emerge based on the visual features, but could be inferred by the context (correlations and topology) among other perspective points.

intermediate representation, defined as the 2D projections of local Manhattan 3D keypoints to locate an object.

The proposed method offers three unique advantages. First, the use of perspective points as the *intermediate representation* bridges the gap between 2D and 3D bounding boxes *without* utilizing any extra category-specific 3D shape priors. As shown in Figure 2.15, it is often challenging for learning-based methods to estimate the 3D bounding boxes from 2D images directly; regressing 3D bounding boxes from 2D input is a highly under-constrained problem and can be easily influenced by appearance variations of shape, texture, lighting, and background. To alleviate this issue, we adopt the perspective points as an intermediate representation to represent the local Manhattan frame that each 3D object aligns with. Intuitively, the perspective points of an object are *3D geometric constraints in the 2D space*. More specifically, the 2D perspective points for each object are defined as the perspective projection of the 3D object bounding box (concatenated with its center), and each 3D box aligns within a 3D local Manhattan frame. These perspective points are fused into the 3D branch to predict the 3D attributes of the 3D bounding boxes.

Second, we devise a *template-based* method to efficiently and robustly estimate the perspective points. Existing methods [NYD16, LBM17, ZCS18, HGD17, SST18] usually exploit heatmap or probability distribution map as the representation to learn the location of visual points (*e.g.*, object keypoint, human skeleton, room layout), relying heavily on the view-dependent visual features, thus insufficient to resolve occlusions or large rotation/viewpoint changes in complex scenes; see an example in Figure 2.15 (b) where the five perspective points (in red) are challenging to emerge from pure visual features but could be inferred by the correlations and topology among other perspective points. To tackle this problem, we treat each set of 2D perspective points as the low dimensional embedding of its corresponding set of 3D points with a constant topology; such an embedding is learned by predicting the perspective points as a mixture of sparse templates. A perspective loss is formulated to impose the perspective constraints; the details are described in Section 2.3.3.2.

Third, the consistency between the 2D perspective points and 3D bounding boxes can be maintained by a *differentiable* projective function; it is end-to-end trainable, from the 2D region proposals, to the 2D bounding boxes, to the 2D perspective points, and to the 3D bounding boxes.

In the experiment, we show that the proposed PerspectiveNet outperforms previous methods with a large margin on SUN RGB-D dataset [SLX15], demonstrating its efficacy on 3D object detection.

2.3.2 Related Work

3D object detection from a single image Detecting 3D objects from a single RGB image is a challenging problem, particularly due to the intrinsic ambiguity of the problem. Existing methods could be categorized into three streams: (i) geometry-based methods that estimate the 3D bounding boxes with geometry and 3D world priors [ZZ11, ZZ13, CCP13, LFU13, ZST14]; (ii) learning-based methods that incorporate category-specific 3D shape prior [ISS17a, HQZ18, HS19] or extra 2.5D information (depth, surface normal, and segmentation) [KLR18, YHZ18, XC18] to detect 3D bounding boxes or reconstruct the 3D object shape; and (iii) deep learning methods that directly estimates the 3D object bounding boxes from 2D bounding boxes [CKZ15, CKZ16, MAF17, HQX18]. To make better estimations, various techniques have been devised to enforce consistencies between the estimated 3D and the input 2D image. [HQX18] proposed a two-stage method to learn the 3D objects and 3D layout cooperatively. [KLR18] proposed a 3D object detection and reconstruction method using category-specific object shape prior by render-and-compare. Different from these methods, the proposed PerspectiveNet is a one-stage end-to-end trainable 3D object detection framework using perspective points as an intermediate representation; the perspective points naturally bridge the gap between the 2D and 3D bounding boxes without any extra annotations, category-specific 3D shape priors, or 2.5D maps.

Manhattan World assumption Human-made environment, from the layout of a city to structures such as buildings, room, furniture, and many other objects, could be viewed as a set of parallel and orthogonal planes, known as the Manhattan World (MW) assumption [CY99]. Formally, it indicates that most human-made structures could be approximated by planar surfaces that are parallel to one of the three principal planes of a common orthogonal coordinate system. This strict Manhattan World assumption is later extended by a Mixture of Manhattan Frame (MMF) [SRF14] to represent more complex real-world scenes (*e.g.*, city layouts, rotated objects). In literature, MW and MMF have been adopted in vanish points (VPs) estimation and camera calibration [SD04, KDV15], orientation estimation [BRL03, SBL15, GTC15], layout estimation [HHF09, LHK09, HHF10, SHP12, ZCS18], and 3D scene reconstruction [DLN07, FCS09, XHR13, XF14, RS16, LZZ17]. In this work, we extend the MW to local Manhattan assumption where the cuboids are aligned with the vertical (gravity) direction but with arbitrary horizontal orientation (also see [XF14]), and perspective points are adopted as the intermediate representation for 3D object detection.

Intermediate 3D representation Intermediate 3D representations are bridges that narrow the gap and maintain the consistency between the 2D image plane and 3D world. Among them, 2.5D sketches have been broadly used in reconstructing the 3D shapes [WWX17, ZZZ18b, ZZZ18a] and 3D scenes [TGF18, HQZ18]. Other recent alternative intermediate 3D representations include: (i) [WXL16] uses pre-annotated and category-specific object keypoints as an intermediate representation, and (ii) [TSF18] uses projected corners of 3D bounding boxes in learning the 6D object pose. In this work, we explore the perspective points as an intermediate representation of 2D and 3D bounding boxes, and provide an efficient learning framework for 3D object detection.

2.3.3 Learning Perspective Points for 3D Object Detection

2.3.3.1 Overall Architecture

As shown in Figure 2.16, the proposed PerspectiveNet contains a backbone architecture for feature extraction over the entire image, a region proposal network (RPN) [RHG15] that proposes regions of interest (RoIs), and a network head including three region-wise parallel branches. For each proposed box, its RoI feature is fed into the three network branches to predict: (i) the object class and the 2D bounding box offset, (ii) the 2D perspective points (projected 3D box corners and object center) as a weighted sum of predicted perspective templates, and (iii) the 3D box size, orientation, and its distance from the camera. Detected 3D boxes are reconstructed by the projected object center, distance, box size, and rotation. The overall architecture of the PerspectiveNet resembles the R-CNN structure, and we refer readers to [RHG15, Gir15, HGD17] for more details of training R-CNN detectors.

During training, we define a multi-task loss on each proposed RoI as

$$\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{2D} + \mathcal{L}_{pp} + \mathcal{L}_{p} + \mathcal{L}_{3D} + \mathcal{L}_{proj}, \tag{2.21}$$

where the classification loss $\mathcal{L}_{cls}$ and 2D bounding box loss $\mathcal{L}_{2D}$ belong to the 2D bounding box branch and are identical to those defined in 2D R-CNNs [RHG15, HGD17]. $\mathcal{L}_{pp}$ and $\mathcal{L}_{p}$ are defined on the perspective point branch (Section 2.3.3.2), $\mathcal{L}_{3D}$ is defined on the 3D bounding box branch (see Section 2.3.3.3), and the $\mathcal{L}_{proj}$ is defined on maintaining the 2D-3D projection consistency (see Section 2.3.3.4).

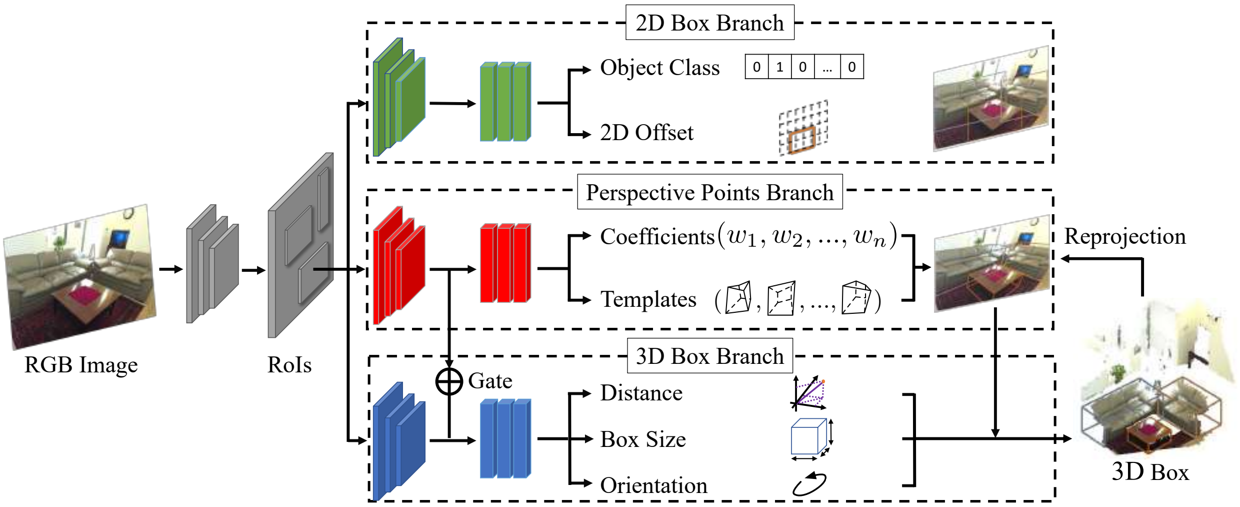


Figure 2.16: The proposed framework of the PerspectiveNet. Given an RGB Image, the backbone of PerspectiveNet extracts global features and propose candidate 2D bounding boxes (RoIs). For each proposed box, its RoI feature is fed into three network branches to predict: (i) the object class and the 2D box offset, (ii) 2D perspective templates (projected 3D box corners and object center) and the corresponding coefficients, and (iii) the 3D box size, orientation, and its distance from the camera. Detected 3D boxes are reconstructed by the projected object center, distance, box size, and rotation. By projecting the detected 3D boxes to 2D and comparing them with 2D perspective points, the network imposes and learns a consistency between the 2D inputs and 3D estimations.

2.3.3.2 Perspective Point Estimation

The perspective point branch estimates the set of 2D perspective points for each RoI. Formally, the 2D perspective points of an object are the 2D projections of local Manhattan 3D keypoints to locate that object, and they satisfy certain geometric constraints imposed by the perspective projection. In our case, the perspective points (Figure 2.15(b)) include the 2D projections of the 3D bounding box corners and the 3D object center. The perspective points are predicted using a template-based regression and learned by a mean squared error and a perspective loss detailed below.

Template-based Regression Most of the existing methods [NYD16, LBM17, ZCS18, HGD17, SST18] estimate visual keypoints with heatmaps, where each map predicts the location for a certain keypoint. However, predicting perspective points by heatmaps has two major problems: (i) Heatmap prediction for different keypoints is independent, thus fail to capture the topology correlations among the perspective points. (ii) Heatmap prediction for each keypoint relies heavily on the visual feature such as corners, which may be difficult to detect (see an example in Figure 2.15(b)). In contrast, each set of 2D perspective points can be treated as a low dimensional embedding of a set of 3D points with a particular topology, thus inferring such points relies more on the relation and topology among the points instead of just the visual features.

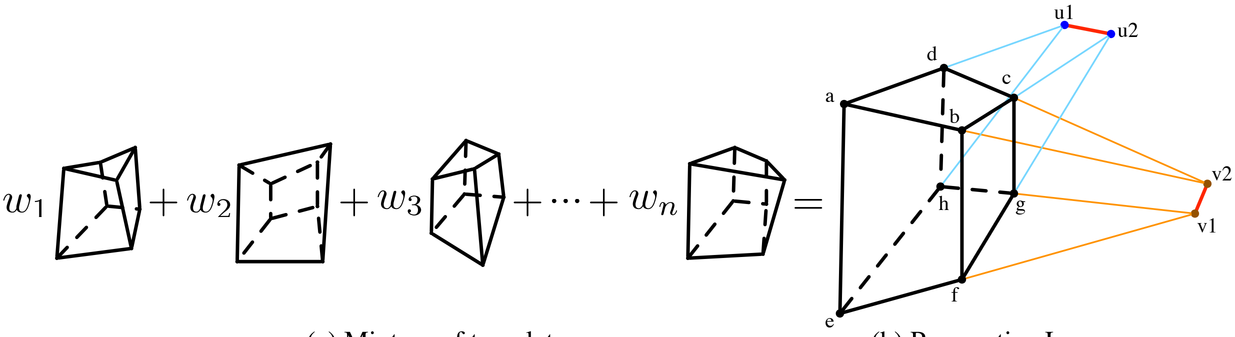
To tackle these problems, we avoid dense per-pixel predictions. Instead, we estimate the perspective points by a mixture of sparse templates [OF96, WSG10]. The sparse templates are more robust when facing unfamiliar scenes or objects. Ablative experiments show that the proposed template-based method provides a more accurate estimation of perspective points than heatmap-based methods; see Section 2.3.5.1.

Specifically, we project both the 3D object center and eight 3D bounding box corners to 2D with camera parameters to generate the ground-truth 2D perspective points $P_{gt} \in \mathbb{R}^{2\times 9}$. Since a portion of the perspective points usually lies out of the RoI, we calculate the location of the perspective points in an extended (doubled) size of RoI and normalize the locations to $[0, 1]$.

We predict the perspective points by a linear combination of templates; see Figure 2.17. The perspective point branch has a $C \times K \times 2 \times 9$ dimensional output for the templates T, and a $C \times K$ dimensional output for the coefficients w, where K denotes the number of templates for each class and C denotes the number of object classes. The templates T is scaled to $[0, 1]$ by a sigmoid nonlinear function, and the coefficients w is normalized by a softmax function. The estimated perspective points $\hat{P} \in \mathbb{R}^{C\times 2\times 9}$ can be computed by a linear combination:

$$\hat{P}_i = \sum_{k=1}^{K} w_{ik}\, T_{ik}, \quad \forall i = 1, \cdots, C. \tag{2.22}$$

The template design is both class-specific and instance-specific: (i) Class-specific: we decouple the prediction of the perspective point and the object class, allowing the network to learn perspective points for every class without competition among classes. (ii) Instance-specific: the templates are inferred for each RoI; hence, they are specific to each object



(a) Mixture of templates

(b) Perspective Loss

Figure 2.17: Perspective point estimation. (a) The perspective points are estimated by a mixture of templates through a linear combination. Each template encodes geometric cues including orientations and viewpoints. (b) The perspective loss enforces each set of 2D perspective points to be the perspective projection of a (vertical) 3D cuboid. For a vertical cuboid, the projected vertical edges (*i.e.*, ae, bf, cg, and dh) should be parallel or near parallel (under small camera tilting angles). For 3D parallel lines that are perpendicular to the gravity direction, the vanishing points of their 2D projections should coincide (*e.g.*, $u1$ and $u2$).

instance. The templates are automatically learned for each object instance from data with the end-to-end learning framework; thus, both the templates and coefficients for each instance are optimizable and can better fit the training data.

The average mean squared error (MSE) loss is defined as $\mathcal{L}_{pp} = \text{MSE}(\hat{P}_c, P_{gt})$. For an RoI associated with ground-truth class c, $\mathcal{L}_{pp}$ is only defined on the c's perspective points during training; perspective point outputs from other classes do not contribute to the loss. In inference, we rely on the dedicated classification branch to predict the class label to select the output perspective points.

Perspective Loss Under the assumption that each 3D bounding box aligns with a local Manhattan frame, we regularize the estimation of the perspective points to satisfy the constraint of perspective projection. Each set of mutually parallel lines in 3D can be projected into 2D as intersecting lines; see Figure 2.17 (b). These intersecting lines should converge at the same vanishing point. Therefore, the desired algorithm would penalize the distance between the intersection points from the two sets of intersecting lines. For example in Figure 2.17 (b), we select line ad and line eh as a pair of lines, bc and fg as another, and compute the distance between their intersection point u_1 and u_2. Additionally, since we assume each 3D local Manhattan frame aligns with the vertical (gravity) direction, we enforce the edges in gravity direction (*i.e.*, ae, bf, cg, and dh) to be parallel by penalizing the large slope variance.

The perspective loss is computed as $\mathcal{L}_p = \mathcal{L}_{d1} + \mathcal{L}_{d2} + \mathcal{L}_{grav}$, where $\mathcal{L}_{grav}$ penalizes the slope variance in gravity direction, $\mathcal{L}_{d1}$ and $\mathcal{L}_{d2}$ penalize the intersection point distance for the two perpendicular directions along the gravity direction.

2.3.3.3 3D Bounding Box Estimation

Estimating 3D bounding boxes is a two-step process. In the first step, the 3D branch estimates the 3D attributes, including the distance between the camera center and the 3D object center, as well as the 3D size and orientation following [HQX18]. Since the perspective point branch encodes rich 3D geometric features, the 3D attribute estimator aggregates the feature from perspective point branch with a soft gated function between $[0, 1]$ to improve the prediction. The gated function serves as a soft-attention mechanism that decides how much information from perspective points should contribute to the 3D prediction.

In the second step, with the estimated projected 3D bounding boxes center (*i.e.*, the first estimated perspective point) and the 3D attributes, we compose the 3D bounding boxes by the inverse projection from the 2D image plane to the 3D world following [HQX18] given camera parameters.

The 3D loss is computed by the sum of individual losses of 3D attributes and a joint loss of 3D bounding box $\mathcal{L}_{3D} = \mathcal{L}_{dis} + \mathcal{L}_{size} + \mathcal{L}_{ori} + \mathcal{L}_{box3d}$.

2.3.3.4 2D-3D Consistency

In contrast to prior work [WXL16, REM16, YYY16, MAF17, WWX17, HQX18] that enforces the consistency between estimated 3D objects and 2D image, we devise a new way to impose a re-projection consistency loss between 3D bounding boxes and perspective points. Specifically, we compute the 2D projected perspective points P_{proj} by projecting the 3D

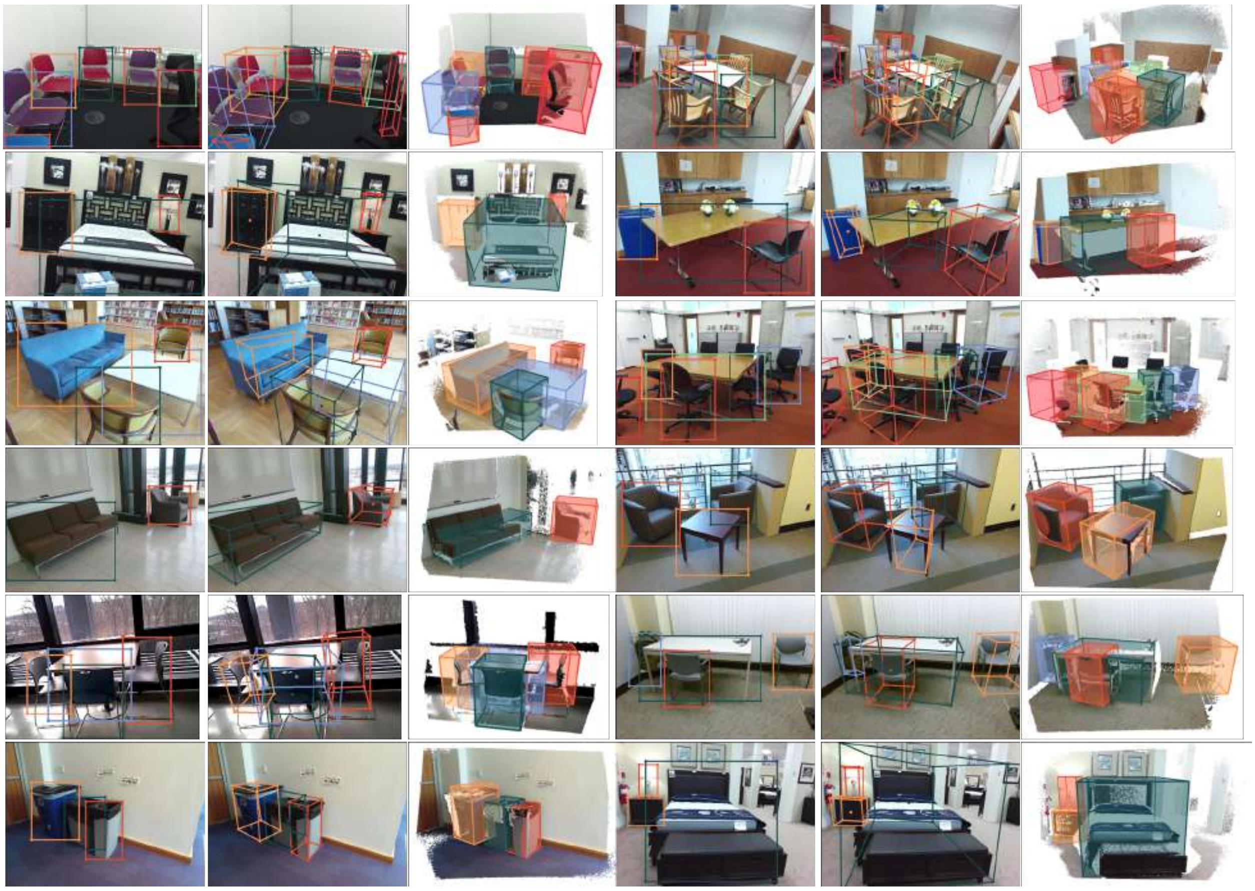


Figure 2.18: Qualitative results (top 50%). For every three columns as a group: (Left) The RGB image with 2D detection results. (Middle) The RGB image with estimated perspective points. (Right) The results in 3D point cloud; point cloud is used for visualization only.

bounding box corners back to 2D image plane and computing the distance with respect to ground-truth perspective points $\mathcal{L}_{proj} = \text{MSE}(P_{proj}, P_{gt})$. Comparing with prior work to maintain the consistency between 2D and 3D bounding boxes by approximating the 2D projection of 3D bounding boxes [MAF17, HQX18], the proposed method uses the *exact* projection of projected 3D boxes to establish the consistency, capturing a more precise 2D-3D relationship.

2.3.4 Implementation Details

Network Backbone Inspired by [HGD17], we use the combination of residual network (ResNet) [HZR16] and feature pyramid network (FPN) [LDG17] to extract the feature from

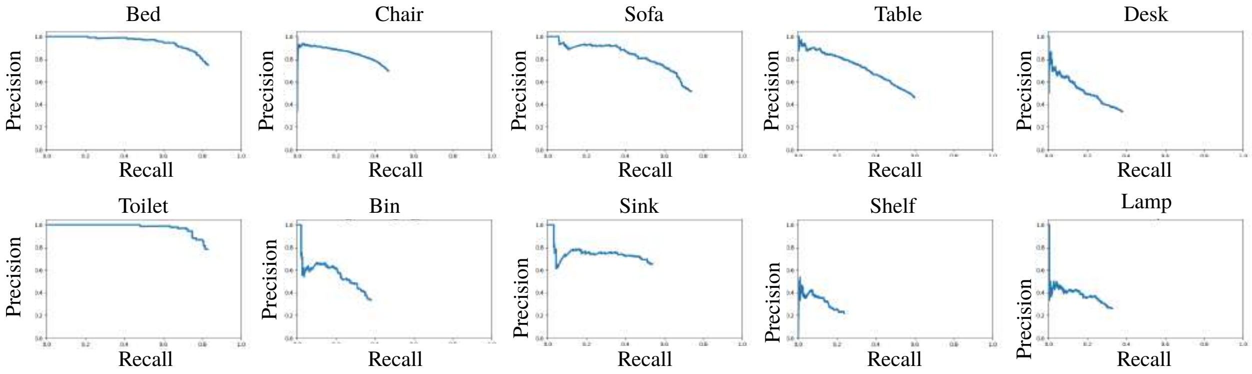


Figure 2.19: Precision-Recall (PR) curves for 3D object detection on SUN RGB-D

the entire image. A region proposal network (RPN) [RHG15] is used to produce object proposals (*i.e.*, RoI). A RoIAlign [HGD17] module is adopted to extract a smaller features map (256 × 7 × 7) for each proposal.

Network Head The network head consists of three branches, and each branch has its individual feature extractor and predictor. Three feature extractors have the same architecture of two fully connected (FC) layers; each FC layer is followed by a ReLU function. The feature extractors take the 256 × 7 × 7 dimensional RoI features as the input and output a 1024 dimensional vector.

The predictor in the 2D branch has two separate FC layers to predict a C dimensional object class probabilities and a $C \times 4$ dimensional 2D bounding box offset. The predictor in the perspective point branch predicts $C \times K \times 2 \times 9$ dimensional templates and $C \times K$ dimensional coefficients with two FC layers and their corresponding nonlinear activation functions (*i.e.*, sigmoid, softmax). The soft gate in the 3D branch consists of an FC layer (1024-1) and a sigmoid function to generate the weight for feature aggregation. The predictor in the 3D branch consists of three FC layers to predict the size, the distance from the camera, and the orientation of the 3D bounding box.

2.3.5 Experiments

Dataset We conduct comprehensive experiments on SUN RGB-D [SLX15] dataset. The SUN RGB-D dataset has a total of 10,335 images, in which 5,050 are test images. It has a rich annotation of scene categories, camera pose, and 3D bounding boxes. We evaluate the 3D object detection results of the proposed PerspectiveNet, make comparisons with the state-of-the-art methods, and further examine the contribution of each module in ablative experiments.

Experimental Setup To prepare valid data for training the proposed model, we discard the images with no 3D objects or incorrect correspondence between 2D and 3D bounding boxes, resulting 4783 training images and 4220 test images. We detect 30 categories of objects following [HQX18].

Reproduciblity Details During training, an RoI is considered positive if it has the IoU with a ground-truth box of at least 0.5. $\mathcal{L}_{pp}$, $\mathcal{L}_{p}$, $\mathcal{L}_{3D}$, and $\mathcal{L}_{proj}$ are only defined on positive RoIs. Each image has N sampled RoIs, where the ratio of positive to negative is 1:3 following the protocol presented in [Gir15].

We resize the images so that the shorter edges are all 800 pixels. To avoid over-fitting, a data augmentation procedure is performed by randomly flipping the images or randomly shifting the 2D bounding boxes with corresponding labels during the training. We use SGD for optimization with a batch size of 32 on a desktop with 4 Nvidia TITAN RTX cards (8 images each card). The learning rate starts at 0.01 and decays by 0.1 at 30,000 and 35,000 iterations. We implement our framework based on the code of [MG18]. It takes 6 hours to train, and the trained PerspectiveNet provides inference in real-time (20 FPS) using a single GPU.

Since the consistency loss and perspective loss can be substantial during the early stage of the training process, we add them to the joint loss when the learning rate decays twice. The hyper-parameter (*e.g.*, the weights of losses, the architecture of network head) is tuned empirically by a local search.

Evaluation Metric We evaluate the performance of 3D object detection using the metric presented in [SLX15]. Specifically, we first calculate the 3D Intersection over Union (IoU) between the predicted 3D bounding boxes and the ground-truth 3D bounding boxes, and then compute the mean average precision (mAP). Following [HQX18], we set the 3D IoU threshold as 0.15 in the absence of depth information.

Qualitative Results The qualitative results of 2D object detection, 2D perspective point estimation, and 3D object detection are shown in Figure 2.18. Note that the proposed method performs accurate 3D object detection in some challenging scenes. For the perspective point estimation, even though some of the perspective points are not aligned with image features, the proposed method can still localize their positions robustly.

Quantitative Results Since the state-of-the-art method [HQX18] learns the camera extrinsic parameters jointly, we provide two protocals for evaluations for a fair comparison: (i) PerspectiveNet given ground-truth camera extrinsic parameter (*full*), and (ii) PerspectiveNet without ground-truth camera extrinsic parameter by learning it jointly following [HQX18] (*w/o. cam*).

We learn the detector for 30 object categories and report the precision-recall (PR) curve of 10 main categories in Figure 2.19. We calculate the area under the curve to compute AP; Table 2.9 shows the comparisons of APs of the proposed models with existing approaches.

Note that the critical difference between the proposed model and the state-of-the-art method [HQX18] is the intermediate representation to learn the 2D-3D consistency. [HQX18] uses 2D bounding boxes to enforce a 2D-3D consistency by minimizing the differences between projected 3D boxes and detected 2D boxes. In contrast, the proposed intermediate representation has a clear advantage since projected 3D boxes often are not 2D rectangles, and perspective points eliminate such errors.

Quantitatively, our full model improves the mAP of the state-of-the-art method [HQX18] by 14.71%, and the model without the camera extrinsic parameter improves by 10.91%. The significant improvement of the mAP demonstrates the efficacy of the proposed intermediate representation. We defer more analysis on how each component contributes to the overall performance in Section 2.3.5.1.

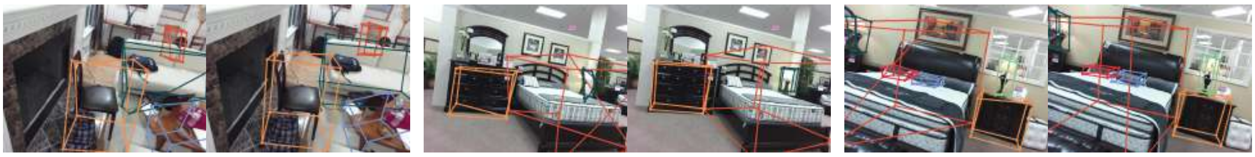


Figure 2.20: Heatmaps vs. templates for perspective point prediction. (Left) Estimated by heatmap-based method. (Right) Estimated by the proposed template-based method.

2.3.5.1 Ablative Analysis

In this section, we analyze each major component of the model to examine its contribution to the overall significant performance gain. Specifically, we design six variants of the proposed model.

- S_1: The model trained without the perspective point branch, using the 2D offset to predict the 3D center of the object following [HQX18].
- S_2: The model that aggregates the feature from the perspective point branch and 3D branch directly without the gate function.
- S_3: The model that aggregates the feature from the perspective point branch and 3D branch with a gate function that only outputs 0 or 1 (hard gate).
- S_4: The model trained without the perspective loss.
- S_5: The model trained without the consistency loss.
- S_6: The model trained without the perspective branch, perspective loss, or consistency loss.

Table 2.10 shows the mAP for each variant of the proposed model. The mAP drops 3.86% without the perspective point branch (S_1), 1.66% without the consistency loss (S_5), indicating that the perspective point and re-projection consistency influence the most to the

Table 2.9: Comparisons of 3D object detection on SUN RGB-D (AP).

	bed	chair	sofa	table	desk	toilet	bin	sink	shelf	lamp	mAP
3DGP [CCP13]	5.62	2.31	3.24	1.23	-	-	-	-	-	-	-
HoPR [HQZ18]	58.29	13.56	28.37	12.12	4.79	16.50	0.63	2.18	1.29	2.41	14.01
CooP [HQX18]	63.58	17.12	41.22	26.21	9.55	58.55	10.19	5.34	3.01	1.75	23.65
Ours (w/o. cam)	71.39	34.94	55.63	34.10	14.23	73.73	17.47	34.41	4.21	9.54	34.96
Ours (full)	**79.69**	**40.42**	**62.35**	**44.12**	**20.19**	**81.22**	**22.42**	**41.35**	**8.29**	**13.14**	**39.09**

proposed framework. In addition, the switch of gate function (S_2, S_3) and perspective loss (S_4) contribute less to the final performance. Since S_6 is still higher than the state-of-the-art result [HQX18] with 9.32%, we conjecture this performance gain may come from the one-stage (vs. two-stage) end-to-end training framework and the usage of ground-truth camera parameter; we will further investigate this in future work.

2.3.5.2 Heatmaps vs. Templates

As discussed in Section 2.3.3.2, we test two different methods for the perspective point estimation: (i) dense prediction as heatmaps following the human pose estimation mechanism in [HGD17] by adding a parallel heatmap prediction branch, and (ii) template-based regression by the proposed method. The qualitative results (see Figure 2.20) show that the heatmap-based estimation suffers severely from occlusion and topology change among the perspective points, whereas the proposed template-based regression eases the problem significantly by learning robust sparse templates, capturing consistent topological relations. We also evaluate the quantitative results by computing the average absolute distance between the ground-truth and estimated perspective points. The heatmap-based method has a 10.25 pix error, while the proposed method only has a 6.37 pix error, which further demonstrates the efficacy of the proposed template-based perspective point estimation.

2.3.5.3 Failure Cases

In a large portion of the failure cases, the perspective point estimation and the 3D box estimation fail at the same time; see Figure 2.21. It implies that the perspective point estimation and the 3D box estimation are highly coupled, which supports the assumptions that the perspective points encode richer 3D information, and the 3D branch learns meaningful

Table 2.10: Ablative analysis of the proposed model on SUN RGB-D. We evaluate the mAP for 3D object detection.

Setting	S_1	S_2	S_3	S_4	S_5	S_6	Full
mAP	35.23	38.63	38.87	39.01	37.43	32.97	**39.09**

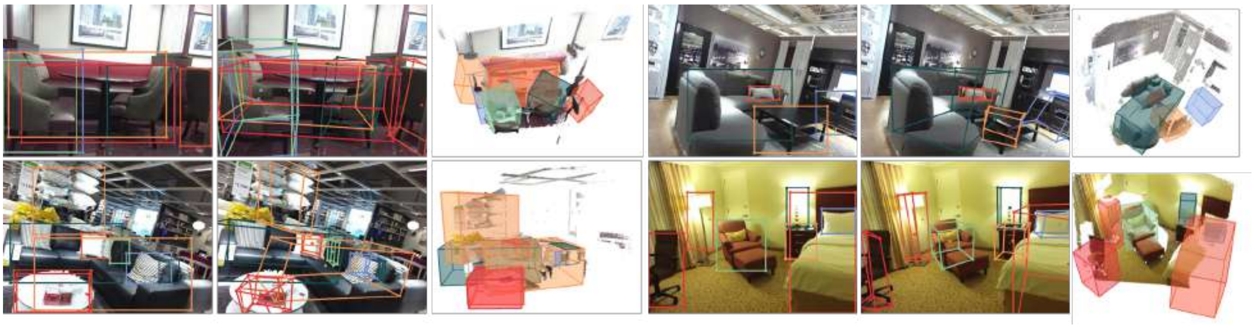


Figure 2.21: Some failure cases. The perspective point estimation and the 3D box estimation fail at the same time.

knowledge from the 2D branch. In future work, we may need a more sophisticated and general 3D prior to infer the 3D locations of objects for such challenging cases.

2.3.5.4 Discussions and Future Work

Comparison with optimization-based methods. Assume the estimated 3D size or distance is given, it is possible to compute the 3D bounding box with an optimization-based method like efficient PnP. However, the optimization-based methods are sensitive to the accuracy of the given known variables. It is more suitable for tasks with smaller solution spaces (*e.g.*, 6-DoF pose estimation where the 3D shapes of objects are fixed). However, it would be difficult for tasks with larger solution spaces (*e.g.*, 3D object detection where the 3D size, distance, and object pose could vary significantly). Therefore, we argue that directly estimating each variable with constraints imposed among them is a more natural and more straightforward solution.

Potential incorporation with depth information. The PerspectiveNet estimates the distance between the 3D object center and camera center based on the color image only (pure RGB without any depth information). If the depth information was also provided, the proposed method should be able to make a much more accurate distance prediction.

Potential application to outdoor environment. It would be interesting to see how the proposed method would perform on outdoor 3D object detection datasets like KITTI [GLS13]. The differences between the indoor and outdoor datasets for the task of 3D object detection lie in various aspects, including the diversity of object categories, the variety of object dimension, the severeness of the occlusion, the range of the camera angles, and the range of the distance (depth). We hope to adopt the PerspectiveNet in future to the outdoor scenarios.

2.3.6 Conclusion

We propose the PerspectiveNet, an end-to-end differentiable framework for 3D object detection from a single RGB image. It uses perspective points as an intermediate representation between 2D input and 3D estimations. The PerspectiveNet adopts an R-CNN structure, where the region-wise branches predict 2D boxes, perspective points, and 3D boxes. Instead of using a direct regression of 2D-3D relations, we further propose a template-based regression for estimating the perspective points, which enforces a better consistency between the predicted 3D boxes and the 2D image input. The experiments show that the proposed method significantly improves existing RGB-based methods.

CHAPTER 3

Human-centric 3D Scene Synthesis with Stochastic Grammar

In this chapter, we present a human-centric method to sample and synthesize 3D room layouts and 2D images thereof, to obtain large-scale 2D/3D image data with the perfect per-pixel ground truth. An attributed spatial And-Or graph (S-AOG) is proposed to represent indoor scenes. The S-AOG is a probabilistic grammar model, in which the terminal nodes are object entities including room, furniture, and supported objects. Human contexts as contextual relations are encoded by Markov Random Fields (MRF) on the terminal nodes. We learn the distributions from an indoor scene dataset and sample new layouts using Monte Carlo Markov Chain. Experiments demonstrate that the proposed method can robustly sample a large variety of realistic room layouts based on three criteria: (i) visual realism comparing to a state-of-the-art room arrangement method, (ii) accuracy of the affordance maps with respect to ground-truth, and (ii) the functionality and naturalness of synthesized rooms evaluated by human subjects.

3.1 Introduction

Traditional methods of 2D/3D image data collection and ground-truth labeling have evident limitations. i) High-quality ground truths are hard to obtain, as depth and surface normal obtained from sensors are always noisy. ii) It is impossible to label certain ground truth information, *e.g.*, 3D objects sizes in 2D images. iii) Manual labeling of massive ground-truth is tedious and error-prone even if possible. To provide training data for modern machine learning algorithms, an approach to generate large-scale, high-quality data with the perfect

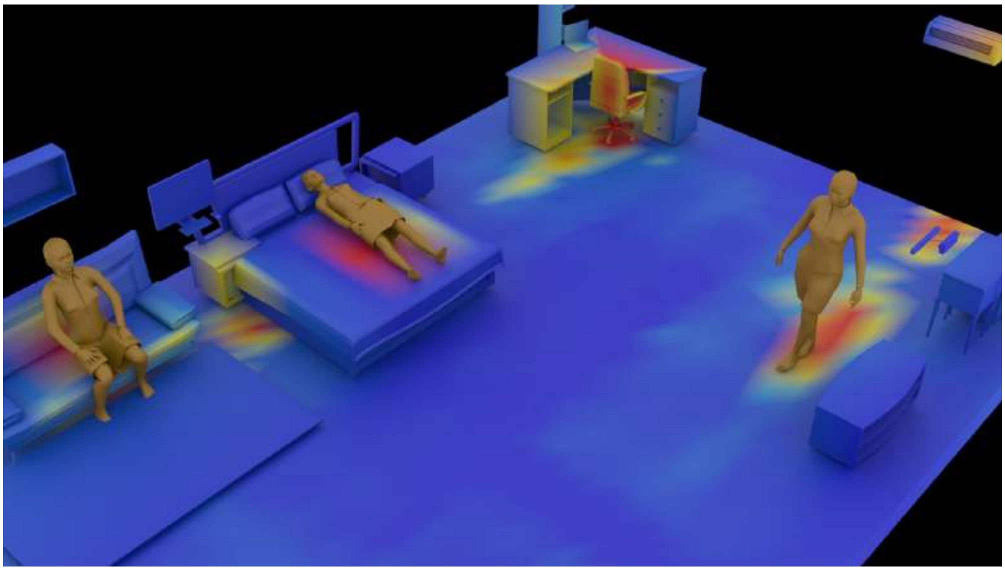


Figure 3.1: An example of synthesized indoor scene (bedroom) with affordance heatmap. The joint sampling of a scene is achieved by alternative sampling of humans and objects according to the joint probability distribution.

per-pixel ground truth is in need.

In this work, we propose an algorithm to automatically generate a large-scale 3D indoor scene dataset, from which we can render 2D images with pixel-wise ground-truth of the surface normal, depth, and segmentation, *etc.*. The proposed algorithm is useful for tasks including but not limited to: i) learning and inference for various computer vision tasks; ii) 3D content generation for 3D modeling and games; iii) 3D reconstruction and robot mappings problems; iv) benchmarking of both low-level and high-level task-planning problems in robotics.

Synthesizing indoor scenes is a non-trivial task. It is often difficult to properly model either the relations between furniture of a functional group, or the relations between the supported objects and the supporting furniture. Specifically, we argue there are four major difficulties. (i) In a functional group such as a dining set, the number of pieces may vary. (ii) Even if we only consider pair-wise relations, there is already a quadratic number of object-object relations. (iii) What makes it worse is that most object-object relations are not obviously meaningful. For example, it is unnecessary to model the relation between a

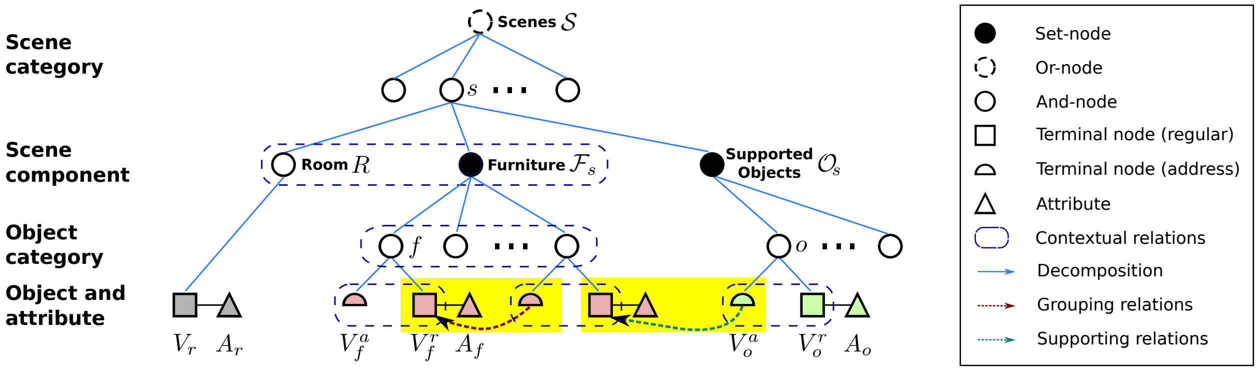


Figure 3.2: Scene grammar as an attributed S-AOG. A scene of different types is decomposed into a room, furniture, and supported objects. Attributes of terminal nodes are internal attributes (sizes), external attributes (positions and orientations), and a human position that interacts with this entity. Furniture and object nodes are combined by an address terminal node and a regular terminal node. A furniture node (*e.g.*, a chair) is grouped with another furniture node (*e.g.*, a desk) pointed by its address terminal node. An object (*e.g.*, a monitor) is supported by the furniture (*e.g.*, a desk) it is pointing to. If the value of the address node is null, the furniture is not grouped with any furniture, or the object is put on the floor. Contextual relations are defined between the room and furniture, between a supported object and supporting furniture, among different pieces of furniture, and among functional groups.

pen and a monitor, even though they are both placed on a desk. (iv) Due to the previous difficulties, an excessive number of constraints are generated. Many of the constraints contain loops, making the final layout hard to sample and optimize.

To address these challenges, we propose a human-centric approach to model indoor scene layout. It integrates human activities and functional grouping/supporting relations as illustrated in Figure 3.1. This method not only captures the human context but also simplifies the scene structure. Specifically, we use a probabilistic grammar model for images and scenes [ZM07] – an attributed spatial And-Or graph (S-AOG), including vertical hierarchy and horizontal contextual relations. The contextual relations encode functional grouping relations and supporting relations modeled by object affordances [Gib79]. For each object, we learn the affordance distribution, *i.e.*, an object-human relation, so that a human can be sampled based on that object. Besides static object affordance, we also consider dynamic human activities in a scene, constraining the layout by planning trajectories from one piece of furniture to another.

In Section 3.3, we define the grammar and its parse graph which represents an indoor scene. We formulate the probability of a parse graph in Section 3.4. The learning algorithm is described in Section 3.5. Finally, sampling an indoor scene is achieved by sampling a parse tree (Section 3.6) from the S-AOG according to the prior probability distribution.

This work makes three major contributions. (i) We jointly model objects, affordances, and activity planning for indoor scene configurations. (ii) We provide a general learning and sampling framework for indoor scene modeling. (iii) We demonstrate the effectiveness of this structured joint sampling by extensive comparative experiments.

3.2 Related Work

3D content generation is one of the largest communities in the game industry and we refer readers to a recent survey [HMV13] and book [STN16]. In this work, we focus on approaches related to our work using probabilistic inference. Yu [YYT11] and Handa [HPS16] optimize the layout of rooms given a set of furniture using MCMC, while Talton [TLL11] and Yeh [YYW12] consider an open world layout using RJMCMC. These 3D room re-arrangement algorithms optimize room layouts based on constraints to generate new room layouts using a given set of objects. In contrast, the proposed method is capable of adding or deleting objects without fixing the number of objects. Some literature focused on fine-grained room arrangement for specific problems, *e.g.*, small objects arrangement using user-input examples [FRS12] and procedural modeling of objects to encourage volumetric similarity to a target shape [RMG15]. To achieve better realism, Merrell [MSL11] introduced an interactive system providing suggestions following interior design guidelines. Jiang [JKS16] uses a mixture of conditional random field (CRF) to model the hidden human context and arrange new small objects based on existing furniture in a room. However, it cannot direct sampling/synthesizing an indoor scene, since the CRF is intrinsically a discriminative model for structured classification instead of generation.

Synthetic data has been attracting an increasing interest to augment or even serve as training data for object detection and correspondence [DMH17, MHL17, QSN16, SX14, SS14, ZSY17b, ZKA16], single-view reconstruction [HWK15], pose estimation [CWL16, SVD03, SQL15, YIK16], depth prediction [SHM14], semantic segmentation [RVR16], scene understanding [HPB16, HPS16, ZBK17], autonomous pedestrians and crowd [OPO10, QZ18, ST05], VQA [JHM17b], training autonomous vehicles [CSK15, DRC17, SDL17], human utility learning [YQK17, ZJZ16] and benchmarks [HWM14, QY16].

Stochastic grammar model has been used for parsing the hierarchical structures from images of indoor [LZZ14, ZZ13] and outdoor scenes [LZZ14], and images/videos involving humans [QHW17, WXS18]. In this work, instead of using stochastic grammar for parsing, we forward sample from a grammar model to generate large variations of indoor scenes.

3.3 Representation of Indoor Scenes

We use an attributed S-AOG [ZM07] to represent an indoor scene. An attributed S-AOG is a probabilistic grammar model with attributes on the terminal nodes. It combines i) a probabilistic context free grammar (PCFG), and ii) contextual relations defined on an Markov Random Field (MRF), *i.e.*, the horizontal links among the nodes. The PCFG represents the hierarchical decomposition from scenes (top level) to objects (bottom level) by a set of terminal and non-terminal nodes, whereas contextual relations encode the spatial and functional relations through horizontal links. The structure of S-AOG is shown in Figure 3.2.

Formally, an S-AOG is defined as a 5-tuple: $\mathcal{G} = \langle S, V, R, P, E \rangle$, where we use notations S the root node of the scene grammar, V the vertex set, R the production rules, P the probability model defined on the attributed S-AOG, and E the contextual relations represented as horizontal links between nodes in the same layer. [1]

Vertex Set V can be decomposed into a finite set of non-terminal and terminal nodes: $V = V_{NT} \cup V_T$.

[1] We use the term "vertices" instead of "symbols" (in the traditional definition of PCFG) to be consistent with the notations in graphical models.

- $V_{NT} = V^{And} \cup V^{Or} \cup V^{Set}$. The non-terminal nodes consists of three subsets. i) A set of **And-nodes** V^{And}, in which each node represents a decomposition of a larger entity (*e.g.*, a bedroom) into smaller components (*e.g.*, walls, furniture and supported objects). ii) A set of **Or-nodes** V^{Or}, in which each node branches to alternative decompositions (*e.g.*, an indoor scene can be a bedroom or a living room), enabling the algorithm to reconfigure a scene. iii) A set of **Set nodes** V^{Set}, in which each node represents a nested And-Or relation: a set of Or-nodes serving as child branches are grouped by an And-node, and each child branch may include different numbers of objects.

- $V_T = V_T^r \cup V_T^a$. The terminal nodes consists of two subsets of nodes: regular nodes and address nodes. i) A **regular terminal node** $v \in V_T^r$ represents a spatial entity in a scene (*e.g.*, an office chair in a bedroom) with attributes. In this work, the attributes include internal attributes A_{int} of object sizes (w, l, h), external attributes A_{ext} of object position (x, y, z) and orientation ($x - y$ plane) θ, and sampled human positions A_h. ii) To avoid excessively dense graphs, an **address terminal node** $v \in V_T^a$ is introduced to encode interactions that only occur in a certain context but are absent in all others [Fri03]. It is a pointer to regular terminal nodes, taking values in the set $V_T^r \cup \{\text{nil}\}$, representing supporting or grouping relations as shown in Figure 3.2.

Contextual Relations E among nodes are represented by the horizontal links in S-AOG forming MRFs on the terminal nodes. To encode the contextual relations, we define different types of potential functions for different cliques. The contextual relations $E = E_f \cup E_o \cup E_g \cup E_r$ are divided into four subsets: i) relations among furniture E_f; ii) relations between supported objects and their supporting objects E_o (*e.g.*, a monitor on a desk); iii) relations between objects of a functional pair E_g (*e.g.*, a chair and a desk); and iv) relations between furniture and the room E_r. Accordingly, the cliques formed in the terminal layer could also be divided into four subsets: $C = C_f \cup C_o \cup C_g \cup C_r$. Instead of directly capturing the object-object relations, we compute the potentials using affordances as a bridge to characterize the object-human-object relations.

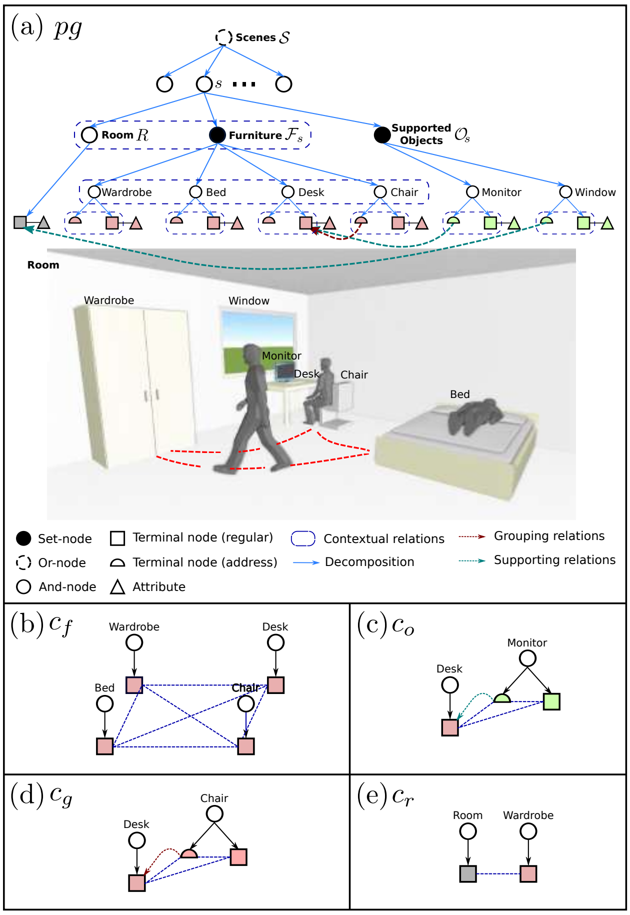


Figure 3.3: (a) A simplified example of a parse graph of a bedroom. The terminal nodes of the parse graph form an MRF in the terminal layer. Cliques are formed by the contextual relations projected to the terminal layer. Examples of the four types of cliques are shown in (b)-(e), representing four different types of contextual relations.

A hierarchical parse tree pt is an instantiation of the S-AOG by selecting a child node for the Or-nodes as well as determining the state of each child node for the Set-nodes. A parse graph pg consists of a parse tree pt and a number of contextual relations E on the parse tree: $pg = (pt, E_{pt})$. Figure 3.3 illustrates a simple example of a parse graph and four types of cliques formed in the terminal layer.

3.4 Probabilistic Formulation of S-AOG

A scene configuration is represented by a parse graph pg, including objects in the scene and associated attributes. The prior probability of pg generated by an S-AOG parameterized by Θ is formulated as a Gibbs distribution:

$$\begin{aligned} p(pg|\Theta) &= \frac{1}{Z}\exp\{-\mathcal{E}(pg|\Theta)\} && (3.1) \\ &= \frac{1}{Z}\exp\{-\mathcal{E}(pt|\Theta) - \mathcal{E}(E_{pt}|\Theta)\}, && (3.2) \end{aligned}$$

where $\mathcal{E}(pg|\Theta)$ is the energy function of a parse graph, $\mathcal{E}(pt|\Theta)$ is the energy function of a parse tree, and $\mathcal{E}(E_{pt}|\Theta)$ is the energy term of the contextual relations.

$\mathcal{E}(pt|\Theta)$ can be further decomposed into the energy functions of different types of non-terminal nodes, and the energy functions of internal attributes of both regular and address terminal nodes:

$$\mathcal{E}(pt|\Theta) = \underbrace{\sum_{v \in V^{Or}} \mathcal{E}_{\Theta}^{Or}(v) + \sum_{v \in V^{Set}} \mathcal{E}_{\Theta}^{Set}(v)}_{\text{non-terminal nodes}} + \underbrace{\sum_{v \in V_T^r} \mathcal{E}_{\Theta}^{A_{in}}(v)}_{\text{terminal nodes}}, \tag{3.3}$$

where the choice of the child node of an Or-node $v \in V^{Or}$ and the child branch of a Set-node $v \in V^{Set}$ follow different multinomial distributions. Since the And-nodes are deterministically expanded, we do not have an energy term for the And-nodes here. The internal attributes A_{in} (size) of terminal nodes follows a non-parametric probability distribution learned by kernel density estimation.

$\mathcal{E}(E_{pt}|\Theta)$ combines the potentials of the four types of cliques formed in the terminal layer, integrating human attributes and external attributes of regular terminal nodes:

$$\begin{aligned} p(E_{pt}|\Theta) &= \frac{1}{Z}\exp\{-\mathcal{E}(E_{pt}|\Theta)\} && (3.4) \\ &= \prod_{c \in C_f} \phi_f(c) \prod_{c \in C_o} \phi_o(c) \prod_{c \in C_g} \phi_g(c) \prod_{c \in C_r} \phi_r(c). && (3.5) \end{aligned}$$

Human Centric Potential Functions:

- Potential function $\phi_f(c)$ is defined on relations between furniture (Figure 3.3(b)). The clique $c = \{f_i\} \in C_f$ includes all the terminal nodes representing furniture:

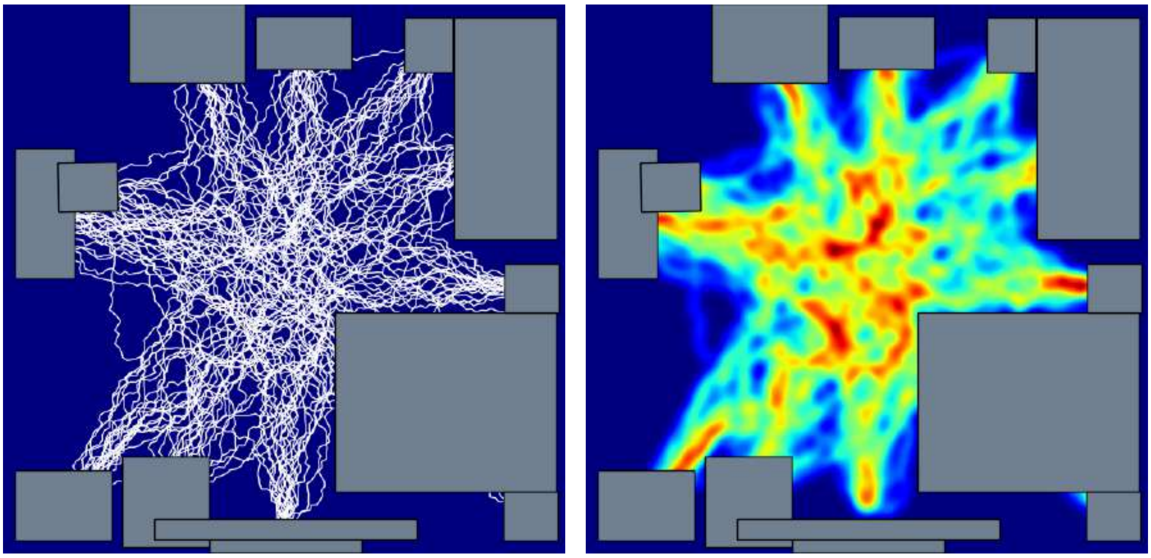
$$\phi_f(c) = \frac{1}{Z} \exp\{-\lambda_f \cdot \langle \sum_{f_i \neq f_j} l_{\text{col}}(f_i, f_j), l_{\text{ent}}(c) \rangle\}, \tag{3.6}$$

  where λ_f is a weight vector, $< \cdot, \cdot >$ denotes a vector, and the cost function $l_{\text{col}}(f_i, f_j)$ is the overlapping volume of the two pieces of furniture, serving as the penalty of collision. The cost function $l_{\text{ent}}(c) = -H(\Gamma) = \Sigma_i p(\gamma_i) \log p(\gamma_i)$ yields better utility of the room space by sampling human trajectories, where Γ is the set of planned trajectories in the room, and $H(\Gamma)$ is the entropy. The trajectory probability map is first obtained by planning a trajectory γ_i from the center of every piece of furniture to another one using bi-directional rapidly-exploring random tree (RRT) [LaV98], which forms a heatmap. The entropy is computed from the heatmap as shown in Figure 3.4.

- Potential function $\phi_o(c)$ is defined on relations between a supported object and the supporting furniture (Figure 3.3(c)). A clique $c = \{f, a, o\} \in C_o$ includes a supported object terminal node o, the address node a connected to the object, and the furniture terminal node f pointed by a:

$$\phi_o(c) = \frac{1}{Z} \exp\{-\lambda_o \cdot \langle l_{\text{hum}}(f, o), l_{\text{add}}(a) \rangle\}, \tag{3.7}$$

  where the cost function $l_{\text{hum}}(f, o)$ defines the human usability cost—a favorable human position should enable an agent to access or use both the furniture and the object. To compute the usability cost, human positions h_i^o are first sampled based on position, orientation, and the affordance map of the supported object. Given a piece of furniture,


(a) Planned trajectories

(b) Probability map

Figure 3.4: Given a scene configuration, we use bi-directional RRT to plan from every piece of furniture to another, generating a human activity probability map.

the probability of the human positions is then computed by:

$$l_{\text{hum}}(f, o) = \max_i p(h_i^o | f). \tag{3.8}$$

The cost function $l_{\text{add}}(a)$ is the negative log probability of an address node $v \in V_T^a$, treated as a certain regular terminal node, following a multinomial distribution.

- Potential function $\phi_g(c)$ is defined on functional grouping relations between furniture (Figure 3.3(d)). A clique $c = \{f_i, a, f_j\} \in C_g$ consists of terminal nodes of a core functional furniture f_i, pointed by the address node a of an associated furniture f_j. The grouping relation potential is defined similarly to the supporting relation potential

$$\phi_g(c) = \frac{1}{Z} \exp\{-\lambda_c \cdot \langle l_{\text{hum}}(f_i, f_j), l_{\text{add}}(a) \rangle\}. \tag{3.9}$$

Other Potential Functions:

- Potential function $\phi_r(c)$ is defined on relations between the room and furniture (Figure 3.3(e)). A clique $c = \{f, r\} \in C_r$ includes a terminal node f and r representing a piece of furniture and a room, respectively. The potential is defined as

$$\phi_r(c) = \frac{1}{Z}\exp\{-\lambda_r \cdot \langle l_{\text{dis}}(f,r), l_{\text{ori}}(f,r)\rangle\}, \tag{3.10}$$

where the distance cost function is defined as $l_{\text{dis}}(f,r) = -\log p(d|\Theta)$, in which $d \sim \ln\mathcal{N}(\mu,\sigma^2)$ is the distance between the furniture and the nearest wall modeled by a log normal distribution. The orientation cost function is defined as $l_{\text{ori}}(f,r) = -\log p(\theta|\Theta)$, where $\theta \sim p(\mu,\kappa) = \frac{e^{\kappa\cos(x-\mu)}}{2\pi I_0(\kappa)}$ is the relative orientation between the model and the nearest wall modeled by a von Mises distribution.

3.5 Learning S-AOG

We use the SUNCG dataset [SYZ17b] as training data. It contains over 45K different scenes with manually created realistic room and furniture layouts. We collect the statistics of room types, room sizes, furniture occurrences, furniture sizes, relative distances, orientations between furniture and walls, furniture affordance, grouping occurrences, and supporting relations. The parameters Θ of the probability model P can be learned in a supervised way by maximum likelihood estimation (MLE).

Weights of Loss Functions: Recall that the probability distribution of cliques formed in the terminal layer is

$$\begin{aligned} p(E_{pt}|\Theta) &= \frac{1}{Z}\exp\{-\mathcal{E}(E_{pt}|\Theta)\} \qquad (3.11)\\ &= \frac{1}{Z}\exp\{-\langle\lambda, l(E_{pt})\rangle\}, \qquad (3.12)\end{aligned}$$

where λ is the weight vector and $l(E_{pt})$ is the loss vector given by four different types of potential functions.

To learn the weight vector, the standard MLE maximizes the average log-likelihood:

$$\mathcal{L}(E_{pt}|\Theta) = -\frac{1}{N}\sum_{n=1}^{N}\langle\lambda, l(E_{pt_n})\rangle - \log Z. \tag{3.13}$$

This is usually maximized by following the gradient:

$$\frac{\partial \mathcal{L}(E_{pt}|\Theta)}{\partial \lambda} = -\frac{1}{N}\sum_{n=1}^{N} l(E_{pt_n}) - \frac{\partial \log Z}{\partial \lambda} \tag{3.14}$$

$$= -\frac{1}{N}\sum_{n=1}^{N} l(E_{pt_n}) - \frac{\partial \log \sum_{pt} \exp\{-\langle \lambda, l(E_{pt})\rangle\}}{\partial \lambda} \tag{3.15}$$

$$= -\frac{1}{N}\sum_{n=1}^{N} l(E_{pt_n}) + \sum_{pt} \frac{1}{Z} \exp\{-\langle \lambda, l(E_{pt})\rangle\} l(E_{pt}) \tag{3.16}$$

$$= -\frac{1}{N}\sum_{n=1}^{N} l(E_{pt_n}) + \frac{1}{\widetilde{N}}\sum_{\widetilde{n}=1}^{\widetilde{N}} l(E_{pt_{\widetilde{n}}}), \tag{3.17}$$

where $\{E_{pt_{\widetilde{n}}}\}_{\widetilde{n}=1,\cdots,\widetilde{N}}$ is the set of synthesized examples from the current model.

It is usually computationally infeasible to sample a Markov chain that burns into an *equilibrium distribution* at every iteration of gradient ascent. Hence, instead of waiting for the Markov chain to converge, we adopt the contrastive divergence (CD) learning that follows the gradient of difference of two divergences [Hin02]

$$\mathrm{CD}_{\widetilde{N}} = \mathrm{KL}(p_0||p_\infty) - \mathrm{KL}(p_{\widetilde{n}}||p_\infty), \tag{3.18}$$

where $\mathrm{KL}(p_0||p_\infty)$ is the Kullback-Leibler divergence between the data distribution p_0 and the model distribution p_∞, and $p_{\widetilde{n}}$ is the distribution obtained by a Markov chain started at the data distribution and run for a small number $\widetilde{n}$ of steps. In this work, we set $\widetilde{n} = 1$.

Contrastive divergence learning has been applied effectively to addressing various problems; one of the most notable work is in the context of Restricted Boltzmann Machines [HS06]. Both theoretical and empirical evidences shows its efficiency while keeping bias typically very small [CH05]. The gradient of the contrastive divergence is given by

$$\begin{aligned}\frac{\partial \mathrm{CD}_{\widetilde{N}}}{\partial \lambda} =& \frac{1}{N}\sum_{n=1}^{N} l(E_{pt_n}) - \frac{1}{\widetilde{N}}\sum_{\tilde{n}=1}^{\widetilde{N}} l(E_{pt_{\tilde{n}}}) \\ & - \frac{\partial p_{\tilde{n}}}{\partial \lambda}\frac{\partial \mathrm{KL}(p_{\tilde{n}}||p_{\infty})}{\partial p_{\tilde{n}}}.\end{aligned} \tag{3.19}$$

Extensive simulations [Hin02] showed that the third term can be safely ignored since it is small and seldom opposes the resultant of the other two terms.

Finally, the weight vector is learned by gradient descent computed by generating a small number $\widetilde{N}$ of examples from the Markov chain

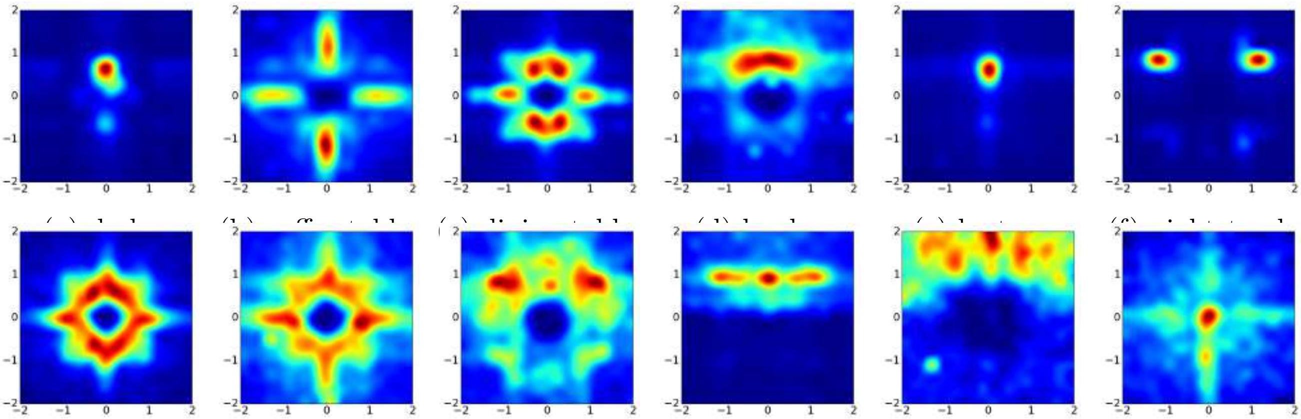
$$\lambda_{t+1} = \lambda_t - \eta_t \frac{\partial \mathrm{CD}_{\widetilde{N}}}{\partial \lambda} \tag{3.20}$$

$$= \lambda_t + \eta_t \left(\frac{1}{\widetilde{N}}\sum_{\tilde{n}=1}^{\widetilde{N}} l(E_{pt_{\tilde{n}}}) - \frac{1}{N}\sum_{n=1}^{N} l(E_{pt_n}) \right). \tag{3.21}$$

Branching Probabilities: The MLE of the branch probabilities ρ_i of Or-nodes, Set-nodes and address terminal nodes is simply the frequency of each alternative choice [ZM07]: $\rho_i = \#(v \to u_i) / \sum_{j=1}^{n(v)} \#(v \to u_j)$.

Grouping Relations: The grouping relations are hand-defined (*i.e.*, nightstands are associated with beds, chairs are associated with desks and tables). The probability of occurrence is learned as a multinomial distribution, and the supporting relations are automatically extracted from SUNCG.

Room Size and Object Sizes: The distribution of the room size and object size among all the furniture and supported objects is learned as a non-parametric distribution. We first extract the size information from the 3D models inside SUNCG dataset, and then fit a non-parametric distribution using kernel density estimation. The distances and relative orientations of the furniture and objects to the nearest wall are computed and fitted into a log normal and a mixture of von Mises distributions, respectively.



(g) fruit bowl (h) vase (i) floor lamp (j) wall lamp (k) fireplace (l) ceiling fan

Figure 3.5: Examples of the learned affordance maps. Given the object positioned in the center facing upwards, *i.e.*, coordinate of $(0, 0)$ facing direction $(0, 1)$, the maps show the distributions of human positions. The affordance maps accurately capture the subtle differences among desks, coffee tables, and dining tables. Some objects are orientation sensitive, *e.g.*, books, laptops, and night stands, while some are orientation invariant, *e.g.*, fruit bowls and vases.

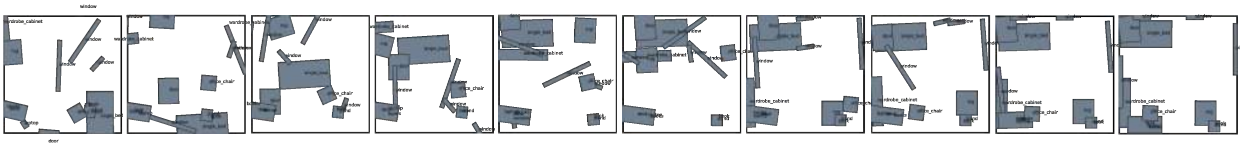


Figure 3.6: MCMC sampling process (from left to right) of scene configurations with simulated annealing.

Affordances: We learn the affordance maps of all the furniture and supported objects by computing the heatmap of possible human positions. These position include annotated humans, and we assume that the center of chairs, sofas, and beds are positions that humans often visit. By accumulating the relative positions, we get reasonable affordance maps as non-parametric distributions as shown in Figure 3.5.

3.6 Synthesizing Scene Configurations

Synthesizing scene configurations is accomplished by sampling a parse graph pg from the prior probability $p(pg|\Theta)$ defined by the S-AOG. The structure of a parse tree pt (*i.e.*, the selection of Or-nodes and child branches of Set-nodes) and the internal attributes (sizes) of objects can be easily sampled from the closed-form distributions or non-parametric distributions. However, the external attributes (positions and orientations) of objects are constrained by multiple potential functions, hence they are too complicated to be directly sampled from. Here, we utilize a Markov chain Monte Carlo (MCMC) sampler to draw a typical state in the distribution. The process of each sampling can be divided into two major steps:

1. Directly sample the structure of pt and internal attributes A_{in}: (i) sample the child node for the Or-nodes; (ii) determine the state of each child branch of the Set-nodes; and (iii) for each regular terminal node, sample the sizes and human positions from learned distributions.

2. Use an MCMC scheme to sample the values of address nodes V^a and external attributes A_{ex} by making proposal moves. A sample will be chosen after the Markov chain converges.

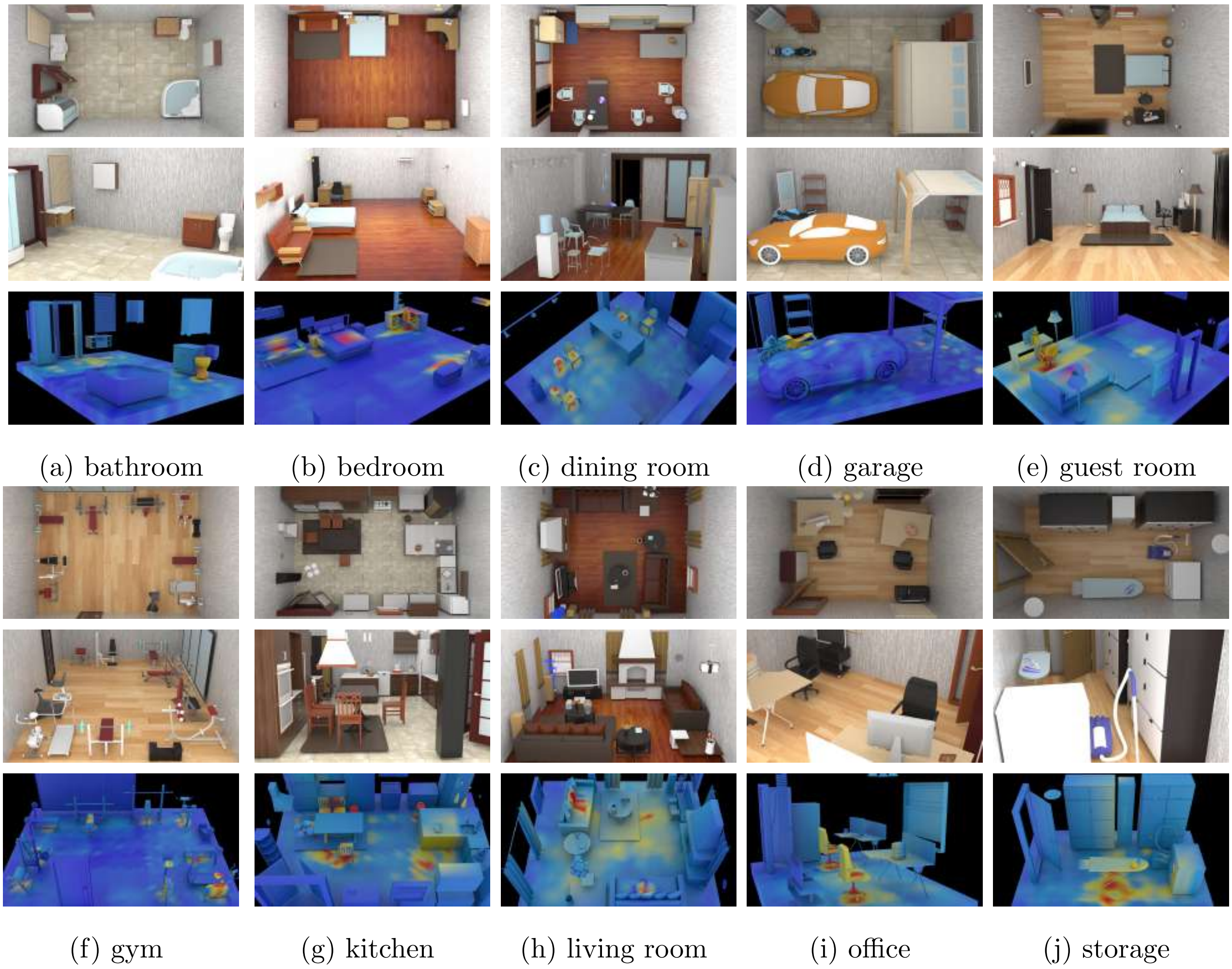
We design two simple types of Markov chain dynamics which are used at random with probabilities $q_i, i = 1, 2$ to make proposal moves:

- Dynamics q_1: translation of objects. This dynamic chooses a regular terminal node, and samples a new position based on the current position x: $x \to x + \delta x$, where δx follows a bivariate normal distribution.

- Dynamics q_2: rotation of objects. This dynamic chooses a regular terminal node, and samples a new orientation based on the current orientation of the object: $\theta \to \theta + \delta\theta$, where $\delta\theta$ follows a normal distribution.

Adopting the Metropolis-Hastings algorithm, the proposed new parse graph pg' is accepted according to the following acceptance probability:

$$\begin{aligned}\alpha(pg'|pg, \Theta) &= \min(1, \frac{p(pg'|\Theta)p(pg|pg')}{p(pg|\Theta)p(pg'|pg)}) &\quad (3.22)\\ &= \min(1, \exp(\mathcal{E}(pg|\Theta) - \mathcal{E}(pg'|\Theta))), &\quad (3.23)\end{aligned}$$



(a) bathroom (b) bedroom (c) dining room (d) garage (e) guest room

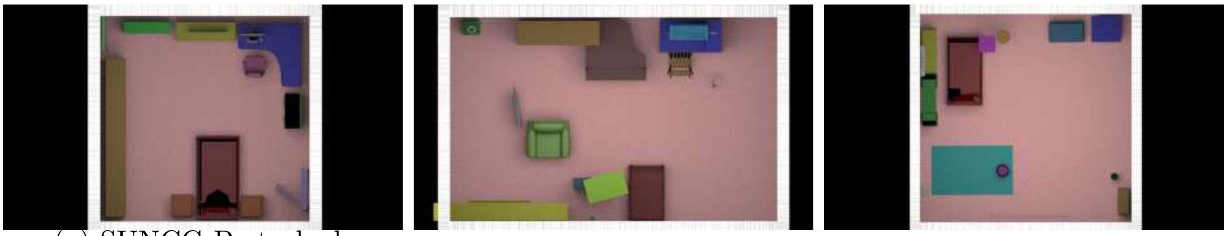
(f) gym (g) kitchen (h) living room (i) office (j) storage

Figure 3.7: Examples of scenes in ten different categories. Top: top-view. Middle: a side-view. Bottom: affordance heatmap.

where the proposal probability rate is canceled since the proposal moves are symmetric in probability. A simulated annealing scheme is adopted to obtain samples with high probability as shown in Figure 3.6.

3.7 Experiments

We design three experiments based on different criteria: i) visual similarity to manually constructed scenes, ii) the accuracy of affordance maps for the synthesized scenes, and iii) functionalities and naturalness of the synthesized scenes. The first experiment compares our



(a) SUNCG Perturbed

(b) Yu *et al.* [YYT11]

(c) Ours

Figure 3.8: Top-view segmentation maps for classification.

method with a state-of-the-art room arrangement method; the second experiment measures the synthesized affordances; the third one is an ablation study. Overall, the experiments show that our algorithm can robustly sample a large variety of realistic scenes that exhibits naturalness and functionality.

Layout Classification. To quantitatively evaluate the visual realism, we trained a classifier on the top-view segmentation maps of synthesized scenes and SUNCG scenes. Specifically, we train a ResNet-152 [HZR16] to classify top view layout segmentation maps (synthesized vs. SUNCG). Examples of top-view segmentation maps are shown in Figure 3.8. The reason to use segmentation maps is that we want to evaluate the room layout excluding rendering factors such as object materials. We use two methods for comparison: i) a state-of-the-art furniture arrangement optimization method proposed by Yu *et al.* [YYT11], and ii) slight perturbation of SUNCG scenes by adding small Gaussian noise ($e.g. \mu = 0, \sigma = 0.1$) to the layout. The room arrangement algorithm proposed by [YYT11] takes one pre-fixed input room and re-organizes the room. 1500 scenes are randomly selected for each method and SUNCG: 800 for training, 200 for validation, and 500 for testing. As shown in Table 3.1, the classifier successfully distinguishes Yu *et al.*vs. SUNCG with an accuracy of 87.49%. Our method achieves a better performance of 76.18%, exhibiting a higher realism and larger

Table 3.1: Classification results on segmentation maps of synthesized scenes using different methods vs. SUNCG.

Method	Yu *et al.* [YYT11]	SUNCG Perturbed	Ours
Accuracy(%) ↓	87.49	63.69	76.18

Table 3.2: Comparison between affordance maps computed from our samples and real data

Metric	Bathroom	Bedroom	Dining Room	Garage	Guest Room	Gym	Kitchen	Living Room	Office	Storage
Total variation	0.431	0.202	0.387	0.237	0.175	0.278	0.227	0.117	0.303	0.708
Hellinger distance	0.453	0.252	0.442	0.284	0.212	0.294	0.251	0.158	0.318	0.703

Table 3.3: Human subjects' ratings (1-5) of the sampled layouts based on functionality (top) and naturalness (bottom)

Method	Bathroom	Bedroom	Dining Room	Garage	Guest Room	Gym	Kitchen	Living Room	Office	Storage
no-context	1.12 ± 0.33	1.25 ± 0.43	1.38 ± 0.48	1.75 ± 0.66	1.50 ± 0.50	3.75 ± 0.97	2.38 ± 0.48	1.50 ± 0.87	1.62 ± 0.48	1.75 ± 0.43
object	3.12 ± 0.60	3.62 ± 1.22	2.50 ± 0.71	3.50 ± 0.71	2.25 ± 0.97	3.62 ± 0.70	3.62 ± 0.70	3.12 ± 0.78	1.62 ± 0.48	4.00 ± 0.71
Yu *et al.* [YYT11]	3.61 ± 0.52	4.15 ± 0.25	3.15 ± 0.40	3.59 ± 0.51	2.58 ± 0.31	2.03 ± 0.56	3.91 ± 0.98	4.62 ± 0.21	3.32 ± 0.81	2.58 ± 0.64
ours	4.58 ± 0.86	4.67 ± 0.90	3.33 ± 0.90	3.96 ± 0.79	3.25 ± 1.36	4.04 ± 0.79	4.21 ± 0.87	4.58 ± 0.86	3.67 ± 0.75	4.79 ± 0.58
no-context	1.00 ± 0.00	1.00 ± 0.00	1.12 ± 0.33	1.38 ± 0.70	1.12 ± 0.33	1.62 ± 0.86	1.00 ± 0.00	1.25 ± 0.43	1.12 ± 0.33	1.00 ± 0.00
object	2.88 ± 0.78	3.12 ± 1.17	2.38 ± 0.86	3.00 ± 0.71	2.50 ± 0.50	3.38 ± 0.86	3.25 ± 0.66	2.50 ± 0.50	1.25 ± 0.43	3.75 ± 0.66
Yu *et al.* [YYT11]	4.00 ± 0.52	3.85 ± 0.92	3.27 ± 1.01	2.99 ± 0.25	3.52 ± 0.93	2.14 ± 0.63	3.89 ± 0.90	3.31 ± 0.29	2.77 ± 0.67	2.96 ± 0.41
ours	4.21 ± 0.71	4.25 ± 0.66	3.08 ± 0.70	3.71 ± 0.68	3.83 ± 0.80	4.17 ± 0.75	4.38 ± 0.56	3.42 ± 0.70	3.25 ± 0.72	4.54 ± 0.71

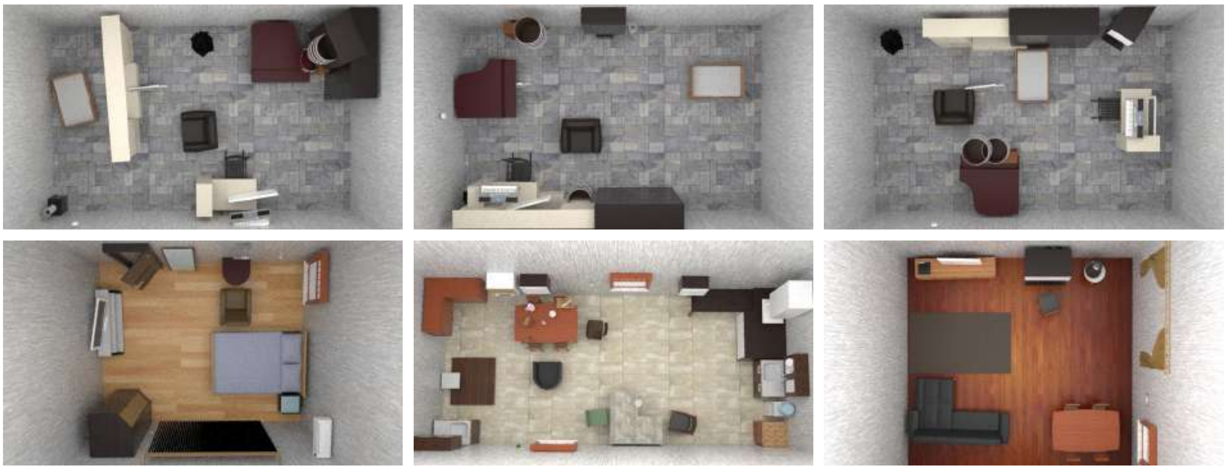


Figure 3.9: **Top**: previous methods [YYT11] only re-arranges a given input scene with a fixed room size and a predefined set of objects. **Bottom**: our method samples a large variety of scenes.

variety. This result indicates our method is much more visually similar to real scenes than the comparative scene optimization method. Qualitative comparisons of Yu *et al.*and our method are shown in Figure 3.9.

Affordance Maps Comparison. We sample 500 rooms of 10 different scene categories summarized in Table 3.2. For each type of room, we compute the affordance maps of the objects in the synthesized samples, and calculate both the total variation distances and Hellinger distances between the affordance maps computed from the synthesized samples and the SUNCG dataset. The two distributions are similar if the distance is close to 0. Most sampled scenes using the proposed method show similar affordance distributions to manually created ones from SUNCG. Some scene types (*e.g.*Storage) show a larger distance since they do not exhibit clear affordances. Overall, the results indicate that affordance maps computed from the synthesized scenes are reasonably close to the ones computed from manually constructed scenes by artists.

Functionality and naturalness. Three methods are used for comparison: (i) direct sampling of rooms according to the statistics of furniture occurrence without adding contextual relation, (ii) an approach that only models object-wise relations by removing the human constraints in our model, and (iii) the algorithm proposed by Yu *et al.* [YYT11]. We showed the sampled layouts using three methods to 4 human subjects. Subjects were told the room category in advance, and instructed to rate given scene layouts without knowing the method used to generate the layouts. For each of the 10 room categories, 24 samples were randomly selected using our method and [YYT11], and 8 samples were selected using both the object-wise modeling method and the random generation. The subjects evaluated the layouts based on two criteria: (i) functionality of the rooms, *e.g.*, can the "bedroom" satisfies a human's needs for daily life; and (ii) the naturalness and realism of the layout. Scales of responses range from 1 to 5, with 5 indicating perfect functionalilty or perfect naturalness and realism. The mean ratings and the standard deviations are summarized in Table 3.3. Our approach outperforms the three methods in both criteria, demonstrating the ability to sample a functionally reasonable and realistic scene layout. More qualitative results are shown in Figure 3.7.

Complexity of synthesis. The time complexity is hard to measure since MCMC sampling is adopted. Empirically, it takes about 20-40 minutes to sample an interior layout (20000 iterations of MCMC), and roughly 12-20 minutes to render a 640×480 image on a normal PC. The rendering speed depends on settings related to illumination, environments, and the size of the scene, *etc.*.

3.8 Conclusion

We propose a novel general framework for human-centric indoor scene synthesis by sampling from a spatial And-Or graph. The experimental results demonstrate the effectiveness of our approach over a large variety of scenes based on different criteria. In the future, to synthesize physically plausible scenes, a physics engine should be integrated. We hope the synthesized data can contribute to the broad AI community.

Part II

Interaction: Human-like 3D Interaction Understanding

CHAPTER 4

Human-object Interaction and Affordance

In this chapter, we study the human-object interaction. Given a single image where humans interact with the scene, the machine is expected to understand the human actions and utilize the interaction between humans and objects to ease the ambiguities of single image parsing. The interaction could serve as general commonsense knowledge for understanding the actions and events, helping the generalization to various environments.

4.1 Joint Scene Parsing with 3D Human-object Interaction

In this section, we propose a new 3D holistic^{++} scene understanding problem, which jointly tackles two tasks from a single-view image: (i) holistic scene parsing and reconstruction—3D estimations of object bounding boxes, camera pose, and room layout, and (ii) 3D human pose estimation. The intuition behind is to leverage the coupled nature of these two tasks to improve the granularity and performance of scene understanding. We propose to exploit two critical and essential connections between these two tasks: (i) human-object interaction (HOI) to model the fine-grained relations between agents and objects in the scene, and (ii) physical commonsense to model the physical plausibility of the reconstructed scene. The optimal configuration of the 3D scene, represented by a parse graph, is inferred using Markov chain Monte Carlo (MCMC), which efficiently traverses through the non-differentiable joint solution space. Experimental results demonstrate that the proposed algorithm significantly improves the performance of the two tasks on three datasets, showing an improved generalization ability.

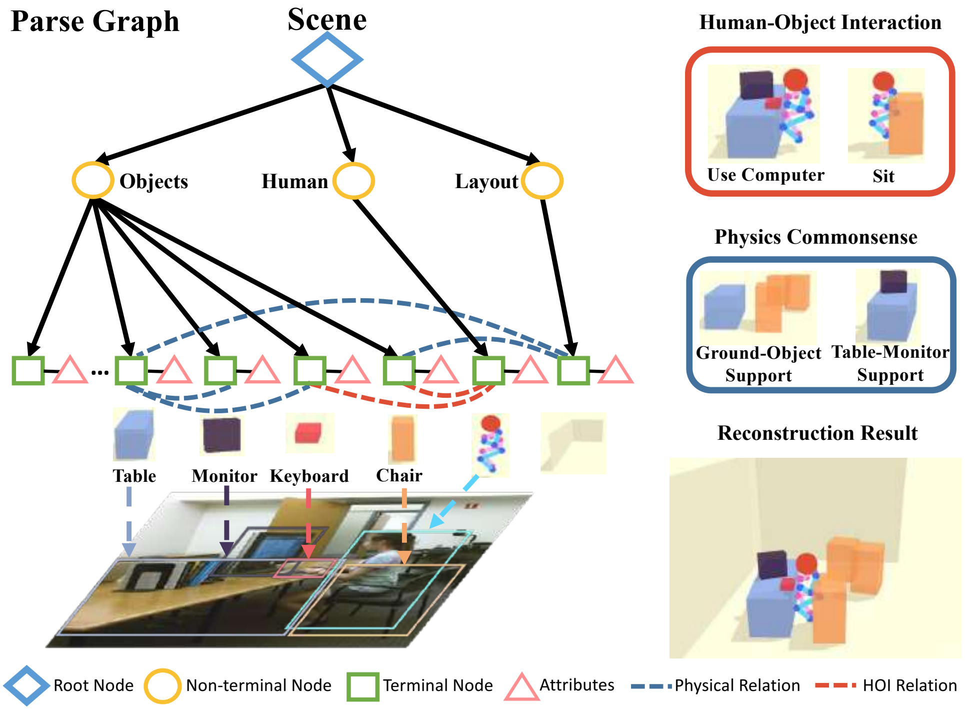


Figure 4.1: **holistic^{++} scene understanding** task requires to jointly recover a parse graph that represents the scene, including human poses, objects, camera pose, and room layout, all in 3D. Reasoning human-object interaction (HOI) helps reconstruct the detailed spatial relations between humans and objects. Physical commonsense (*e.g.*, physical property, plausibility, and stability) further refines relations and improves predictions.

4.1.1 Introduction

Humans, even young infants, are adept at perceiving and understanding complex indoor scenes. Such an incredible vision system not only relies on the data-driven pattern recognition but also roots from the visual reasoning system, known as the core knowledge [SK07], that facilitates the 3D holistic scene understanding tasks. Consider a typical indoor scene shown in Figure 4.1 where a person sits in an office. We can effortlessly extract rich knowledge from the static scene, including 3D room layout, 3D position of all the objects and agents, and correct human-object interaction (HOI) relations in a physically plausible manner. In fact, psychology studies have established that even infants employ at least two constraints—HOI and physical commonsense—in perceiving occlusions [THK87, KS83], tracking small objects even if contained by other objects [FC03], realizing object permanence [BSW85], recognizing rational HOI [Woo99, SCS13], understanding intuitive physics [GBK02a, Nee97, Bai04], and using exploratory play to understand the environment [SF15]. All the evidence calls for a treatment to integrate HOI and physical commonsense with a modern computer vision system for scene understanding.

In contrast, few attempts have been made to achieve this goal. This challenge is difficult partially due to the fact that the algorithm has to *jointly* accomplish both 3D holistic scene understanding task and the 3D human pose estimation task in a *physically plausible* fashion. Since this task is beyond the scope of holistic scene understanding in the literature, we define this comprehensive task as *holistic*$^{++}$ *scene understanding*—to simultaneously estimate human pose, objects, room layout, and camera pose, all in 3D.

Based on one single-view image, existing work either focuses only on 3D holistic scene understanding [HQZ18, ZLH17, BRG16, SYZ17a] or 3D human pose estimation [ZWM17, RKS12, FXW18]. Although one can achieve an impressive performance in a single task by training with an enormous amount of annotated data, we, however, argue that these two tasks are intertwined tightly since the indoor scenes are invented and constructed by human designs to support the daily activities, generating affordance for rich tasks and human activities [Gib79].

To solve the proposed *holistic*$^{++}$ *scene understanding* task, we attempt to address four fundamental challenges:

1. How to utilize the coupled nature of human pose estimation and holistic scene understanding, and make them benefit each other? How to reconstruct the scene with complex human activities and interactions?

2. How to constrain the solution space of the 3D estimations from a single 2D image?

3. How to make a physically plausible and stable estimation for complex scenes with human agents and objects?

4. How to improve the generalization ability to achieve a more robust reconstruction across different datasets?

To address the first two challenges, we take a novel step to incorporate **HOI** as constraints for **joint parsing** of both 3D human pose and 3D scene. The integration of HOI is inspired by crucial observations of human 3D scene perception, which are challenging for existing systems. Take Figure 4.1 as an example; humans are able to impose a constraint and infer the relative position and orientation between the girl and chair by recognizing the girl is sitting in the chair. Similarly, such a constraint can help to recover the small objects (*e.g.*, recognizing keyboard by detecting the girl is using a computer in Figure 4.1). By learning HOI priors and using the inferred HOI as visual cues to adjust the fine-grained spatial relations between human and scene (objects and room layout), the geometric ambiguity (3D estimation solution space) in the single-view reconstruction would be largely eased, and the reconstruction performances of both tasks would be improved.

To address the third challenge, we incorporate **physical commonsense** into the proposed method. Specifically, the proposed method reasons about the physical relations (*e.g.*, support relation) and penalizes the physical violations to predict a physically plausible and stable 3D scene. The HOI and physical commonsense serve as **general prior** knowledge across different datasets, thus help address the fourth issue.

To jointly parse 3D human pose and 3D scene, we represent the configuration of an indoor scene by a parse graph shown in Figure 4.1, which consists of a parse tree with hierarchical structure and a MRF over the terminal nodes, capturing the rich contextual relations among human, objects, and room layout. The optimal parse graph to reconstruct both the 3D scene and human poses is achieved by a MAP estimation, where the prior characterizes the prior distribution of the contextual HOI and physical relations among the nodes. The likelihood measures the similarity between (i) the detection results directly from 2D object and pose detector, and (ii) the 2D results projected from the 3D parsing results. The parse graph can be iteratively optimized by sampling an MCMC with simulated annealing based on posterior probability. The joint optimization relies less on a specific training dataset since it benefits from the prior of HOI and physical commonsense which are almost invariant across environments and datasets, and other knowledge learned from well-defined vision task (*e.g.*, 3D pose estimation, scene reconstruction), improving the generalization ability significantly across different datasets compared with purely data-driven methods.

Experimental results on PiGraphs [SCH16], Watch-n-Patch [WZS15], and SUN RGB-D [SLX15] demonstrate that the proposed method outperforms state-of-the-art methods for both 3D scene reconstruction and 3D pose estimation. Moreover, the ablative analysis shows that the HOI prior improves the reconstruction, and the physical common sense helps to make physically plausible predictions.

This work makes four major contributions:

1. We propose a new *holistic*$^{++}$ *scene understanding* task with a computational framework to jointly infer human poses, objects, room layout, and camera pose, all in 3D.

2. We integrate HOI to bridge the human pose estimation and the scene reconstruction, reducing geometric ambiguities (solution space) of the single-view reconstruction.

3. We incorporate physical commonsense, which helps to predict physically plausible scenes and improve the 3D localization of both humans and objects.

4. We demonstrate the joint inference improves the performance of each sub-module and achieves better generalization ability across various indoor scene datasets compared with purely data-driven methods.

4.1.2 Related Work

Single-view 3D Human Pose Estimation: Previous methods on 3D pose estimation can be divided into two streams: (i) directly learning 3D pose from a 2D image [SRA12, LC14], and (ii) cascaded frameworks that first perform 2D pose estimation and then reconstruct 3D pose from the estimated 2D joints [ZWM17, MSS17, RKS12, WXL16, CLO16, TRA17]. Although these researches have produced impressive results in scenarios with relatively clean background, the problem of estimating the 3D pose in a typical indoor scene with arbitrary cluttered objects has rarely been discussed. Recently, Zanfir *et al.* [ZMS18] adopts constraints of ground plane support and volume occupancy by multiple people, but the detailed relations between human and scene (objects and layout) are still missing. In contrast, the proposed model not only estimates the 3D poses of multiple people with an absolute scale but also models the physical relations between humans and 3D scenes.

Single-view 3D Scene Reconstruction: Single-view 3D scene reconstruction has three main approaches: (i) Predict room layouts by extracting geometric features to rank 3D cuboids proposals [ZLH17, SYZ17a, ISS17b, ZCS18]. (ii) Align object proposals to RGB or depth image by treating objects as geometric primitives or CAD models [BRG16, SX14, ZLX14]. (iii) Joint estimation of the room layout and 3D objects with contexts [SYZ17a, ZZ13, CCP13, ZSY17a, ZLH17]. A more recent work by Huang *et al.* [HQZ18] models the hierarchical structure, latent human context, physical constraints, and jointly optimizes in an analysis-by-synthesis fashion; although human context and functionality were taken into account, indoor scene reconstruction with human poses and HOI remains untouched.

Human-Object Interaction: Reasoning fine-grained human interactions with objects is essential for a more holistic indoor scene understanding as it provides crucial cues for human activities and physical interactions. In robotics and computer vision, prior work has exploited human-object relations in event, object, and scene modeling, but most work focuses on human-object relation detection in images [CLL18, QWJ18a, ML16, KRK11], probabilistic modeling from multiple data sources [WZZ13, SCH14, GKD09], and snapshots generation or scene synthesis [SCH16, MLZ16, QZH18, JQZ18]. Different from all previous work, we use the learned 3D HOI priors to refine the relative spatial relations between human and scene, enabling a top-down prediction of interacted objects.

Physical Commonsense: The ability to infer hidden physical properties is a well-established human cognitive ability [MWF83, KHL17]. By exploiting the underlying physical properties of scenes and objects, recent efforts have demonstrated the capability of estimating both current and future dynamics of static scenes [WYL15, MBR16] and objects [ZZC15], understanding the support relationships and stability of objects [ZZY13], volumetric and occlusion reasoning [SHK12, ZZY15], inferring the hidden force [ZJZ16], and reconstructing the 3D scene [HQX18, DLB18] and 3D pose [ZMS18]. In addition to the physical properties and support relations among objects adopted in previous methods, we further model the physical relations (i) between human and objects, and (ii) between human and room layout, resulting in a physically plausible and stable scene.

4.1.3 Representation

The configuration of an indoor scene is represented by a parse graph $pg = (pt, E)$; see Figure 4.1. It combines a parse tree pt and contextual relations E among the leaf nodes. Here, a parse tree $pt = (V, R)$ includes the vertex set with a three-level hierarchical structure $V = V_r \cup V_m \cup V_t$ and the decomposing rules R, where the root node V_r represents the overall scene, the middle node V_m has three types of nodes (objects, human, and room layout), and the terminal nodes V_t contains child nodes of the middle nodes, representing the detected instances of the parent node in this scene. $E \subset V_t \times V_t$ is the set of contextual relations among the terminal nodes, represented by horizontal links.

Terminal Nodes $\mathbf{V_t}$ in pg can be further decomposed as $V_t = V_{\text{layout}} \cup V_{\text{object}} \cup V_{\text{human}}$. Specifically:

- The room layout $v \in V_{\text{layout}}$ is represented by a 3D bounding box $X^L \in \mathbb{R}^{3\times 8}$ in the world coordinate. The 3D bounding box is parametrized by the node's attributes, including its 3D size $S^L \in \mathbb{R}^3$, center $C^L \in \mathbb{R}^3$, and orientation $Rot(\theta^L) \in \mathbb{R}^{3\times 3}$.

- Each 3D object $v \in V_{\text{object}}$ is represented by a 3D bounding box with its semantic label. We use the same 3D bounding box parameterization as the one for the room layout.

- Each human $v \in V_{\text{human}}$ is represented by 17 3D joints $X^H \in \mathbb{R}^{3\times 17}$ with their action labels. These 3D joints are parametrized by the pose scale $S^H \in \mathbb{R}$, pose center $C^H \in \mathbb{R}^3$ (*i.e.*, hip), local joint position $Rel^H \in \mathbb{R}^{3\times 17}$, and pose orientation $Rot(\theta^H) \in \mathbb{R}^{3\times 3}$. Each person is also attributed by a concurrent action label a, which is a multi-hot vector representing the current actions of this person: one can "sit" and "drink", or "walk" and "make phone call" at the same time.

Contextual Relations E contains three types of relations in the scene $E = \{E_s, E_c, E_{hoi}\}$. Specifically:

- E_s and E_c denote support relation and physical collision, respectively. These two relations penalize the physical violations among objects, between objects and layout, and between human and layout, resulting in a physically plausible and stable prediction.

- E_{hoi} models HOI and provides strong and fine-grained constraints for holistic scene understanding. For instance, if a person is detected as sitting on a chair, we can constrain the relative 3D positions between this person and chair using a pre-learned spatial relation of "sitting."

4.1.4 Probabilistic Formulation

The parse graph pg is a comprehensive interpretation of the observed image I [ZM07]. The goal of the holistic^{++} scene understanding is to infer the optimal parse graph pg^* given I by an MAP estimation:

$$\begin{aligned} pg^* &= \arg\max_{pg} p(pg|I) = \arg\max_{pg} p(pg) \cdot p(I|pg) \\ &= \arg\max_{pg} \frac{1}{Z} \exp\{-\mathcal{E}_{phy}(pg) - \mathcal{E}_{hoi}(pg) - \mathcal{E}(I|pg)\}. \end{aligned} \tag{4.1}$$

We model the joint distribution by a Gibbs distribution, where the prior probability of parse graph can be decomposed into physical prior $\mathcal{E}_{phy}(pg)$ and HOI prior $\mathcal{E}_{hoi}(pg)$; balancing factors are neglected for simplicity.

Physical Prior $\mathcal{E}_{phy}(pg)$ represents physical commonsense in a 3D scene. We consider two types of physical relations among the terminal nodes: support relation E_s and collision relation E_c. Therefore, the energy of physical prior is defined as $\mathcal{E}_{phy}(pg) = \mathcal{E}_s(pg) + \mathcal{E}_c(pg)$. Specifically:

- *Support Relation* $\mathcal{E}_s(pg)$ defines the energy between the supported object/human and the supporting object/layout:

$$\mathcal{E}_s(pg) = \sum_{(v_i,v_j)\in E_s} \mathcal{E}_o(v_i, v_j) + \mathcal{E}_{\text{height}}(v_i, v_j), \tag{4.2}$$

where $\mathcal{E}_o(v_i, v_j) = 1 - \text{area}(v_i \cap v_j)/\text{area}(v_i)$ is the overlapping ratio in the xy-plane, and $\mathcal{E}_{\text{height}}(v_i, v_j)$ is the absolute height difference between the lower surface of the supported object v_i and the upper surface of the supporting object v_j; $\mathcal{E}_o(v_i, v_j) = 0$ when the supporting object is the floor and $\mathcal{E}_{\text{height}}(v_i, v_j) = 0$ when the supporting object is the wall.

- *Physical Collision* $\mathcal{E}_c(pg)$ denotes the physical violations. We penalize the intersection among human, objects, and room layout except the objects in HOI and objects that could be a container. The potential function is defined as:

$$\mathcal{E}_c(pg) = \sum_{v\in(V_{\text{object}}\cup V_{\text{human}})} \mathcal{C}(v, V_{\text{layout}}) + \sum_{\substack{v_i\in V_{\text{object}}\\ v_j\in V_{\text{human}}\\ (v_i,v_j)\notin E_{hoi}}} \mathcal{C}(v_i, v_j) + \sum_{\substack{v_i,v_j\in V_{\text{object}}\\ v_i,v_j\notin V_{\text{container}}}} \mathcal{C}(v_i, v_j), \tag{4.3}$$

where $\mathcal{C}()$ denotes the volume of intersection between entities. $V_{\text{container}}$ denotes the objects that can be a container, such as a cabinet, desk, and drawer.

Human-object Interaction Prior $\mathcal{E}_{hoi}(pg)$ is defined by the interactions between human and objects:

$$\mathcal{E}_{hoi}(pg) = \sum_{(v_i,v_j)\in E_{hoi}} \mathcal{K}(v_i, v_j, a_{v_j}), \tag{4.4}$$

where $v_i \in V_{\text{object}}, v_j \in V_{\text{human}}$, and $\mathcal{K}$ is an HOI function that evaluates the interaction between an object and a human given the action label a:

$$\mathcal{K}(v_i, v_j, a_{v_j}) = -\log l(v_i, v_j | a_{v_j}), \tag{4.5}$$

where $l(v_i, v_j | a_{v_j})$ is the likelihood of the relative position between node v_i and v_j given an action label a. We formulate the action detection as a *multi-label classification*; see Section 4.1.5.3 for details. The likelihood $l(\cdot)$ models the distance between key joints and the center of the object; *e.g.*, for "sitting," it models the relative spatial relation between

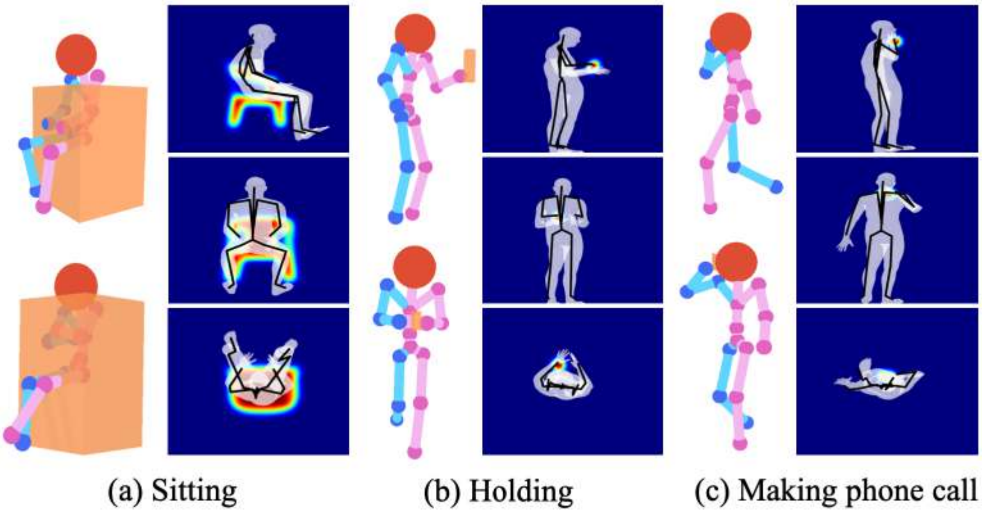


Figure 4.2: Examples of typical HOIs and examples from the SHADE dataset. The heatmap indicates the probable locations of HOI.

the hip and the center of a chair. The likelihood can be learned from 3D HOI datasets with a multivariate Gaussian distribution $(\Delta x, \Delta y, \Delta z) \sim \mathcal{N}_3(\mu, \Sigma)$, where $\Delta x, \Delta y$, and Δz are the relative distances in the directions of three axes.

Likelihood $\mathcal{E}(I|pg)$ characterizes the consistency between the observed 2D image and the inferred 3D result. The projected 2D object bounding boxes and human poses can be computed by projecting the inferred 3D objects and human poses onto a 2D image plane. The likelihood is obtained by comparing the directly detected 2D bounding boxes and human poses with projected ones from inferred 3D results:

$$\mathcal{E}(I|pg) = \sum_{v \in V_{\text{object}}} \cdot \mathcal{D}_o(B(v), B'(v)) + \sum_{v \in V_{\text{human}}} \cdot \mathcal{D}_h(Po(v), Po'(v)), \tag{4.6}$$

where $B()$ and $B'()$ are the bounding boxes of detected and projected 2D objects, $Po()$ and $Po'()$ the poses of detected and projected 2D humans, $\mathcal{D}_o(\cdot)$ the IoU between the detected 2D bounding box and the convex hull of the projected 3D bounding box, and $\mathcal{D}_h(\cdot)$ the average pixel-wise Euclidean distance between two 2D poses.

4.1.4.1 SHADE Dataset

We collect SHADE (Synthetic Human Activities with Dynamic Environment), a self-annotated dataset that consists of dynamic 3D human skeletons and objects, to learn the prior model for each HOI. It is collected from a video game Grand Theft Auto V with various daily activities and HOIs. Currently, there are over 29 million frames of 3D human poses, where 772,229 frames are annotated. On average, each annotated frame is associated with 2.03 action labels and 0.89 HOIs. The SHADE dataset contains 19 fine-grained HOIs for both indoor and outdoor activities. By selecting most frequent HOIs and merging similar HOIs, we choose 6 final HOIs: *read [phone, notebook, tablet], sit-at [human-table relation], sit [human-chair relation], make-phone-call, hold, use-laptop.* Figure 4.2 shows some typical examples and relations in the dataset.

4.1.5 Joint Inference

Given a single RGB image as the input, the goal of joint inference is to find the optimal parse graph that maximizes the posterior probability $p(pg|I)$. The joint parsing is a four-step process: (i) 3D scene initialization of the camera pose, room layout, and 3D object bounding boxes, (ii) 3D human pose initialization that estimates rough 3D human poses in a 3D scene, (iii) concurrent action detection, and (iv) joint inference to optimize the objects, layout, and human poses in 3D scenes by maximizing the posterior probability.

4.1.5.1 3D Scene Initialization

Following [HQX18], we initialize the 3D objects, room layout, and camera pose cooperatively, where the room layout and objects are parametrized by 3D bounding boxes. For each object $v_i \in V_{\text{object}}$, we find its supporting object/layout by minimizing the supporting energy:

$$v_j^* = \arg\min_{v_j} \mathcal{E}_o(v_i, v_j) + \mathcal{E}_{\text{height}}(v_i, v_j) - \lambda_s \log p_{spt}(v_i, v_j), \tag{4.7}$$

where $v_j \in (V_{\text{object}}, V_{\text{layout}})$ and $p_{spt}(v_i, v_j)$ are the prior probabilities of the supporting relation modeled by multinoulli distributions, and λ_s a balancing constant.

4.1.5.2 3D Human Pose Initialization

We take 2D poses as the input and predict 3D poses in a local 3D coordinate following [TRA17], where the 2D poses are detected and estimated by [CSW17]. The local 3D coordinate is centered at the human hip joint, and the z-axis is aligned with the up direction of the world coordinate.

To transform this local 3D pose into the world coordinate, we find the 3D world coordinate $\mathbf{v_{3D}} \in \mathbb{R}^3$ of one visible 2D joint $\mathbf{v_{2D}} \in \mathbb{R}^2$ (*e.g.*, head) by solving a linear equation with the camera intrinsic parameter K and estimated camera pose R. Per the pinhole camera projection model, we have

$$\alpha \begin{bmatrix} \mathbf{v_{2D}} \\ 1 \end{bmatrix} = K \cdot R \cdot \mathbf{v_{3D}}, \tag{4.8}$$

where α is a scaling factor in the homogeneous coordinate. To make the function solvable, we assume a pre-defined height h_0 for the joint position $\mathbf{v_{3D}}$ in the world coordinate. Lastly, the 3D pose initialization is obtained by aligning the local 3D pose and the corresponding joint position with $\mathbf{v_{3D}}$.

4.1.5.3 Concurrent Action Detection

We formulate the concurrent action detection as a multi-label classification problem to ease the ambiguity in describing the action. We define a portion of the action labels (*e.g.*, “eating”, “making phone call”) as the HOI labels, and the remaining action labels (*e.g.*, “standing”, “bending”) as general human poses without HOI. The mixture of HOI actions and non-HOI actions covers most of the daily human actions in indoor scenes. We manually map each of the HOI action labels to a 3D HOI relation learned from the SHADE dataset, and use the HOI actions as cues to improve the accuracy of 3D reconstruction by integrating it as prior knowledge in our model. The concurrent action detector takes 2D skeletons as the input and

Algorithm 2 Joint Inference Algorithm

```
Given: Image $I$, initialized parse graph $pg_{init}$
procedure Phase 1
    for Different temperatures do
        Inference with physical commonsense $\mathcal{E}_{phy}$ but without HOI $\mathcal{E}_{hoi}$: randomly select from room layout, objects, and human poses to optimize $pg$
procedure Phase 2
    Match each agent with their interacting objects
procedure Phase 3
    for Different temperatures do
        Inference with total energy $\mathcal{E}$, including physical commonsense and HOI: randomly select from layout, objects, and human poses to optimize $pg$
procedure Phase 4
    Top-down sampling by HOIs
```

predicts multiple action labels with a three-layer multi-layer perceptron (MLP).

The dataset for training the concurrent action detectors consists of both synthetic data and real-world data. It is collected from: (i) The synthetic dataset described in Section 4.1.4.1. We project the 3D human poses of different HOIs into 2D poses with random camera poses. (ii) The dataset proposed and collected by [JSL17], which also contains 3D poses of multiple persons in social interactions. We project 3D poses into 2D following the same method as in (i). (iii) The 2D poses in an action recognition dataset [YJK11]. Our results show that the synthetic data can significantly expand the training set and help to avoid overfitting in concurrent action detection.

4.1.5.4 Inference

Given an initialized parse graph, we use MCMC with simulated annealing to jointly optimize the room layout, 3D objects, and 3D human poses through the non-differentiable energy space; see Algorithm 2 as a summary. To improve the efficiency of the optimization process, we adopt a scheduling strategy that divides the optimization process into following four phases with different focuses: (i) Optimize objects, room layout, and human poses without HOIs. (ii) Assign HOI labels to each agent in the scene, and search the interacting objects of each agent. (iii) Optimize objects, room layout, and human poses jointly with HOIs. (iv)

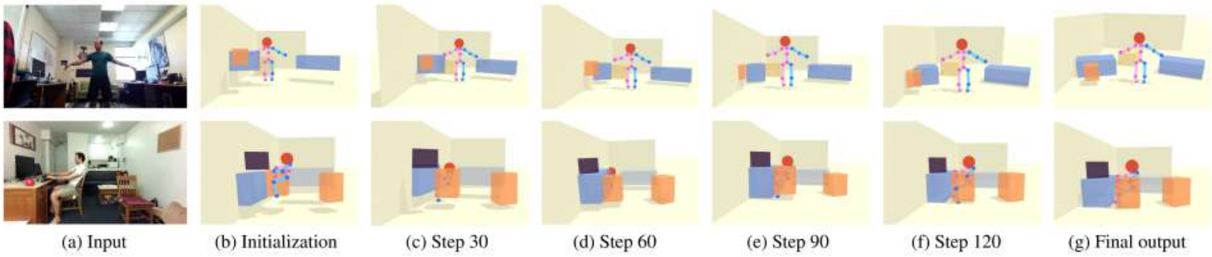


Figure 4.3: The optimization process of the scene configuration by simulated annealing MCMC. Each step is the number of accepted proposal.

Generate possible miss-detected objects by top-down sampling.

Dynamics: In Phase (i) and (iii), we use distinct MCMC processes. To traverse non-differentiable energy spaces, we design Markov chain dynamics q_1^o, q_2^o, q_3^o for objects, q_1^l, q_2^l for room layout, and q_1^h, q_2^h, q_3^h for human poses.

• Object Dynamics: Dynamics q_1^o adjusts the position of an object, which translates the object center in one of the three Cartesian coordinate axes or along the depth direction; the depth direction starts from the camera position and points to the object center. Translation along depth is effective with proper camera pose initialization. Dynamics q_2^o proposes rotation of the object with a specified angle. Dynamics q_3^o changes the scale of the object by expanding or shrinking corner positions of the cuboid with respect to the object center. Each dynamic can diffuse in two directions: translate in the direction of '$+x$' and '$-x$,' or rotate in the direction of clockwise and counterclockwise. To better traverse in energy space, the dynamics may propose to move along the gradient descent direction with a probability of 0.95 or the gradient ascent direction with a probability of 0.05.

• Human Dynamics: Dynamics q_1^h proposes to translate 3D human joints along x, y, z, or depth direction. Dynamics q_2^h rotates the human pose with a certain angle. Dynamics q_3^h adjusts the scale of human poses by a scaling factor on the 3D joints with respect to the pose center.

• Layout Dynamics: Dynamics q_1^l translates the wall towards or away from the layout center. Dynamics q_2^l adjusts the floor height, equivalent to changing the camera height.

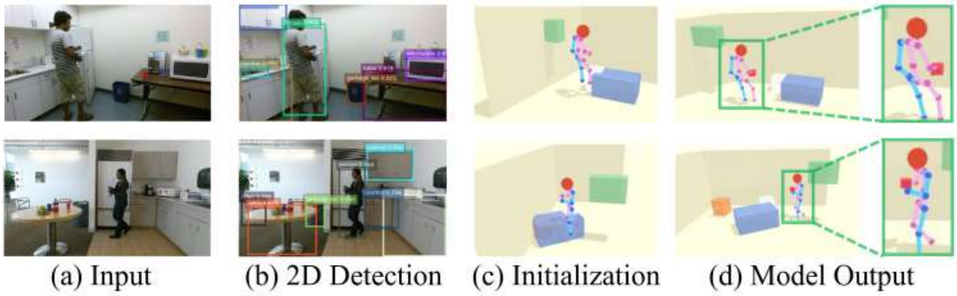


Figure 4.4: Illustration of the top-down sampling process. The object detection module misses the detection of the bottle held by the person, but our model can still recover the bottle by reasoning HOI.

In each sampling iteration, the algorithm proposes a new pg' from current pg under the proposal probability of $q(pg \to pg'|I)$ by applying one of the above dynamics. The generated proposal is accepted with respect to an acceptance rate $\alpha(\cdot)$ as in the Metropolis-Hastings algorithm [Has70]:

$$\alpha(pg \to pg') = \min(1, \frac{q(pg' \to pg) \cdot p(pg'|I)}{q(pg \to pg') \cdot p(pg|I)}), \tag{4.9}$$

A simulated annealing scheme is adopted to obtain pg with a high probability.

Top-down sampling: By top-down sampling objects from HOIs relations, the proposed method can recover the interacting 3D objects that are too small or novel to be detected by the state-of-the-art 2D object detector. In Phase (iv), we propose to sample an interacting object from the person if the confidence of HOI is higher than a threshold; we minimize the HOI energy in Equation (4.4) to determine the category and location of the object; see examples in Figure 4.4.

Implementation Details: In Phase (ii), we search the interacting objects for each agent involved in HOI by minimizing the energy in Equation (4.4). In Phase (iii), after matching each agent with their interacting objects, we can jointly optimize objects, room layout, and human poses with the constraint imposed by HOI. Figure 4.3 shows examples of the simulated annealing optimization process.

4.1.6 Experiments

Since the proposed task is new and challenging, limited data and state-of-the-art methods are available for the proposed problem. For fair evaluations and comparisons, we evaluate the proposed algorithm on three types of datasets: (i) Real data with full annotation on PiGraphs dataset [SCH16] with limited 3D scenes. (ii) Real data with partial annotation on daily activity dataset Watch-n-Patch [WZS15], which only contains ground-truth depth information and annotations of 3D human poses. (iii) Synthetic data with generated annotations to serve as the ground truth: we sample 3D human poses of various activities in SUN RGB-D dataset [SLX15] and project the sampled skeletons back onto the 2D image plane.

4.1.6.1 Comparative methods

To the best of our knowledge, no previous algorithm jointly optimizes the 3D scene and 3D human pose from a single image. Therefore, we compare our model against state-of-the-art methods for each task. Particularly, we compare with [HQX18] for single-image 3D scene reconstruction and VNect [MSS17] for 3D pose estimation in the world coordinate.

Since VNect can only estimate a single person, we design an additional baseline for 3D multi-person human pose estimation in the world coordinate. We first extract a 2048-D image feature vector using the Global Geometry Network (GGN) [HQX18] to capture the global geometry of the scene. The concatenated vector (GGN image feature, 2D pose, 3D pose in the local coordinate, and the camera intrinsic matrix) is fed into a 5-layer fully connected network to predict the 3D pose. The fully-connected layers are trained using the mean squared error loss. We train the network on the training set of the synthetic SUN RGB-D dataset.

4.1.6.2 Dataset

PiGraphs [SCH16] contains 30 scenes and 63 video recordings obtained by Kinect v2, designed to associate human poses with object arrangements. There are 298 actions available in

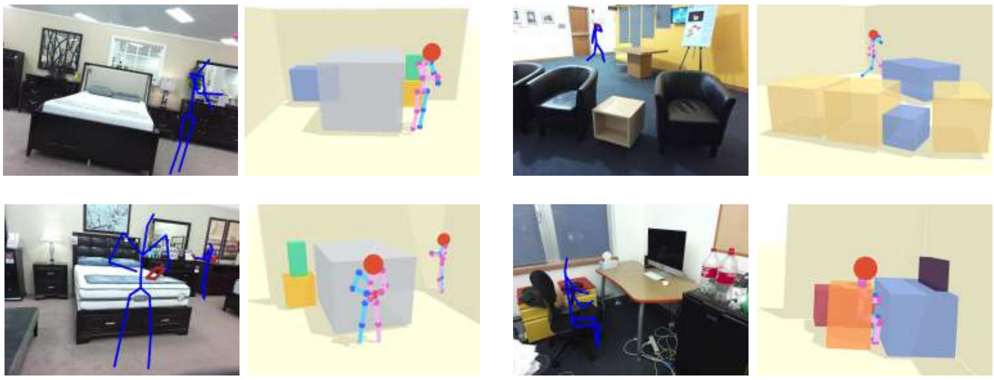


Figure 4.5: Augmenting SUN RGB-D with synthetic human poses.

approximately 2-hours of recordings. Each recording is about 2-minute long, with an average 4.9 action annotation. We removed the frames with no human appearance or annotations, resulting in 36,551 test images.

Watch-n-Patch (WnP) [WZS15] is an activity video dataset recorded by Kinect v2. It contains several human daily activities as compositions of multiple actions interacting with various objects. The dataset comes with activity annotations, depth maps, and 3D human poses. We test our algorithm on 1,210 randomly selected frames.

SUN RGB-D [SLX15] contains rich indoor scenes that are densely annotated with 3D bounding boxes, room layouts, and camera poses. The original dataset has 5,050 testing images, but we discarded images with no detected 2D objects, invalid 3D room layout annotation, limited space, or small field of view, resulting in 3,476 testing images.

Synthetic SUN RGB-D is augmented from SUN RGB-D dataset by sampling human poses in the scenes. Following methods of sampling imaginary human poses in [HQZ18], we extend the sampling to more generalized settings for various poses. The augmented human is represented by a 6-tuple $\langle a, \mu, t, r, s, \hat{\mu} \rangle$, where a is the action type, μ the pose template, t translation, r rotation, s scale, and $\hat{\mu} = \mu \cdot r \cdot s + t$ the imagined human skeleton. For each action label, we sample an imagined human pose inside a 3D scene: $\langle t^*, r^*, s^* \rangle = \arg\min_{t,r,s} \mathcal{E}_{phy} + \mathcal{E}_{hoi}$. If a is involved with any HOI unit, we further augment the 3D bounding box of the object. After sampling a human pose, we project the augmented 3D scenes back onto the 2D image plane using the ground truth camera matrix and camera pose; see examples in Figure 4.5. For a fair comparison of 3D human pose estimation on synthetic SUN RGB-D, all the algorithms are provided with the ground truth 2D skeletons as the input.

For 3D scene reconstruction, both [HQX18] and the proposed 3D scene initialization are learned using SUN RGB-D training data and tested on the above three datasets. For 3D pose estimation, both [MSS17] and the initialization of the proposed method are trained on public datasets, while the baseline is trained on synthetic SUN RGB-D. Note that we only use the SHADE dataset for learning a dictionary of HOIs.

4.1.6.3 Quantitative and Qualitative Results

We evaluate the proposed model on holistic^{++} scene understanding task by comparing the performances on both 3D scene reconstruction and 3D pose estimation.

Scene Reconstruction: We compute the 3D IoU and 2D IoU of object bounding boxes to evaluate the 3D scene reconstruction and the consistency between the 3D world and 2D image. Following the metrics described in [HQX18], we compute the 3D IoU between the estimated 3D bounding boxes and the annotated 3D bounding boxes on PiGraphs and SUN RGB-D. For dataset without ground-truth 3D bounding boxes (*i.e.*, Watch-n-Patch), we evaluate the distance between the camera center and the 3D object center. To evaluate the 2D-3D consistency, the 2D IoU is computed between the projected 2D boxes of the 3D object bounding boxes and the ground-truth 2D boxes or detected 2D boxes (*i.e.*, Watch-n-Patch). As shown in Table 4.1, the proposed method improves the state-of-the-art 3D scene reconstruction results on all three datasets without specific training on each of them. More importantly, it significantly improves the results on PiGraphs and Watch-n-Patch compared with [HQX18]. The most likely reason is: [HQX18] is trained on SUN RGB-D dataset in a purely data-driven fashion, therefore difficult to generalize across to other datasets (*i.e.*, PiGraphs, and Watch-n-Patch). In contrast, the proposed model incorporates more general prior knowledge of HOI and physical commonsense, and combines such knowledge with 2D-3D consistency (likelihood) for joint inference, avoiding the over-fitting caused by the direct

Table 4.1: Quantitative Results of 3D Scene Reconstruction

Methods	Huang *et al.*[HQX18]			Ours		
Metric	2D IoU (%)	3D IoU (%)	Depth (m)	2D IOU (%)	3D IoU (%)	Depth (m)
PiGraphs	68.6	21.4	-	**75.1**	**24.9**	-
SUN RGB-D	63.9	17.7	-	**72.9**	**18.2**	-
WnP	67.3	-	0.375	**73.6**	-	**0.162**

Table 4.2: Quantitative Results of Global 3D Pose Estimation

Methods	VNect[MSS17]		Baseline		Ours	
Metrics	2D (pix)	3D (m)	2D (pix)	3D (m)	2D (pix)	3D (m)
PiGraphs	63.9	0.732	284.5	2.67	**15.9**	**0.472**
SUNRGBD	-	-	45.81	**0.435**	**14.03**	0.517
WnP	50.51	0.646	325.2	2.14	**20.5**	**0.330**

3D estimation from 2D. Figure 4.6 shows the qualitative results on all three datasets.

Pose Estimation: We evaluate the pose estimation in both 3D and 2D. For 3D evaluation, we compute the Euclidean distance between the estimated 3D joints and the 3D ground-truth and average it over all the joints. For 2D evaluation, we project the estimated 3D pose back to the 2D image plane and compute the pixel distance against the ground truth. See Table 4.2 for quantitative results. The proposed method outperforms two other methods in both 2D and 3D. On the synthetic SUN RGB-D dataset, all algorithms are given the ground truth 2D poses as the input for a fair comparison. Although the baseline model achieves better performances since the baseline model fits well for the 3D human poses synthesized with limited templates, the 3D poses estimated by VNect and baseline model deviate a lot from the ground truth for datasets with real human poses (*i.e.*, PiGraph, and Watch-n-Patch). In contrast, the proposed algorithm performs consistently well, demonstrating an outstanding generalization ability across various datasets.

Ablative Analysis: To analyze the contributions of HOI and physical commonsense, we compare two variants of the proposed full model: (i) model *w/o HOI*: without HOI $\mathcal{E}_{hoi}(pg)$, and (ii) model *w/o phy.*: without physical commonsense $\mathcal{E}_{phy}(pg)$.

- *Human-Object Interaction.* We compare our full model with model *w/o hoi* to evaluate the effects of each category of HOI. Evaluation metrics include 3D pose estimation error,

Table 4.3: Ablative results of HOI on 3D object IoU (%), 3D pose estimation error (m), and miss-detection rate (MR, %)

Methods	*w/o hoi*			*Full model*		
HOI Type	Object ↑	Pose ↓	MR ↓	Object ↑	Pose ↓	MR ↓
Sit	26.9	0.590	15.2	**27.8**	**0.521**	**13.1**
Hold	17.4	0.517	78.9	**17.6**	**0.490**	**54.6**
Use Laptop	14.1	0.544	58.8	**15.0**	**0.534**	**43.3**
Read	**14.5**	0.466	65.3	14.3	**0.453**	**41.9**

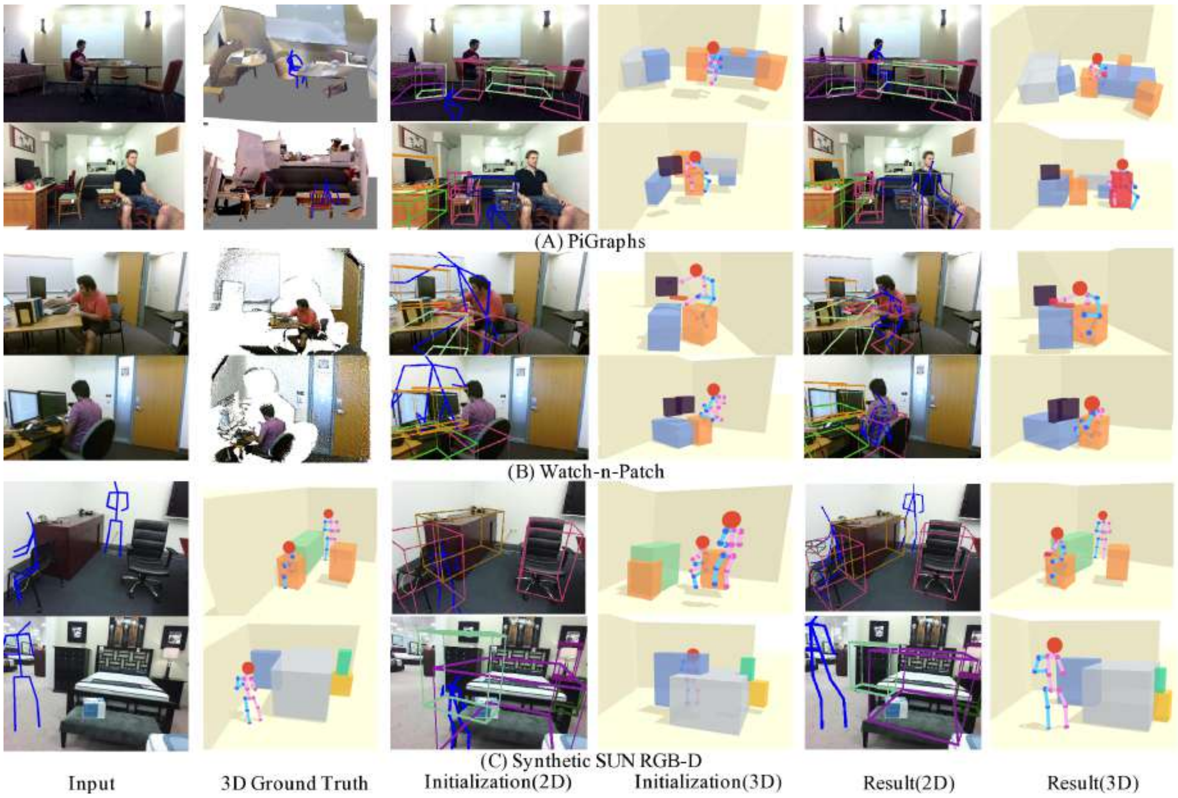


Figure 4.6: Qualitative results of the proposed method on three datasets. The proposed model improves the initialization with accurate spatial relations and physical plausibility and demonstrates an outstanding generalization across various datasets.

3D bounding box IoU, and miss-detection rate (MR) of the objects interacted with agents. The experiments are conducted on PiGraphs dataset and Synthetic SUN RGB-D dataset with the annotated HOI labels. Note that for the consistency of the ablative analysis across three different datasets, we merge the *sit* and *sit-at* into *sit*, and eliminate the *make-phone-call*. As shown in Table 4.3, the performances of both scene reconstruction and human pose estimation are hindered without reasoning HOI, indicating HOI helps to infer the relative

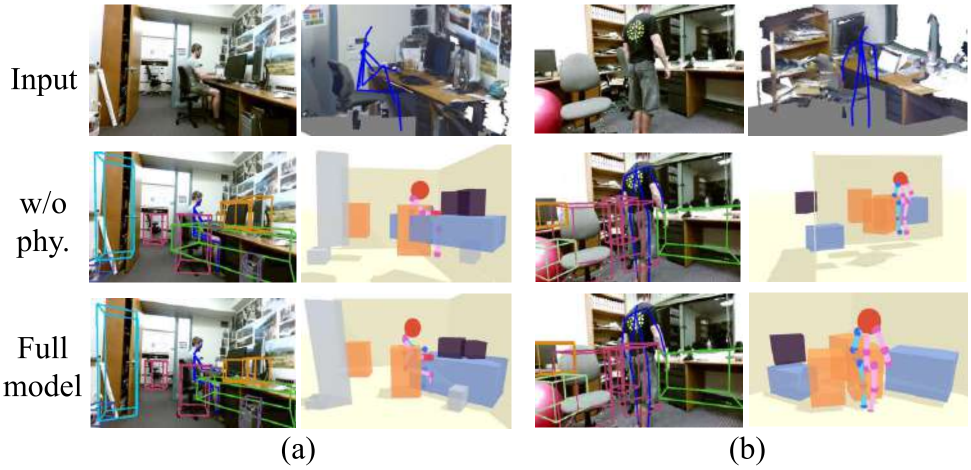


Figure 4.7: Qualitative comparison between (a) model *w/o phy.* and (b) the full model on PiGraphs dataset.

spatial relationship between agents and objects to improve the performance of both two tasks further. Moreover, a marked performance gain of miss-detection rate implies the effectiveness of the top-down sampling process during the joint inference.

• *Physical Commonsense.* Reasoning about physical commonsense drives the reconstructed 3D scene to be physically plausible and stable. We test 3D estimation of object bounding boxes on the PiGraphs dataset using *w/o phy.* and the full model. The full model outperforms *w/o phy.* in two aspects: (i) 3D object detection IoU (from 23.5% to 24.9%), and (ii) physical violation (from 0.223m to 0.150m); see qualitative comparisons in Figure 4.7. The physical violation is computed as the distance between the lower surface of an object and the upper surface of its supporting object. Objects detected by model *w/o phy.* may float in the air or penetrate each other, while the full model yields physically plausible results.

4.1.7 Conclusion

This work tackles a challenging holistic^{++} scene understanding problem to jointly solve 3D scene reconstruction and 3D human pose estimation from a single RGB image. By incorporating physical commonsense and reasoning about HOI, our approach leverages the coupled nature of these two tasks and goes beyond merely reconstructing the 3D scene or human pose by reasoning about the concurrent action of human in the scene. We design a joint inference algorithm which traverses the non-differentiable solution space with MCMC and optimizes the scene configuration. Experiments on PiGraphs, Watch-n-Patch, and Synthetic SUN RGB-D demonstrate the efficacy of the proposed algorithm and the general prior knowledge of HOI and physical commonsense.

4.1.8 Appendix: Additional Results

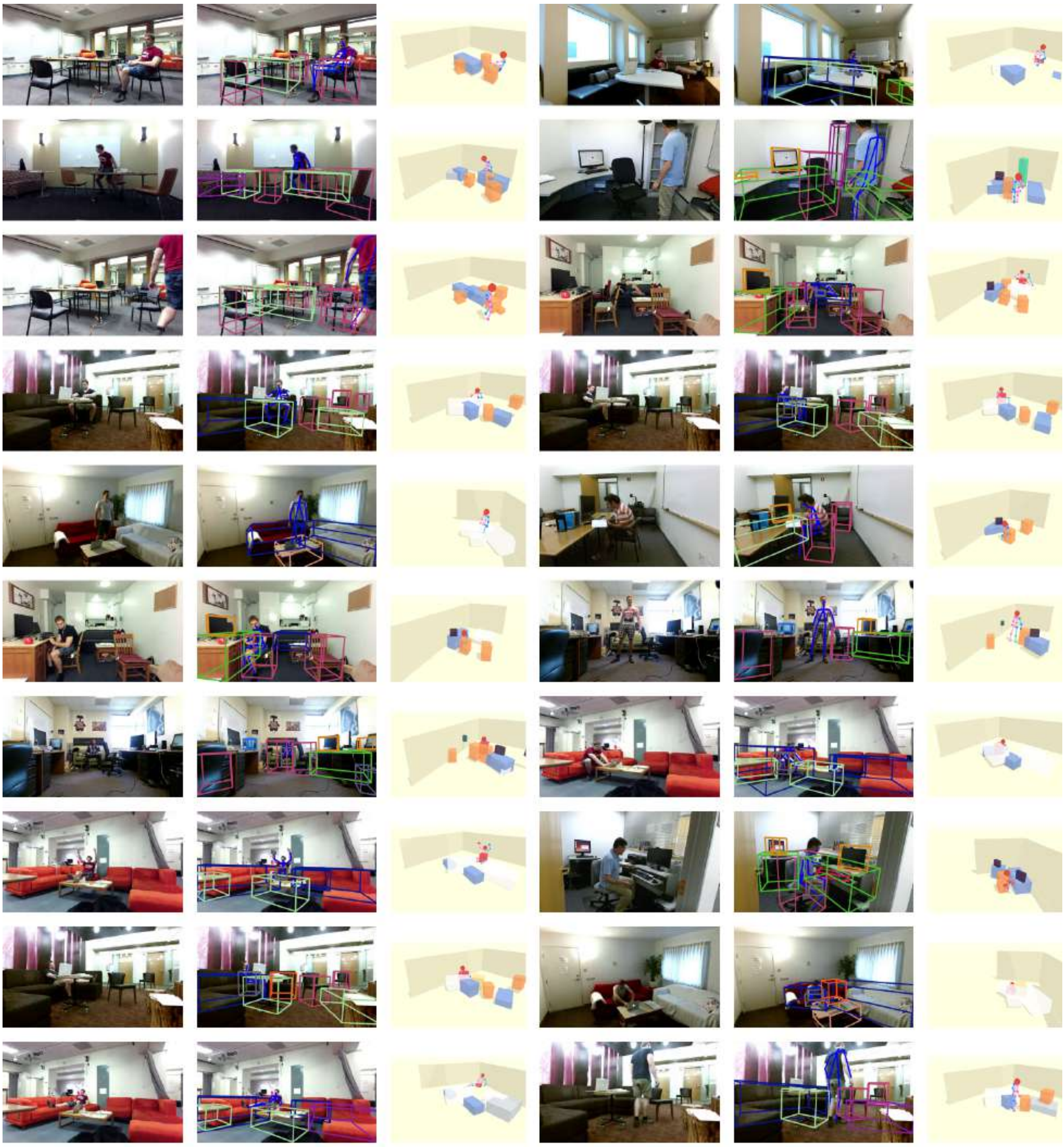


Figure 4.8: Additional results on Watch-n-Patch and PiGraphs dataset.

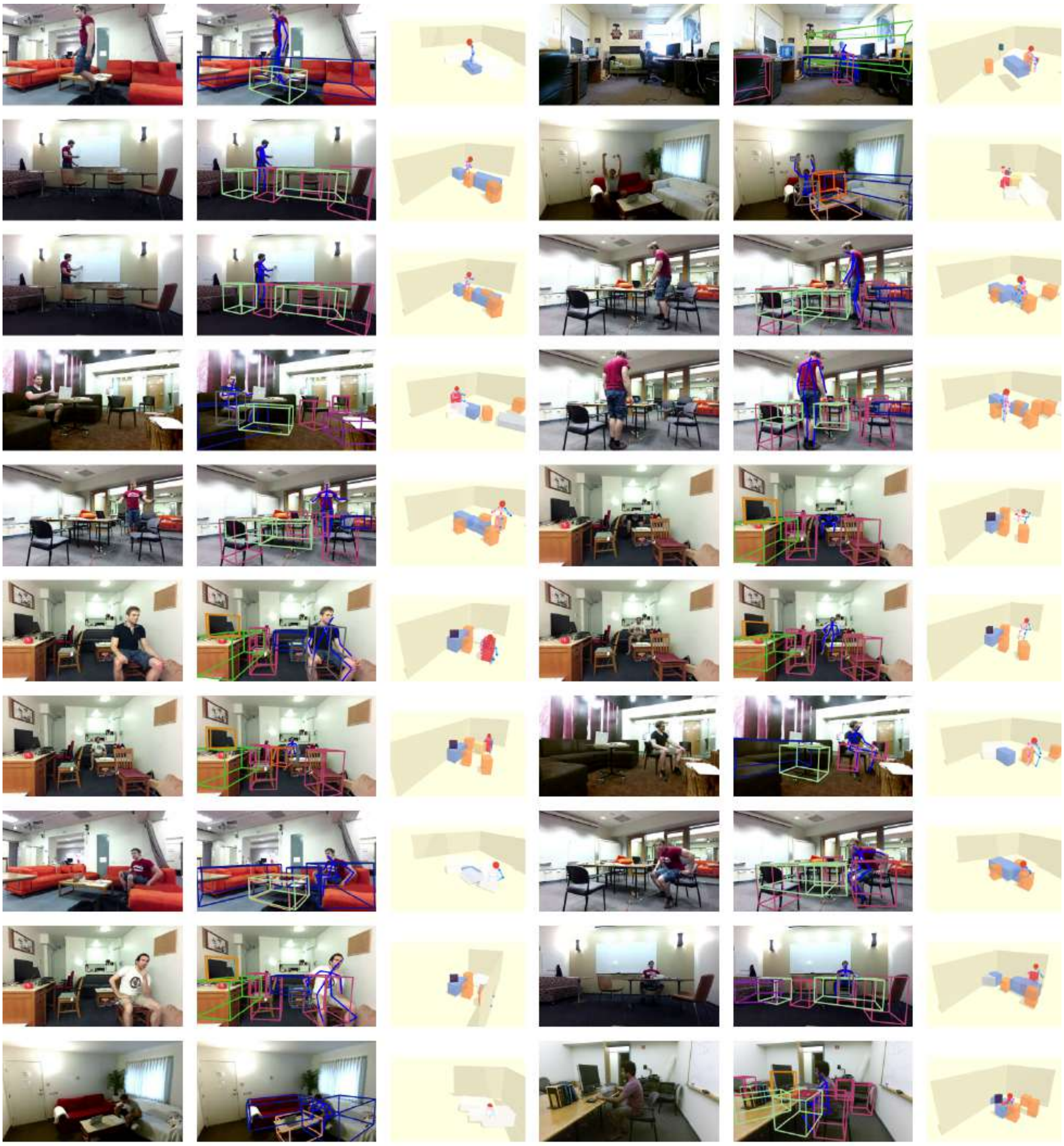


Figure 4.9: Additional results on Watch-n-Patch and PiGraphs dataset.

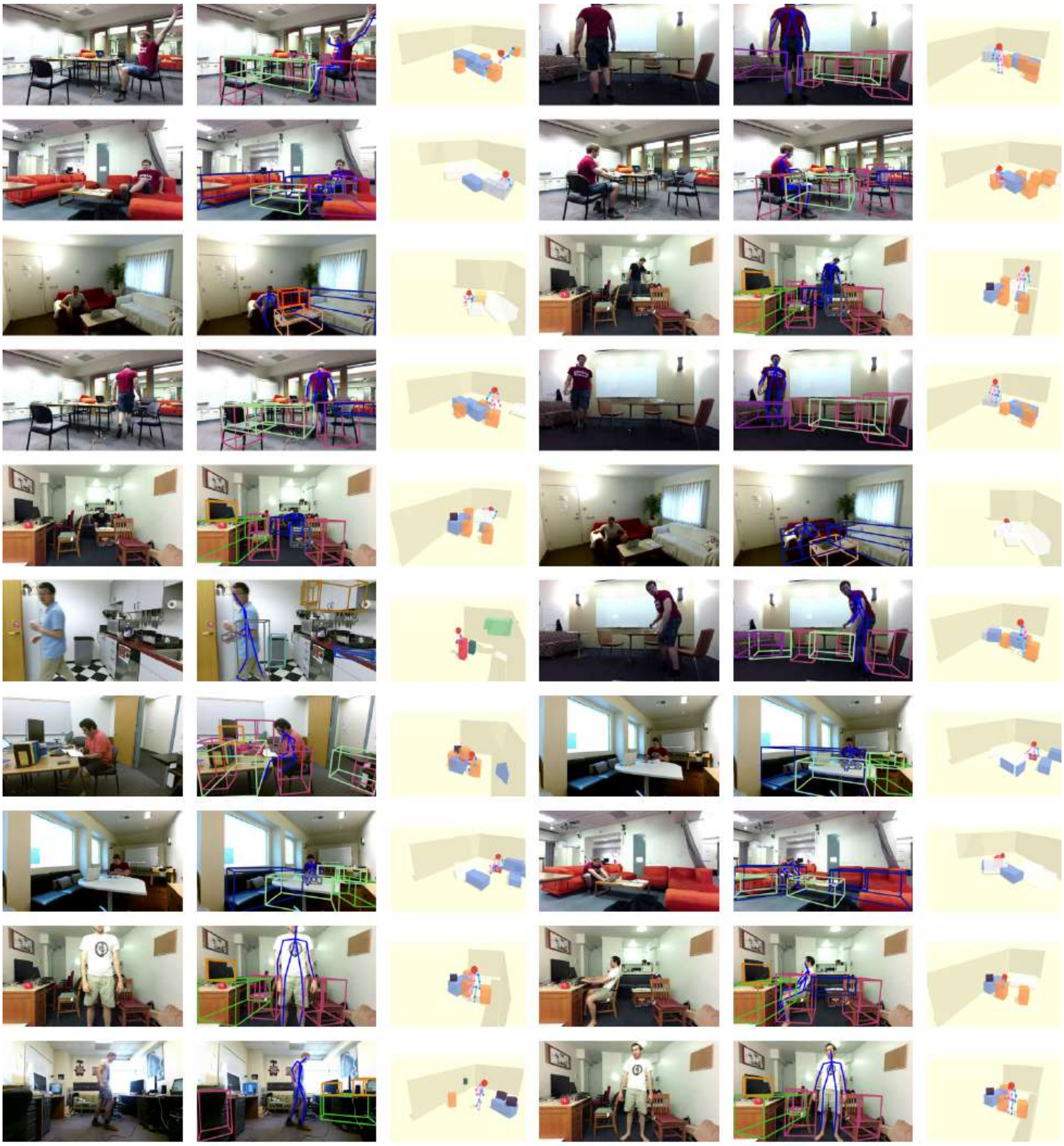


Figure 4.10: Additional results on Watch-n-Patch and PiGraphs dataset.

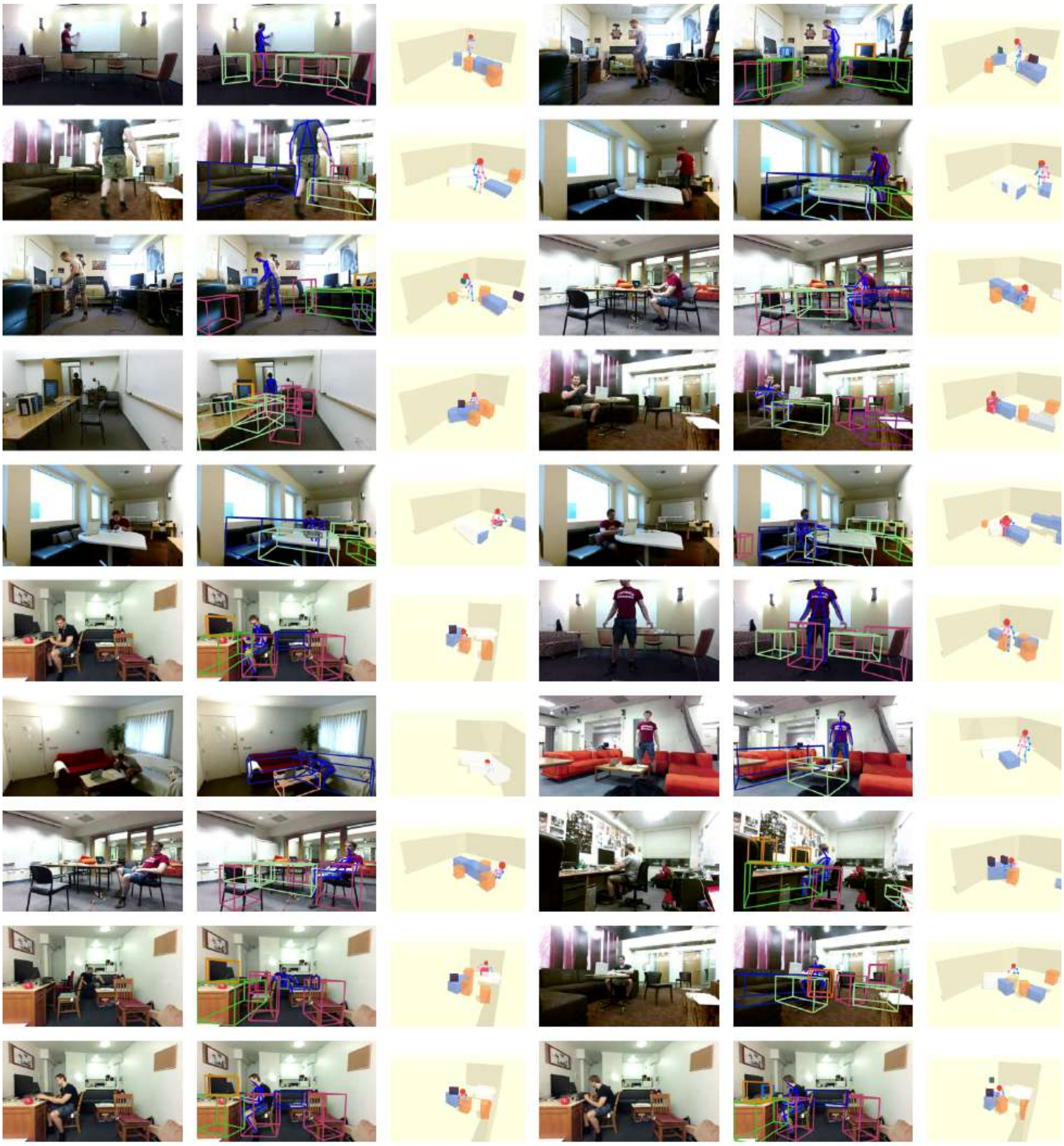


Figure 4.11: Additional results on Watch-n-Patch and PiGraphs dataset.

CHAPTER 5

Human-human Interaction and Collaboration

In this chapter, we explore an important area of social scene understanding, human-human interaction. Understanding human-human interaction is critical for the machine to sense the social relations and activities in the scenes, providing potential channels for actively helping humans. Specifically, we study the human gaze communication in social videos from both atomic-level and event-level in Section 5.1 and the multi-task multi-agent activities in the context of human collaboration in Section 5.2.

5.1 Understanding Human Gaze Interaction by Spatio-Temporal Graph Reasoning

In this section, we addresses a new problem of understanding human gaze communication in social videos from both atomic-level and event-level, which is significant for studying human social interactions. To tackle this novel and challenging problem, we contribute a large-scale video dataset, *VACATION*, which covers diverse daily social scenes and gaze communication behaviors with complete annotations of objects and human faces, human attention, and communication structures and labels in both atomic-level and event-level. Together with *VACATION*, we propose a spatio-temporal graph neural network to explicitly represent the diverse gaze interactions in the social scenes and to infer atomic-level gaze communication by message passing. We further propose an event network with encoder-decoder structure to predict the event-level gaze communication. Our experiments demonstrate that the proposed model improves various baselines significantly in predicting the atomic-level and event-level gaze communications.

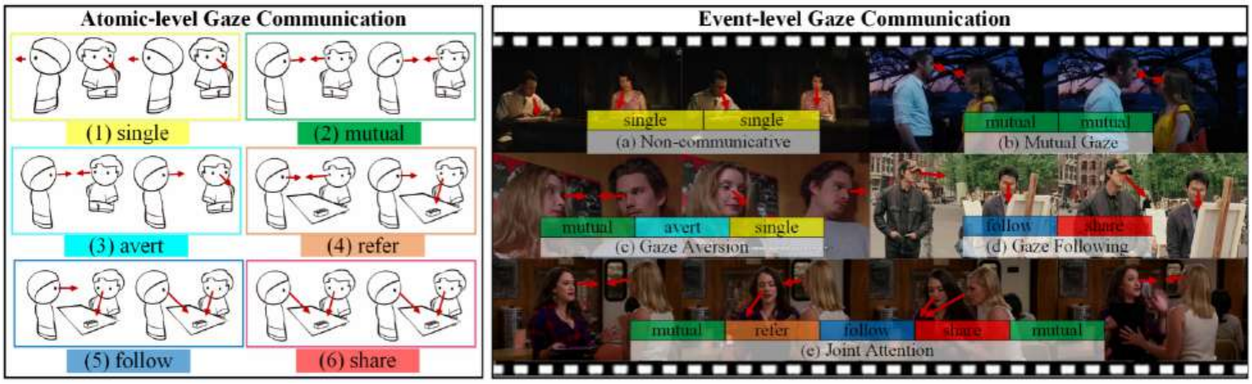


Figure 5.1: We study human gaze communication dynamics in two hierarchical levels: atomic-level and event-level. Atomic-level gaze communication describes the fine-grained structures in human gaze interactions, *i.e.*, *single*, *mutual*, *avert*, *refer*, *follow* and *share* (as shown in the left part). Event-level gaze communication refers to high-level, complex social communication events, including *Non-communicative*, *Mutual Gaze*, *Gaze Aversion*, *Gaze Following* and *Joint Attention*. Each gaze communication event is a temporal composition of some atomic-level gaze communications (as shown in the right part).

5.1.1 Introduction

In this work, we introduce the task of understanding human *gaze communication* in social interactions. Evidence from psychology suggests that eyes are a cognitively special stimulus, with unique "hard-wired" pathways in the brain dedicated to their interpretation and humans have the unique ability to infer others' intentions from eye gazes [Eme00]. Gaze communication is a primitive form of human communication, whose underlying social-cognitive and social-motivational infrastructure acted as a psychological platform on which various linguistic systems could be built [Tom10]. Though verbal communication has become the primary form in social interaction, gaze communication still plays an important role in conveying hidden mental state and augmenting verbal communication [AS17]. To better understand human communication, we not only need natural language processing (NLP), but also require a systematical study of human gaze communication mechanism.

The study of human gaze communication in social interaction is essential for the following several reasons: 1) it helps to better understand multi-agent gaze communication behaviors in realistic social scenes, especially from social and psychological views; 2) it provides evidences for robot systems to learn human behavior patterns in gaze communication and further facilitates intuitive and efficient interactions between human and robot; 3) it enables simulation of more natural human gaze communication behaviors in Virtual Reality environment; 4) it builds up a common sense knowledge base of human gaze communication for studying human mental state in social interaction; 5) it helps to evaluate and diagnose children with autism.

Over the past decades, lots of research [HBM77, KK97, IB09, JHB18] on the types and effects of social gazes have been done in cognitive psychology and neuroscience communities. With previous efforts and established terminologies, we distinguish atomic-level gaze communications into six classes:

- *Single* refers to individual gaze behavior without any social communication intention (see Figure 5.1 (1)).

- *Mutual* [AS17, AC76] gaze occurs when two agents look into eyes of each other (see Figure 5.1 (2)), which is the strongest mode of establishing a communicative link between human agents. *Mutual* gaze can capture attention, initialize a conversation, maintain engagement, express feelings of trust and extroversion, and signal availability for interaction in cases like passing objects to a partner.

- *Avert* [Rie49, GSR98] refers to averted gaze and happens when gaze of one agent is shifted away from another in order to avoid mutual gaze (see Figure 5.1 (3)). *Avert* gaze expresses distrust, introversion, fear, and can also modulate intimacy, communicate thoughtfulness or signal cognitive effort such as looking away before responding to a question.

- *Refer* [SJC06] means referential gaze and happens when one agent tries to induce another agent's attention to a target via gaze (see Figure 5.1 (4)). Referential gaze shows intents to inform, share or request sth. We can use *refer* gaze to eliminate uncertainty about reference and respond quickly.

- *Follow* [She10, Zub08, BM05] means following gaze and happens when one agent perceives gaze from another and follows to contact with the stimuli the other is attending to (see Figure 5.1 (5)). Gaze following is to figure out partner's intention.

- *Share* [OT06a] means shared gaze and appears when two agents are gazing at the same stimuli (see Figure 5.1 (6)).

The above atomic-level gazes capture the most general, core and fine-grained gaze communication patterns in human social interactions. We further study the long-term, coarse-grained temporal compositions of the above six atomic-level gaze communication patterns, and generalize them into totally five gaze communication events, *i.e.*, *Non-communicative*, *Mutual Gaze*, *Gaze Aversion*, *Gaze Following* and *Joint Attention*, as illustrated in the right part of Figure 5.1. Typically the temporal order of atomic gazes means different phases of each event. *Non-communicative* (see Figure 5.1 (a)) and *Mutual Gaze* (see Figure 5.1 (b)) are one-phase events and simply consist of *single* and *mutual* respectively. *Gaze Aversion* (see Figure 5.1 (c)) starts from *mutual*, then *avert* to *single*, demonstrating the avoidance of mutual eye contact. *Gaze Following* (see Figure 5.1 (d)) is composed of *follow* and *share*, but without *mutual*, meaning that there is only one-way awareness and observation, no shared attention nor knowledge. *Joint Attention* (see Figure 5.1 (e)) is the most advanced and appears when two agents have the same intention to share attention on a common stimuli and both know that they are sharing something as common ground. Such event consists of several phases, typically beginning with *mutual* gaze to establish communication channel, proceeding to *refer* gaze to draw attention to the target, and *follow* gaze to check the referred stimuli, and cycling back to *mutual* gaze to ensure that the experience is shared [MDD14]. Clearly, recognizing and understanding atomic-level gaze communication patterns is necessary and significant first-step for comprehensively understanding human gaze behaviors.

To facilitate the research of gaze communication understanding in computer vision community, we propose a large-scale social video dataset named *VACATION* (Video gAze CommunicATION) with complete gaze communication annotations. With our dataset, we aim to build spatio-temporal attention graph given a third-person social video sequence with human face and object bboxes, and predict gaze communication relations for this video in both atomic-level and event-level. Clearly, this is a structured task that requires a comprehensive modeling of human-human and human-scene interactions in both spatial and temporal domains.

Inspired by recent advance in graph neural network [QWJ18b, VCC18], we propose a novel spatio-temporal reasoning graph network for atomic-level gaze communication detection as well as an event network with encoder-decoder structure for event-level gaze communication understanding. The reasoning model learns the relations among social entities and iteratively propagates information over a social graph. The event network utilizes the encoder-decoder structure to eliminate the noises in gaze communications and learns the temporal coherence for each event to classify event-level gaze communication.

This work makes **three major contributions**:

- It proposes and addresses a new task of gaze communication learning in social interaction videos. To the best of our knowledge, this is the first work to tackle such problem in computer vision community.
- It presents a large-scale video dataset, named *VACATION*, covering diverse social scenes with complete gaze communication annotations and benchmark results for advancing gaze communication study.
- It proposes a spatio-temporal graph neural network and an event network to hierarchically reason both atomic- and event-level gaze communications in videos.

5.1.2 Related Work

5.1.2.1 Gaze Communication in HHI

Eye gaze is closely tied to underlying attention, intention, emotion and personality [Kle86]. Gaze communication allows people to communicate at the most basic level regardless of their verbal language system. Such gaze functions thus transcend cultural differences, forming a universal language [BGF16]. During conversations, eye gaze can be used to convey information, regulate social intimacy, manage turn-taking [Kle86]. People are also good at identifying the target of their partner's referential gaze and use this information to predict what their partner is going to say [SC11, BPL12].

In a nutshell, gaze communication is omnipresent and multifunctional [BGF16]. Exploring the role of gaze communication in HHI is essential but has been rarely touched by computer vision researchers. Current research in computer vision community [IKN98, BI13, WS18, FWC19, WLF19] mainly focuses on studying the salient properties of environment to model human visual attention mechanism. Only a few [PJS12, PS15, FCW18] studied human shared attention behaviors in social scenes.

5.1.2.2 Gaze Communication in HRI

To improve human-robot collaboration, the field of HRI strives to develop effective gaze communication for robots [AS17]. Researchers in robotics tried to incorporate responsive, meaningful and convincing eye gaze into HRI [AS14, AMT15], which helps the humanoid agent to engender the desired familiarity and trust, and makes HRI more intuitive and fluent. Their efforts vary widely [SM11, ATG14, AS17], including human-robot visual dialogue interaction [MKF12, SC09, LII12], storytelling [MFH06], and socially assistive robotics [TMS07]. For example, a tutoring or assistive robot can demonstrate attention to and engagement with the user by performing proper *mutual* and *follow* gazes [MBS10], direct user attention to a target using *refer* gaze, and form joint attention with humans [HT11]. A collaborative assembly-line robot can also enable object reference and joint attention by gazes. Robots can also serve as therapy tools for children with autism.

5.1.2.3 Graph Neural Networks

Recently, graph neural networks [SGT09, LTB16, JZS16, GSR17] received increased interests since they inherit the complementary advantages of graphs (with strong representation ability) and neural networks (with end-to-end learning power). These models typically pass local messages on graphs to explicitly capture the relations among nodes, which are shown to be effective at a large range of structured tasks, such as graph-level classification [BZS14, DDS16, VCC18], node-level classification [HYL17], relational reasoning [SRB17, KFW18], multi-agent communications [SSF16, BPL16], human-object interactions [QWJ18b, FCT18], and scene understanding [MSG17, LTL17]. Some others [DMI15, NAK16, KW17, SK17, CLF18] tried to generalize convolutional architecture over graph-structured data. Inspired by above

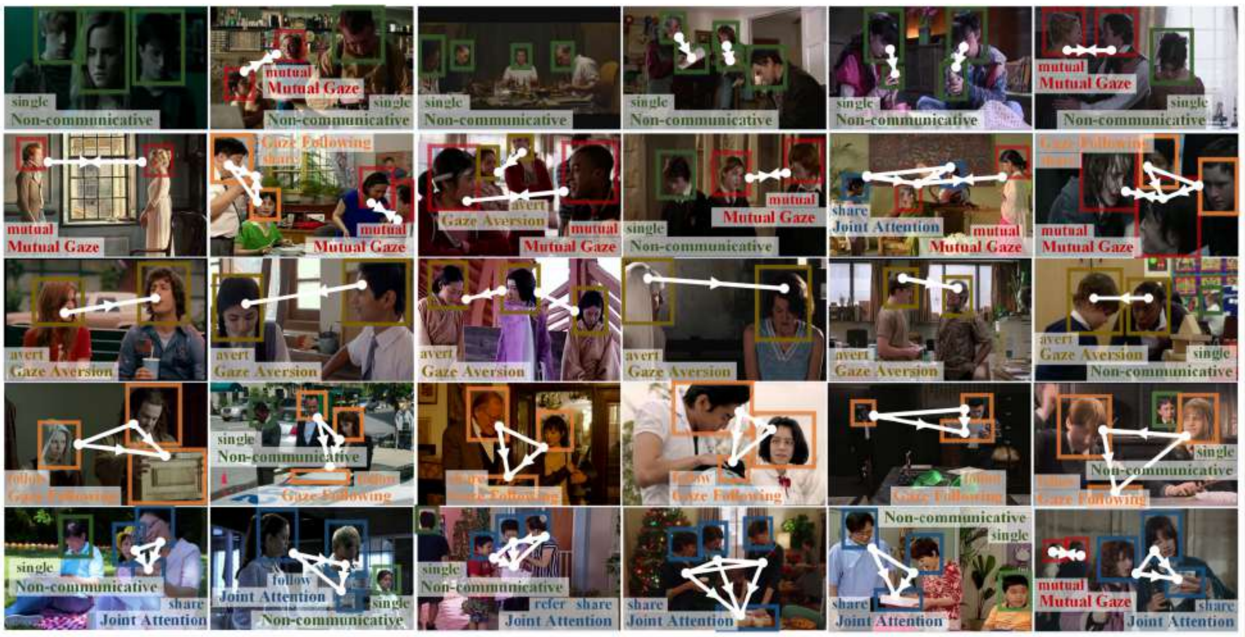


Figure 5.2: **Example frames and annotations of our *VACATION* dataset**, showing that our dataset covers rich gaze communication behaviors, diverse general social scenes, different cultures, *etc.*. It also provides rich annotations, *i.e.*, human face and object bboxes, gaze communication structures and labels. Human faces and related objects are marked by boxes with the same color of corresponding communication labels. White lines link entities with gaze relations in a temporal sequence and white arrows indicate gaze directions in the current frame. There may exist various number of agents, many different gaze communication types and complex communication relations in one frame, resulting in a highly-challenging and structured task. See Section 5.1.3 for details.

efforts, we build a spatio-temporal social graph to explicitly model the rich interactions in dynamic scenes. Then a spatio-temporal reasoning network is proposed to learn gaze communications by passing messages over the social graph.

5.1.3 The Proposed *VACATION* Dataset

VACATION contains 300 social videos with diverse gaze communication behaviors. Example frames can be found in Figure 5.2. Next we will elaborate *VACATION* from the following essential aspects.

Event-level (%)		*Non-Comm.*	*Mutual Gaze*	*Gaze Aversion*	*Gaze Following*	*Joint Attention*
		28.16	24.00	10.00	10.64	27.20
Atomic-level(%)	*single*	92.20	15.99	3.29	39.26	26.91
	mutual	0.76	75.64	14.15	0.00	16.90
	avert	1.34	6.21	81.71	0.00	1.18
	refer	0.00	0.37	0.15	0.62	7.08
	follow	1.04	0.29	0.00	10.71	2.69
	share	4.66	1.50	0.70	49.41	45.24

Table 5.1: **Statistics of gaze communication categories** in our *VACATION* dataset, including the distribution of event-level gaze communication category over full dataset and the distribution of atomic-level gaze communication for each event-level category.

5.1.3.1 Data Collection

Quality and diversity are two essential factors considered in our data collection.

High quality. We searched the Youtube engine for more than 50 famous TV shows and movies (*e.g.*, The Big Bang Theory, Harry Potter, *etc.*). Compared with self-shot social data in laboratory or other limited environments, these stimuli provide much more natural and richer social interactions in general and representative scenes, and are closer to real human social behaviors, which helps to better understand and model real human gaze communication behaviors. After that, about $1,000$ video clips are roughly split from the retrieved results. We further eliminate the videos with big logo or of low-quality. Each of the rest videos is then cropped with accurate shot boundaries and uniformly stored in MPEG-4 format with 640×360 spatial resolution. *VACATION* finally comprises a total of 300 high-quality social video sequences with 96,993 frames and 3,880-second duration. The lengths of videos span from 2.2 to 74.56 seconds and are 13.28 seconds on average.

Diverse social scenes. The collected videos cover diverse daily social scenes (*e.g.*, party, home, office, *etc.*), with different cultures (*e.g.*, American, Chinese, Indian, *etc.*). The appearances of actors/actresses, costume and props, and scenario settings, also vary a lot, which makes our dataset more diverse and general. By training on such data, algorithms are supposed to have better generalization ability in handling diverse realistic social scenes.

VACATION	# Video	# Frame	# Human	# GCR
training	180	57,749	123,812	97,265
validation	60	22,005	49,012	42,066
testing	60	17,239	33,950	25,034
full dataset	300	96,993	206,774	164,365

Table 5.2: **Statistics of dataset splitting**. GCR refers to Gaze Communication Relation. See Section 5.1.3.2 for more details.

5.1.3.2 Data Annotation and Statistics

Our dataset provides rich annotations, including human face and object bounding boxes, human attention, atomic-level and event-level gaze communication labels. The annotation takes about 1,616 hours in total, considering an average annotation time of 1 minute per frame. Three extra volunteers are included in this process.

Human face and object annotation. We first annotate each frame with bounding boxes of human face and key object, using the online video annotation platform Vatic [VPR13]. 206,774 human face bounding boxes (avg. 2.13 per frame) and 85,441 key object bounding boxes (avg. 0.88 per frame) are annotated in total.

Human attention annotation. We annotate the attention of each person in each frame, *i.e.*the bounding box (human face or object) this person is gazing at.

Gaze communication labeling. The annotators are instructed to annotate both atomic-level and event-level gaze communication labels for every group of people in each frame. To ensure the annotation accuracy, we used cross-validation in the annotation process, *i.e.*, two volunteers annotated all the persons in the videos separately, and the differences between their annotations were judged by a specialist in this area. See Table 5.1 for the information regarding the distributions of gaze communication categories.

Dataset splitting. Our dataset is split into training, validation and testing sets with the ratio of 6:2:2. We arrive at a unique split consisting of 180 training (57,749 frames), 60 validation (22,005 frames), and 60 testing videos (17,239 frames). To avoid over-fitting, there is no source-overlap among videos in different sets (see Table 5.2 for more details).

5.1.4 Our Approach

We design a spatio-temporal graph neural network to explicitly represent the diverse interactions in social scenes and infer atomic-level gaze communications by passing messages over the graph. Given the atomic-level gaze interaction inferences, we further design an event network with encoder-decoder structure for event-level gaze communication reasoning. As shown in Figure 5.3, gaze communication entities, *i.e.*, human, social scene, are represented by graph nodes, gaze communication structures are represented by edges. We introduce notations and formulations in Section 5.1.4.1 and provide more implementation details in Section 5.1.4.2.

5.1.4.1 Model Formulation

Social Graph. We first define a social graph as a *complete graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where node $v \in \mathcal{V}$ takes unique value from $\{1, \cdots, |\mathcal{V}|\}$, representing the entities (*i.e.*, scene, human) in social scenes, and edge $e = (v, w) \in \mathcal{E}$ indicates a directed edge $v \to w$, representing all the possible human-human gaze interactions or human-scene relations. There is a special node $s \in \mathcal{V}$ representing the social scene. For node v, its *node representation/embedding* is denoted by a V-dimensional vector: $\mathbf{x}_v \in \mathbb{R}^V$. Similarly, the *edge representation/embedding* for edge $e = (v, w)$ is denoted by an E-dimensional vector: $\mathbf{x}_{v,w} \in \mathbb{R}^E$. Each human node $v \in \mathcal{V} \backslash s$ has an output state $l_v \in \mathcal{L}$ that takes a value from a set of atomic gaze labels: $\mathcal{L} = \{$*single*, *mutual*, *avert*, *refer*, *follow*, *share*$\}$. We further define an adjacency matrix $\mathbf{A} \in [0, 1]^{|\mathcal{V}| \times |\mathcal{V}|}$ to represent the communication structure over our complete social graph $\mathcal{G}$, where each element $a_{v,w}$ represents the connectivity from node v to w.

Different from most previous graph neural networks that only focus on inferring graph- or node-level labels, our model aims to learn the graph structure $\mathbf{A}$ and the visual labels $\{l_v\}_{v \in \mathcal{V} \backslash s}$ of all the human nodes $\mathcal{V} \backslash s$ simultaneously.

To this end, our spatio-temporal reasoning model is designed to have two steps. First, in spatial domain, there is a message passing step (Figure 5.3 (b)) that iteratively learns gaze communication structures $\mathbf{A}$ and propagates information over $\mathbf{A}$ to update node representa-

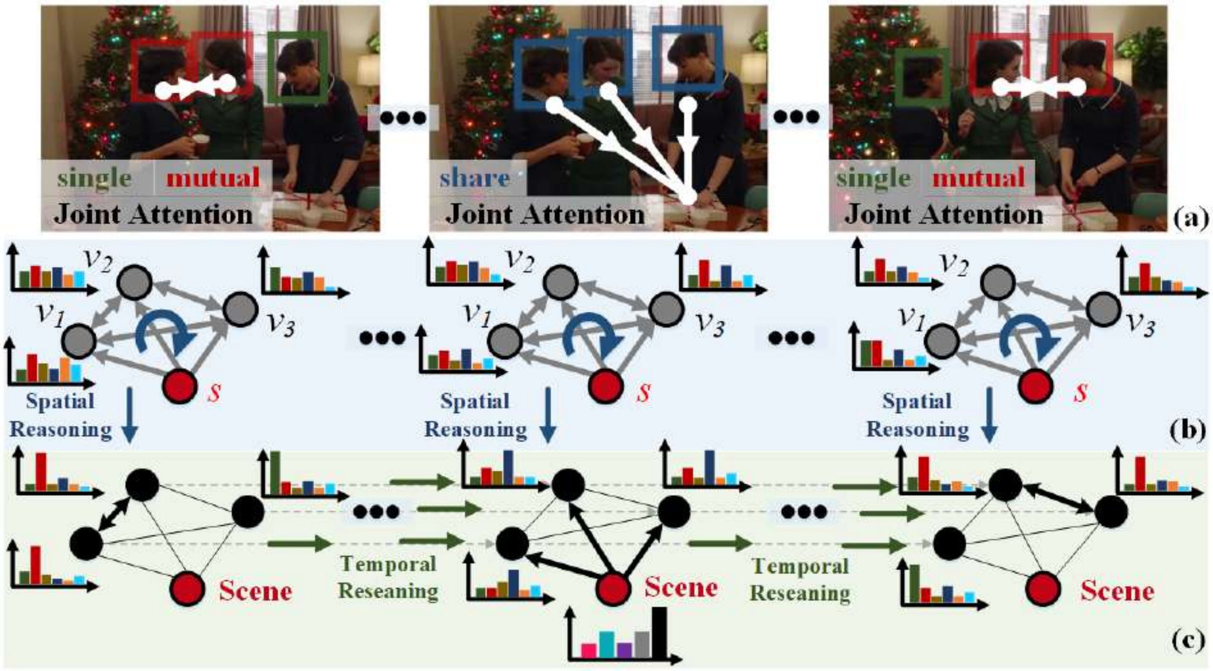


Figure 5.3: **Illustration of the proposed spatio-temporal reasoning model** for gaze communication understanding. Given an input social video sequence (a), for each frame, a spatial reasoning process (b) is first performed for simultaneously capturing gaze communication relations (social graph structure) and updating node representations through message propagation. Then, in (c), a temporal reasoning process is applied for each node to dynamically update node representation over temporal domain, which is achieved by an LSTM. Bolder edges represent higher connectivity weight inferred in spatial reasoning step (b). See Section 5.1.4.1 for details.

tions. Second, as shown in Figure 5.3 (c), an LSTM is incorporated into our model for more robust node representation learning by considering temporal dynamics. A more detailed model architecture is schematically depicted in Figure 5.4. In the following, we describe the above two steps in detail.

Message Passing based Spatial Reasoning. Inspired by previous graph neural networks [GSR17, QWJ18b, KFW18], our message passing step is designed to have three phases, an *edge update* phase, a *graph structure update* phase, and a *node update* phase. The whole message passing process runs for N iterations to iteratively propagate information. In n-th iteration step, we first perform the edge update phase that updates edge representations $\mathbf{y}_{v,w}^{(n)}$

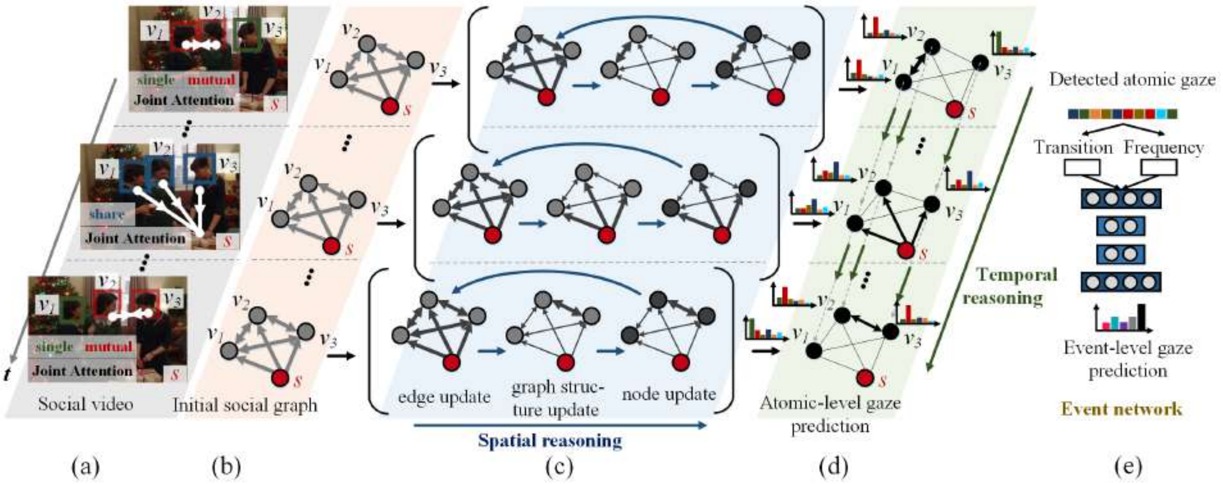


Figure 5.4: **Detailed architecture of the proposed spatio-temporal reasoning model** for gaze communication understanding. See the last paragraph in Section 5.1.4.1 for detailed descriptions.

by collecting information from connected nodes:

$$\mathbf{y}_{v,w}^{(n)} = f_E(\langle \mathbf{y}_v^{(n-1)}, \mathbf{y}_w^{(n-1)}, \mathbf{x}_{v,w} \rangle), \tag{5.1}$$

where $\mathbf{y}_v^{(n-1)}$ indicates the node representation of v in $(n-1)$-th step, and $\langle \cdot, \cdot \rangle$ denotes concatenation of vectors. f_E represents an *edge update function* $f_E : \mathbb{R}^{2V+E} \to \mathbb{R}^E$, which is implemented by a neural network.

After that, the graph structure update phase updates the adjacency matrix $\mathbf{A}$ to infer the current social graph structure, according to the updated edge representations $\mathbf{y}_{v,w}^{(n)}$:

$$a_{v,w}^{(n)} = \sigma(f_A(\mathbf{y}_{v,w}^{(n)})), \tag{5.2}$$

where the connectivity matrix $\mathbf{A}^{(n)} = [a_{v,w}^{(n)}]_{v,w}$ encodes current visual communication structures. $f_A : \mathbb{R}^E \to \mathbb{R}$ is a *connectivity readout network* that maps an edge representation into the connectivity weight, and σ denotes nonlinear activation function.

Finally, in the node update phase, we update node representations $\mathbf{y}_v^{(n)}$ via considering all the incoming edge information weighted by the corresponding connectivity:

$$\mathbf{y}_v^{(n)} = f_V(\langle \sum\nolimits_w a_{v,w}^{(n)} \mathbf{y}_{v,w}^{(n)}, \mathbf{x}_v \rangle), \tag{5.3}$$

where $f_V : \mathbb{R}^{V+E} \rightarrow \mathbb{R}^V$ represents a *node update network* .

The above functions $f(\cdot)$ are all learned differentiable functions. In the above message passing process, we infer social communication structures in the graph structure update phase (Equation (5.2)), where the relations between each social entities are learned through updated edge representations (Equation (5.1)). Then, the information is propagated through the learned social graph structure and the hidden state of each node is updated based on its history and incoming messages from its neighborhoods (Equation (5.3)). If we know whether there exist interactions between nodes (human, object), *i.e.*, given the groundtruth of $\mathbf{A}$, we can learn $\mathbf{A}$ in an *explicit* manner, which is similar to the graph parsing network [QWJ18b]. Otherwise, the adjacent matrix $\mathbf{A}$ can be viewed as an attention or gating mechanism that automatically weights the messages and can be learned in an *implicit* manner; this shares a similar spirit with graph attention network [VCC18]. More implementation details can be found in Section 5.1.4.2.

Recurrent Network based Temporal Reasoning. Since our task is defined on a spatio-temporal domain, temporal dynamics should be considered for more comprehensive reasoning. With the updated human node representations $\{\mathbf{y}_v \in \mathbb{R}^V\}_{v \in \mathcal{V} \backslash s}$ from our message passing based spatial reasoning model, we further apply LSTM to each node for temporal reasoning. More specifically, our temporal reasoning step has two phases: a *temporal message passing* phase and a *readout* phase. We denote by $\mathbf{y}_v^t$ the feature of a human node $v \in \mathcal{V} \backslash s$ at time t, which is obtained after N-iteration spatial message passing. In the temporal message passing phase, we propagate the information over the temporal axis using LSTM:

$$\mathbf{h}_v^t = f_{\text{LSTM}}(\mathbf{y}_v^t | \mathbf{h}_v^{t-1}), \tag{5.4}$$

where $f_{\text{LSTM}} : \mathbb{R}^V \rightarrow \mathbb{R}^V$ is an LSTM based temporal reasoning function that updates the node representation using temporal information. $\mathbf{y}_v^t$ is used as the input of the LSTM at time t, and $\mathbf{h}_v^t$ indicates the corresponding hidden state output via considering previous information $\mathbf{h}_v^{t-1}$.

Then, in the readout phase, for each human node v, a corresponding gaze label $\hat{l}_v^t \in \mathcal{L}$ is predicted from the final node representation $\mathbf{h}_v^t$:

$$\hat{l}_v^t = f_R(\mathbf{h}_v^t), \tag{5.5}$$

where $f_R : \mathbb{R}^V \to \mathcal{L}$ maps the node feature into the label space $\mathcal{L}$, which is implemented by a classifier network.

Event Network. The event network is designed with an encoder-decoder structure to learn the correlation of the atomic gazes and classify the event-level gaze communication for each video sequence. To reduce the large variance of video length, we pre-process the input atomic gaze sequence into two vectors: i) the transition vector that records each transition from one category of atomic gaze to another, and ii) the frequency vector that computes the frequency of each atomic type. The encoder individually encodes the transition vector and frequency vector into two embedded vectors. The decoder decodes the concatenation of these two embedded vectors and makes final event label prediction. Since the atomic gaze communications are noisy within communicative activities, the encoder-decoder structure will try to eliminate the noise and improve the prediction performance. The encoder and decoder are both implemented by fully-connected layers.

Before going deep into our model implementation, we offer a short summary of the whole spatio-temporal reasoning process. As shown in Figure 5.4, with an input social video (a), for each frame, we build an initial complete graph $\mathcal{G}$ (b) to represent the gaze communication entities (*i.e.*, humans and social scene) by nodes and their relations by edges. During the spatial reasoning step (c), we first update edge representations using Equation (5.1) (note the changed edge color compared to (b)). Then, in the graph structure update phase, we infer the graph structure through updating the connectivities between each node pairs using Equation (5.2) (note the changed edge thickness compared to (b)). In the node update phase, we update node embeddings using Equation (5.3) (note the changed node color compared

Task	Atomic-level Gaze Communication (Precision & F1-score)													
	single		*mutual*		*avert*		*refer*		*follow*		*share*		*Avg. Acc.*	
Metric	$\mathcal{P}$ (%) ↑	$\mathcal{F}$ (%) ↑	$\mathcal{P}$ (%) ↑	$\mathcal{F}$ (%) ↑	$\mathcal{P}$ (%) ↑	$\mathcal{F}$ (%) ↑	$\mathcal{P}$ (%) ↑	$\mathcal{F}$ (%) ↑	$\mathcal{P}$ (%) ↑	$\mathcal{F}$ (%) ↑	$\mathcal{P}$ (%) ↑	$\mathcal{F}$ (%) ↑	top-1 (%) ↑	top-2 (%) ↑
Ours-full *(iteration 2)*	22.10	26.17	98.68	98.60	59.20	74.28	56.90	53.16	32.83	18.05	61.51	46.61	**55.02**	76.45
Chance	16.50	16.45	16.42	16.65	16.65	16.51	16.07	16.06	16.80	16.74	16.20	16.25	16.44	-
CNN	21.32	27.89	15.99	14.48	47.81	50.82	0.00	0.00	19.21	23.10	11.70	2.80	23.05	40.32
CNN+LSTM	22.10	11.78	18.55	16.37	64.24	59.57	13.69	18.55	22.70	29.13	17.18	3.61	24.65	45.50
CNN+SVM	19.92	23.63	28.46	38.30	68.53	76.07	15.15	6.32	23.28	16.87	40.76	49.24	36.23	-
CNN+RF	53.12	57.98	20.78	0.24	0.00	0.00	51.88	27.31	15.90	19.39	35.56	44.42	37.68	-
PRNet	0.00	0.00	47.52	52.54	89.63	58.00	19.49	21.52	19.72	22.05	48.69	62.40	39.59	61.45
VGG16	35.55	48.93	99.70	99.85	76.95	13.04	37.02	31.88	26.62	20.89	53.05	59.88	49.91	72.18
Resnet50 (192-d)	33.61	38.19	78.22	85.66	62.27	76.75	18.58	11.21	35.89	18.55	57.82	60.26	53.72	77.16
AdjMat-only	34.00	22.63	31.46	22.81	38.06	52.42	27.70	26.79	25.42	25.25	32.32	28.69	32.64	46.48
2 branch-iteration 2	20.43	8.93	92.65	76.03	47.57	59.47	40.34	45.35	36.36	35.77	55.15	57.93	49.57	80.33
2 branch-iteration 3	18.92	19.67	99.72	97.18	57.69	60.18	11.92	6.19	31.10	20.40	39.67	53.22	46.39	66.77
Ours-iteration 1	6.69	4.66	49.39	47.96	36.56	39.44	25.89	27.82	35.05	31.93	36.71	42.22	33.67	53.97
Ours-iteration 3	44.83	0.77	51.29	66.41	47.09	64.03	0.00	0.00	25.95	26.20	47.42	46.74	44.52	72.77
Ours-iteration 4	28.01	5.77	99.59	93.15	42.06	59.06	38.46	14.02	22.02	17.54	43.69	55.77	48.35	72.35
Ours w/o. temporal reason.	13.74	10.80	98.64	98.54	54.54	53.17	55.87	53.75	40.83	25.00	45.89	61.55	53.73	80.33
Ours w. implicit learn.	30.60	9.15	33.00	34.56	43.39	56.00	21.50	26.98	22.43	18.63	58.30	39.33	33.74	56.54

Table 5.3: **Quantitative results of atomic-level gaze communication prediction**. The best scores are marked in **bold**.

to (b)). Iterating above processes leads to efficient message propagation in spatial domain. After several spatial message passing iterations, we feed the enhanced node feature into a LSTM based temporal reasoning module, to capture the temporal dynamics (Equation (5.4)) and predict final atomic gaze communication labels (Equation (5.5)). We then use event network to reason about event-level labels based on previous inferred atomic-level label compositions for a long sequence in a larger time scale.

5.1.4.2 Detailed Network Architecture

Attention Graph Learning. In our social graph, the adjacency matrix $\mathbf{A}$ stores the attention relations between nodes, *i.e.*, representing the interactions between the entities in the social scene. Since we have annotated all the directed human-human interactions and human-scene relations (Section 5.1.3.2), we learn the adjacency matrix $\mathbf{A}$ in an explicit manner (under the supervision of ground-truth). Additionally, for the scene node s, since it's a 'dummy' node, we enforce $a_{v,s}$ as 0, where $v \in \mathcal{V}$. In this way, other human nodes cannot influence the state of the scene node during message passing. In our experiments, we will offer more detailed results regarding learning $\mathbf{A}$ in an implicit (*w/o.* ground-truth) or explicit manner.

Node/Edge Feature Initialization. For each node $v \in \mathcal{V} \backslash s$, the 4096-$d$ features (from the

Task	Event-level Gaze Communication (Precision & F1-score)											
	Non-Comm.		*Mutual Gaze*		*Gaze Aversion*		*Gaze Following*		*Joint Attention*		*Avg. Acc.*	
Metric	$\mathcal{P}$ (%) ↑	$\mathcal{F}$ (%) ↑	$\mathcal{P}$ (%) ↑	$\mathcal{F}$ (%) ↑	$\mathcal{P}$ (%) ↑	$\mathcal{F}$ (%) ↑	$\mathcal{P}$ (%) ↑	$\mathcal{F}$ (%) ↑	$\mathcal{P}$ (%) ↑	$\mathcal{F}$ (%) ↑	top-1 (%) ↑	top-2 (%) ↑
Chance	21.3	29.3	25.0	23.0	20.0	14.8	36.3	15.1	20.3	22.1	22.7	45.0
FC-w/o. GT	43.7	44.3	16.9	23.3	6.2	10.0	8.3	9.1	60.9	40.2	35.6	69.1
Ours-w/o. GT	50.7	49.3	16.7	21.0	8.2	11.3	6.2	7.7	60.9	40.0	**37.1**	65.5
FC-w. GT	90.7	70.7	12.3	30.8	22.2	30.8	15.0	48.3	56.8	57.1	52.6	86.5
Ours-w. GT	91.4	72.7	14.5	32.3	18.5	45.5	20.0	66.7	62.2	30.8	**55.9**	79.4

Table 5.4: **Quantitative results of event-level gaze communication prediction**. The best scores are marked in **bold**.

fc7 layer of a pre-trained ResNet50 [HZR16]) are extracted from the corresponding bounding box as its initial feature $\mathbf{x}_v$. For the scene node s, the *fc7* feature of the whole frame is used as its node representation $\mathbf{x}_s$. To decrease the amount of parameter, we use fully connected layer to compress all the node features into 6-d and then encode a 6-d node position info with it. For an edge $e=(v,w)\in\mathcal{V}$, we just concatenate the related two node features as its initial feature $\mathbf{x}_{v,w}$. Thus, we have $V=12$ and $E=24$.

Graph Network Implementations. The functions $f(\cdot)$ in Equation (5.1), Equation (5.2) and Equation (5.5) are all implemented by fully connected layers, whose configurations can be determined according to their corresponding definitions. The function in Equation (5.3) is implemented by gated recurrent unit (GRU) network.

Loss functions. When explicitly learning the adjacency matrix, we treat it as a binary classification problem and use the *cross entropy* loss. We also employ standard *cross entropy* loss for the multi-class classification of gaze communication labels.

5.1.5 Experiments

5.1.5.1 Experimental Setup

Evaluation Metrics. Four evaluation metrics, we use precision, F1-score, top-1 Avg. Acc. and top-2 Avg. Acc. in our experiments. Precision $\mathcal{P}$ refers to the ratio of true-positive classifications to all positive classifications. F1-score $\mathcal{F}$ is the harmonic mean of the precision and recall: $2\times\text{precision}\times\text{recall}/(\text{precision}+\text{recall})$. Top-1 Avg. Acc. and top-2 Avg. Acc. calculate the average label classification accuracy over all the test set.

Implementation Details. Our model is implemented by PyTorch. During training phase,

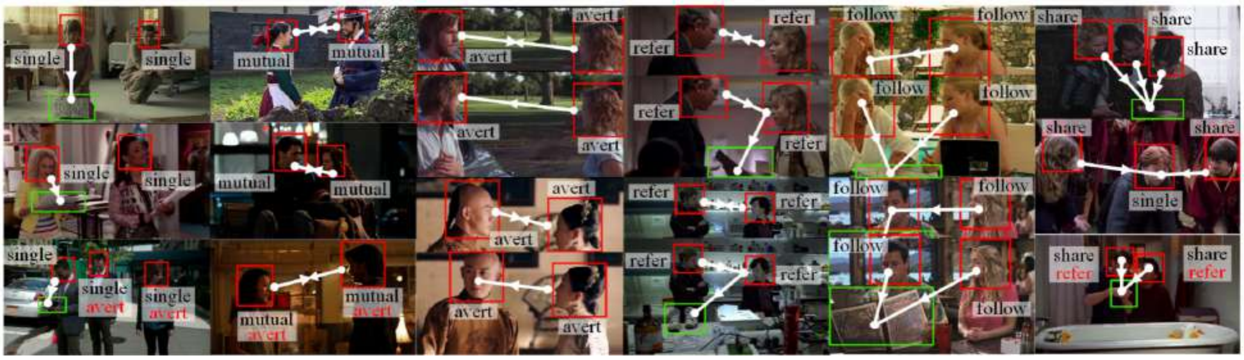


Figure 5.5: **Qualitative results of atomic-level gaze communication prediction**. Correctly inferred labels are shown in black while error examples are shown in red.

the learning rate is set to 1e-1, and decays by 0.1 per epoch. For the atomic-gaze interaction temporal reasoning module, we set the sequential length to 5 frames according to our dataset statistics. The training process takes about 10 epochs (5 hours) to roughly converge with an NVIDIA TITAN X GPU.

Baselines. To better evaluate the performance of our model, we consider the following baselines:

• *Chance* is a weak baseline, *i.e.*, randomly assigning an atomic gaze communication label to each human node.

• *CNN* uses three *Conv2d* layers to extract features for each human node and concatenates the features with position info. for label classification (no spatial communication structure, no temporal relations).

• *CNN+LSTM* feeds the CNN-based node feature to an LSTM (only temporal dynamics, no spatial structures).

• *CNN+SVM* concatenates the CNN-based node features and feeds it into a Support Vector Machine classifier.

• *CNN+RF* replaces the above SVM classifier with a Random Forest classifier.

• *FC-w/o. GT & FC-w. GT* are fully connected layers without or with ground truth atomic gaze labels.

Ablation Study. To assess the effectiveness of our essential model components, we derive the following variants:

• *Different node feature.* We try different ways to extract node features. *PRNet* uses 68 3D face keypoints extracted by PRNet [FWS18]. *VGG16* replaces Resnet50 with VGG16 [SZ14]. *Resnet50 (192-d)* compresses the 4096-d features from fc7 layer of Resnet50 [HZR16] to 192-d.

• *AdjMat-only* directly feeds the explicitly learned adjacency matrix into some *Conv3d* layers for classification.

• *2 branch* concatenates a second adjacency matrix branch alongside the GNN branch for classification. We test with different message passing iterations.

• *Ours-iteration 1,2,3,4* test different message passing iterations in the spatial reasoning phase of our full model.

• *Ours w/o. temporal reason.* replaces LSTM with *Cond3d* layers in the temporal reasoning phase of our full model.

• *Ours w. implicit learn.* is achieved by unsupervisedly learning adjacent matrix $\mathbf{A}$ (*w/o.* attention ground truths).

5.1.5.2 Results and Analyses

Overall Quantitative Results. The quantitative results are shown in Table 5.3 and Table 5.4 respectively for the atomic-level and event-level gaze communication classification experiments. For the atomic-level task, our full model achieves the best top-1 avg. acc. (55.02%) on the test set and shows good and balanced performance for each atomic type instead of overfitting to certain categories. For the event-level task, our event network improves the top-1 avg. acc. on the test set, achieving 37.1% with the predicted atomic labels and 55.9% with the ground truth atomic labels.

In-depth Analyses. For atomic-level task, we examined different ways to extract node features and find Restnet50 the best. Also, compressing the Resnet50 feature to a low dimension still performs well and efficiently (full model vs. Resnet50 192-d). The performance of *AdjMat-only* which directly uses the concatenated adjacency matrix can obtain some reasonable results compared to the weak baselines but not good enough, which is probably because that gaze communication dynamic understanding is not simply about geometric attention relations, but also depends on a deep and comprehensive understanding of spatial-temporal scene context. We examine the effect of iterative message passing and find it is able to gradually improve the performance in general. But with iterations increased to a certain extent, the performance drops slightly.

Qualitative Results. Figure 5.5 shows some visual results of our full model for atomic-level gaze communication recognition. The predicted communication structures are shown with bounding boxes and arrows. Our method can correctly recognize different atomic-level gaze communication types (shown in black) with effective spatial-temporal graph reasoning. We also present some failure cases (shown in red), which may be due to the ambiguity and subtlety of gaze interactions, and the illegibility of eyes. Also, the shift between gaze phases could be fast and some phases are very short, making it hard to recognize.

5.1.6 Conclusion

We address a new problem of inferring human gaze communication from both atomic-level and event-level in third-person social videos. We propose a new video dataset *VACATION* and a spatial-temporal graph reasoning model, and show benchmark results on our dataset. We hope our work will serve as important resources to facilitate future studies related to this important topic.

5.2 A Multi-view Dataset for Learning Multi-task Multi-agent Activities

Understanding and interpreting human actions is a long-standing challenge and a critical indicator of perception in artificial intelligence. However, a few imperative components of daily human activities are largely missed in prior literature, including the goal-directed actions, concurrent multi-tasks, and collaborations among multi-agents.

In this section, we introduce the LEMMA dataset to provide a single home to address these missing dimensions with meticulously designed settings, wherein the number of tasks and agents varies to highlight different learning objectives. We densely annotate the atomic-actions with human-object interactions to provide ground-truths of the compositionality, scheduling, and assignment of daily activities. We further devise challenging compositional action recognition and action/task anticipation benchmarks with baseline models to measure the capability of compositional action understanding and temporal reasoning. We hope this effort would drive the machine vision community to examine goal-directed human activities and further study the task scheduling and assignment in the real world.

5.2.1 Introduction

Activity understanding is one of the most fundamental problems in artificial intelligence and computer vision. As the most readily available learning source, videos of daily human activities could be used to train intelligent agents and, in turn, to assist humans. However, compared to recent progress in learning from static images [AAL15, HZR16, HGD17, RHG15], current machine vision's ability to understand activities from videos still falls short. Admittedly, activity understanding is inherently more challenging, which requires reason about the complex structures in activities along the additional temporal dimension; but we argue there are more profound reasons that we must look back to the origin of activity understanding.

The study and analysis of human motion perception are rooted in the field of neuroscience [TCS08]. Using a dot-representation of human motions, Johansson [Joh73] adopted

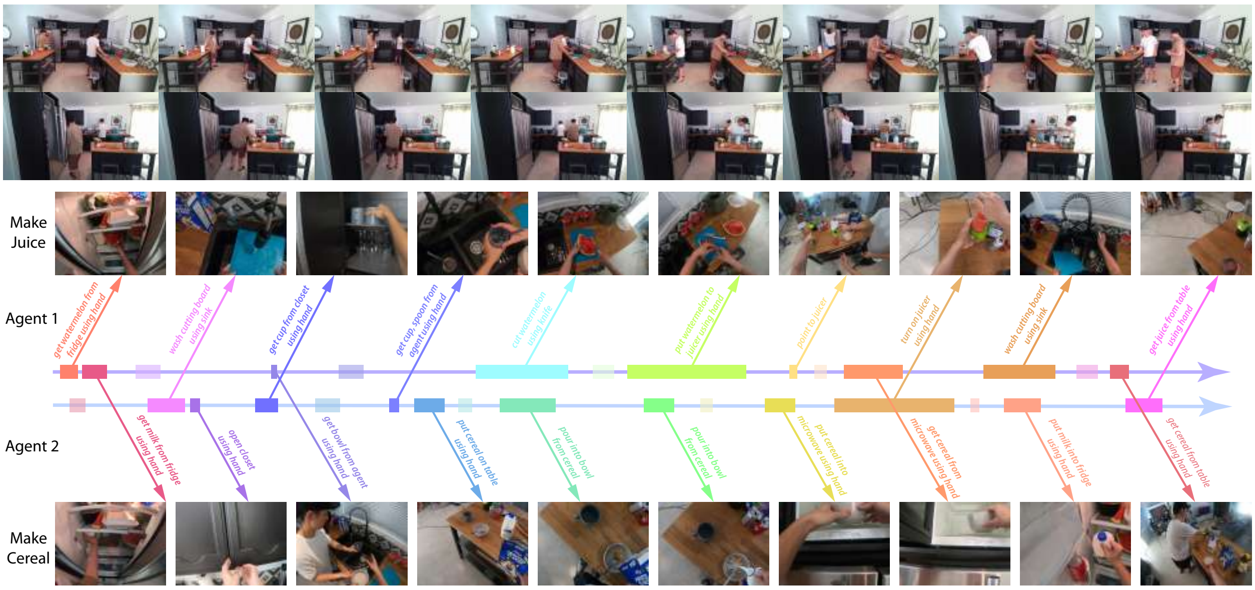

Make Juice
Agent 1
Agent 2
Make Cereal
get watermelon from fridge using hand
wash cutting board using sink
get cup from closet using hand
get cup, spoon from agent using hand
cut watermelon using knife
put watermelon to juicer using hand
point to juicer
turn on juicer using hand
wash cutting board using sink
get juice from table using hand
get milk from fridge using hand
open closet using hand
get bowl from agent using hand
put cereal on table using hand
pour into bowl from cereal
pour into bowl from cereal
put cereal into microwave using hand
get cereal from microwave using hand
put milk into fridge using hand
get cereal from table using hand

Figure 5.6: Illustrations of the proposed multi-view dataset with annotations. From top to bottom: frames captured from the third-person primary view, frames captured from the third-person side view, annotated segments of each agent executing tasks, and corresponding frames captured from the first-person view.

a method to produce proximal patterns (*i.e.*, the moving light display experiment), which demonstrated that human perception of activities does not tightly couple with *pixel-based features*; human subjects can still perceive the semantics of activities from *sparse* representations of motions. Evidence from developmental psychology, the classic Heider-Simmel experiment, further suggests that we perceive human activities from as *goal-directed* behaviors [Woo98, BBS01, GBK02b, CG07]; it is the underlying intent, rather than the surface pixels or behavior, that matters when we observe motions [BB01]. Such a **goal-directed [LMR99] perspective** of activity understanding has been largely left untouched in computer vision.

Daily human activities are intrinsically multi-tasked [Mon03, RME01]; understanding activity naturally demands a learning system to interpret concurrent interactions. As agents' decision-making processes are deeply affected by their unique social values, task scheduling is significantly affected by interactions (*e.g.*, cooperation, competition, subordination) among multi-agents [KHA16]. These observations implicate that the machine vision system must objectively understand how a given task should be decomposed into atomic-actions, how multi-tasks should be executed and coordinated in parallel among multi-agents, and take the perspective from human agents to understand why the observed human activities are optimal solutions. Such a **decompositional, multi-task, multi-agent, diagnostic-driven, social perspective** of activity understanding is critical for an intelligent agent to understand human behavior and team with humans collaboratively; yet it is broadly missing in activity understanding literature.

The semantics of human actions are intrinsically ambiguous when described in natural language. For instance, although both "opening the fridge" and "opening a book" use the action verb "open," their semantics of the actions are utterly different. In this paper, we take the stance of Grice's influential work on language act [Gri75]—technical tools for reasoning about rational action should elucidate linguistic phenomena [GF16]. Specifically, the compositional relations between the verbs and nouns could reveal the functionality of the object and the patterns of human-object interactions, which subsequently facilitate the understanding of the observed human activities and the language that describes them. Though the previous work [GKM17] attempted to address this issue, more general and flexible **compositional relations for describing human actions interacting with objects** are requisite for a goal-directed activity understanding.

Motivated by these deficiencies in prior work, we introduce the LEMMA dataset to explore the essence of complex human activities in a goal-directed, multi-agent, multi-task setting with ground-truth labels of compositional atomic-actions and their associated tasks. By quantifying the scenarios to up to two multi-step tasks with two agents, we strive to address human multi-task and multi-agent interactions in four scenarios: single-agent single-task (1×1), single-agent multi-task (1×2), multi-agent single-task (2×1), and multi-agent multi-task (2×2). Task instructions are only given to one agent in the 2×1 setting to resemble the robot-helping scenario, hoping that the learned perception models could be applied in robotic tasks (especially in HRI) in the near future.

Both the third-person views (TPVs) and the first-person views (FPVs) were recorded to account for different perspectives of the same activities; see Figure 5.6. We densely annotate atomic-actions (in the form of compositional verb-noun pairs) and tasks of each atomic-action, to facilitate the learning of multi-agent multi-task task scheduling and assignment; see more details in Section 5.2.3.

5.2.2 Related Work

In this section, we review and compare prior indoor activity datasets on the basis of tasks and captured video contents; see a detailed summary in Table 5.5.

Crowd-sourced from online videos and movie sharing platforms, typical large-scale video datasets [SZS12, KTS14, CEG15, CZ17, FKE18] focus on **video-level summarization and classification**. Although activity classes exhibit a large inter-class variability, spanning from outdoor sports activities to indoor household activities, they generally lack sequential, goal-directed activities. Notably, they suffer from a major drawback [GR20]; activities are highly correlated to the general scene and object context, possessing a strong dataset bias for activity understanding.

Some datasets tackle the **human atomic-actions** using short clips or limited tasks, with a focus on the semantics of action verbs and objects [GKM17], 3D action analysis [LZL10, IPO13, SCH16], and action grounding with multi-modality inputs [MAZ19]. Although such datasets are suitable for atomic-actions, they are intrinsically impaired at studying the long-term reasoning of goal-directed human activities.

Recently, **concurrent actions** have been taken into consideration. For instance, Charades [SVW16] is a large-scale benchmark for household activities, and Charades-Ego [SGS18] steps further with both FPVs and TPVs. However, the activities involved are mostly unre-

Table 5.5: Comparisons between LEMMA and relevant indoor activity datasets.

Dataset	Task Annotation	Multi-agent	Multi-task	Multi-view	Samples	Frames	Action Classes	Action Segments	Actions per Video	Modality	Year
MPII Cooking [RAA12]	✓	✗	✗	✗	273	2.9M	88	14,105	51.7	RGB	2012
ADL [PR12]	✗	✗	✓	✗	20	1.0M	32	436	13.6	RGB	2012
50Salads [SM13]	✓	✗	✗	✗	50	0.5M	17	966	19.3	RGB-D	2013
CAD-120 [KGS13]	✗	✗	✗	✗	120	0.1M	10	1,175	9.8	RGB-D	2013
Breakfast [KAS14]	✓	✗	✗	✓	433	3.0M	50	3,078	7.1	RGB	2014
Watch-n-Patch [WZS15]	✓	✗	✗	✗	458	0.1M	21	2978	6.5	RGB-D	2015
Charades [SVW16]	✗	✗	✓	✗	9,848	7.4M	157	67,000	6.8	RGB	2016
Something-Something [GKM17]	✗	✗	✗	✗	108,499	-	174	108,499	1.0	RGB	2017
EGTEA GAZE+ [LLR18]	✓	✗	✗	✗	86	2.4M	106	10,325	120.1	RGB	2018
EPIC-KITCHENS [DDM18]	✗	✗	✓	✗	432	11.5M	149	39,596	91.7	RGB	2018
LEMMA (proposed)	✓	✓	✓	✓	324	4.6M	641	11,781	36.4	RGB-D	2020

lated to specific goals due to the crowdsourced script generation process. Similarly, although Multi-THUMOS [YRJ18] and AVA [GSR18] focus on highly paralleled activities, and some datasets look at the temporal order of activities [BLB14, TZS16], the unnaturally scripted activities result in the lack of meaningful goal-directed tasks exhibited in our daily life.

Conversely, **instructional video** datasets [ABA16, SM13, KAS14, KGS13, RRR16] tackle goal-directed multi-step tasks, mostly in cooking, repairing, and assembling activities. In spite of their relevance, they fail to account for multi-agent or multi-task problems. EPIC-KITCHENS [DDM18] is perhaps the only exception; it records naturally paralleled task execution of agents in kitchen environments, but with no task specification or multi-agent interactions. Additionally, prior instructional video datasets have either drastic view perspective changes [ZXC18, ABA16, TDR19, TCH17] or limited egocentric view with severe occlusions [PR12, LLR18], hindering the activity understanding.

Another related stream of work is the learning of group-level activities in a **multi-agent** setting [IMD16], such as detecting key actors [RHA16], predicting future trajectories [PES09, LCL07], and recognizing collective activities [CSS09, OHP11, SXR15]. However, such coarse-grained multi-agent interactions leave the latent subtlety of collaboration and task assignment untouched. Although simulation-based multi-agent environments [BKM20, VBC19, BBC19] can partially address such an issue, learning from noisy and real visual input in physical work is still essential for understanding collaborative planning behaviors of agents in the context of complex daily tasks.

The collected LEMMA dataset strives to address the shortcomings of the aforementioned works, capturing goal-directed, decompositional, multi-task activities with multi-agent collaborations. As shown in Table 5.5, the size, annotation, and actions per video of LEMMA are at a comparable scale to state-of-the-art benchmarks. We hope such a design will boost the study of human activity understanding and potentially motivate new cross-disciplinary research insights.

5.2.2.1 Contributions

This paper's contribution is three-fold. (i) We design and collect a multi-view video dataset, capturing multi-agent, multi-task activities with goal-directed daily tasks. (ii) We annotate the dataset, focusing on the compositionality of actions and the governing task for each atomic-action. (iii) We provide compositional action recognition and action/task anticipation benchmarks by considering the aforementioned features; we also compare and analyze multiple baseline models to promote future research on human activity understanding.

5.2.3 The LEMMA Dataset

This section describes the design, data collection, and data annotation process of the LEMMA dataset. The dataset is profiled by various statistics from diversified perspectives to highlight its potentials in activity understanding.[1]

5.2.3.1 Activities and Scenarios

We first build a task pool of 15 common tasks in the kitchen (*e.g.*, "make juice," "make cereal") and living room (*e.g.*"watch TV," "water plant"). On top of these tasks, we design four types of scenarios (with a different focus) to study goal-directed multi-step multi-task indoor activities in multi-agent settings.

1. **Single-agent Single-task (1 × 1):** Each participant was first asked to perform all tasks from the task pool independently; this ensures participants are clear with the goal of each task and could schedule and assign tasks efficiently in later multi-task or multi-agent scenarios. Participants were asked to read the instructions and walk around to get familiarized with the new environments.

2. **Single-agent Multi-task (1 × 2):** Each participant was then asked to simultaneously perform two tasks, randomly sampled from the task pool. The participants determined

[1] The dataset will be made publicly available at the following website with download links and util code: `https://sites.google.com/view/lemma-activity`.

the order of task executions without any restrictions.

3. **Multi-agent Single-task (2×1):** Two participants were asked to perform a single task cooperatively; the task is randomly selected from the task pool. To emulate human-robot teaming accurately, only one participant (leader) was provided with task instructions; the other participant (helper), with no knowledge of the task, was asked to collaborate with the leader agent to finish the task efficiently. Only nonverbal communications (*e.g.*, gestures) were allowed between two participants; this design would open up new venues on nonverbal communications and the emergence of language in real-world environments.

4. **Multi-agent Multi-task (2×2):** Both participants were provided with task instructions. Since both participants were asked to accomplish two complex multi-step tasks collaboratively, this scenario has the most natural activity/task patterns and richest mechanisms for learning task scheduling and assignment.

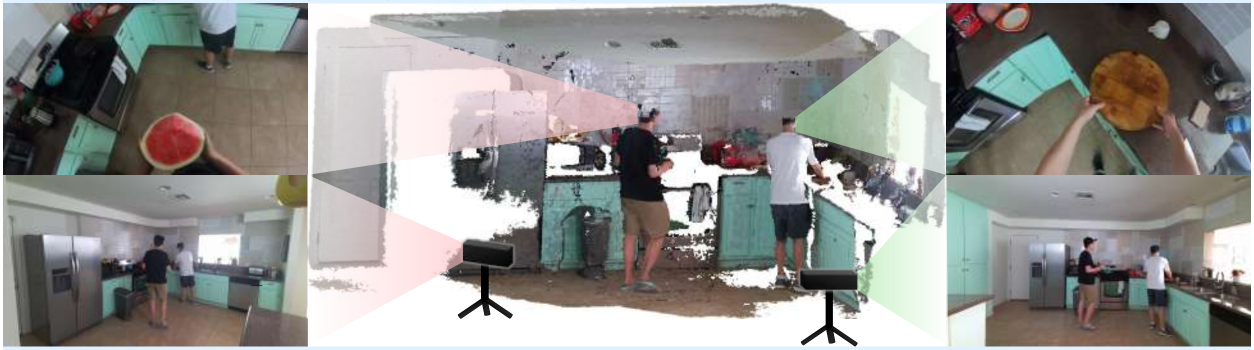
In total, the LEMMA dataset includes 37 unique task combinations in the multi-task scenarios. Participants were explicitly instructed to perform tasks efficiently and provided with a brief task instruction with basic environment information. Except for the specification of the goal states for each task, we add no additional constraint to the order of task execution; participants perform tasks naturally and freely. Figure 5.7 shows a sample instruction for the 2×1 scenario.

5.2.3.2 Data Collection

We recorded the data in 7 different Airbnb houses, performed by 8 individuals in 14 unique kitchens/living rooms. To provide different views of performing the daily activities and avoid occlusion in narrow spaces, we set up two Kinect Azure cameras to capture the RGB-D videos of the global scene and human bodies. In addition, each participant was instructed to wear a head-mounted GoPro camera to capture detailed agent-specific actions in an egocentric view.

Leader

In this task, you are asked to **make watermelon juice**. Here are things to know before your start:
- All the items needed for this task can be found either in the **fridge**, on the **table**, or in one of the **drawers** or **closets**.
- Please **cut** the **watermelon** into pieces before blending it with the **juicer**.
- Please keep the kitchen clean; wash all the **tools/objects** you used.
- You will have an additional **helper** to collaborate with you.
  - Do **Not** speak with them. They do **NOT** know anything about the task you are working on.
  - Feel free to ask them for help, but only using **non-verbal** communication (e.g., gestures). For instance, you may point to something, or any other gestues you think may help instruct them.



Helper

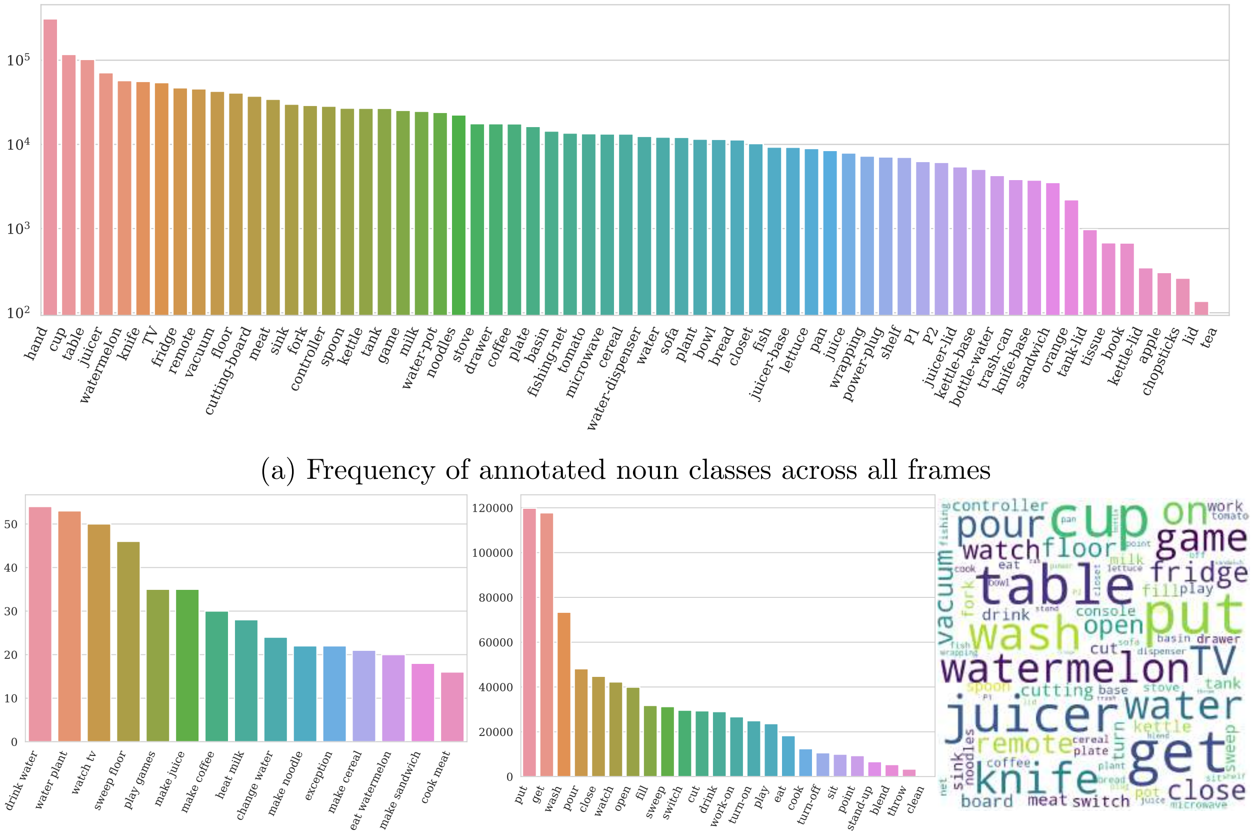
In this task, you are asked to **collaborate** with your friend to finish a task in the kitchen.
Here are things to know before your start:
- All the items needed for this task can be found either in the **fridge**, on the **table**, or in one of the **drawers** or **closets**.
- Please keep the kitchen clean; wash all the **tools/objects** you used.
- As only your friend knows the task instruction, please try to infer what the task is and offer helps.
- **You may not speak with your friend**. You can only use **non-verbal** communication (e.g., gestures).

Figure 5.7: An exemplar task instruction of making juice for two agents in a **Multi-agent Single-task** (2×1) scenario. Middle: Point clouds, TPVs, and FPVs.

In post-processing, we synchronize the camera recordings of all views at a frame rate of 24 FPS. Figure 5.7 shows an example of a scene with a point cloud merged from two Kinects and four RGB views from both Kinects and GoPros. Combining TPVs and FPVs captures most of the details of performing daily activities, provides sufficient data for understanding human activities, and benefits future research in embodied vision. The additional depth information and 3D human skeletons captured by Kinects can also be adopted for future 3D understanding tasks.

5.2.3.3 Ground-truth Annotation

We used the Amazon Mechanical Turk (AMT) to annotate both human bounding boxes and action information in the synchronized recordings. Specifically, action information includes the temporal localization of segments, semantic labels, and the governing task of each atomic-action. The semantic labels of atomic-actions are composed of verbs and nouns, representing



(a) Frequency of annotated noun classes across all frames

(b) figures/lemma/Frequency of recorded tasks

(c) Frequency of annotated verb

(d) Action wordle

Figure 5.8: Statistics of the LEMMA dataset.

flexible compositional relations to describe human actions. Additional details are provided below.

Bounding Boxes and Segments: Bounding boxes of humans are annotated on the primary view of TPVs. Skeletons captured by Kinects are used to provide initial estimations of bounding boxes. Next, we use Vatic [VPR13] to adjust bounding boxes and annotate the segments of atomic-actions. The segments of atomic-actions are defined by verbs without corresponding nouns, for example, “put __ to __ using __,” “pour into __ from __.” Each video was first annotated by two AMT workers; task-irrelevant actions (*e.g.*, “walking,” “holding”) are ignored. We then compute the Intersection over Union (IoU) of both bounding boxes and temporal segments. A third AMT worker is asked to fine-tune the annotations if the IoU of bounding boxes or segments annotated is lower than 0.5.

Atomic-actions and Activities: Given the verbs of the atomic-action segments, two AMT workers were asked to fill in the blanks of the verb patterns and annotate the governing tasks in multi-task scenarios with a self-developed interactive annotation tool. We allow concurrent actions for each agent with multiple nouns for the same verb; for example, "get spoon, cup from table using hand." As there might exist ambiguities in describing the atomic-actions with natural languages, such as the possible annotations of "wash cup using water" *vs.* "wash cup using sink," we manually go through all the annotations and resolve the ambiguous action annotations following a uniform criterion.

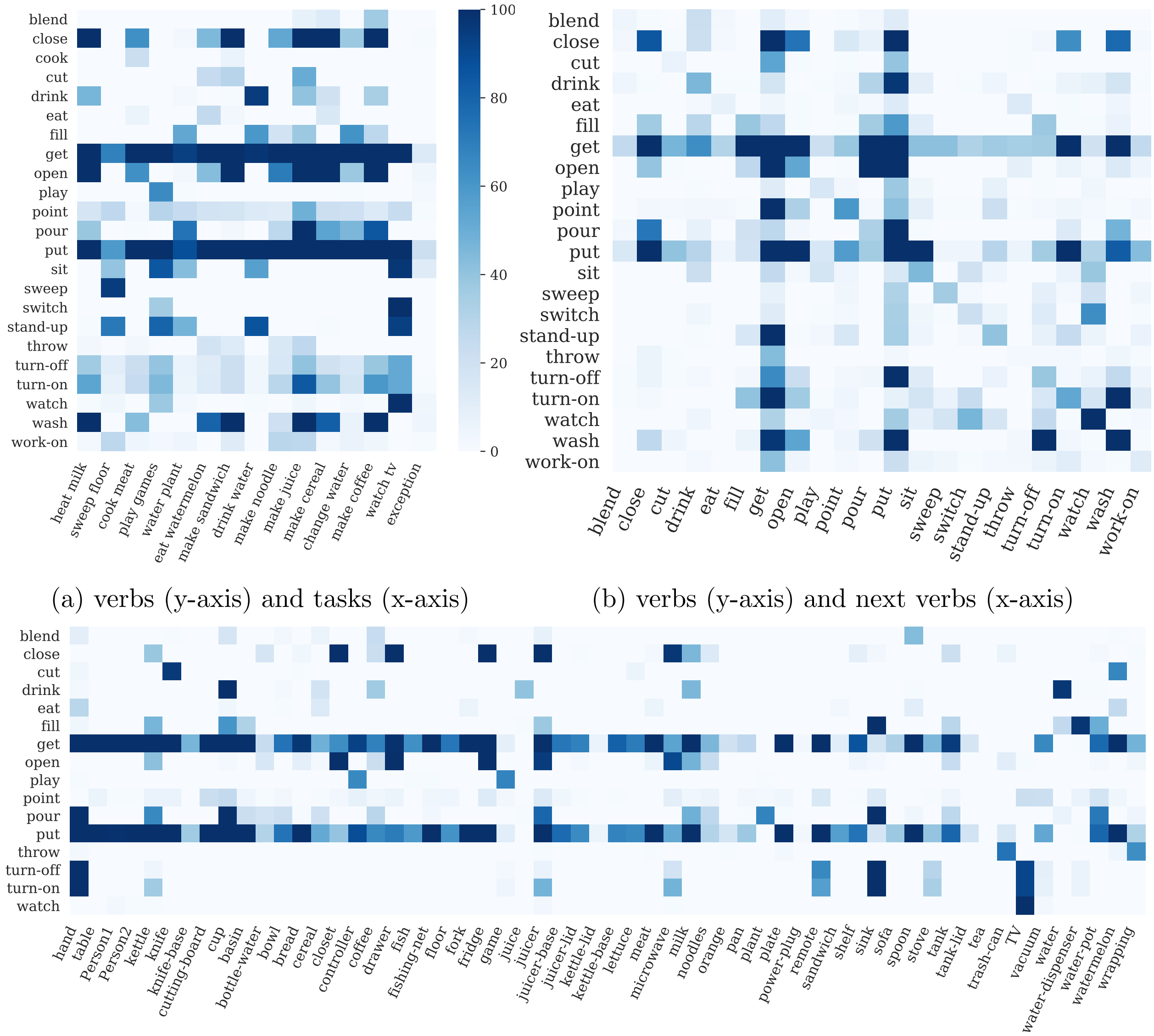
5.2.3.4 Dataset Statistics

In total, we recorded 324 activities, generating 324 × 2 TPV videos (from both Kinects) and 445 FPV videos. Among them, 136 activities were performed in kitchens and the remaining 188 in the living rooms. The collected LEMMA dataset consists of 127 1 × 1 activities, 76 1 × 2 activities, 66 2 × 1 activities, and 55 2 × 2 activities. The frequency of the recorded tasks is shown in Figure 5.8b. The total duration of all the activities is 10.1 hours, with an average duration of 2 minutes per video and the longest activity of 7 minutes.

We retrieved a total of 4.6 million images during post-processing, including 2.9 million RGB images captured by both GoPros and Kinects and 1.7 million depth images captured by Kinects. We annotated 0.9 million RGB frames captured by the primary view Kinect and gathered 0.8 million annotated frames with one or more actions performed by each of the agents (if multiple).

After resolving annotation ambiguities, we collected 24 verb classes and 64 noun classes, resulting in 862 compositional atomic-action labels, of which 641 appear more than 50 times. We show the frequencies of annotated verbs and nouns in Figures 5.8a and 5.8c; both distributions roughly follow the Zipf's law.

Co-occurrence relations among annotated verbs, nouns, and tasks are shown in Figure 5.9. As we can see from Figures 5.9a and 5.9c, verbs like "get" and "put" co-occur with various nouns in almost all of the tasks, which aligns with our intuition that moving objects around



(a) verbs (y-axis) and tasks (x-axis)

(b) verbs (y-axis) and next verbs (x-axis)

(c) verbs (y-axis) and nouns (x-axis)

Figure 5.9: The co-occurrence statistics for verbs, nouns, and tasks in LEMMA.

consists a large portion of our daily activities. Interactive actions between participants are captured by verbs (*e.g.*, "point-to") and nouns (*e.g.*, "P1," short for "participant 1") in the form of annotations like "get knife from P1 using hand" or "point-to sink."

5.2.4 Benchmarks

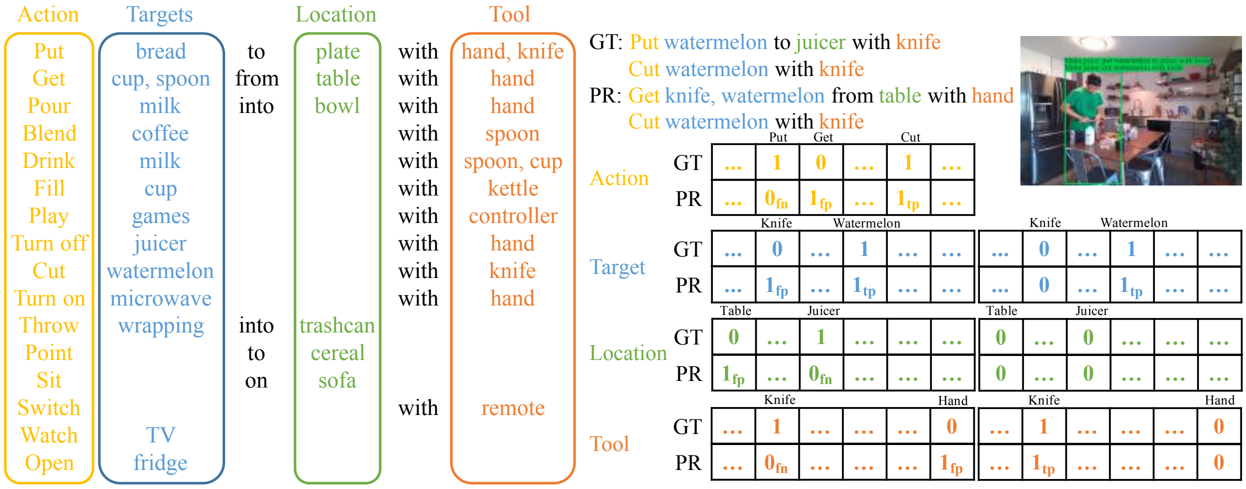
Aligned with our motivations, two general goals are constructed to evaluate indoor human activity understanding on the collected LEMMA dataset: (i) recognize atomic-actions and their semantics; and (ii) understand the goal-directed activities and monitor multiple concurrent tasks, especially in multi-agent scenarios. Specifically, we define two challenging benchmarks to test the capability of understanding complex goal-directed activities for computer vision algorithms.

5.2.4.1 Compositional Action Recognition

Human indoor activities are composed of fine-grained action segments with rich semantics. As mentioned by Goyal *et al.* [GKM17], interactions with objects are highly purposive. From the simplest verb of "put," we can generate a plethora of combinations of objects and target places, such as "put cup onto table," "put fork into drawer." Situations could become even more challenging when objects were used as tools; for example, "put meat into pan using fork."

Motivated by the above observation, we propose the compositional action recognition benchmark on the collected LEMMA dataset with each object attributed to a specific semantic position in the action label. Specifically, we build 24 compositional action templates; see Figure 5.10a for some examples. In these action templates, each noun could denote an interacting object, a target or a source location, or a tool used by a human agent to perform certain actions.

The proposed compositional action recognition benchmark is challenging; it requires computational models to correctly detect the ongoing concurrent action verbs as well as the nouns at their correct semantic positions. We evaluate model performances by metrics on compositional action recognition in both FPVs and TPVs. Specifically, the model is asked to predict (i) multiple labels in verb recognition for concurrent actions (*e.g.*, "watch tv" and "drink with cup" at the same time), and (ii) multiple labels in noun recognition for each semantic position given verbs, representing the interactions with multiple objects using the same action (*e.g.*, "wash spoon, cup using sink"). Figure 5.10b shows the schematics of the evaluation process. For training and testing on TPVs, we provide ground-truth bounding boxes of humans as additional information on spatial localization.



(a) Compositional action templates

(b) Prediction of verbs and nouns

Figure 5.10: Compositional action recognition benchmark on LEMMA. (a) Examples of Compositional action templates. Yellow denotes verbs. Blue, green, and brown denote nouns for an interacting object, target/source location, and tool, respectively. (b) Examples of predictions of the verbs and nouns in compositional action recognition. Verbs and nouns are evaluated through multi-label classification.

5.2.4.2 Action and Task Anticipation

As emphasized throughout the paper, the most significant factor of human activities is the goal-directed, teleological stand. An in-depth understanding of goal-directed tasks demands a predictive ability of latent goals, action preferences, and potential outcomes. To tackle these challenges, we propose the action and task anticipation benchmark on the collected LEMMA dataset. Specifically, we evaluate model performances for the anticipation (*i.e.*, predictions for the next action segment) of action and task with both FPV and TPV videos.

This benchmark provides both the training and testing data in all four scenarios of activities to study the goal-directed multi-task multi-agent problem. As there is an innate discrepancy of prediction difficulties among these four scenarios, we gradually increase the overall prediction difficulty, akin to a curriculum learning process, by setting the percentage of training videos to be 3/4, 1/4, 1/4, and 1/4 for 1×1, 1×2, 2×1 and 2×2 scenarios, respectively. Intuitively, with sufficient clean demonstrations of tasks in 1×1 scenario, interpreting tasks in more complex settings (*i.e.*, 1×2, 2×1, and 2×2) should be easier, thus requiring less learning samples; such a design encourages the model to generalize. The model performance is evaluated individually for each scenario.

5.2.5 Experiments

In this section, we conduct experiments on the two proposed benchmarks with details on evaluation metrics, experimental settings, and baseline results. We further discuss the results to highlight the underlying challenges of each task.

5.2.5.1 Compositional Action Recognition

Experimental Setup: We randomly split all the video samples into training and test sets with a ratio of 3:1, resulting in 243 recorded activities for training and the remaining 81 for testing. Due to the multi-agent setup, each activity may have multiple FPVs; 333 (out of 445) FPV videos are split into training. In TPVs, the recordings of the primary view with the ground-truth human bounding box annotations are given for both training and testing videos. Results are evaluated on two separate sources of inputs: FPVs and TPVs.

Evaluation Metrics: Model performances are evaluated separately for verbs, nouns, and compositional action recognition. Verb and compositional action recognition are treated as multi-label classifications with 25 verb classes and 863 compositional action classes (including a "null" action). After generating multi-hot labels for each semantic position in the presented verb, noun recognition is evaluated as multi-label classification (64 object classes). Average precision, recall, and F1-score for all predictions are reported on testing sets. During the evaluation, we sample image frames at 5 FPS and evaluate on these frames.

Methods: We adopt two recent 3D-CNN networks, I3D [CZ17] and SlowFast Network [FFM19], as the baseline models. The baseline models predict the compositional action directly. Considering compositionality of verbs and nouns, we propose two variants of the baseline models: (i) a multi-branch network (branching model) that builds on the bottleneck layer of the backbone models to leverage both verb and noun supervision, and (ii) a multi-step inference model (sequential model), wherein verbs are first inferred with a beam search and then fed into object inference with their verb embeddings for joint learning.

Implementation Details: The training procedure utilizes all annotated segments in the training set. Additionally, we re-scale all the images with the short side to 256 pixels. To feed data into 3D-CNN models, 4 frames are first sampled for each action segment as center frames, and an additional 8 frames are then uniformly sampled around center frames with a window length of 32. We train each model on 8 Titan RTX GPUs on a single computing node for 50 epochs (20k iterations) with a batch size of 96. We use warm-up strategy and perform large mini-batch batch normalization, as suggested in [GDG17]. The learning rate is initially set to 0.0125 for each parallel branch and decays with a cosine annealing. Other settings of the backbone models are the same as in [FFM19]. For the proposed sequential model, we use the beam search with a size of 5 for action inference. We extract bounding box features of humans with ROIAlign [HGD17] for frames in TPVs.

Results and Discussion: Table 5.6 shows quantitative results of predicting verbs, nouns, and compositional actions for the compositional action recognition task. For FPVs, rather than directly predicting the compositional actions (baseline models), predicting the verbs and nouns with their semantic positions boosts the performance on all metrics, indicating that understanding the compositional structures of human actions indeed supports the prediction. We also observe that the results of compositional action recognition in the sequential models are slightly lower than the branching model due to the aggregated error brought in by a relatively low precision ($\sim$25%) of the verb recognition.

In comparison, the results of compositional action recognition in TPVs are significantly lower than those in the FPVs due to severe occlusion. It also shows that predicting the composition of verbs and nouns makes no significant improvement compared with predicting compositional action directly. Such a result implies that current models could not capture the details of compositions between verbs and nouns from TPVs. Taken together, the results indicate that fusion among the representations of visual embodiment between TPVs and

Table 5.6: Comparisons of compositional action recognition on LEMMA.

View Type	Method	Verb			Noun			Compositional Action		
		Avg.Prec	Avg.Rec	Avg.F1	Avg.Prec	Avg.Rec	Avg.F1	Avg.Prec	Avg.Rec	Avg.F1
FPV	I3D	17.09	43.89	24.60	3.42	16.15	5.72	11.07	39.49	17.30
	Slowfast	22.27	56.42	31.94	4.31	20.60	7.13	18.68	**50.65**	27.3
	I3D sequential	25.04	**57.00**	34.80	**19.36**	**75.29**	**30.80**	18.00	50.04	26.47
	Slowfast sequential	24.30	49.71	32.64	17.95	59.11	27.54	26.80	38.41	31.57
	I3D branching	25.73	55.62	**35.8**	18.63	69.76	29.41	22.29	48.46	30.53
	Slowfast branching	**26.16**	56.33	35.73	18.18	73.46	29.15	**27.97**	48.87	**35.58**
TPV	I3D	14.18	36.34	20.40	2.29	11.05	3.79	6.85	**23.82**	10.64
	Slowfast	14.28	**37.38**	20.66	2.32	11.14	3.83	**7.76**	23.25	**16.31**
	I3D sequential	16.17	30.17	21.05	7.79	**25.41**	11.93	2.23	12.67	3.79
	Slowfast sequential	15.31	28.84	20.00	6.37	22.39	9.92	3.27	9.16	4.82
	I3D branching	12.92	32.09	18.43	12.75	17.70	14.82	4.67	20.76	7.6
	Slowfast branching	**16.64**	33.40	**22.21**	**17.29**	18.36	**17.81**	6.52	21.55	10.01

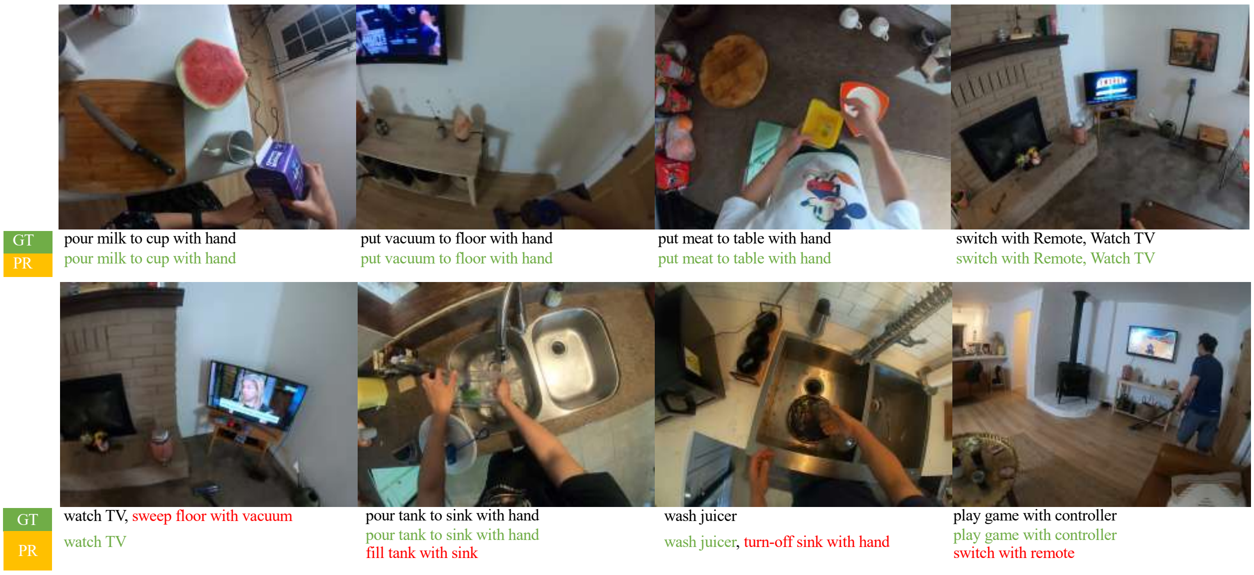

GT
PR
pour milk to cup with hand
pour milk to cup with hand
put vacuum to floor with hand
put vacuum to floor with hand
put meat to table with hand
put meat to table with hand
switch with Remote, Watch TV
switch with Remote, Watch TV
GT
PR
watch TV, sweep floor with vacuum
watch TV
pour tank to sink with hand
pour tank to sink with hand
fill tank with sink
wash juicer
wash juicer, turn-off sink with hand
play game with controller
play game with controller
switch with remote

Figure 5.11: Qualitative results of compositional action recognition on LEMMA. From top to bottom, we show correct predictions and failure examples. Red marks wrong verb or noun predictions, green indicates correct verb or noun predictions.

FPVs might be a crucial ingredient to tackle this problem in the future.

Figure 5.11 shows qualitative results for the composed action recognition task.

5.2.5.2 Action and Task Anticipations

Experimental Setup: We split the training and test sets with ratios 3 : 1, 1 : 3, 1 : 3, 1 : 3 for the four scenarios 1 × 1, 1 × 2, 2 × 1, 2 × 2, respectively. Such a spit results in training set with (96, 19, 16, 13) activities and a test set with (31, 57, 50, 42) activities in four scenarios. During training and testing, the computational models have access to both FPVs and TPVs, together with the ground-truth human bounding boxes annotations of the TPV primary view.

Evaluation Metrics: Model performances are evaluated individually (per agent) for the action and task anticipations task. Specifically, both action and task anticipations are evaluated as multi-label classifications with 863 compositional action classes (including a "null" action) and 15 task classes. Average precision, recall, and F1-score are reported individually for each of the four scenarios on the testing sets. Similar to the protocol used in the above compositional action recognition task, we re-sample image frames at 5 FPS and evaluate these sub-sampled frames during the testing phase.

Methods: We leverage the visual features extracted by the pre-trained SlowFast model in compositional action recognition for baseline models. Specifically, we compare two backbone models: (i) using segment-level recognition feature (SF) directly by adding an MLP on top of the features, and (ii) using long-term feature bank (LFB) with max pooling [WFF19]. For activities with multi-agent interactions, we use the other agent's FPV features together with their own's to capture the joint task execution progress for learning and inference; these variants are denoted as M-SF (FPV) and M-LFB (FPV) For comparison, we also use the concatenation of the FPV feature and primary TPV feature as the input; the corresponding models are denoted as M-SF (TPV) and M-LFB (TPV).

Implementation Details: For the LFB model, we use a history window size of 10 and aggregate the features using max-pooling, as described in [WFF19]. For the multi-agent variants, we use max-pooling to fuse features of two views and process them with a different branch as another temporal inference module. We train models on a single Titan Xp GPU for 50 epochs with a learning rate of 0.001.

Results and Discussion: Table 5.7 shows quantitative results of action and task anticipation. The proposed multi-agent variants (M-) of baseline models perform the best among

Table 5.7: Comparisons of the action and task anticipations on LEMMA.

Scenario	Method	1 × 1			1 × 2			2 × 1			2 × 2		
		Avg.Prec	Avg.Rec	Avg.F1	Avg.Prec	Avg.Rec	Avg.F1	Avg.Prec	Avg.Rec	Avg.F1	Avg.Prec	Avg.Rec	Avg.F1
Compositional action	SF	23.42	22.25	22.82	20.13	20.06	20.10	18.89	19.22	19.05	18.31	16.67	17.45
	LFB	23.03	28.67	25.54	20.48	25.4	22.67	18.31	22.30	20.11	18.53	20.97	19.68
	M-SF (TPV)	**24.22**	28.05	25.99	20.10	24.48	22.08	19.15	16.71	17.85	19.64	15.18	17.12
	M-LFB (TPV)	23.54	**37.81**	**29.01**	21.10	**31.86**	25.39	19.67	21.03	20.33	**20.11**	20.30	**20.15**
	M-SF (FPV)	23.30	25.41	24.31	**21.34**	23.18	22.22	**19.70**	17.46	18.51	19.82	15.8	17.58
	M-LFB (FPV)	23.26	31.07	26.60	20.78	27.40	23.63	19.42	**21.73**	**20.51**	19.49	20.12	19.8
Task	SF	50.53	79.08	61.66	48.07	67.78	56.25	39.05	57.43	46.49	44.88	62.09	52.1
	LFB	57.57	**84.31**	68.42	52.12	68.94	59.36	38.40	53.08	44.56	48.17	64.61	55.19
	M-SF (TPV)	58.61	79.96	67.05	55.45	67.24	60.78	**45.73**	58.98	**51.51**	**49.66**	64.47	56.10
	M-LFB (TPV)	**60.27**	82.19	**69.54**	**56.2**	**72.46**	**63.30**	43.94	61.41	51.23	48.85	**67.48**	**56.67**
	M-SF (FPV)	51.12	79.18	62.13	48.42	69.04	56.92	41.00	58.11	48.08	46.04	65.97	54.24
	M-LFB (FPV)	55.56	82.83	66.51	52.22	70.01	59.82	41.33	**64.49**	50.38	46.65	69.59	55.86

all models. For single-agent activities (1 × 1, 1 × 2), we have the following crucial observations. First, models that consider temporal relations between frames generally perform better than the models using segment features. Second, adding additional TPV features to single-agent activities slightly helps interpret the task being executed and therefore promotes anticipation. This result matches the intuition that computational models having access to both FPVs and TPVs would perceive more holistic scene information. We also find that the performances of task anticipation in the 1 × 1 single-task scenario are better than the one in the 1 × 2 multi-task scenario, matching what we would expect from more complicated task execution patterns.

For multi-agent activities (2 × 1, 2 × 2), we observe that the aggregation of FPV and TPV features generally performs better. It supports our hypothesis that observing the other agents' actions helps the computational models to "understand" task scheduling and assignment. We also observe that, models' performances in 2 × 1 activities are slightly worse than in 2 × 2 activities. We hypothesize that task plans in the 2 × 2 scenarios change less frequently, with a clear task assignment coordinates the individual tasks. In comparison, in the 2 × 1 scenarios, the sequential ordering of the task requires more frequent communications between agents to coordinate. Such a performance gap calls for better modeling of multi-agent task assignments.

5.2.6 Conclusions

In this paper, we introduce the LEMMA dataset with a focus on natural multi-agent multi-task daily activities. Dense annotations are provided on both compositional action and task for learning and inference on four different activity scenarios with increasing difficulty. Additionally, we propose two challenging tasks on LEMMA to measure existing models' competence in action understanding and temporal reasoning: (i) compositional action recognition, and (ii) action/task anticipations. We hope this effort would attract the computer vision community to look into natural and realistic goal-directed human activities and further study the task scheduling and assignment in real-world scenarios.

Part III

Learning and Reasoning: Human-like Representation and Concept Learning

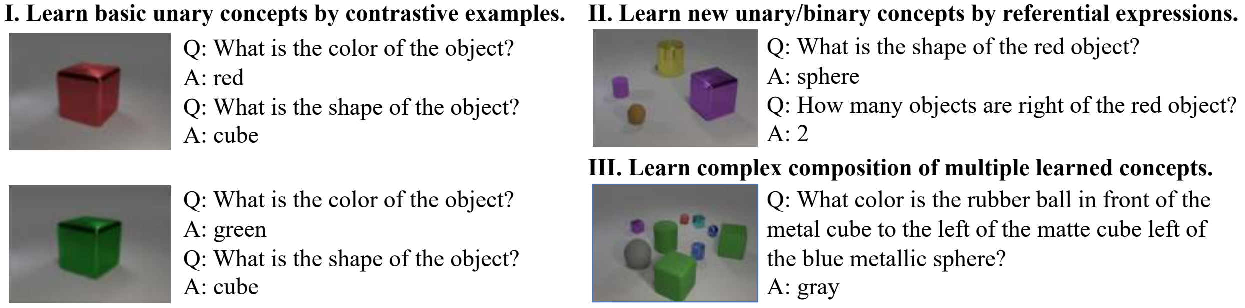
CHAPTER 6

Neural-Symbolic Learning and Reasoning

The goal of neural-symbolic computation is to integrate the connectionist and symbolist paradigms. It naturally bridges the perception and reasoning systems with flexibility between implicit and explicit computation, boosting performance and interpretability on many visual reasoning tasks. In this chapter, we introduce two neural-symbolic approaches that can efficiently learn and reason [LHH20b, LHH20a]. Specifically, Section 6.1 mimics humans' learning curriculum in concept learning and Section 6.2 closes the loop of neural-symbolic learning by integrating a grammar parsing module. .

6.1 Visual Concept Learning with Competence-aware Curriculum

In this section, we propose a competence-aware curriculum for visual concept learning in a question-answering manner, to mimic humans' ability in progressively learning visual concepts from easy to hard questions. Specifically, we design a neural-symbolic concept learner for learning the visual concepts and a multi-dimensional Item Response Theory (mIRT) model for guiding the learning process with an adaptive curriculum. The mIRT effectively estimates the concept difficulty and the model competence at each learning step from accumulated model responses. The estimated concept difficulty and model competence are further utilized to select the most profitable training samples. Experimental results on CLEVR show that with a competence-aware curriculum, the proposed method achieves state-of-the-art performances with superior data efficiency and convergence speed. Specifically, the proposed model only uses **40% of training data** and converges **three times faster** compared with other state-of-the-art methods.



I. Learn basic unary concepts by contrastive examples.

Q: What is the color of the object?
A: red
Q: What is the shape of the object?
A: cube

Q: What is the color of the object?
A: green
Q: What is the shape of the object?
A: cube

II. Learn new unary/binary concepts by referential expressions.

Q: What is the shape of the red object?
A: sphere
Q: How many objects are right of the red object?
A: 2

III. Learn complex composition of multiple learned concepts.

Q: What color is the rubber ball in front of the metal cube to the left of the matte cube left of the blue metallic sphere?
A: gray

Figure 6.1: The incremental learning of visual concepts in a question-answering manner. Three difficulty levels can be categorized into I) unary concepts from simple questions, II) binary (relational) concepts based on the learned concepts, and III) compositions of visual concepts from comprehensive questions.

6.1.1 Introduction

Humans excel at learning visual concepts and their compositions in a question-answering manner [FAS10, CKA15, GLT18, ZCN17, ZRH20], which requires a joint understanding of vision and language. The essence of such learning skill is the superior capability to connect linguistic symbols (words/phrases) in question-answer pairs with visual cues (appearance/geometry) in images. Imagine a person without prior knowledge of colors is presented with two contrastive examples in Figure 6.1-I. The left images are the same except for color, and the right question-answer pairs differ only in the descriptions about color. By assuming that the differences in the question-answer pairs capture the differences in appearances, he can learn the concept of color and the appearance of specific colors (*i.e.*, red and green). Besides learning the basic unary concepts from contrastive examples, compositional relations from complex questions consisting of multiple concepts can be further learned, as shown in Figure 6.1-II and -III.

Another crucial characteristic of the human learning process is to start *small* and learn *incrementally*. More specifically, the human learning process is well-organized with a curriculum that introduces concepts progressively and facilitates the learning of new abstract knowledge by exploiting learned concepts. A good curriculum serves as an experienced teacher. By ranking and selecting examples according to the learning state, it can guide the training process of the learner (student) and significantly increase the learning speed. This idea is originally examined in animal training as *shaping* [Ski58, Pet04, KD09] and then applied to machine learning as *curriculum learning* [Elm93, BLC09, GBM17, GHZ18, PSL14].

Inspired by the efficient curriculum, Mao *et al.* [MGK19] proposes a neural-symbolic approach to learn visual concepts with a *fixed* curriculum. Their approach learns from image-question-answer triplets and does not require annotation on images or programs generated from questions. The model is trained with a manually-designed curriculum that includes four stages: (1) learning unary visual concepts; (2) learning relational concepts; (3) learning more complex questions with visual perception fixed; (4) joint fine-tuning all modules. They select questions for each stage by the depths of the latent programs. Their curriculum heavily relies on the manually-designed heuristic that measures the question difficulty and discretizes the curriculum. Such heuristic suffers from three limitations. First, it ignores the variance of difficulties for questions with the same program depths, where different concepts might have various difficulties. Second, the manually-designed curriculum relies on strong human prior knowledge for the difficulties, while such prior may conflict with the inherent difficulty distribution of the training examples. Last but most importantly, it neglects the progress of the learner that evolves along with the training process. More specifically, the order of training samples in the curriculum is nonadjustable based on the model state. This scheme is in stark contrast to the way that humans learn – by *actively* selecting learning samples based on our current learning state, instead of *passively* accepting specific training samples. A desirable learning system should be capable of automatically adjusting the curriculum during the learning process without requiring any prior knowledge, which makes the learning procedure more efficient with less data redundancy and faster convergence speed.

To address these issues and mimic human ability in adaptive learning, we propose a **competence-aware** curriculum for visual concept learning via question answering, where competence represents the capability of the model to recognize each concept. The proposed approach utilizes multi-dimensional Item Response Theory (mIRT) to estimate the **concept difficulty** and **model competence** at each learning step from accumulated model responses. Item Response Theory (IRT) [Bak01, BK04] is a widely adopted method in psychometrics that estimates the human ability and the item difficulty from human responses on various items. We extend the IRT to a mIRT that matches the compositional nature of visual reasoning, and apply variational inference to get a Bayesian estimation for the parameters in mIRT. Based on the estimations of concept difficulty and model competence, we further define a continuous adaptive curriculum (instead of a discretized fixed regime) that selects the most profitable training samples according to the current learning state. More specifically, the learner can filter out samples with either too naive or too challenging questions. These questions bring either negligible or sharp gradients to the learner, which makes it slower and harder to converge.

With the proposed competence-aware curriculum, the learner can address the aforementioned limitations brought by a fixed curriculum with the following advantages:

1. The concept difficulty and the model competence at each learning step can be inferred effectively from accumulated model responses. It enables the model to distinguish difficulties among various concepts and be aware of its own capability for recognizing these concepts.

2. The question difficulty can be calculated with the estimated concept difficulty and model competence without requiring any heuristics.

3. The adaptive curriculum significantly contributes to the improvement of learning efficiency by relieving the data redundancy and accelerating the convergence, as well as the improvement of the final performance.

We explore the proposed method on the CLEVR dataset [JHM17a], an artificial universe where visual concepts are clearly defined and less correlated. We opt for this synthetic environment because there is little prior work on curriculum learning for visual concepts and there lacks a clear definition of visual concepts in real-world setting. CLEVR allows us to perform controllable diagnoses of the proposed mIRT model in building an adaptive curriculum. Section 6.1.5 further discusses the potentials and challenges of generalizing our method to other domains such as real-world images and natural language processing.

Experimental results show that the visual concept learner with the proposed competence-aware curriculum converges three times faster and consumes only 40% of the training data while achieving similar or even higher accuracy compared with other state-of-the-art models. We also evaluate individual modules in the proposed method and demonstrate their efficacy in Section 6.1.4.

6.1.2 Related Work

6.1.2.1 Neural-symbolic Visual Question Answering

Visual question answering (VQA) [MF14, TML14, QWL15, JHM17a, GLL17] is a popular task for gauging the capability of visual reasoning systems. Some recent studies [ARD15, ARD16, HAR17, JHM17c, YGL20] focus on learning the neural module networks (NMNs) on the CLEVR dataset. NMNs translate questions into programs, which are further executed over image features to predict answers. The program generator is typically trained on human annotations. Several recent works target on reducing the supervision or increasing the generalization ability to new tasks in NMNs. For example, Johnson *et al.* [JHM17c] replaces the hand-designed syntactic parsers by a learned program generator. Neural-Symbolic VQA [YWG18] explores an object-based visual representation and uses a symbolic executor for inferring the answer. Neural-symbolic concept learner [MGK19] uses a symbolic reasoning process and manually-defined curriculum to bridge the learning of visual concepts, words, and the parsing of questions without explicit annotations. In this work, we build our model on the neural-symbolic concept learner [MGK19] and learn an adaptive curriculum to select the most profitable training samples.

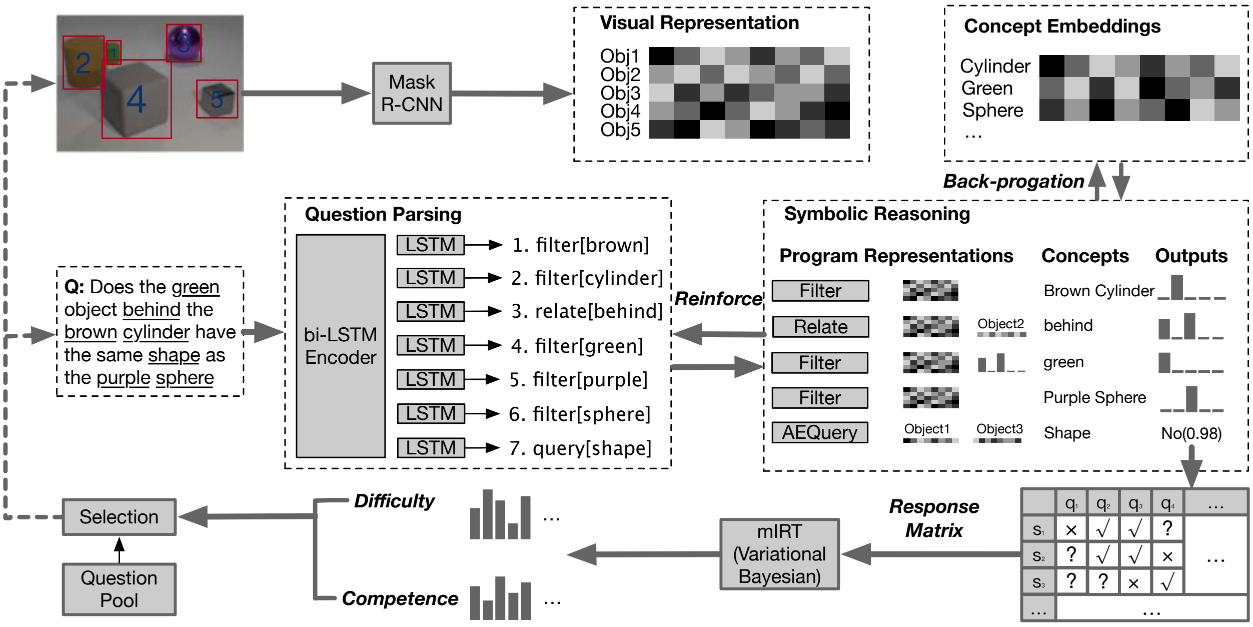
Learning-by-asking (LBA) [MGF17] proposes an interactive learning framework that allows the model to actively query an oracle and discover an easy-to-hard curriculum. LBA uses the expected accuracy improvement over candidate answers as an informativeness measure to pick questions. However, it is costly to compute the expected accuracy improvement for sampled questions since it requires to process all the questions and images through a VQA model. Moreover, the expected accuracy improvement cannot help to learn which specific component of the question contributes to the performance, especially while learning from the answers with little information such as "yes/no". In contrast, we select questions by explicitly modeling the difficulty of visual concepts, combined with model competence to infer the difficulty of each question.

6.1.2.2 Curriculum Learning and Machine Teaching

The competence-aware curriculum in our work is related to *curriculum learning* [BLC09, SAJ10, TFL16, GBM17, Sac16, PSL14, GHZ18, PSN19] and *machine teaching* [Zhu15, ZSZ18, LDH17, DHP19, MCV19, Fan18, Wu18]. *Curriculum learning* is firstly proposed by Bengio *et al.* [BLC09] and demonstrates that a dataset order from easy instances to hard ones benefits learning process. The measures of hardness in curriculum learning approaches are usually determined by hand-designed heuristics [SAJ10, TFL16, Sac16, MGK19]. Graves *et al.* [GBM17] explore learning signals based on the increase rates in prediction accuracy and network complexity to adjust data distributions along with training. Self-paced learning [Kum10, Jia14, Jia15, Sac16] quantifies the sample hardness by the training loss and formulates curriculum learning as an optimization problem by jointly modeling the sample selection and the learning objective. These hand-designed heuristics are usually task-specific without any generalization ability to other domains.

Machine teaching [Zhu15, ZSZ18, LDH17] introduces a teacher model that receives feedback from the student model and guides the learning of the student model accordingly. Zhu *et al.* [Zhu15, ZSZ18] assume that the teacher knows the ground-truth model (*i.e.*, the Oracle) beforehand and constructs a minimal training set for the student model. The recent works *learning to teach* [Fan18, Wu18] break this strong assumption of the existence of the oracle model and endow the teacher with the capability of learning to teach via a reinforcement learning framework.

Our work explores curriculum learning in visual reasoning, which is highly compositional and more complex than tasks studied before. Different from previous works, our method requires neither hand-designed heuristics nor an extra teacher model. We combine the idea



	q_1	q_2	q_3	q_4	…
s_1	×	√	√	?	…
s_2	?	√	√	×	
s_3	?	?	×	√	
…	…				

Figure 6.2: The overview of the proposed approach. We use neural symbolic reasoning as a bridge to jointly learn concept embeddings and question parsing. The model responses in the training process are accumulated to estimate concept difficulty and model competence at each learning step with mIRT. The estimations help to select appropriate training samples for the current model. In the response matrix,'✓' or '✕' denotes that the snapshot predicts a correct or wrong answer, and '?' means the snapshot has no response to this question.

of *competence* with curriculum learning and propose a novel mIRT model that estimates the concept difficulty and model competence from accumulated model responses.

6.1.3 Methodology

In this section, we will discuss the proposed competence-aware curriculum for visual concept learning, as also shown in Figure 6.2. We first describe a neural-symbolic approach to learn visual concepts from image-question-answer triplets. Next, we introduce the background of IRT model and discuss how we derive a mIRT model for estimating concept difficulty and model competence. Finally, we present how to select training samples based on the estimated concept difficulty and model competence to make the training process more efficient.

6.1.3.1 Neural-Symbolic Concept Learner

We briefly describe the neural-symbolic concept learner. It uses a symbolic reasoning process to bridge the learning of visual concepts and the semantic parsing of textual questions without any intermediate annotations except for the final answers. We refer readers to [MGK19, YWG18] for more details on this model.

Scene Parsing. A scene parsing module develops an object-based representation for each image. Concretely, we adopt a pre-trained Mask R-CNN [HGD17] to generate object proposals from the image. The detected bounding boxes with the original image are sent to a ResNet-34 [HZR16] to extract the object-based features.

Concept Embeddings. By assuming each visual attribute (*e.g.*, shape) contains a set of visual concepts (*e.g.*, cylinder), the extracted visual features are embedded into concept spaces by learnable neural operators of the attributes.

Question Parsing. The question parsing module translates a question in natural language into an executable program in a domain-specific language designed for VQA. The question parser generates the latent program from a question in a sequence-to-sequence manner. A bi-directional LSTM is used to encode the input question into a fixed-length representation. The decoder is an attention-based LSTM, which produces the operations in the program step-by-step. Some operations take concepts as their parameters, such as *Filter[Cube]* and *Relate[Left]*. These concepts are selected from the concepts appearing in the question by the attention mechanism.

Symbolic Reasoning. Given the latent program, the symbolic executor runs the operations in the program with the object-based image representation to derive an answer for the input question. The execution is fully differentiable with respect to the concept embeddings since the intermediate results are represented in a probabilistic manner. Specifically, we keep an attention mask on all object proposals, with each element in the mask denoting the probability that the corresponding object contains certain concepts. The attention mask is fed into the next operation, and the execution continues. The final operation predicts an answer to the question.

Joint Optimizing. We formulate the problem of jointly learning the question parser and the concept embeddings without the annotated programs. Suppose we have a training sample consisting of image I, question Q, and answer A, and we do not observe the latent program l. The goal of training the whole system is to maximize the following conditional probability:

$$p(A|I,Q) = \mathbb{E}_{l\sim p(l|Q)}\ [p(A|l,I)], \tag{6.1}$$

where $p(l|Q)$ is parametrized by the question parser with the parameters θ_l and $p(A|l,I)$ is parametrized by the concept embeddings θ_e (there are no learnable parameters in the symbolic reasoning module). Considering the expectation over the program space in Equation (6.1) is intractable, we approximate the expectation with Monte Carlo sampling. Specifically, we first sample a program $\hat{l}$ from the question parser $p(l|Q;\theta_l)$ and then apply $\hat{l}$ to obtain a probability distribution over possible answers $p(A|\hat{l},I;\theta_e)$.

Recalling the program execution is fully differentiable w.r.t. the concept embeddings, we learn the concept embeddings by directly maximizing $\log p(A|\hat{l},I;\theta_e)$ using gradient descent and the gradient $\nabla_{\theta_e}\log p(A|\hat{l},I;\theta_e)$ can be calculated through back-propagation. Since the hard selection of $\hat{l}$ through Monte Carlo sampling is non-differentiable, the gradients of the question parser cannot be computed by back-propagation. Instead we optimize the question parser using the REINFORCE algorithm [Wil92]. The gradient of the reward function J over the parameters of the policy is:

$$\nabla J(\theta_l) = \mathbb{E}_{l\sim p(l|Q;\theta_l)}\left[\nabla \log p\left(l|Q;\theta_l\right)\cdot r\right], \tag{6.2}$$

where r denotes the reward. Defining the reward as the log-probability of the correct answer and again, we rewrite the intractable expectation with one Monte Carlo sample $\hat{l}$:

$$\nabla J(\theta_l) = \nabla \log p\left(\hat{l}|Q;\theta_l\right)\cdot[\log p(A|\hat{l},I;\theta_e) - b], \tag{6.3}$$

where b is the exponential moving average of $\log p(A|\hat{l},I;\theta_e)$, serving as a simple baseline to reduce the variance of gradients. Therefore, the update to the question parser at each learning step is simply the gradient of the log-probability of choosing the program, multiplied by the probability of the correct answer using that program.

6.1.3.2 Background of Item Response Theory (IRT)

Item response theory (IRT) [Bak01, BK04] was initially created in the fields of educational measurement and psychometrics. It has been widely used to measure the latent abilities of subjects (*e.g.*, human beings, robots or AI models) based on their responses to items (*e.g.*, test questions) with different levels of difficulty. The core idea of IRT is that the probability of a correct response to an item can be modeled by a mathematical function of both individual ability and item characteristics. More formally, if we let i be an individual and j be an item, then the probability that the individual i answers the item j correctly can be modeled by a logistic model as:

$$p_{ij} = c_j + \frac{1 - c_j}{1 + e^{-a_j(\theta_i - b_j)}}, \tag{6.4}$$

where θ_i is the latent ability of the individual i and a_j, b_j, c_j are the characteristics of the item j. The item parameters can be interpreted as changing the shape of the standard logistic function: a_j (the discrimination parameter) controls the slope of the curve; b_j (the difficulty parameter) is the ability level, it is the point on θ_i where the probability of a correct response is the average of c_j (min) and 1 (max), also where the slope is maximized; c_j (the guessing parameter) is the asymptotic minimum of this function, which accounts for the effects of guessing on the probability of a correct response for a multi-choice item. Equation (6.4) is often referred to as the three-parameter logistic (3PL) model since it has three parameters describing the characteristics of items. We refer the readers to [Bak01, BK04, ER13] for more background and details on IRT.

6.1.3.3 Multi-dimensional IRT using Model Responses

Traditional IRT is proposed to model the human responses to several hundred items. However, datasets used in machine learning, especially deep neural networks, often consist of hundreds of thousands of samples or even more. It is costly to collect human responses for large datasets, and more importantly, human responses are not distinguishable enough to estimate the sample difficulties since samples in machine learning datasets are usually straightforward for humans. Lalor *et al.* [LWY16, LWY19] empirically shows on two NLP tasks that IRT models can be fit using machine responses by comparing item parameters learned from the human responses and the responses from an artificial crowd of thousands of machine learning models.

Similarly, we propose to fit IRT models with accumulated model responses (*i.e.*, the predictions of model snapshots) from the training process. Considering the compositional nature of visual reasoning, we propose a multi-dimensional IRT (mIRT) model to estimate the concept difficulty and model competence (corresponding to the subject ability in original IRT), from which the question difficulty can be further calculated.

Formally, we have C concepts, M model snapshots saved from all time steps, and N questions. Let $\Theta = \{\theta_{ic}\}_{i=1..M}^{c=1...C}$, where θ_{ic} is the i-th snapshot's competence on the c-th concept, and $B = \{b_c\}^{c=1...C}$, where b_c is the difficulty of the c-th concept, $\mathcal{Q} = \{q_{jc}\}_{j=1...N}^{c=1...C}$, where q_{jc} is the number of the c-th concept in the j-th question and g_j is the probability of guessing the correct answer to the j-th question, $\mathcal{Z} = \{z_{ij}\}_{i=1...M}^{j=1...N}$, where $z_{ij} \in \{0, 1\}$ be the response of the i-th snapshot to the j-th question (1 if the model answers the question correctly and 0 otherwise). The probability that the snapshot i can correctly recognize the concept c is formulated by a logistic function:

$$p_{ic}(\theta_{ic}, b_c) = \frac{1}{1 + e^{-(\theta_{ic} - b_c)}}. \tag{6.5}$$

Then the probability that the snapshot i answers the question j correctly is calculated as:

$$p(z_{ij} = 1|\theta_i, B) = g_j + (1 - g_j) \prod_{c=1}^{C} p_{ic}^{q_{jc}}. \tag{6.6}$$

The probability that the snapshot i answers the question j incorrectly is:

$$p(z_{ij} = 0|\theta_i, B) = 1 - p(z_{ij} = 1|\theta_i, B). \tag{6.7}$$

The total data likelihood is:

$$p(\mathcal{Z}|\Theta, B) = \prod_{i=1}^{M} \prod_{j=1}^{N} p(z_{ij}|\theta_i, B). \tag{6.8}$$

This formulation is also referred to as conjunctive multi-dimensional IRT [Rec85, Rec09].

6.1.3.4 Variational Bayesian Inference for mIRT

The goal of fitting an IRT model on observed responses is to estimate the latent subject abilities and item parameters. In traditional IRT, the item parameters are usually estimated by Marginal Maximum Likelihood (MML) via an Expectation-Maximization (EM) algorithm [BA81], where the subject ability parameters are randomly sampled from a normal distribution and marginalized out. Once the item parameters are estimated, the subject abilities are scored by maximum a posterior (MAP) estimation based on their responses to items. However, the EM algorithm is not computational efficient on large datasets. One feasible way for scaling up is to perform variational Bayesian inference on IRT [NNM16, LWY19]. The posterior probability of the parameters in mIRT can be written as:

$$p(\Theta, B|\mathcal{Z}) = \frac{p(\mathcal{Z}|\Theta, B)p(\Theta)p(B)}{\int_{\Theta,B} p(\Theta, B, \mathcal{Z})}, \tag{6.9}$$

where $p(\Theta), p(B)$ are the priors distribution of Θ and B. The integral over the parameter space in Equation (6.9) is intractable. Therefore, we approximate it by a factorized variational distribution on top of an independence assumption of Θ and B:

$$q(\Theta, B) = \prod_{i=1,c=1}^{M,C} \pi_{ic}^{\theta}\left(\theta_{ic}\right) \prod_{c=1}^{C} \pi_{c}^{b}\left(b_{c}\right), \tag{6.10}$$

where π^{θ}_{ic} and π^{b}_{c} denote Gaussian distributions for model competences and concept difficulties, respectively. We adopt the Kullback-Leibler divergence (KL-divergence) to measure the distance of p from q, which is defined as:

$$D_{\mathrm{KL}}(q \| p) := \mathbb{E}_{q(\Theta,B)} \log \frac{q(\Theta, B)}{p(\Theta, B|\mathcal{Z})}, \tag{6.11}$$

where $p(\Theta, B|\mathcal{Z})$ is still intractable. We can further decompose the KL-divergence as:

$$D_{\mathrm{KL}}(q \| p) = \mathbb{E}_{q(\Theta,B)} \left[\log \frac{q(\Theta, B)}{p(\Theta, B, \mathcal{Z})} + \log p(\mathcal{Z}) \right]. \tag{6.12}$$

In other words, we also have:

$$\begin{aligned} \log p(\mathcal{Z}) &= D_{\mathrm{KL}}(q \| p) - \mathbb{E}_{q(\Theta,B)} \log \frac{q(\Theta, B)}{p(\Theta, B, \mathcal{Z})} & (6.13) \\ &= D_{\mathrm{KL}}(q \| p) + \mathcal{L}(q). & (6.14) \end{aligned}$$

As the log evidence $\log p(\mathcal{Z})$ is fixed with respect to q, maximizing the final term $\mathcal{L}(q)$ minimizes the KL divergence of q from p. And since $q(\Theta, B)$ is a parametric distribution we can sample from, we can use Monte Carlo sampling to estimate this quantity. Since the KL-divergence is non-negative, $\mathcal{L}(q)$ is an evidence lower bound (ELBO) of $\log p(\mathcal{Z})$. By maximizing the ELBO with an Adam optimizer [KB14] in Pyro [BCJ18], we can estimate the parameters in mIRT.

6.1.3.5 Training Samples Selection Strategy

The proposed model can estimate the question difficulty for the current model competence without looking at the ground-truth images and answers. It facilitates the active selection for future training samples. More specifically, we can easily calculate the probability that the model answers a given question correctly from Equation (6.5) and Equation (6.6) (without guessing) using estimated Θ and b. This probability serves as an indicator of the question difficulty for the learner in each stage. The higher the probability, the easier the question. To

Algorithm 3 Competence-aware Curriculum Learning.

```
**Initialization**: the training set $\mathcal{D} = \{(I_j, Q_j, A_j)\}_{j=1}^{N}$, concept difficulty $B^{(0)}$, model competence $\Theta^{(0)}$, concept learner $\phi^{(0)}$, accumulated responses $\mathcal{Z} = \{\}$
**for** $t = 1$ to $T$ **do**
    $\Theta^{(t)}, B^{(t)} = \arg\max_{\Theta,B} \mathcal{L}(q; \Theta^{(t-1)}, B^{(t-1)}, \mathcal{Z})$
    $\mathcal{D}^{(t)} = \{(I, Q, A) : \mathrm{LB} \leqslant p(Q; \Theta^{(t)}, B^{(t)}) \leqslant \mathrm{UB}\}$
    $\phi^{(t)}, \mathcal{Z}^{(t)} = \mathrm{Train}(\phi^{(t-1)}, \mathcal{D}^{(t)})$
    $\mathcal{Z} = \mathcal{Z} \cup \mathcal{Z}^{(t)}$
```

select appropriate training samples, we rank the questions and filter out the hardest questions by setting a probability lower bound (LB) and the easiest questions by a probability upper bound (UB). Algorithm 3 summarizes the overall training process. We will discuss the influence of LB and UB on the learning process in Section 6.1.4.5.

6.1.4 Experiments

6.1.4.1 Experimental Setup

Dataset. We evaluate the proposed method on the CLEVR dataset [JHM17a], which consists of a training set of 70k images and ∼700k questions, and a validation set of 15k images and ∼150k questions. The proposed model selects questions from the training set during learning, and we evaluate our model on the entire validation set.

Models. To analyze the performance of the proposed approach, We conduct experiments by comparing with several model variants:

- **FiLM-LBA**: the best model from [MGF17].
- **NSCL**: the neural-symbolic concept learner [MGK19] without using any curriculum. Questions are randomly sampled from the training set.
- **NSCL-Fixed**: NSCL following a manually-designed discretized curriculum.
- **NSCL-mIRT**: NSCL following a continuous curriculum built by the proposed mIRT estimator.

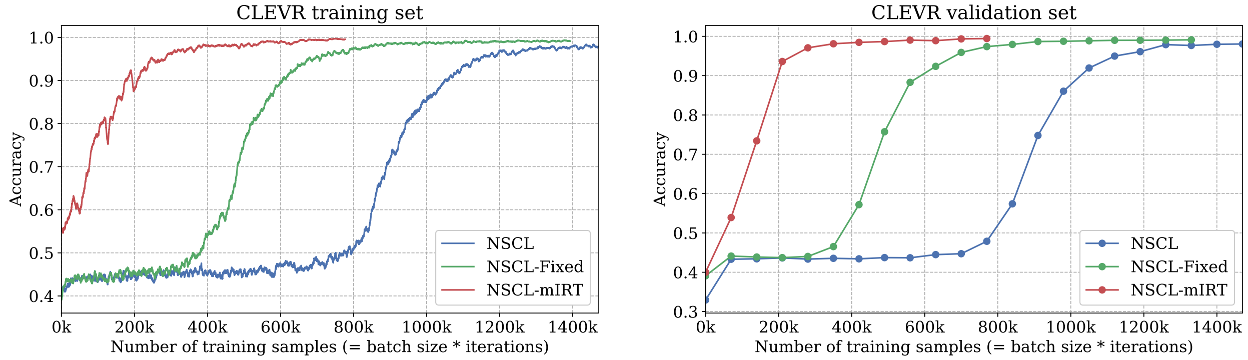


Figure 6.3: The learning curves of different model variants on the CLEVR dataset.

6.1.4.2 Training Process & Model Performance

Figure 6.3 shows the accuracies of the model variants at different timesteps on the training set (left) and validation set (right). Notably, the proposed NSCL-mIRT converges almost 2 times faster than NSCL-Fixed and 3 times faster than NSCL (*i.e.*, 400k v.s. 800k v.s. 1200k). Although NSCL-mIRT spends extra time to estimate the parameters of the mIRT model, such time cost is negligible compared to other time spent in training (less than 1%). From Table 6.1, we can see that NSCL-mIRT consistently outperforms FilM-LBA at various iterations, which demonstrates the preeminence of mIRT in building an adaptive curriculum.

Besides, NSCL-mIRT consumes less than 300k unique questions for training when it converges. It indicates that NSCL-mIRT saves about 60% of the training data, which largely eases the data redundancy problems. It provides a promising direction for designing a data-efficient curriculum and helping current data-hungry deep learning models save time and money cost during data annotation and model training.

Moreover, NSCL-mIRT obtains even higher accuracy than NSCL and NSCL-Fixed. This indicates that the adaptive curriculum built by the multi-dimensional IRT model not only remarkably increases the speed of convergence and reduces the data consumption during the training process, but also leads to better performance, which also verifies the hypothesis made by Bengio *et al.* [BLC09].

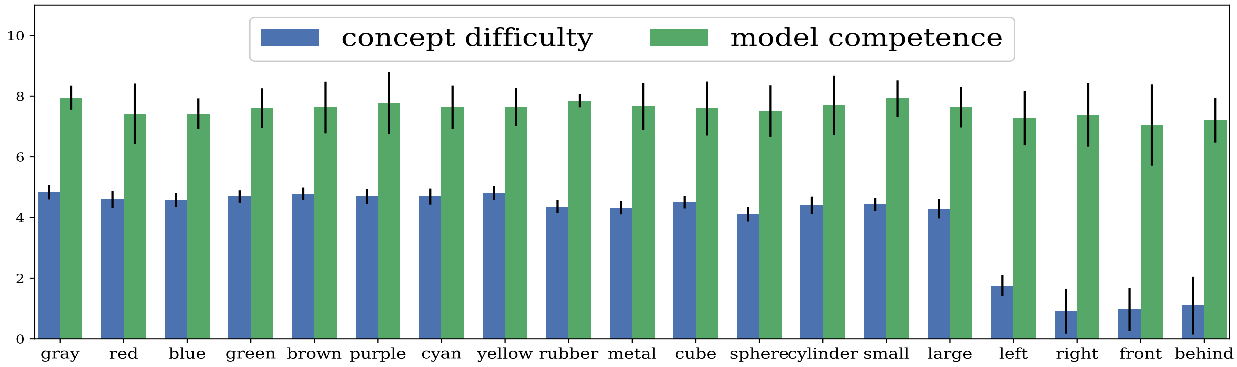


Figure 6.4: The estimated concept difficulty and model competence at the final iteration.

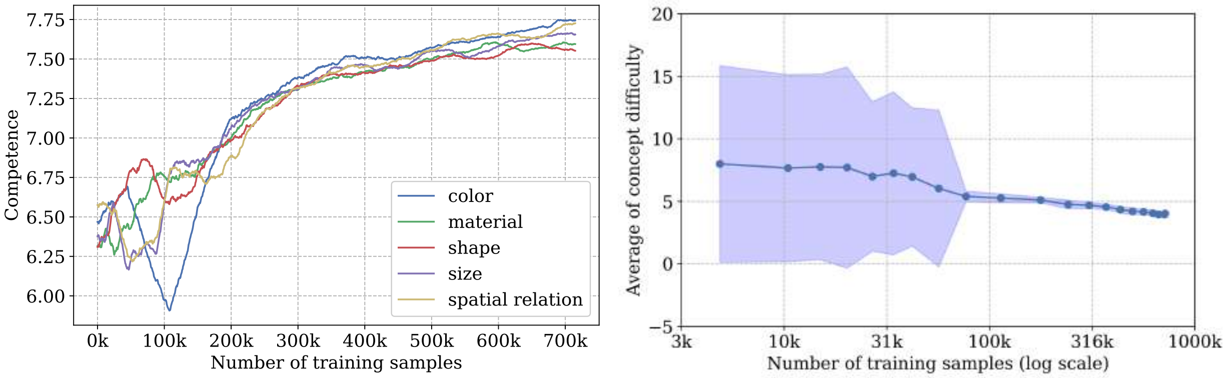


Figure 6.5: (a) The estimated model competence at various iterations for different attributes. The value for each attribute type is averaged from the visual concept it contains. (b) The estimated concept difficulty at various iterations. The shaded area represents the variance of the estimations.

6.1.4.3 Multi-dimensional IRT

The estimated concept difficulty and model competence after converging is shown in Figure 6.4 for studying the performance of the mIRT model. Several critical observations are: (1) The spatial relations (*i.e.*, left/right/front/behind) are the easiest concepts. It satisfies our intuition since the model only needs to exploit the object positions to determine their spatial relations without dealing with appearance. The spatial relations are learned during the late stages since they appear more frequently in complex questions to connect multiple concepts. (2) Colors are the most difficult concepts. The model needs to capture the subtle differences in the appearance of objects to distinguish eight different colors. (3) The model competence scores surpass the concept difficulty scores for all the concepts. This result

Table 6.1: The VQA accuracy of different models on the CLEVR validation set at various iterations. NSCL and NSCL-Fixed continue to improve with longer training steps, which is not shown for space limit.

Models	70k	140k	280k	420k	630k	700k
FiLM-LBA [MGF17]	51.2	**76.2**	92.9	94.8	95.2	97.3
NSCL	43.3	43.4	43.3	43.4	44.5	44.7
NSCL-Fixed	44.1	43.9	44.0	57.2	92.4	95.9
NSCL-mIRT	**53.9**	73.4	**97.1**	**98.5**	**98.9**	**99.3**

Table 6.2: The accuracy of the visual attributes of different models.

Model	Overall	Color	Material	Shape	Size
IEP [JHM17a]	90.6	91.0	90.0	89.9	90.6
MAC [HM18]	95.9	98.0	91.4	94.4	94.2
NSCL-Fixed [MGK19]	98.7	99.0	98.7	98.1	99.1
NSCL-mIRT	**99.5**	**99.5**	**99.7**	**99.4**	**99.6**

Table 6.3: Comparisons of the VQA accuracy on the CLEVR validation set with other models.

Model	Overall	Count	Cmp Num.	Exist	Query Attr.	Cmp Attr.
Human	92.6	86.7	86.4	96.6	95.0	96.0
IEP [JHM17a]	96.9	92.7	98.7	97.1	98.1	98.9
FiLM [PSV17]	97.6	94.5	93.8	99.2	99.2	99.0
MAC [HM18]	98.9	97.2	99.4	99.5	99.3	99.5
NSCL [MGK19]	98.9	98.2	99.0	98.8	99.3	99.1
NS-VQA [YWG18]	**99.8**	**99.7**	**99.9**	**99.9**	**99.8**	**99.8**
NSCL-mIRT	99.5	98.9	99.0	99.7	99.7	99.6

Table 6.4: The VQA accuracy on CLEVR validation set with different LBs and UBs in the question selection strategy. Both LB and UB are in log scale.

(LB,UB)	70k	140k	210k	280k	560k	770k
(-10, 0)	44.39	52.01	63.04	73.5	97.93	99.01
(-5, 0)	53.75	69.55	82.44	95.31	98.92	99.27
(-3, 0)	51.38	55.97	58.33	65.11	69.57	70.01
(-5, -0.5)	42.06	52.67	80.46	95.54	98.41	99.06
(-5, -0.75)	**53.91**	**73.42**	**93.6**	**97.07**	99.04	**99.50**
(-5, -1)	44.57	63.65	82.95	94.38	**99.15**	99.48

corresponds to the nearly perfect accuracy (> 99%) on all questions and concepts.

Section 6.1.4.2 shows the estimation of the model competence for each attribute type at various iterations. We can observe that model competence consistently increases throughout the training. Section 6.1.4.2 shows the estimations of the concept difficulty at different learning steps. As the training progresses, the estimations become more stable with smaller variance since more model responses are accumulated.

6.1.4.4 Concept Learner

We apply the count-based concept evaluation metric proposed in [MGK19] to measure the performance of the concept learner, which evaluates the visual concepts on synthetic questions with a single concept such as "How many *red* objects are there?" Table 6.2 presents the results by comparing with several state-of-the-art methods, which includes methods based on neural module network with programs (IEP [JHM17a]) and neural attentions without programs (MAC [HAR17]). Our model achieves nearly perfect performance across visual concepts and outperforms all other approaches. This means the model can learn visual concepts better with an adaptive curriculum. Our model can also be applied to the VQA. Table 6.3 summarizes the VQA accuracy on the CLEVR validation split. Our approach achieves comparable performance with state-of-the-art methods.

6.1.4.5 Question Selection strategy

The question selection strategy is controlled by two hyper-parameters: the lower bound (LB) and upper bound (UB). We conduct experiments by learning with different LBs and UBs, and Table 6.4 shows the VQA accuracy at various iterations. It reveals that the proper lower bound can effectively filter out too hard questions and accelerate the learning at the early stage of the training, as shown in the first three rows. Similarly, a proper upper bound helps to filter out too easy questions at the late stage of the training when the model has learned most concepts.

6.1.5 Conclusions and Discussions

We propose a competence-aware curriculum for visual concepts learning via question answering. We design a multi-dimensional IRT model to estimate concept difficulty and model competence at each training step from the accumulated model responses generated by different model snapshots. The estimated concept difficulty and model competence are further used to build an adaptive curriculum for the visual concept learner. Experiments on the CLEVR dataset show that the concept learner with the proposed competence-aware curriculum converges three times faster and consumes only 40% of the training data while achieving similar or even higher accuracy compared with other state-of-the-art models.

In the future, our work can be potentially applied to *real-world images* like GQA [HM19] and VQA-v2 [GKS17] datasets, by explicitly modeling the relationship among visual concepts. However, there are still unsolved challenges for real-world images. Specifically, compared with synthetic images in CLEVR, real-world images have a much larger vocabulary of visual concepts. For example, as shown in [AHB18], there are over 2,000 visual concepts in MSCOCO images. Usually, these concepts are automatically mined from image captions and scene graphs. Thus some of them are highly correlated like “huge” and “large”, and some of them are very subjective like “busy” and “calm”. Such a large and noisy vocabulary of visual concepts is challenging for the mIRT model since current visual concepts are assumed to be independent. It also requires a much longer time to converge when maximizing the ELBO to fit the mIRT model with more concepts. A potential solution is to consider the hierarchical structure of visual concept space and correlations among the concepts and incorporate commonsense knowledge to handle subjective concepts.

More importantly, the competence-aware curriculum can be adapted to other domains that possess compositional structures such as natural language processing. Specifically, in neural machine translation task [SVL14, BCB15], mIRT can be used to model the difficulty and competence of translating different words/phrases and build a curriculum to increase learning speed and data efficiency. mIRT can also be used in the task of semantic parsing [DL16, LBL16a, LNB18a] that transforms natural language sentences (*e.g.*, instructions or queries) into logic forms (*e.g.*, lambda-calculus or SQL). The difficulty and competence of different logic predicates can also be estimated by the mIRT model.

6.2 Close-loop Neural-symbolic Learning with Grammar Model

The goal of neural-symbolic computation is to integrate the connectionist and symbolist paradigms. Prior methods learn the neural-symbolic models using reinforcement learning (RL) approaches, which ignore the error propagation in the symbolic reasoning module and thus converge slowly with sparse rewards.

In this section, we address these issues and close the loop of neural-symbolic learning by (1) introducing the **grammar** model as a *symbolic prior* to bridge neural perception and symbolic reasoning, and (2) proposing a novel **back-search** algorithm which mimics the top-down human-like learning procedure to propagate the error through the symbolic reasoning module efficiently. We further interpret the proposed learning framework as maximum likelihood estimation using Markov chain Monte Carlo sampling and the back-search algorithm as a Metropolis-Hastings sampler. The experiments are conducted on two weakly-supervised neural-symbolic tasks: (1) handwritten formula recognition on the newly introduced HWF dataset; (2) visual question answering on the CLEVR dataset. The results show that our approach significantly outperforms the RL methods in terms of performance, converging speed, and data efficiency.

6.2.1 Introduction

Integrating robust connectionist learning and sound symbolic reasoning is a key challenge in modern Artificial Intelligence. Deep neural networks [LBH15, LB95, HS97] provide us powerful and flexible representation learning that has achieved state-of-the-art performances across a variety of AI tasks such as image classification [KSH12, SLJ15, HZR16], machine translation [SVL14], and speech recognition [GMH13]. However, it turns out that many aspects of human cognition, such as systematic compositionality and generalization [FP88, Mar98, FL02, CS14, Mar18, LB18], cannot be captured by neural networks. On the other hand, symbolic reasoning supports strong abstraction and generalization but is fragile and inflexible. Consequently, many methods have focused on building neural-symbolic models to combine the best of deep representation learning and symbolic reasoning [Sun94, GLG08,

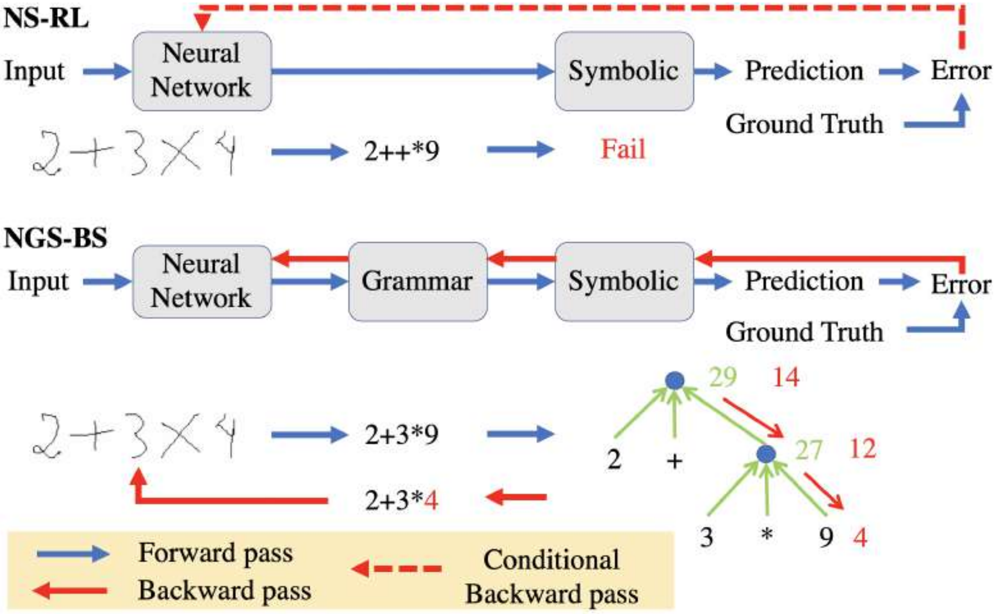


Figure 6.6: Comparison between the original neural-symbolic model learned by REINFORCE (NS-RL) and the proposed neural-grammar-symbolic model learned by back-search (NGS-BS). In NS-RL, the neural network predicts an invalid formula, causing a failure in the symbolic reasoning module. There is no backward pass in this example since it generates zero reward. In contrast, NGS-BS predicts a valid formula and searches a correction for its prediction. The neural network is updated using this correction as the pseudo label.

BGH09, BGB17b, YWG18].

Recently, this neural-symbolic paradigm has been extensively explored in the tasks of the visual question answering (VQA) [YWG18, VDL19, MGK19], vision-language navigation [AWT18, FHC18], embodied question answering [DDG18, DGL18], and semantic parsing [LBL16b, YZH18], often with weak supervision. Concretely, for these tasks, neural networks are used to map raw signals (images/questions/instructions) to symbolic representations (scenes/programs/actions), which are then used to perform symbolic reasoning/execution to generate final outputs. Weak supervision in these tasks usually provides pairs of raw inputs and final outputs, with intermediate symbolic representations unobserved. Since symbolic reasoning is non-differentiable, previous methods usually learn the neural-symbolic models by policy gradient methods like REINFORCE. The policy gradient methods generate samples and update the policy based on the generated samples that happen to hit high cumulative rewards. No efforts are made to improve each generated sample to increase its cumulative reward. Thus the learning has been proved to be time-consuming because it requires generating a large number of samples over a large latent space of symbolic representations with sparse rewards, in the hope that some samples may be lucky enough to hit high rewards so that such lucky samples can be utilized for updating the policy. As a result, policy gradients methods converge slowly or even fail to converge without pre-training the neural networks on fully-supervised data.

To model the recursive compositionality in a sequence of symbols, we introduce the **grammar** model to bridge neural perception and symbolic reasoning. The structured symbolic representation often exhibits compositional and recursive properties over individual symbols in it. Correspondingly, the grammar models encode *symbolic prior* about composition rules, thus can dramatically reduce the solution space by parsing the sequence of symbols into valid sentences. For example, in the handwritten formula recognition problem, the grammar model ensures that the predicted formula is always valid, as shown in Figure 6.6.

To make the neural-symbolic learning more efficient, we propose a novel **back-search** strategy which mimics human's ability to learn from failures via abductive reasoning [Mag09, Zho19]. Specifically, the back-search algorithm propagates the error from the root node to the leaf nodes in the reasoning tree and finds the most probable *correction* that can generate the desired output. The correction is further used as a pseudo label for training the neural network. Figure 6.6 shows an exemplar backward pass of the back-search algorithm. We argue that the back-search algorithm makes a first step towards closing the learning loop by propagating the error through the non-differentiable grammar parsing and symbolic reasoning modules. We also show that the proposed multi-step back-search algorithm can serve as a Metropolis-Hastings sampler which samples the posterior distribution of the symbolic representations in the maximum likelihood estimation in Section 6.2.3.2.

We conduct experiments on two weakly-supervised neural-symbolic tasks: (1) handwritten formula recognition on the newly introduced HWF dataset (Hand-Written Formula), where the input image and the formula result are given during training, while the formula is hidden; (2) visual question answering on the CLEVR dataset. The question, image, and answer are given, while the functional program generated by the question is hidden. The evaluation results show that the proposed Neural-Grammar-Symbolic (NGS) model with back-search significantly outperforms the baselines in terms of performance, convergence speed, and data efficiency. The ablative experiments also demonstrate the efficacy of the multi-step back-search algorithm and the incorporation of grammar in the neural-symbolic model.

6.2.2 Related Work

Neural-symbolic Integration. Researchers have proposed to combine statistical learning and symbolic reasoning in the AI community, with pioneer works devoted to different aspects including representation learning and reasoning [Sun94, GLG08, MDK18], abductive learning [DZ17, DXY19, Zho19, HLC20, HLG20], knowledge abstraction [HOT06, BGH09], knowledge transfer [FFG89, YCX09], *etc.*. Recent research shifts the focus to the application of neural-symbolic integration, where a large amount of heterogeneous data and knowledge descriptions are needed, such as neural-symbolic VQA [YWG18, VDL19, MGK19, LFY18, LTJ18, LHH20b], semantic parsing in Natural Language Processing (NLP) [LBL16b, YZH18], math word problem [LC19, LSR19] and program synthesis [EG18, KMP18, MDK18]. Different from previous methods, the proposed NGS model considers the compositionality and recursivity in natural sequences of symbols and brings together the neural perception and symbolic reasoning module with a grammar model.

Grammar Model. Grammar model has been adopted in various tasks for its advantage in modeling compositional and recursive structures, like image parsing [TCY05, HZ05, ZM07, ZZ11, FD18], video parsing [GSS09, QJZ18, QJH20], scene understanding [HQZ18, HQX18, QZH18, JQZ18, CHY19], and task planning [XLE18]. By integrating the grammar into the neural-symbolic task as a symbolic prior for the first time, the grammar model ensures the desired dependencies and structures for the symbol sequence and generates valid sentences for symbolic reasoning. Furthermore, it improves the learning efficiency significantly by shrinking the search space with the back-search algorithm.

Policy Gradient. Policy gradient methods like REINFORCE [Wil92] are the most commonly used algorithm for the neural-symbolic tasks to connect the learning gap between neural networks and symbolic reasoning [MTS18, MGK19, AKL17, DGL18, BHD18, GPL17]. However, original REINFORCE algorithm suffers from large sample estimate variance, sparse rewards from cold start and exploitation-exploration dilemma, which lead to unstable learning dynamics and poor data efficiency. Many papers propose to tackle this problem [LBL16b, GPL17, LNB18b, WZG18, ALS19]. Specifically, [LBL16b] uses iterative maximum likelihood to find pseudo-gold symbolic representations, and then add these representations to the REINFORCE training set. [GPL17] combines the systematic beam search employed in maximum marginal likelihood with the greedy randomized exploration of REINFORCE. [LNB18b] proposes Memory Augmented Policy Optimization (MAPO) to express the expected return objective as a weighted sum of an expectation over the high-reward history trajectories, and a separate expectation over new trajectories. Although utilizing positive representations from either beam search or past training process, these methods still cannot learn from negative samples and thus fail to explore the solution space efficiently. On the contrary, we propose to diagnose and correct the negative samples through the back-search algorithm under the constraint of grammar and symbolic reasoning rules. Intuitively speaking, the proposed back-search algorithm traverses around the negative sample and find a nearby positive sample to help the training.

6.2.3 Neural-Grammar-Symbolic Model (NGS)

In this section, we will first describe the inference and learning algorithms of the proposed neural-grammar-symbolic (NGS) model. Then we provide an interpretation of our model based on maximum likelihood estimation (MLE) and draw the connection between the proposed back-search algorithm and Metropolis-Hastings sampler. We further introduce the task-specific designs in Section 6.2.4.

6.2.3.1 Inference

In a neural-symbolic system, let x be the input (*e.g.*, an image or question), z be the hidden symbolic representation, and y be the desired output inferred by z. The proposed NGS model combines neural perception, grammar parsing, and symbolic reasoning modules efficiently to perform the inference.

Neural Perception. The neural network is used as a perception module which maps the high-dimensional input x to a normalized probability distribution of the hidden symbolic representation z:

$$\begin{aligned} p_\theta(z|x) &= softmax(\phi_\theta(z,x)) \qquad (6.15) \\ &= \frac{\exp(\phi_\theta(z,x))}{\sum_{z'} \exp(\phi_\theta(z',x))}, \qquad (6.16) \end{aligned}$$

where $\phi_\theta(z,x)$ is a scoring function or a negative energy function represented by a neural network with parameters θ.

Grammar Parsing. Take z as a sequence of individual symbols: $z = (z_1, z_2, ..., z_l), z_i \in \Sigma$, where Σ denotes the vocabulary of possible symbols. The neural network is powerful at modeling the mapping between x and z, but the recursive compositionality among the individual symbols z_i is not well captured. Grammar is a natural choice to tackle this problem by modeling the compositional properties in sequence data.

Take the *context-free grammar* (CFG) as an example. In formal language theory, a CFG is a type of formal grammar containing a set of production rules that describe all possible sentences in a given formal language. Specifically, a context-free grammar G in Chomsky Normal Form is defined by a 4-tuple $G = (V, \Sigma, R, S)$, where

- V is a finite set of non-terminal symbols that can be replaced by/expanded to a sequence of symbols.
- Σ is a finite set of terminal symbols that represent actual words in a language, which cannot be further expanded. Here Σ is the vocabulary of possible symbols.

- R is a finite set of production rules describing the replacement of symbols, typically of the form $A \to BC$ or $A \to \alpha$, where $A, B, C \in V$ and $\alpha \in \Sigma$. A production rule replaces the left-hand side non-terminal symbols by the right-hand side expression. For example, $A \to BC|\alpha$ means that A can be replaced by either BC or α.

- $S \in V$ is the start symbol.

Given a formal grammar, *parsing* is the process of determining whether a string of symbolic nodes can be accepted according to the production rules in the grammar. If the string is accepted by the grammar, the parsing process generates a parse tree. A parse tree represents the syntactic structure of a string according to certain CFG. The root node of the tree is the grammar root. Other non-leaf nodes correspond to non-terminals in the grammar, expanded according to grammar production rules. The leaf nodes are terminal nodes. All the leaf nodes together form a sentence.

In neural-symbolic tasks, the objective of parsing is to find the most probable z that can be accepted by the grammar:

$$\hat{z} = \arg\max_{z \in L(G)} p_\theta(z|x) \tag{6.17}$$

where $L(G)$ denotes the language of G, i.e., the set of all valid z that accepted by G.

Traditional grammar parsers can only work on symbolic sentences. [QJZ18] proposes a generalized version of Earley Parser, which takes a probability sequence as input and outputs the most probable parse. We use this method to compute the best parse $\hat{z}$ in Equation (6.17).

Symbolic Reasoning. Given the parsed symbolic representation $\hat{z}$, the symbolic reasoning module performs deterministic inference with $\hat{z}$ and the domain-specific knowledge Δ. Formally, we want to find the entailed sentence $\hat{y}$ given $\hat{z}$ and Δ:

$$\hat{y} : \hat{z} \ \wedge \ \Delta \models \hat{y} \tag{6.18}$$

Since the inference process is deterministic, we re-write the above equation as:

$$\hat{y} = f(\hat{z}; \Delta), \tag{6.19}$$

where f denotes complete inference rules under the domain Δ. The inference rules generate a reasoning path $\hat{\tau}$ that leads to the predicted output $\hat{y}$ from $\hat{z}$ and Δ. The reasoning path $\hat{\tau}$ has a tree structure with the root node $\hat{y}$ and the leaf nodes from $\hat{z}$ or Δ.

6.2.3.2 Learning

It is challenging to obtain the ground truth of the symbolic representation z, and the rules (*i.e.*grammar rules and the symbolic inference rules) are usually designed explicitly by human knowledge. We formulate the learning process as a weakly-supervised learning of the neural network model θ where the symbolic representation z is missing, and the grammar model G, domain-specific language Δ, the symbolic inference rules f are given.

1-step back-search (1-BS) As shown in Figure 6.6, previous methods using policy gradient to learn the model discard all the samples with zero reward and learn nothing from them. It makes the learning process inefficient and unstable. However, humans can learn from the wrong predictions by *diagnosing* and *correcting* the wrong answers according to the desired outputs with top-down reasoning. Based on such observation, we propose a 1-step back-search (1-BS) algorithm which can *correct* wrong samples and use the corrections as pseudo labels for training. The 1-BS algorithm closes the learning loop since the error can also be propagated through the non-differentiable grammar parsing and symbolic reasoning modules. Specifically, we find the most probable correction for the wrong prediction by back-tracking the symbolic reasoning tree and propagating the error from the root node into the leaf nodes in a top-down manner.

The 1-BS algorithm is implemented with a priority queue as shown in Algorithm 4. The 1-BS gradually searches down the reasoning tree $\hat{\tau}$ starting from the root node S to the leaf nodes. Specifically, each element in the priority queue represents a valid change, defined as a 3-tuple (A, α_A, p):

- $A \in V \cup \Sigma$ is the current visiting node.
- α_A is the expected value on this node, which means if the value of A is changed to α_A, $\hat{z}$ will execute to the ground-truth answer y, $i.e. y = f(\hat{z}(A \rightarrow \alpha_A); \Delta))$.
- p is the visiting priority, which reflects the potential of changing the value of A.

Formally, the priority for this change is defined as the probability ratio:

$$p(A \rightarrow \alpha_A) = \begin{cases} \frac{1-p(A)}{p(A)}, & \text{if } A \notin \Sigma \\ \frac{p(\alpha_A)}{p(A)}, & \text{if } A \in \Sigma \ \& \ \alpha_A \in \Sigma. \end{cases} \tag{6.20}$$

where $p(A)$ is calculated as Equation 6.15,if $A \in \Sigma$; otherwise, it is defined as the product of the probabilities of all leaf nodes in A. If $A \in \Sigma$ and $\alpha_A \notin \Sigma$, it means we need to correct the terminal node to a value that is not in the vocabulary. Therefore, this change is not possible and thus should be discarded.

The error propagation through the reasoning tree is achieved by a $solve(B, A, \alpha_A|\Delta, G)$ function, which aims at computing the expected value α_B of the child node B from the expected value α_A of its parent node A, *i.e.*, finding α_B satisfying $f(\hat{z}(B \rightarrow \alpha_B); \Delta)) = f(\hat{z}(A \rightarrow \alpha_A); \Delta)) = y$.

In the 1-BS, we make a greedy assumption that only one symbol can be replaced at a time. This assumption implies only searching the neighborhood of $\hat{z}$ at one-step distance. In Section 6.2.3.2, the 1-BS is extended to the multi-step back-search algorithm, which allows searching beyond one-step distance.

Maximum Likelihood Estimation Since z is conditioned on x and y is conditioned on z, the likelihood for the observation (x, y) marginalized over z is:

$$p(y|x) = \sum_z p(y, z|x) = \sum_z p(y|z) p_\theta(z|x). \tag{6.21}$$

Algorithm 4 1-step back-search (1-BS)

```
Input: $\hat{z}, S, y$
$q = PriorityQueue()$
$q.push(S, y, 1)$
while $A, \alpha_A, p = q.pop()$ do
    if $A \in \Sigma$ then
        $z^* = \hat{z}(A \rightarrow \alpha_A)$
        return $z^*$
    for $B \in child(A)$ do
        $\alpha_B = solve(B, A, \alpha_A|\Delta, G)$
        $q.push(B, \alpha_B, p(B \rightarrow \alpha_B))$
return $\varnothing$
```

The learning goal is to maximize the observed-data log likelihood $L(x, y) = \log p(y|x)$.

By taking derivative, the gradient for the parameter θ is given by

$$\begin{aligned}
\nabla_\theta L(x, y) &= \nabla_\theta \log p(y|x) \\
&= \frac{1}{p(y|x)} \nabla_\theta p(y|x) \\
&= \sum_z \frac{p(y|z)p_\theta(z|x)}{\sum_{z'} p(y|z')p_\theta(z'|x)} \nabla_\theta \log p_\theta(z|x) \\
&= \mathbb{E}_{z\sim p(z|x,y)}[\nabla_\theta \log p_\theta(z|x)], \qquad (6.22)
\end{aligned}$$

where $p(z|x, y)$ is the posterior distribution of z given x, y. Since $p(y|z)$ is computed by the symbolic reasoning module and can only be 0 or 1, $p(z|x, y)$ can be written as:

$$\begin{aligned}
p(z|x, y) &= \frac{p(y|z)p_\theta(z|x)}{\sum_{z'} p(y|z')p_\theta(z'|x)} \\
&= \begin{cases} 0, & \text{for} \quad z \notin Q \\ \frac{p_\theta(z|x)}{\sum_{z' \in Q} p_\theta(z'|x)}, & \text{for } z \in Q \end{cases} \qquad (6.23)
\end{aligned}$$

where $Q = \{z : p(y|z) = 1\} = \{z : f(z; \Delta) = y\}$ is the set of z that generates y. Usually Q is a very small subset of the whole space of z.

Equation (6.23) indicates that z is sampled from the posterior distribution $p(z|x, y)$, which only has non-zero probabilities on Q, instead of the whole space of z.

Unfortunately, computing the posterior distribution is not efficient as evaluating the normalizing constant for this distribution requires summing over all possible z, and the computational complexity of the summation grows exponentially.

Nonetheless, it is feasible to design algorithms that sample from this distribution using Markov chain Monte Carlo (MCMC). Since z is always trapped in the modes where $p(z|x, y) = 0$, the remaining question is how we can sample the posterior distribution $p(z|x, y)$ efficiently to avoid redundant random walk at states with zero probabilities.

Algorithm 5 m-step back-search (m-BS)

```
**Hyperparameters**: $T$, $\lambda$
**Input**: $\hat{z}, y$
$z^{(0)} = \hat{z}$
**for** $t \leftarrow 0$ to $T - 1$ **do**
    $z^* = \text{1-BS}(z^t, y)$
    draw $u \sim \mathcal{U}(0, 1)$
    **if** $u \leqslant \lambda$ and $z^* \neq \varnothing$ **then**
        $z^{t+1} = z^*$
    **else**
        $z^{t+1} = \textsc{RandomWalk}(z^t)$
**return** $z^T$

**function** $\textsc{RandomWalk}(z^t)$
    sample $z^* \sim g(\cdot|z^t)$
    compute acceptance ratio $a = min(1, \frac{p_\theta(z^*|x)}{p_\theta(z^t|x)})$
    draw $u \sim \mathcal{U}(0, 1)$
    $z^{t+1} = \begin{cases} z^*, & \text{if } u \leqslant a \\ z^t, & \text{otherwise.} \end{cases}$
```

m-BS as Metropolis-Hastings Sampler In order to perform efficient sampling, we extend the 1-step back search to a multi-step back search (m-BS), which serves as a Metropolis-Hastings sampler.

A Metropolis-Hastings sampler for a probability distribution $\pi(s)$ is a MCMC algorithm that makes use of a proposal distribution $Q(s'|s)$ from which it draws samples and uses an acceptance/rejection scheme to define a transition kernel with the desired distribution $\pi(s)$. Specifically, given the current state s, a sample $s' \neq s$ drawn from $Q(s'|s)$ is accepted as the next state with probability

$$A(s, s') = min\left\{1, \frac{\pi(s')Q(s|s')}{\pi(s)Q(s'|s)}\right\}. \tag{6.24}$$

Since it is impossible to jump between the states with zero probability, we define $p'(z|x, y)$ as a smoothing of $p(z|x, y)$ by adding a small constant ϵ to $p(y|z)$:

$$p'(z|x, y) = \frac{[p(y|z) + \epsilon]p_\theta(z|x)}{\sum_{z'}[p(y|z') + \epsilon]p_\theta(z'|x)} \tag{6.25}$$

As shown in Algorithm 5, in each step, the m-BS proposes 1-BS search with probability of λ ($\lambda < 1$) and random walk with probability of $1 - \lambda$. The combination of 1-BS and random walk helps the sampler to traverse all the states with non-zero probabilities and ensures the Markov chain to be ergodic.

Random Walk: Defining a Poisson distribution for the random walk as

$$g(z_1|z_2) = Poisson(d(z_1, z_2); \beta), \tag{6.26}$$

where $d(z_1, z_2)$ denotes the edit distance between z_1, z_2, and β is equal to the expected value of d and also to its variance. β is set as 1 in most cases due to the preference for a short-distance random walk. The acceptance ratio for sampling a z^* from $g(\cdot|z^t)$ is $a = min(1, r(z^t, z^*))$, where

$$\begin{aligned} r(z^t, z^*) &= \frac{q(z^*)(1-\lambda)g(z^t|z^*)}{q(z^t)(1-\lambda)g(z^*|z^t)} \\ &= \frac{p_\theta(z^*|x)}{p_\theta(z^t|x)}. \end{aligned} \tag{6.27}$$

1-BS: While proposing the z^* with 1-BS, we search a z^* that satisfies $p(y|z^*) = 1$. If z^*

is proposed, the acceptance ratio for is $a = min(1, r(z^t, z^*))$, where

$$\begin{aligned} r(z^{(t)}, z^*) &= \frac{q(z^*)[0 + (1-\lambda)g(z^t|z^*)]}{q(z^t) \cdot [\lambda + (1-\lambda)g(z^*|z^{(t)})]} \\ &= \frac{1+\epsilon}{\epsilon} \cdot \frac{p_\theta(z^*|x)}{p_\theta(z^t|x)} \cdot \frac{(1-\lambda)g(z^t|z^*)}{\lambda + (1-\lambda)g(z^*|z^t)}. \end{aligned} \tag{6.28}$$

$q(z) = [p(y|z) + \epsilon]p_\theta(z|x)$ is denoted as the numerator of $p'(z|x, y)$. With an enough small ϵ, $\frac{1+\epsilon}{\epsilon} \gg 1$, $r(z^t, z^*) > 1$, we will always accept z^*.

Notably, the 1-BS algorithm tries to transit the current state into a state where $z^* =$ 1-BS(z^t, y), making movements in directions of increasing the posterior probability. Similar to the gradient-based MCMCs like Langevin dynamics [DK86, WT11], this is the main reason that the proposed method can sample the posterior efficiently.

Comparison with Policy Gradient Since grammar parsing and symbolic reasoning are non-differentiable, most of the previous approaches for neural-symbolic learning use policy gradient like REINFORCE to learn the neural network. Treat $p_\theta(z|x)$ as the policy function and the reward given z, y can be written as:

$$r(z, y) = \begin{cases} 0, & \text{if } f(z; \Delta) \neq y. \\ 1, & \text{if } f(z; \Delta) = y. \end{cases} \tag{6.29}$$

The learning objective is to maximize the expected reward under current policy p_θ:

$$R(x, y) = \mathbb{E}_{z \sim p_\theta(z|x))} r(z, y) = \sum_z p_\theta(z|x) r(z, y). \tag{6.30}$$

Then the gradient for θ is:

$$\begin{aligned} \nabla_\theta R(x, y) &= \sum_z r(z, y) p_\theta(z|x) \nabla_\theta \log p_\theta(z|x) \\ &= \mathbb{E}_{z \sim p_\theta(z|x))}[r(z, y) \nabla_\theta \log p_\theta(z|x)]. \end{aligned} \tag{6.31}$$

We can approximate the expectation using one sample at each time, and then we get the

REINFORCE algorithm:

$$\begin{aligned}\nabla_\theta &= r(z,y)\nabla_\theta \log p_\theta(z|x), z \sim p_\theta(z|x) \\ &= \begin{cases} 0, & \text{if } f(z;\Delta) \neq y. \\ \nabla_\theta \log p_\theta(z|x), & \text{if } f(z;\Delta) = y. \end{cases}\end{aligned} \tag{6.32}$$

Equation (6.32) reveals the gradient is non-zero only when the sampled z satisfies $f(z;\Delta) = y$. However, among the whole space of z, only a very small portion can generate the desired y, which implies that *the REINFORCE will get zero gradients from most of the samples*. This is why the REINFORCE method converges slowly or even fail to converge, as also shown from the experiments in Section 6.2.4.

6.2.4 Experiments and Results

6.2.4.1 Handwritten Formula Recognition

Task definition. The handwritten formula recognition task tries to recognize each mathematical symbol given a raw image of the handwritten formula. We learn this task in a weakly-supervised manner, where raw image of the handwritten formula is given as input data x, and the computed results of the formulas is treated as outputs y. The symbolic representation z that represent the ground-truth formula composed by individual symbols is hidden. Our task is to predict the formula, which could further be executed to calculate the final result.

HWF Dataset. We generate the HWF dataset based on the CROHME 2019 Offline Handwritten Formula Recognition Task[1]. First, we extract all symbols from CROHME and only keep ten digits (0∼9) and four basic operators (+,−,×, ÷). Then we generate formulas by sampling from a pre-defined grammar that only considers arithmetic operations over single-digit numbers. For each formula, we randomly select symbol images from CROHME. Overall, our dataset contains 10K training formulas and 2K test formulas.

[1] `https://www.cs.rit.edu/~crohme2019/task.html`

Evaluation Metrics. We report both the calculation accuracy (*i.e.*whether the calculation of predicted formula yields to the correct result) and the symbol recognition accuracy (*i.e.*whether each symbol is recognized correctly from the image) on the synthetic dataset.

Models. In this task, we use LeNet [LeC15] as the neural perception module to process the handwritten formula. Before feeding into LeNet, the original image of an formula is pre-segmented into a sequence of sub-images, and each sub-image contains only one symbol. The symbolic reasoning module works like a calculator, and each inference step computes the parent value given the values of two child nodes (left/right) and the operator. The $solve(B, A, \alpha_A)$ function in 1-step back-search algorithm works in the following way for mathematical formulas:

- If B is A's left or right child, we directly solve the equation $\alpha_B \bigoplus child_R(A) = \alpha_A$ or $child_L(A) \bigoplus \alpha_B = \alpha_A$ to get α_B, where $\bigoplus$ denotes the operator.
- If B is an operator node, we try all other operators and check whether the new formula can generate the correct result.

We conduct experiments by comparing the following variants of the proposed model:

- **NGS-RL**: learning the NGS model with REINFORCE.
- **NGS-MAPO**: learning the NGS model by Memory Augmented Policy Optimization (MAPO) [LNB18b], which leverages a memory buffer of rewarding samples to reduce the variance of policy gradient estimates.
- **NGS-RL-Pretrain**: NGS-RL with LeNet pre-trained on a small set of fully-supervised data.
- **NGS-MAPO-Pretrain**: NGS-MAPO with pre-trained LeNet.
- **NGS-m-BS**: learning the NGS model with the proposed m-step back-search algorithm.

Learning Curve. Figure 6.7 shows the learning curves of different models. The proposed NGS-m-BS converges much faster and achieves higher accuracy compared with other models.

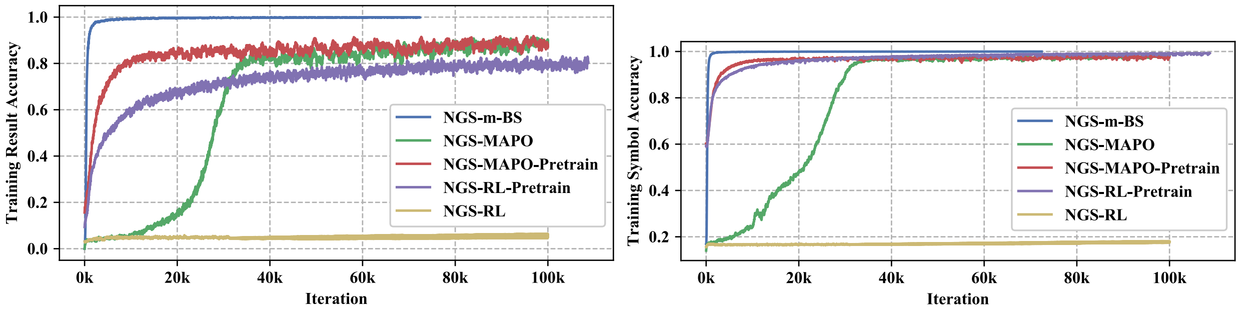


Figure 6.7: The learning curves of the calculation accuracy and the symbol recognition accuracy for different models.

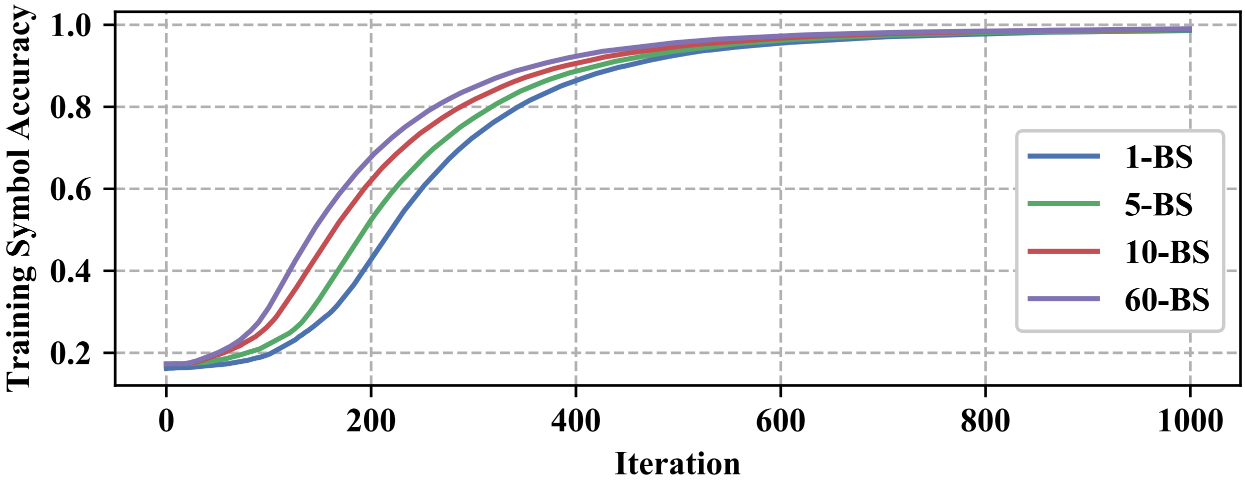


Figure 6.8: The training curves of NGS-m-BS with different steps.

NGS-RL fails without pre-training and rarely improves during the entire training process. NGS-MAPO can learn the model without pre-training, but it takes a long time to start efficient learning, which indicates that MAPO suffers from the cold-start problem and needs time to accumulate rewarding samples. Pre-training the LeNet solves the cold start problem for NGS-RL and NGS-MAPO. However, the training curves for these two models are quite noisy and are hard to converge even after 100k iterations. Our NGS-m-BS model learns from scratch and avoids the cold-start problem. It converges quickly with nearly perfect accuracy, with a much smoother training curve than the RL baselines.

Back-Search Step. Figure 6.8 illustrates the comparison of the various number of steps in the multi-step back-search algorithm. Generally, increasing the number of steps will increase the chances of correcting wrong samples, thus making the model converge faster.

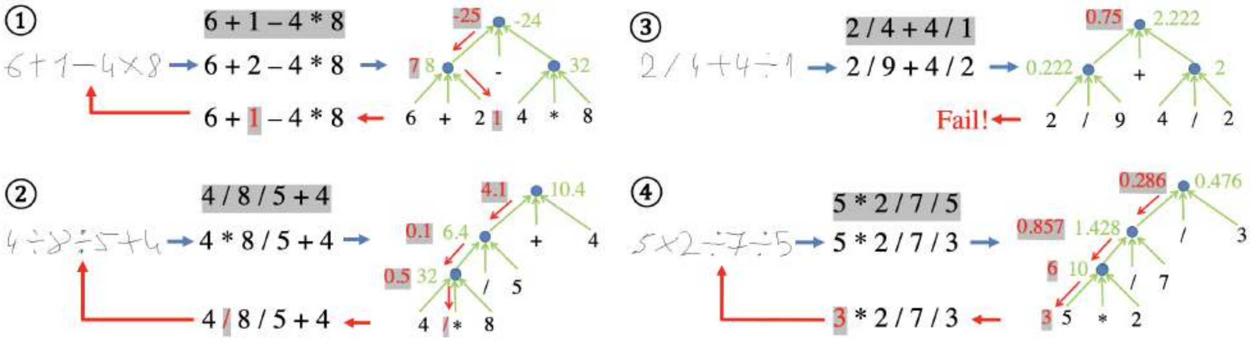


Figure 6.9: Examples of correcting wrong predictions using the one-step back-search algorithm.

However, increasing the number of steps will also increase the time consumption of each iteration.

Data Efficiency. Table 6.5 and Table 6.6 show the accuracies on the test set while using various percentage of training data. All models are trained with 15K iterations. It turns out the NGS-m-BS is much more data-efficient than the RL methods. Specifically, when only using 25% of the training data, NGS-m-BS can get a calculation accuracy of 93.3%, while NGS-MAPO only gets 5.1%.

Table 6.5: The calculation accuracy on the test set using various percentage of training data.

Model	25%	50 %	75 %	100%
NGS-RL	0.035	0.036	0.034	0.034
NGS-MAPO	0.051	0.095	0.305	0.717
NGS-RL-Pretrain	0.534	0.621	0.663	0.685
NGS-MAPO-Pretrain	0.687	0.773	0.893	0.956
NGS-m-BS	**0.933**	**0.957**	**0.975**	**0.985**

Table 6.6: The symbol recognition accuracy on the test set using various percentage of training data.

Model	25%	50 %	75 %	100%
NGS-RL	0.170	0.170	0.170	0.170
NGS-MAPO	0.316	0.481	0.785	0.967
NGS-RL-Pretrain	0.916	0.945	0.959	0.964
NGS-MAPO-Pretrain	0.962	0.983	0.985	0.991
NGS-m-BS	**0.988**	**0.992**	**0.995**	**0.997**

Qualitative Results. Figure 6.9 illustrates four examples of correcting the wrong predictions with 1-BS. In the first two examples, the back-search algorithm successfully corrects the wrong predictions by changing a digit and an operator, respectively. In the third example, the back-search fails to correct the wrong sample. However, if we increase the number of search steps, the model could find a correction for the example. In the fourth example, the back-search finds a spurious correction, which is not the same as the ground-truth formula but generates the same result. Such spurious correction brings a noisy gradient to the neural network update. It remains an open problem for how to avoid similar spurious corrections.

6.2.4.2 Neural-Symbolic Visual Question Answering

Task. Following [YWG18], the neural-symbolic visual question answering task tries to parse the question into functional program and then use a program executor that runs the program on the structural scene representation to obtain the answer. The functional program is hidden.

Dataset. We evaluate the proposed method on the CLEVR dataset [JHM17a]. The CLEVR dataset is a popular benchmark for testing compositional reasoning capability of VQA models in previous works [JHV17, VDL19]. CLEVR consists of a training set of 70K images and ∼700K questions, and a validation set of 15K images and ∼150K questions. We use the VQA accuracy as the evaluation metric.

Models. We adopt the NS-VQA model in [YWG18] and replace the attention-based seq2seq question parser with a Pointer Network [VFJ15]. We store a dictionary to map the keywords in each question to the corresponding functional modules. For example, "red"→"filter color [red]", "how many"→ "count", and "what size" → "query size" *etc.* Therefore, the Pointer Network can point to the functional modules that are related to the input question. The grammar model ensures that the generated sequence of function modules can form a valid program, which indicates the inputs and outputs of these modules can be strictly matched with their forms. We conduct experiments by comparing following models: **NS-RL**, **NGS-RL**, **NGS-1-BS**, **NGS-m-BS**.

Learning Curve. Figure 6.10 shows the learning curves of different model variants. NGS-

BS converges much faster and achieves higher VQA accuracy on the test set compared with the RL baselines. Though taking a long time, NGS-RL does converge, while NS-RL fails. This fact indicates that the grammar model plays a critical role in this task. Conceivably, the latent functional program space is combinatory, but the grammar model rules out all invalid programs that cannot be executed by the symbolic reasoning module. It largely reduces the solution space in this task.

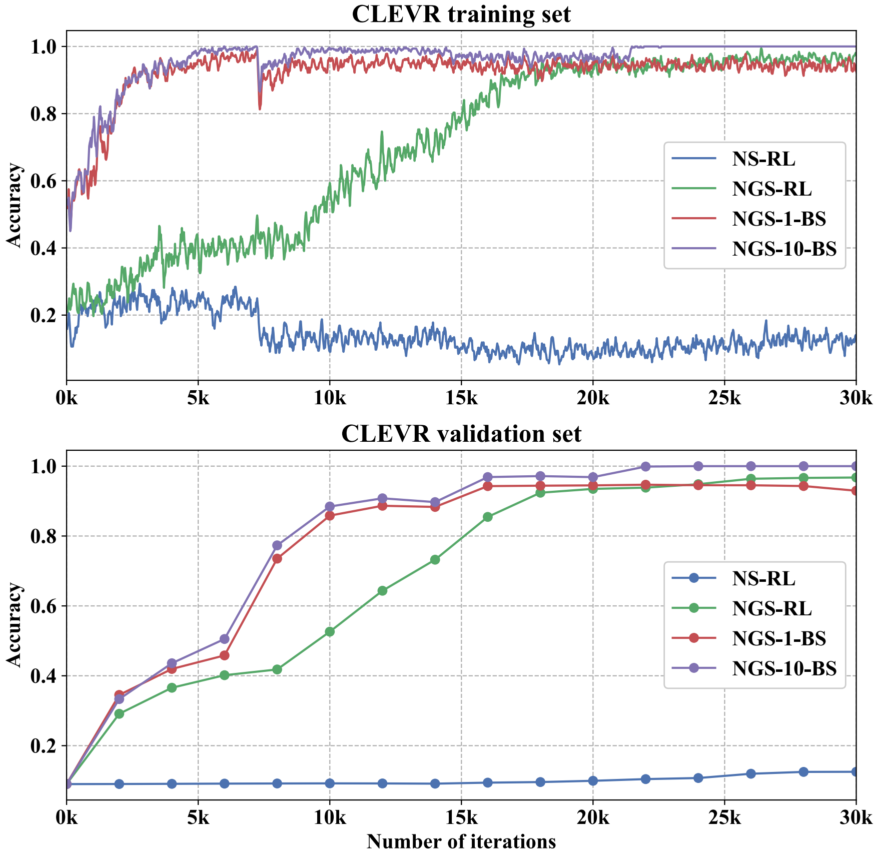


Figure 6.10: The learning curve of different model variants on training and validation set of the CLEVR dataset.

Back-Search Step. As shown in Figure 6.10, NGS-10-BS performs slightly better than the NGS-1-BS, which indicates that searching multiple steps does not help greatly in this task. One possible reason is that there are more ambiguities and more spurious examples compared with the handwritten formula recognition task, making it less efficient to do the m-BS. For example, for the answer “yes”, there might be many possible programs for this question that can generate the same answer given the image.

Data Efficiency Table 6.7 shows the accuracies on the CLEVR validation set when different portions of training data are used. With less training data, the performances decrease for both NGS-RL and NGS-m-BS, but NGS-m-BS still consistently obtains higher accuracies.

Table 6.7: The VQA accuracy on the CLEVR validation set using different percentage of training data. All models are trained 30k iterations.

Model	25%	50 %	75 %	100%
NS-RL	0.090	0.091	0.099	0.125
NGS-RL	0.678	0.839	0.905	0.969
NGS-m-BS	**0.873**	**0.936**	**1.000**	**1.000**

6.2.5 Conclusions

In this work, we propose a neural-grammar-symbolic model and a back-search algorithm to close the loop of neural-symbolic learning. We demonstrate that the grammar model can dramatically reduce the solution space by eliminating invalid possibilities in the latent representation space. The back-search algorithm endows the NGS model with the capability of learning from wrong samples, making the learning more stable and efficient. One future direction is to learn the symbolic prior (*i.e.*the grammar rules and symbolic inference rules) automatically from the data.

CHAPTER 7

Systematic Generalization of Perception, Syntax, and Semantics

Inspired by humans' remarkable ability to master arithmetic and generalize to unseen problems, we present a new dataset, HINT, to study machines' capability of learning generalizable concepts at three different levels: *perception*, *syntax*, and *semantics*. In particular, concepts in HINT, including both digits and operators, are required to learn in a weakly-supervised fashion: Only the final results of handwriting expressions are provided as supervision. Learning agents need to reckon how concepts are perceived from raw signals such as images (*i.e.*, perception), how multiple concepts are structurally combined to form a valid expression (*i.e.*, syntax), and how concepts are realized to afford various reasoning tasks (*i.e.*, semantics). With a focus on systematic generalization, we carefully design a five-fold test set to evaluate both the *interpolation* and the *extrapolation* of learned concepts. To tackle this challenging problem, we propose a neural-symbolic system by integrating neural networks with grammar parsing and program synthesis, learned by a novel deduction–abduction strategy. In experiments, the proposed neural-symbolic system can successfully learn the three-level meanings of concepts with weak supervision and generalize much better than end-to-end neural methods like RNN and Transformer.

An additional study of few-shot learning also indicates that the proposed model can learn new concepts with limited examples.

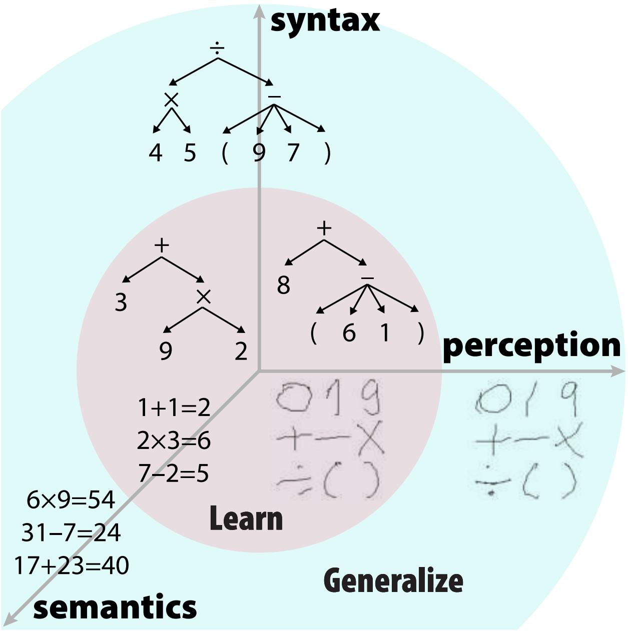


Figure 7.1: Concept learning and generalization at three different levels. A learning agent needs to simultaneously master (i) **perception**, how concepts are perceived from raw signals such as images, (ii) **syntax**, how multiple concepts are structurally combined to form a valid expression, and (iii) **semantics**, how concepts are realized to afford various reasoning tasks.

7.1 Introduction

Humans possess a versatile mechanism for learning concepts [FS16]. Take the arithmetic examples in Figure 7.1: When we master concepts like digits and operators, we not only know how to recognize, write, and pronounce them—what these concepts mean at the *perceptual* level, but also know how to compose them into valid expressions—at the *syntactic* level, and how to calculate the results by reasoning over these concepts—at the *semantic* level. Learning concepts heavily rely on these three-level interweaving meanings. Such observation also conforms with the classic view of human cognition, which postulates at least three distinct levels of organizations in computation systems [Pyl84, FP88].

Crucially, a unique property of human concept learning is its systematic generalization. Once we master the syntax of arithmetic using short expressions, we can parse novel, long expressions. Similarly, once we master operators' semantics using small numbers, we can apply them over novel, large numbers. This property corresponds to the classic idea of the *systematicity* (interpolation) and *productivity* (extrapolation) in cognition: An infinite number of representations can be constructed from a finite set of primitives, just as the mind can think an infinite number of thoughts, understand an infinite number of sentences, or learn new concepts from a seemingly infinite space of possibilities [LUT17, Mar18, Fod75].

To examine the versatile humanlike capabilities of concept learning with a focus on systematic generalization, we take inspiration from arithmetic and introduce a new benchmark HINT, Handwritten arithmetic with INTegers. The task of HINT is intuitive and straightforward: Machines take as input images of handwritten expressions and predict the final results of expressions, restricted in the integer space. The task of HINT is also challenging: Concepts in HINT, including digits and operators, are learned in a weakly-supervised manner. Using final results as the only supervision, machines are tasked to learn the three-level meanings simultaneously—perception, syntax, and semantics of these concepts—to correctly predict the results. Since there is no supervision on any intermediate values or representations, the three-level meanings are presumably intertwined during learning. To provide a holistic and rigorous test on whether learning machines can generalize the learned concepts, we introduce a carefully designed evaluation scheme instead of using a typical i.i.d. test split. This new scheme includes five subsets, focusing on generalization capabilities (*i.e.*, interpolation and extrapolation) at different levels of meanings (*i.e.*, perception, syntax, and semantics).

We evaluate popular state-of-the-art deep learning methods, such as GRU [CGC14] and Transformer [VSP17], on HINT. Our experiment shows that such end-to-end neural networks' performance drops significantly on examples requiring interpolation and extrapolation, even though these models can very well fit the training set. This finding echoes the long-standing arguments against connectionist models, which are believed to lack systematic generalization prevailing in human cognition [LB18, FP88].

Inspired by the superb generalization capability demonstrated in symbolic systems with combinatorial structure [FP88] and recent advances in neural-symbolic integration [LHH20a, MGK19, YWG18, MDK18], we propose an ANS system to approach the HINT challenge. The proposed ANS system integrates the learning of perception, syntax, and semantics in a principled framework; see an illustration in Figure 7.3. Specifically, we first utilize ResNet-18 [HZR16] as a perception module to translate a handwritten expression into a symbolic sequence. This symbolic sequence is then parsed by a transition-based neural dependency parser [CM14], which encodes the syntax of concepts. Finally, we adopt *functional programs* to realize the semantic meaning of concepts, thus view learning semantics as program induction [EWN20].

It is infeasible to perform an end-to-end optimization for our model since syntactic parsing and semantic reasoning are non-differentiable. Inspired by prior arts on abductive learning [LHH20a, Zho19, DXY19], we derive a novel *deduction-abduction* strategy to coordinate the learning of different modules. Specifically, during learning, the system first performs greedy deduction over these modules to propose an initial, rough solution, which is likely to produce a wrong result. A one-step abduction over perception, syntax, and semantics is then applied in a top-down manner to search the initial solution's neighborhood, which updates the solution to explain the ground-truth result better. This revised solution provides *pseudo* supervision on the intermediate values and representations, which are then used to train each module individually.

Evaluated on Hint, ANS can successfully learn the three-level meanings of concepts with weak supervision, obtaining an overall accuracy of 72% and outperforming end-to-end neural methods by nearly 33 percents. A detailed analysis shows that the strong generalization of ANS relies on the learned *symbol system* [FP88], which facilitates the extrapolation on syntax and semantics in a symbolic manner. A preliminary study of few-shot learning further demonstrates that ANS can quickly learn new concepts with limited examples, obtaining an average accuracy of 62% on four new concepts with a hundred training examples.

7.2 Related Work

Three Levels of Concept Learning The surge of deep neural networks [LBH15] in the last decade has significantly advanced the accuracy of **perception learning** from raw signals across multiple modalities, such as image classification from image pixels [HZR16, KSH12] and automatic speech recognition from audio waveforms [PCZ19, HDY12, GMH13].

The goal of **syntax analysis** is to understand the compositional and recursive structures in various tasks, such as natural language parsing [CM14, KK18], image and video parsing [TCY05, ZM07, ZZ11, GSS09, QJZ18, QJH20, JCH20], and scene understanding [HQZ18, HQX18, QZH18, JQZ18, CHY19, YLF20]. There exist two major structural types: constituency structures [KK18] and dependency structures [CM14]. Constituency structures use phrase structure grammar to organize input tokens into nested constituents, whereas dependency structures show which tokens depend on which other tokens.

Semantics of concepts essentially describe its causal effect. There are two primary semantic representations in symbolic reasoning. The first is *logic* [Llo12, MDK18], which regards the semantic learning as inductive logic programming [MD94, EG18]—a general framework to induce first-order logic theory from examples.

The other representation is *program*, which treats the semantic learning as inductive program synthesis [KKT15, LST15, BGB17a, DUB17, ERS18, EMS18]. Recently, [EWN20] release a neural-guided program induction system, *DreamCoder*, which can efficiently discover interpretable, reusable, and generalizable knowledge across a wide range of domains.

However, aforementioned literature tackles *only one or two levels* of concept learning and usually requires *direct* supervision on model outputs.

In contrast, in this work we offer a more holistic perspective that addresses all three levels of concept learning, *i.e.*, perception, syntax, and semantics, taking one step closer to realize a versatile mechanism of concept learning under weak supervision.

Systematic Generalization The central question in systematic generalization is: How well can a learning agent perform in unseen scenarios given limited exposure to the underlying configurations [Gre93]?

This question is also connected to the Language of Thought Hypothesis [Fod75]: The systematicity, productivity, and inferential coherence characterize compositional generalization of concepts [LST15]. As a prevailing property of human cognition, systematicity poses a central argument against connectionist models [FP88]. Recently, there have been several works to explore the systematic generalization of deep neural networks in different tasks

[LB18, BMN18, KSS19, GLB19, XMY21].

By going beyond traditional i.i.d. train/test split, the proposed HINT benchmark well-captures the characteristics of systematic generalization across different aspects of concepts w.r.t. perception, syntax, and semantics.

Neural-Symbolic Integration Researchers have proposed to combine statistical learning and symbolic reasoning, with pioneer efforts devoted to different directions, including representation learning and reasoning [Sun94, GLG08, MDK18], abductive learning [LHH20a, DXY19, Zho19], knowledge abstraction [HOT06, BGH09], *etc.*. There also have been recent works on the application of neural-symbolic methods, such as neural-symbolic visual reasoning and program synthesis [YWG18, MGK19, LHH20b, PMS16], semantic parsing [LBL16b, YZH18], and math word problems [LC20, LSR20]. Current neural-symbolic approaches often require a perfect domain-specific language, including both the syntax and semantics of the targeted domain. In comparison, the proposed model relaxes such a strict requirement and enables the learning of syntax and semantics.

7.3 Benchmark

Task Definition The task of HINT is intuitive and straightforward: It is tasked to predict the final results of handwritten arithmetic expressions in a weakly-supervised manner. Only the final results are given as supervision; all intermediate values and representations are latent, including symbolic expressions, parse trees, and execution traces.

Data Generation The data generation process follows three steps; see Figure 7.2 for an illustration. First, we extract handwritten images from CROHME[1] to obtain primitive concepts, including digits $0 \sim 9$, operators $+, -, \times, \div$, and parentheses $(,)$. Second, we randomly sample *prefix* expressions and convert them to *infix* expressions with necessary parentheses based on the operator precedence; we only allow single-digit numbers in expressions. These

[1] `https://www.cs.rit.edu/~crohme2019/`

symbolic expressions are fed into a solver to calculate the final results. Third, we randomly sample handwritten images for symbols in an expression and concatenate them to construct final handwritten expressions. We only keep the handwritten expressions as input and the corresponding final results as supervision; all intermediate results are discarded.

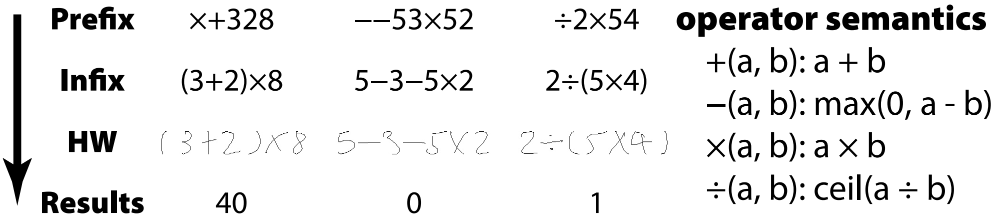


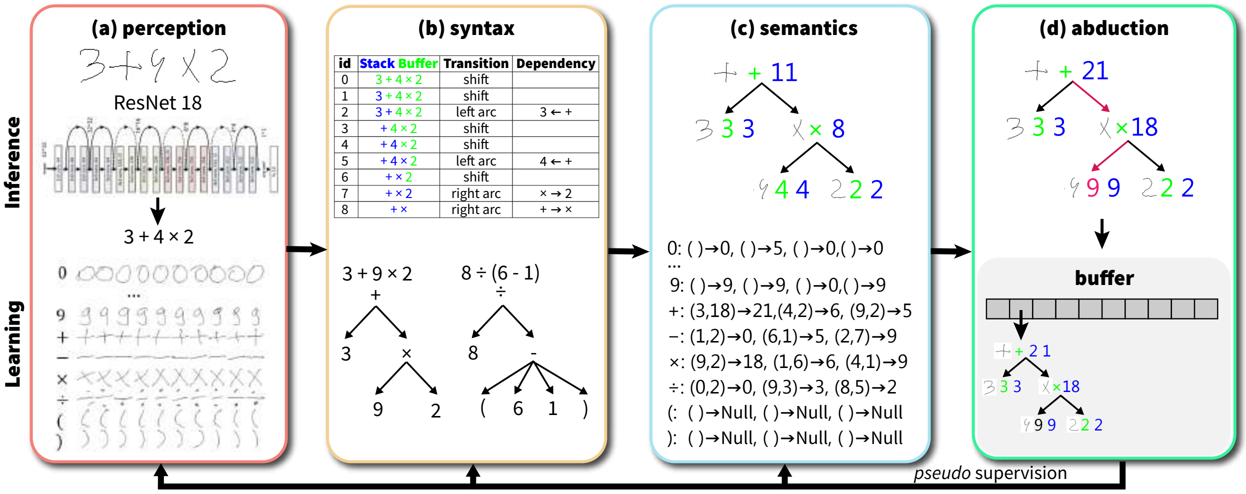
Figure 7.2: Illustrations of the data generation pipeline.

Train and Evaluation To rigorously evaluate how well the learned concepts are systematically generalized, we replace the typical i.i.d. train/test split with a carefully designed evaluation scheme: (i) all handwritten images in the test set are unseen in training, (ii) at most 1,000 samples are generated for each number of operators in expressions, (iii) limit the maximum number of operators to 10 and the maximum values to 100 in the training set:

$$D_{train} \subset \mathcal{D}_{train} = \{(x, y) : |x| \leqslant 10, \max(v) \leqslant 100\}, \tag{7.1}$$

where x is the handwritten expression, $|x|$ its number of operators, y the final result, and v all the intermediate values generated when calculating the final result.

We carefully devise the test set to evaluate different generalization capabilities (*i.e.*, interpolation and extrapolation) on different levels of meanings (*i.e.*, perception, syntax and



id	Stack Buffer	Transition	Dependency
0	3 + 4 × 2	shift	
1	3 + 4 × 2	shift	
2	3 + 4 × 2	left arc	3 ← +
3	+ 4 × 2	shift	
4	+ 4 × 2	shift	
5	+ 4 × 2	left arc	4 ← +
6	+ × 2	shift	
7	+ × 2	right arc	× → 2
8	+ ×	right arc	+ → ×

0: ( )→0, ( )→5, ( )→0,( )→0
...
9: ( )→9, ( )→9, ( )→0,( )→9
+: (3,18)→21,(4,2)→6, (9,2)→5
−: (1,2)→0, (6,1)→5, (2,7)→9
×: (9,2)→18, (1,6)→6, (4,1)→9
÷: (0,2)→0, (9,3)→3, (8,5)→2
(: ( )→Null, ( )→Null, ( )→Null
): ( )→Null, ( )→Null, ( )→Null

Figure 7.3: The Arithmetic Neural-Symbolic model (ANS). ANS consists of three modules for perception, syntax, and semantics. During inference, the model performs greedy deduction over three modules and directly proposes a solution. During learning, the proposed solution is further revised by performing abduction based on the ground-truth supervision. The updated solution is stored in a buffer, providing *pseudo* supervisions to train three modules individually. Each node in the solution tree is an (image, symbol, value) triplet.

semantics). Specifically, the test set is composed of five subsets, formally defined as:

$$
\begin{aligned}
D_{test} &= D_{test}^{(1)} \cup D_{test}^{(2)} \cup D_{test}^{(3)} \cup D_{test}^{(4)} \cup D_{test}^{(5)}, \text{where} \\
D_{test}^{(1)} &= D_{train}, \\
D_{test}^{(2)} &\subset \mathcal{D}_{train} \backslash D_{train}, \\
D_{test}^{(3)} &\subset \{(x, y) : |x| \leqslant 10, \max(v) > 100\}, \\
D_{test}^{(4)} &\subset \{(x, y) : |x| > 10, \max(v) \leqslant 100\}, \\
D_{test}^{(5)} &\subset \{(x, y) : |x| > 10, \max(v) > 100\}.
\end{aligned}
\tag{7.2}
$$

All above subsets requires generalization on perception of learned concepts. $D_{test}^{(1)}$ requires no generalization on either syntax or semantics, $D_{test}^{(2)}$ requires interpolation on both syntax and semantics, $D_{test}^{(3)}$ requires interpolation on syntax and extrapolation on semantics, $D_{test}^{(4)}$ requires extrapolation on syntax and interpolation on semantics, and $D_{test}^{(5)}$ requires extrapolation on both syntax and semantics.

In total, the training and test set includes 11,170 and 48,910 samples, respectively. Subsets in the test set are balanced to be 23%, 23%, 22%, 16%, and 16%.

7.4 A Neural-Symbolic Approach

Below we first describe a general framework from a probabilistic perspective for learning the HINT task as a neural-symbolic approach. This general framework implies a *symbol system* with combinatorial syntactic and semantic structures, initially introduced by [FP88], as a feasible representation of the human mind. Such a symbol system provides a principled integration of perception, syntax, and semantics. Guided by this general framework, we next provide a concrete instantiation of such a neural-symbolic system and introduce a novel deduction-abduction strategy to learn it with weak supervision; see Figure 7.3 for overview.

A General Framework Given a neural-symbolic system, let $x \in \Omega_x$ denote the input (images of handwritten expression in the HINT dataset), $s \in \Omega_s$ the symbolic expression, $pt \in \Omega_t$ the parse tree of the symbolic expression, $et \in \Omega_e$ the execution trace, and $y \in \Omega_y$ the output.

During learning, (x, y) are observed but (s, pt, et) are latent. The likelihood of the observation (x, y) marginalized over (s, pt, et) can be decomposed as:

$$\begin{aligned} p(y|x;\Theta) &= \sum_{s,pt,et} p(s,pt,et,y|x;\Theta) \\ &= \sum_{s,pt,et} p(s|x;\theta_p)p(pt|s;\theta_s)p(et|pt;\theta_l)p(y|et), \end{aligned} \tag{7.3}$$

where (i) $s|x$ denotes the process of perceiving symbols from raw signals, guided by the perceptual model θ_p of learned concepts; (ii) $pt|s$ denotes the process of parsing the symbolic expression into a parse tree, guided by the syntactic model θ_s; (iii) $et|pt$ denotes the process of reasoning over the parse tree, guided by the semantic model θ_l; and (iv) $y|et$ is a deterministic process: If the final output of et equals to y, $p(y|et) = 1$, otherwise 0.

From a maximum likelihood prospective, the learning objective is to maximize the observed-data log likelihood $L(x, y) = \log p(y|x)$. Take the derivative of L w.r.t. $\theta_p, \theta_s, \theta_l$, we have:

$$\begin{aligned}
\nabla_{\theta_p} L(x, y) &= \mathbb{E}_{s,pt,et \sim p(s,pt,et|x,y)}[\nabla_{\theta_p} \log p(s|x; \theta_p)], \\
\nabla_{\theta_s} L(x, y) &= \mathbb{E}_{s,pt,et \sim p(s,pt,et|x,y)}[\nabla_{\theta_s} \log p(pt|s; \theta_s)], \\
\nabla_{\theta_l} L(x, y) &= \mathbb{E}_{s,pt,et \sim p(s,pt,et|x,y)}[\nabla_{\theta_l} \log p(et|pt; \theta_l)],
\end{aligned} \tag{7.4}$$

where $p(s, pt, et|x, y)$ is the posterior distribution of (s, pt, et) given (x, y). Since $p(y|et)$ can only be 0 or 1, $p(s, pt, et|x, y)$ can be rewritten as:

$$\begin{aligned}
p(s, pt, et|x, y) &= \frac{p(s, pt, et, y|x; \Theta)}{\sum_{s',pt',et'} p(s', pt', et', y|x; \Theta)} \\
&= \begin{cases} 0, \quad \text{for} \quad s, pt, et \notin Q \\ \frac{p(s,pt,et|x;\Theta)}{\sum_{s',pt',et' \in Q} p(s',pt',et'|x;\Theta)}, \quad \text{for } s, pt, et \in Q \end{cases}
\end{aligned} \tag{7.5}$$

where $Q = \{(s, pt, et) : p(y|et) = 1, s \in \Omega_s, pt \in \Omega_t, et \in \Omega_e\}$ is the set of (s, pt, et) that generates y. Usually, Q is a very small subset of the entire space of (s, pt, et), *i.e.*, $Q \subseteq \Omega_s \times \Omega_t \times \Omega_e$, where × denotes the Cartesian product.

Since taking expectation w.r.t. this posterior distribution is intractable, we use Monte Carlo sampling to approximate it. Therefore, the learning procedure for an example (x, y) can be depicted as following:

1. sample $\hat{s}, \hat{pt}, \hat{et} \sim p(s, pt, et|x, y)$;

2. use $(x, \hat{s})$ to update the perception model (θ_p);

3. use $(\hat{s}, \hat{pt})$ to update the parsing model (θ_s);

4. use $(\hat{pt}, \hat{et})$ to update the reasoning model (θ_l).

Instantiation: Arithmetic Neural-Symbolic (ANS) The general framework of the desired neural-symbolic system described above is agnostic to the choice of functions and algorithms. Below we delineate a learnable implementation, named ANS, capable of learning generalizable concepts in arithmetic on the proposed Hint dataset.

7.4.0.1 Perception: Neural Network (NN)

The role of the perception module is to map a handwritten expression x into a symbolic expression s. Since disentangling visual symbols from handwritten expressions is trivial in this domain , we assume the input as a sequence of handwritten images, where each image contains one symbol. We adopt a standard ResNet-18 [HZR16] as the perception module to map each handwritten image into a probability distribution over the concept space Σ. Formally,

$$p(s|x;\theta_p) = \prod_i p(w_i|x_i;\theta_p) = \prod_i \texttt{softmax}(\phi(w_i, x_i;\theta_p)), \tag{7.6}$$

where $\phi(s,x;\theta_p)$ is a scoring function parameterized by a NN with parameters θ_p. Since learning such an NN from scratch is prohibitively challenging, the ResNet-18 is pre-trained unsupervisedly [VVG20] on unlabeled handwritten images.

7.4.0.2 Syntax: Dependency Parsing

To parse the symbolic sequence into a parse tree, we adopt a greedy transition-based neural dependency parser [CM14], commonly used for parsing natural language sentences. The transition-based dependency parser relies on a state machine that defines the possible transitions to parse the input sequence into a dependency tree; see panel (b) of Figure 7.3. The learning process induces a model to predict the next transition in the state machine based on the transition history. The parsing process constructs the optimal sequence of transitions for the input sequence. A dependency parser for arithmetic expressions is essentially approximating the Shunting-yard algorithm.

In our parser, a *state* $c = (\alpha, \beta, A)$ consists of a *stack* α, a *buffer* β, and a set of *dependency arcs* A. The initial state for a sequence $s = w_0 w_1 ... w_n$ is $\alpha = [\texttt{Root}], \beta = [w_0 w_1 ... w_n], A = \varnothing$. A state is regarded as terminal if the buffer is empty *and* the stack only contains the node `Root`. The parse tree can be derived from the dependency arcs A. Let α_i denote the i-th top element on the stack, and β_i the i-th element on the buffer. The parser defines three types of transitions between states:

- Left-Arc: add an arc $\alpha_1 \rightarrow \alpha_2$ to A and remove α_2 from the stack α. Precondition: $|\alpha| \geqslant 2$.

- Right-Arc: add an arc $\alpha_2 \rightarrow \alpha_1$ to A and remove α_1 from the stack α. Precondition: $|\alpha| \geqslant 2$.

- Shift: move β_1 from the buffer β to the stack α. Precondition: $|\beta| \geqslant 1$.

The goal of the parser is to predict a transition sequence from an initial state to a terminal state. As the parser is greedy, it attempts to predict one transition from $\mathcal{T} = \{\text{Left-Arc}, \text{Right-Arc}, \text{Shift}\}$ at a time, based on the current state $c = (\alpha, \beta, A)$. The features for a state c contains following three elements: (i) The top three words on the stack and buffer: $\alpha_i, \beta_i, i = 1, 2, 3$; (ii) The first and second leftmost/rightmost children of the top two words on the stack: $lc_1(\alpha_i), rc_1(\alpha_i), lc_2(\alpha_i), rc_2(\alpha_i), i = 1, 2$; (iii) The leftmost of leftmost/rightmost of rightmost children of the top two words on the stack: $lc_1(lc_1(\alpha_i)), rc_1(rc_1(\alpha_i)), i = 1, 2$. We use a special `Null` token for non-existent elements. Each element in the state representation is embedded to a d-dimensional vector $e \in R^d$, and the full embedding matrix is denoted as $E \in R^{|\Sigma| \times d}$, where Σ is the concept space. The embedding vectors for all elements in the state are concatenated as its representation: $c = [e_1 \ e_2 ... e_n] \in R^{nd}$. Given the state representation, we adopt a two-layer feed-forward NN to predict a transition.

7.4.0.3 Semantics: Program Synthesis

Inspired by recent advances in program synthesis [EWN20, BGB17a, DUB17], we adopt *functional programs* to represent the semantics of concepts and view learning as program induction. The semantics of a concept is treated as a function, mapping certain inputs to an output. Learning semantics is equivalent to searching for a program that approximates this unknown function. Compare to purely statistical approaches, symbolic programs exhibit better generalizability and interpretability, and the learning is also more sample-efficient.

To learn semantics as programs, we start from DreamCoder [EWN20], a machine learning system that can efficiently synthesize interpretable, reusable, and generalizable programs across a wide range of domains. DreamCoder embodies a wake-sleep Bayesian program induction approach to progressively learn multiple tasks in a domain, given a set of primitives and input-out pairs for each task. For arithmetic reasoning, the Peano axioms [Pea89] define four primitives: (1) `0`; (2) `inc`: $a \rightarrow a + 1$; (3) `dec`: $a \rightarrow \max(0, a - 1)$; (4) `if`: $(a, b, c) \rightarrow b$ (if a is 0) or c (else). Any arithmetic function can be provably composed from these four primitives. This set of primitives is augmented with a recursion primitive, `Y`-combinator (*a.k.a.*, fixed-point combinator). The `Y`-combinator enables the derivation of recursive functions and is the crux of extrapolating to large numbers.

The semantics of concepts in Hint, including digits, operators, and parentheses, are all represented as programs composed from these primitives $L = \{\mathtt{0}, \mathtt{inc}, \mathtt{dec}, \mathtt{if}, \mathtt{Y}\}$. During inference, these programs are used for reasoning to obtain the results. The learning for a concept c is to find a program ρ_c to maximize the following objective:

$$\rho_c = \arg\max_{\rho} p(\rho|D_c, L) \propto (D_c|\rho)\ p(\rho|L), \tag{7.7}$$

where D_c denotes the input-output pairs of the concept c for program induction, $p(D_c|\rho)$ the likelihood of the program ρ explaining D_c, and $p(\rho|L)$ the prior of ρ under the library L, which defines a generative model over programs. The maximization in Equation (7.7) is achieved by a stochastic search process guided by a neural network, which is trained to approximate the posterior distribution $p(\rho|D_c, L)$.

7.4.0.4 Learning by Deduction-Abduction

In Section 7.4, we derive a general learning procedure for such a neural-symbolic system. The key is to perform efficient sampling from the posterior distribution $p(s, pt, et|x, y)$. In short, we generalize the back-search algorithm in [LHH20a] to a *deduction-abduction* strategy to enable efficient sampling from the posterior distribution of perception, syntax, and semantics.

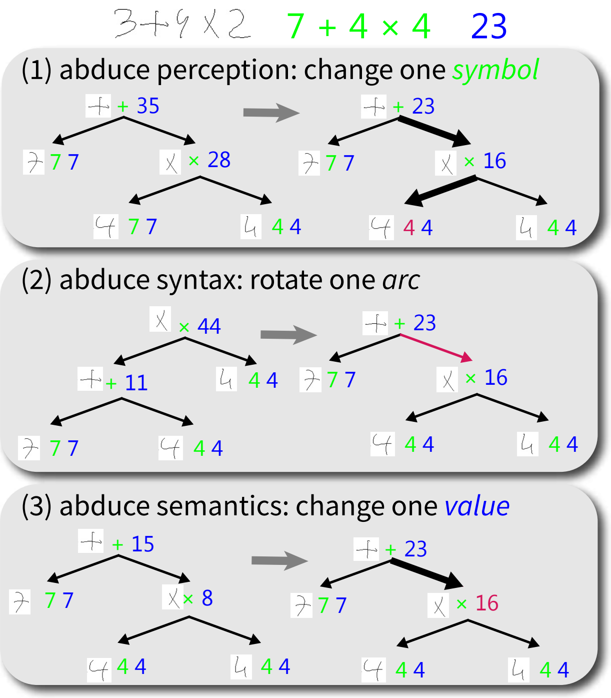


Figure 7.4: Abduction over perception, syntax, and semantics. Each node in the solution tree is a triplet of (image, symbol, value). Parts revised during abduction are highlighted in red.

Deduction For a given example (x, y), we first perform greedy deduction from x to obtain a candidate solution of a compound tree $ct = (x, \hat{s}, \hat{pt}, \hat{et})$. This process is likely to produce a wrong result, thus requiring a separate abduction process to further correct it, detailed below.

Abduction To find a revised solution ct^* that can reach the goal y, we search the neighbors of ct in a top-down manner by performing abduction over perception (s), syntax (pt), and semantics (et), as illustrated in Figure 7.4. Our abduction strategy generalizes the perception-only, one-step back-search algorithm described in [LHH20a] to all three levels. The SOLVE function and the priority used in the top-down search are similarly to the ones in [LHH20a]. The abduction can also be extended to multiple steps, but we only use one step for lower computation overhead.

The above deduction-abduction strategy likely behaves as a Metropolis-Hastings sampler for the posterior distribution [LHH20a].

7.5 Experiments and Results

Experimental Setup

Training Both the ResNet-18 and the dependency parser in the proposed ANS model are trained by an Adam optimizer [KB15] with a learning rate of 10^{-4} and a batch size of 512. The program synthesis module is adapted from DreamCoder [EWN20].

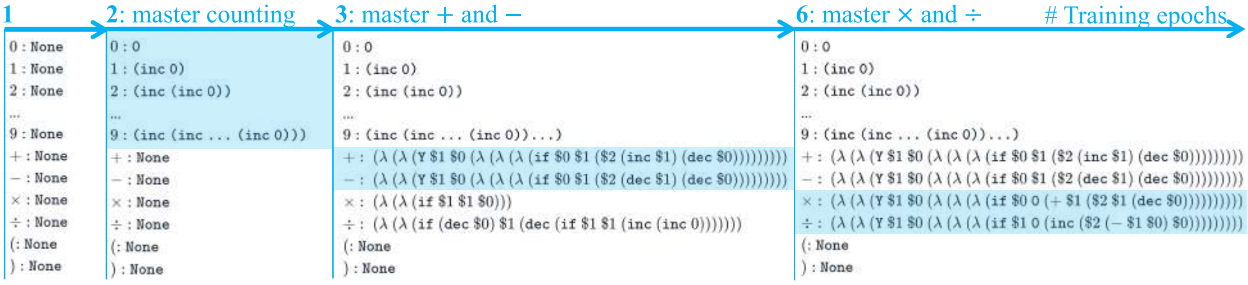
Evaluation Metric We evaluate the models with the accuracy of final results. Note that a predicted result is considered correct when it *exactly* equals to the ground-truth.

Baselines For end-to-end NN baselines, the task of HINT is formulated as a sequence-to-sequence problem: The input is an expression sequence, and the output is a sequence of digits, which is then converted to an integer as the predicted result. We test two popular seq2seq models: (1) BiGRU: the encoder is a bi-directional GRU [CGC14] with three layers, and the decoder is a one-layer GRU; (2) TRAN: a Transformer model [VSP17] with three encoder-layers, three decoder-layers, and four attention heads for each layer. Before being fed into these models, the handwritten expressions are processed by the same ResNet-18 used in ANS. We test models with varied numbers of layers and report ones with the best results. To speed up the convergence, we train all models with a simple curriculum from short expressions to long ones.

Neural-Symbolic v.s. End-to-End Neural Networks

We compare the performance of the proposed neural-symbolic model ANS with end-to-end neural baselines on HINT. As shown in Table 7.1, both BiGRU and TRAN obtain high accuracy on the test subset 1, which indicates that they can generalize over perception very well.

However, their performances drop significantly on the test subsets 2∼5, which require systematic generalization over syntax and semantics. Notably, their accuracy is less than 10% on test subsets 3 and 5 that involve larger numbers compared to the training set.



```
0 : None
1 : None
2 : None
...
9 : None
+ : None
− : None
× : None
÷ : None
( : None
) : None
```

```
0 : 0
1 : (inc 0)
2 : (inc (inc 0))
...
9 : (inc (inc ... (inc 0)))
+ : None
− : None
× : None
÷ : None
( : None
) : None
```

```
0 : 0
1 : (inc 0)
2 : (inc (inc 0))
...
9 : (inc (inc ... (inc 0))...)
+ : (λ (λ (Y $1 $0 (λ (λ (λ (if $0 $1 ($2 (inc $1) (dec $0)))))))))
− : (λ (λ (Y $1 $0 (λ (λ (λ (if $0 $1 ($2 (dec $1) (dec $0)))))))))
× : (λ (λ (if $1 $1 $0)))
÷ : (λ (λ (if (dec $0) $1 (dec (if $1 $1 (inc (inc 0)))))))
( : None
) : None
```

```
0 : 0
1 : (inc 0)
2 : (inc (inc 0))
...
9 : (inc (inc ... (inc 0))...)
+ : (λ (λ (Y $1 $0 (λ (λ (λ (if $0 $1 ($2 (inc $1) (dec $0)))))))))
− : (λ (λ (Y $1 $0 (λ (λ (λ (if $0 $1 ($2 (dec $1) (dec $0)))))))))
× : (λ (λ (Y $1 $0 (λ (λ (λ (if $0 0 (+ $1 ($2 $1 (dec $0))))))))))
÷ : (λ (λ (Y $1 $0 (λ (λ (λ (if $1 0 (inc ($2 (− $1 $0) $0)))))))))
( : None
) : None
```

Figure 7.5: **The evolution of semantics in ANS from initial primitives** {`0`,`inc`,`dec`,`if`,`Y`}. The programs representing the semantics of concepts are denoted by lambda calculus (a.k.a. λ-calculus) with De Bruijn indexing. Note that there might be different yet functionally-equivalent programs to represent the same semantics of concepts. Here, we only show one possibility for each concept.

This result indicates that the pure neural models do not learn the semantics of concepts in a generalizable way and fail to extrapolate to large numbers. In contrast, the proposed ANS model consistently outperforms BiGRU and TRAN by at least 30 absolute percent across all test subsets 2∼5. This superb performance demonstrates the strong systematic generalization of ANS, including both interpolation and extrapolation w.r.t. syntax and semantics.

Table 7.1: The performance comparison of ANS and end-to-end neural networks, *i.e.*, GRU (BiGRU) and Transformer (TRAN).

Input	**Model**	**Test Accuracy** (%)					
		Overall	**1**	**2**	**3**	**4**	**5**
Symbol (Embedding)	BiGRU	49.71	97.05	63.67	11.58	52.41	12.57
	TRAN	34.58	98.31	29.79	2.91	26.39	2.76
	ANS	**88.36**	**99.26**	**97.56**	**84.66**	**87.65**	**65.37**
Image (ResNet-18)	BiGRU	39.39	87.02	46.17	6.51	40.44	6.47
	TRAN	32.95	87.31	30.74	2.67	31.17	2.55
	ANS	**71.97**	**89.10**	**84.29**	**66.77**	**68.19**	**40.73**

How do models extrapolate? Among the generalization capability, we are particularly interested in extrapolation. Based on the experimental results, we firmly believe that the key is *recursion*. In ANS, the extrapolation on syntax is achieved by the transition system of the dependency parser, which recursively applies transition actions to parse arbitrarily long expressions. The extrapolation on semantics is realized by the recursion primitive, *i.e.*, `Y`-combinator. It allows programs to represent recursive functions, which can decompose large numbers into smaller ones by recursively invoking themselves. For BiGRU, although the recurrent structure in its hidden cells serves as a recursive prior on syntax, no such prior in its representation for semantics. This deficiency explains why BiGRU would achieve a decent accuracy (40.44%) on the test subset 3 (extrapolation only on syntax) but a much lower accuracy (6.51%) on the test subset 4 (extrapolation only on semantics). Taken together, these observations strongly imply that the recursive prior on task-specific representations is the crux of extrapolation, which is also in line with the recent analysis of Graph Neural Network, where it successfully extrapolates algorithmic tasks due to the *task-specific non-linearities* in the architecture or features [XLZ20a, XLZ20b].

Ablation Study

Table 7.2 shows an ablation study on the proposed ANS model. In general, providing the ground-truth meaning of concepts can ease the learning and lead to higher test accuracy. Among the three levels of concepts, perception is the hardest to learn since the handwriting images possess a large variance in terms of the visual appearance. The syntax and semantics are relatively easier to learn, since the recursive prior of the transition-based dependency parser and Y-combinator fits the task well.

Table 7.2: Ablation study on ANS. ✓indicates that the ground-truth labels are given during training. For each setting (row), we perform three experiments with different random seeds and report the results of the model with the highest *training* accuracy.

Training Setting			**Test Accuracy (%)**					
Per.	**Syn.**	**Sem.**	**Overall**	**1**	**2**	**3**	**4**	**5**
			71.97	89.10	84.29	66.77	68.19	40.73
		✓	86.44	94.53	91.62	89.58	78.22	71.18
	✓		80.14	92.51	90.16	71.32	84.27	56.27
✓			88.36	99.26	97.56	84.66	87.65	65.37
✓	✓		97.81	100.00	100.00	96.66	100.00	90.97
✓		✓	95.84	99.60	98.23	98.09	91.50	88.20
	✓	✓	88.93	94.30	92.19	90.06	82.99	80.88

Figure 7.5 illustrates the typical pattern of the evolution of semantics in ANS. This pattern is highly in accord with how children learn arithmetic in developmental psychology [CFF99]: The model first masters the semantics of digits as `counting`, then learns + and − as recursive counting, and finally it figures out how to define × and ÷ based on the learned

programs for + and −. Crucially, × and ÷ are impossible to be correctly learned before mastering + and −. The model is endowed with such an incremental learning capability since the program induction module allows the semantics of concepts to be built compositionally from those learned earlier [EWN20].

Few-shot Concept Learning

We further conduct a preliminary study of few-shot learning to demonstrate the ANS's potential in learning new concepts with limited examples. As shown in Table 7.3, we define four new concepts with common semantics. Their visual appearances are denoted by four unseen handwritten symbols $\{\alpha, \beta, \gamma, \phi\}$, and their syntax is decided by their precedence (*i.e.*, 1 is for $\{+, -\}$ and 2 is for $\{\times, \div\}$). We randomly sample a hundred examples from short to long expressions for training each new concept and fine-tune the ANS model on the new training data.

Table 7.3 shows the test accuracy for each new concept. The proposed ANS model obtains a decent performance with an average overall accuracy of 61.92%. Concepts with more complex semantics ($\{\gamma, \phi\}$) are generally harder to learn than those with simpler semantics ($\{\alpha, \beta\}$).

Table 7.3: Few-shot concept learning with ANS.

Per.	**Syn.**	**Sem.**	**Test Accuracy** (%)					
			Overall	**1**	**2**	**3**	**4**	**5**
α	1	$\max(x, y)$	64.08	70.91	81.98	70.79	50.56	40.66
β	1	$\min(x, y)$	72.45	85.45	83.93	81.82	65.91	40.22
γ	2	$(x + y)/2$	56.73	76.36	70.09	61.80	41.94	27.47
ϕ	2	$xy - (x + y)$	54.40	76.36	68.81	41.35	56.04	22.09
avg.	-	-	61.92	77.27	76.20	63.94	53.61	32.61

7.6 Discussion: Contributions and Limitations

In this work, we take inspiration from how humans learn arithmetic and present a new challenge for the machine learning community, Hint, which serves as a minimal yet complete benchmark towards studying systematic generalization of concepts w.r.t. perception, syntax, and semantics. Additionally, we propose a neural-symbolic system, Arithmetic Neural-

Symbolic (ANS), to approach this challenge. ANS integrates recent efforts from the disciplines of neural networks, grammar parsing, and program synthesis and successfully learns the three-level meanings of concepts with weak supervision.

In this work our discussions focus on a relatively simple domain with context-free semantics. One potential future work is to extend our observations to more realistic and complex domains, such as visual reasoning [JHM17a, HM19] and question answering [RZL16]. We may consider to inject contexts into the semantics of concepts and capture their stochastic nature with probabilistic programs [Gha15, CGH17, GXG18, BCJ19, HBM20].

CHAPTER 8

Conclusion

This dissertation introduces our contributions in building up and integrating perception, interaction, learning, and reasoning modules to solve the human-like holistic 3D scene understanding problem. We are still missing several key dimensions in this dissertation, such as studies about how brains work and humans behave, how to do long-horizon planning, and how to actively perceive and interact with the environments. We would like to investigate these problems in the future.

To sum up, it requires interdisciplinary expertise across computer vision, natural language understanding, computer graphics, machine learning, robotics, neuroscience, and cognitive science to build up a human-like intelligence system, which is beyond the Statistics and Computer Science themselves. Machine's capability in solving the general tasks, *i.e.*, the core of human intelligence, would be easily trapped by training on the manually-created tasks and data in a specific field, losing the generalizability to other tasks and domains. Therefore, we believe it is necessary to solve the interdisciplinary AI problems involved with multiple interacting modalities and fields. By teaching machines to solve the holistic tasks with a designed curriculum using a unified framework, it is hopeful to gradually obtain the core commonsense knowledge for human intelligence and improve the generalization capability across scenes and skills.

In the end, we summarize several fundamental and promising research directions.

- Perception. Most current perception models have reached performance bottleneck on single-modal input and are not generalizable to novel scenarios and tasks. Therefore, how to efficiently learn from multi-modal data becomes the next critical challenge for the community. Moreover, we still lack a general-purpose multi-modal perception benchmark for evaluating the performance and generalization capability in multi-modal perception.

- Interaction. In order to help machines understand humans, we should study more fine-grained level human-object interactions, for example, the 4D interactions between our body shape and object, as well as human-human interactions, such as social interactions under various situations. Besides, goal-driven active interaction with the 3D scenes and long-horizon planning also worth exploration. They require policies for interacting and planning from the first-person view. Recent embodied AI platforms [XZH18, SKM19] create several simulated environments for evaluating these capabilities.

- Learning and Reasoning. Humans excel at learning efficiently with less or without supervision, abstracting symbolic and hierarchical representations from observations, and generalizing concepts and knowledge to novel domains and environments. Therefore, learning efficient computational model [GXH19, ZGH21], building a powerful self-supervised representation learning model [DCL18], learning symbolic, hierarchical, compositional representation from data [HLZ21], and developing concepts for achieving systematic generalization [LHH21] are the fundamental problems we hope to address in the next decade.

- Interdisciplinary AI Research. It requires enormous efforts from neuroscience and cognitive science for interpreting and understanding human intelligence [ZGF20, GLG20]. These efforts serve as primary inspirations for AI researchers in different fields (*e.g.*, computer vision, natural language processing, computer graphics, robotics). With rapidly developed computational power and data, the new era has begun for us to integrate our expertise in various areas, tackle challenging interdisciplinary tasks, and build machines that can truly think and behave like humans.

REFERENCES

[AAL15] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. "Vqa: Visual question answering." In *International Conference on Computer Vision (ICCV)*, 2015.

[ABA16] Jean-Baptiste Alayrac, Piotr Bojanowski, Nishant Agrawal, Josef Sivic, Ivan Laptev, and Simon Lacoste-Julien. "Unsupervised learning from narrated instruction videos." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[AC76] Michael Argyle and Mark Cook. *Gaze and mutual gaze.* Cambridge U Press, 1976.

[AHB18] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. "Bottom-up and top-down attention for image captioning and visual question answering." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[AKL17] Jacob Andreas, Dan Klein, and Sergey Levine. "Modular multitask reinforcement learning with policy sketches." In *International Conference on Machine Learning (ICML)*. JMLR. org, 2017.

[ALS19] Rishabh Agarwal, Chen Liang, Dale Schuurmans, and Mohammad Norouzi. "Learning to generalize from sparse and underspecified rewards." *arXiv preprint arXiv:1902.07198*, 2019.

[AME14] Mathieu Aubry, Daniel Maturana, Alexei A Efros, Bryan C Russell, and Josef Sivic. "Seeing 3d chairs: exemplar part-based 2d-3d alignment using a large dataset of cad models." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[AMT15] Sean Andrist, Bilge Mutlu, and Adriana Tapus. "Look like me: matching robot personality via gaze to increase motivation." In *CHI*, 2015.

[ARD15] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. "Neural Module Networks." *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 39–48, 2015.

[ARD16] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. "Learning to Compose Neural Networks for Question Answering." In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016.

[AS14] Henny Admoni and Brian Scassellati. "Data-driven model of nonverbal behavior for socially assistive human-robot interactions." In *ICMI*, 2014.

[AS17] Henny Admoni and Brian Scassellati. "Social Eye Gaze in Human-robot Interaction: A Review." *JHRI*, **6**(1), 2017.

[ATG14] Sean Andrist, Xiang Zhi Tan, Michael Gleicher, and Bilge Mutlu. "Conversational gaze aversion for humanlike robots." In *HRI*, 2014.

[AWT18] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. "Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3674–3683, 2018.

[BA81] R Darrell Bock and Murray Aitkin. "Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm." *Psychometrika*, 1981.

[Bai04] Renée Baillargeon. "Infants' physical world." *Current directions in psychological science*, **13**(3):89–94, 2004.

[Bak01] Frank B Baker. *The basics of item response theory.* ERIC, 2001.

[BB01] Dare A Baldwin and Jodie A Baird. "Discerning intentions in dynamic human action." *Trends in Cognitive Sciences*, **5**(4):171–178, 2001.

[BBC19] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Debiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. "Dota 2 with Large Scale Deep Reinforcement Learning." *arXiv preprint arXiv:1912.06680*, 2019.

[BBS01] Dare A Baldwin, Jodie A Baird, Megan M Saylor, and M Angela Clark. "Infants parse dynamic action." *Child development*, **72**(3):708–717, 2001.

[BCB15] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate." In *International Conference on Learning Representations (ICLR)*, 2015.

[BCJ18] Eli Bingham, Jonathan P. Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D. Goodman. "Pyro: Deep Universal Probabilistic Programming." *Journal of Machine Learning Research*, 2018.

[BCJ19] Eli Bingham, Jonathan P Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D Goodman. "Pyro: Deep universal probabilistic programming." *Journal of Machine Learning Research*, **20**(1):973–978, 2019.

[BGB17a] Matej Balog, Alexander L Gaunt, Marc Brockschmidt, Sebastian Nowozin, and Daniel Tarlow. "Deepcoder: Learning to write programs." In *International Conference on Learning Representations (ICLR)*, 2017.

[BGB17b] Tarek R Besold, Artur d'Avila Garcez, Sebastian Bader, Howard Bowman, Pedro Domingos, Pascal Hitzler, Kai-Uwe Kühnberger, Luis C Lamb, Daniel Lowd, Priscila Machado Vieira Lima, et al. "Neural-symbolic learning and reasoning: A survey and interpretation." *arXiv preprint arXiv:1711.03902*, 2017.

[BGF16] Judee K. Burgoon, Laura K. Guerrero, and Kory Floyd. *Nonverbal communication.* Routledge, 2016.

[BGH09] Sebastian Bader, Artur S d'Avila Garcez, and P Hitzler. "Extracting propositional rules from feed-forward neural networks by means of binary decision diagrams." In *Proceedings of the 5th International Workshop on Neural-Symbolic Learning and Reasoning, NeSy*, volume 9, pp. 22–27. Citeseer, 2009.

[BHD18] Rudy Bunel, Matthew Hausknecht, Jacob Devlin, Rishabh Singh, and Pushmeet Kohli. "Leveraging grammar and reinforcement learning for neural program synthesis." *arXiv preprint arXiv:1805.04276*, 2018.

[BI13] Ali Borji and Laurent Itti. "State-of-the-art in visual attention modeling." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **35**(1):185–207, 2013.

[Bin71] I Binford. "Visual perception by computer." In *IEEE Conference of Systems and Control*, 1971.

[BK04] Frank B Baker and Seock-Ho Kim. *Item response theory: Parameter estimation techniques.* CRC Press, 2004.

[BKC17] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. "Segnet: A deep convolutional encoder-decoder architecture for scene segmentation." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **39**(12):2481–2495, 2017.

[BKM20] Bowen Baker, Ingmar Kanitscheider, Todor Markov, Yi Wu, Glenn Powell, Bob McGrew, and Igor Mordatch. "Emergent tool use from multi-agent autocurricula." In *International Conference on Learning Representations (ICLR)*, 2020.

[BLB14] Piotr Bojanowski, Rémi Lajugie, Francis Bach, Ivan Laptev, Jean Ponce, Cordelia Schmid, and Josef Sivic. "Weakly supervised action labeling in videos under ordering constraints." In *European Conference on Computer Vision (ECCV)*, 2014.

[BLC09] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. "Curriculum learning." In *International Conference on Machine Learning (ICML)*, 2009.

[BM05] Rechele Brooks and Andrew N. Meltzoff. "The development of gaze following and its relation to language." *Developmental Science*, **8**(6):535–543, 2005.

[BMN18] Dzmitry Bahdanau, Shikhar Murty, Michael Noukhovitch, Thien Huu Nguyen, Harm de Vries, and Aaron Courville. "Systematic generalization: what is required and can it be learned?" In *International Conference on Learning Representations (ICLR)*, 2018.

[BPL12] Jean-David Boucher, Ugo Pattacini, Amelie Lelong, Gerard Bailly, Frederic Elisei, Sascha Fagel, Peter F. Dominey, and Jocelyne Ventre-Dominey. "I reach faster when I see you look: gaze effects in human-human and human-robot face-to-face cooperation." *Frontiers in Neurorobotics*, **6**:3, 2012.

[BPL16] Peter W. Battaglia, Razvan Pascanu, Matthew Lai, Danilo Rezende, and Koray Kavukcuoglu. "Interaction networks for learning about objects, relations and physics." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[BRG16] Aayush Bansal, Bryan Russell, and Abhinav Gupta. "Marr revisited: 2d-3d alignment via surface normal prediction." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[BRL03] Michael Bosse, Richard Rikoski, John Leonard, and Seth Teller. "Vanishing points and three-dimensional lines from omni-directional video." *The Visual Computer*, 2003.

[Bro81] Rodney A Brooks. "Symbolic reasoning among 3-D models and 2-D images." *Artificial Intelligence*, **17**(1-3):285–348, 1981.

[Bro85] Donald Broadbent. *A question of levels: Comment on McClelland and Rumelhart.* American Psychological Association, 1985.

[BSC17] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S. Davis. "Soft-NMS – Improving Object Detection With One Line of Code." In *International Conference on Computer Vision (ICCV)*, 2017.

[BSW85] Renee Baillargeon, Elizabeth S Spelke, and Stanley Wasserman. "Object permanence in five-month-old infants." *Cognition*, **20**(3):191–208, 1985.

[BT81] Harry G Barrow and Jay M Tenenbaum. "Interpreting line drawings as three-dimensional surfaces." *Artificial Intelligence*, **17**(1-3):75–116, 1981.

[BZS14] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann Lecun. "Spectral networks and locally connected networks on graphs." In *International Conference on Learning Representations (ICLR)*, 2014.

[CCP13] Wongun Choi, Yu-Wei Chao, Caroline Pantofaru, and Silvio Savarese. "Understanding indoor scenes using 3d geometric phrases." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[CEG15] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. "Activitynet: A large-scale video benchmark for human activity understanding." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[CFF99] Thomas P Carpenter, Elizabeth Fennema, M Loef Franke, Linda Levi, and Susan B Empson. "Children's mathematics." *Cognitively Guided*, 1999.

[CFG15] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. "Shapenet: An information-rich 3d model repository." *arXiv preprint arXiv:1512.03012*, 2015.

[CG07] Gergely Csibra and György Gergely. "'Obsessed with goals': Functions and mechanisms of teleological interpretation of actions in humans." *Acta psychologica*, 2007.

[CGC14] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. "Empirical evaluation of gated recurrent neural networks on sequence modeling." *arXiv preprint arXiv:1412.3555*, 2014.

[CGH17] Bob Carpenter, Andrew Gelman, Matthew D Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus A Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. "Stan: a probabilistic programming language." *Grantee Submission*, **76**(1):1–32, 2017.

[CH05] Miguel A Carreira-Perpinan and Geoffrey E Hinton. "On contrastive divergence learning." In *AI Stats*, 2005.

[CHY19] Yixin Chen, Siyuan Huang, Tao Yuan, Siyuan Qi, Yixin Zhu, and Song-Chun Zhu. "Holistic++ scene understanding: Single-view 3d holistic scene parsing and human pose estimation with human-object interaction and physical commonsense." In *International Conference on Computer Vision (ICCV)*, 2019.

[CKA15] Grzegorz Chrupała, Akos Kádár, and Afra Alishahi. "Learning language through pictures." *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2015.

[CKZ15] Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Andrew G Berneshawi, Huimin Ma, Sanja Fidler, and Raquel Urtasun. "3d object proposals for accurate object class detection." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

[CKZ16] Xiaozhi Chen, Kaustav Kundu, Ziyu Zhang, Huimin Ma, Sanja Fidler, and Raquel Urtasun. "Monocular 3d object detection for autonomous driving." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[CLF18] Xinlei Chen, Li-Jia Li, Li Fei-Fei, and Abhinav Gupta. "Iterative Visual Reasoning Beyond Convolutions." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[CLL18] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. "Learning to detect human-object interactions." In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2018.

[CLO16] Jungchan Cho, Minsik Lee, and Songhwai Oh. "Complex non-rigid 3d shape recovery using a procrustean normal distribution mixture model." *International Journal of Computer Vision (IJCV)*, **117**(3):226–246, 2016.

[CM14] Danqi Chen and Christopher D Manning. "A fast and accurate dependency parser using neural networks." In *Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[CPK17] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2017.

[CS14] Paco Calvo and John Symons. *The architecture of cognition: Rethinking Fodor and Pylyshyn's systematicity challenge.* MIT Press, 2014.

[CSK15] Chenyi Chen, Ari Seff, Alain Kornhauser, and Jianxiong Xiao. "Deepdriving: Learning affordance for direct perception in autonomous driving." In *International Conference on Computer Vision (ICCV)*, 2015.

[CSS09] Wongun Choi, Khuram Shahid, and Silvio Savarese. "What are they doing?: Collective activity classification using spatio-temporal relationship among people." In *International Conference on Computer Vision Workshops (ICCV Workshops)*, 2009.

[CSW17] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. "Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[CWL16] Wenzheng Chen, Huan Wang, Yangyan Li, Hao Su, Dani Lischinsk, Daniel Cohen-Or, Baoquan Chen, et al. "Synthesizing Training Images for Boosting Human 3D Pose Estimation." In *International Conference on 3D Vision (3DV)*, 2016.

[CY99] James M Coughlan and Alan L Yuille. "Manhattan world: Compass direction from a single image by bayesian inference." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 1999.

[CY03] James M Coughlan and Alan L Yuille. "Manhattan world: Orientation and outlier detection by bayesian inference." *Neural Computation*, 2003.

[CZ17] Joao Carreira and Andrew Zisserman. "Quo vadis, action recognition? a new model and the kinetics dataset." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[DCL18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805*, 2018.

[DDG18] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. "Embodied question answering." In *Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*, 2018.

[DDM18] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. "Scaling egocentric vision: The epic-kitchens dataset." In *European Conference on Computer Vision (ECCV)*, 2018.

[DDS16] Hanjun Dai, Bo Dai, and Le Song. "Discriminative embeddings of latent variable models for structured data." In *International Conference on Machine Learning (ICML)*, 2016.

[DFC16] Saumitro Dasgupta, Kuan Fang, Kevin Chen, and Silvio Savarese. "Delay: Robust spatial layout estimation for cluttered indoor scenes." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[DGL18] Abhishek Das, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. "Neural modular control for embodied question answering." *arXiv preprint arXiv:1810.11181*, 2018.

[DHP19] Sanjoy Dasgupta, Daniel Hsu, Stefanos Poulis, and Xiaojin Zhu. "Teaching a black-box learner." In *International Conference on Machine Learning (ICML)*, 2019.

[DHS15] Jifeng Dai, Kaiming He, and Jian Sun. "Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation." In *International Conference on Computer Vision (ICCV)*, 2015.

[DK86] Simon Duane and John B Kogut. "The theory of hybrid stochastic algorithms." *Nuclear Physics B*, **275**(3):398–420, 1986.

[DL16] Li Dong and Mirella Lapata. "Language to logical form with neural attention." *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2016.

[DL17] Zhuo Deng and Longin Jan Latecki. "Amodal detection of 3d objects: Inferring 3d bounding boxes from 2d ones in rgb-depth images." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[DLB18] Yilun Du, Zhijian Liu, Hector Basevi, Ales Leonardis, Bill Freeman, Josh Tenenbaum, and Jiajun Wu. “Learning to Exploit Stability for 3D Scene Parsing.” In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[DLN07] Erick Delage, Honglak Lee, and Andrew Y Ng. “Automatic single-image 3d reconstructions of indoor manhattan world scenes.” In *Robotics Research*, pp. 305–321. Springer, 2007.

[DMH17] Debidatta Dwibedi, Ishan Misra, and Martial Hebert. “Cut, Paste and Learn: Surprisingly Easy Synthesis for Instance Detection.” *International Conference on Computer Vision (ICCV)*, 2017.

[DMI15] David K. Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P. Adams. “Convolutional networks on graphs for learning molecular fingerprints.” In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

[DQX17] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. “Deformable convolutional networks.” In *International Conference on Computer Vision (ICCV)*, 2017.

[DRC17] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. “CARLA: An Open Urban Driving Simulator.” In *Conference on Robot Learning*, 2017.

[DUB17] Jacob Devlin, Jonathan Uesato, Surya Bhupatiraju, Rishabh Singh, Abdelrahman Mohamed, and Pushmeet Kohli. “Robustfill: Neural program learning under noisy i/o.” In *International Conference on Machine Learning (ICML)*, 2017.

[DXY19] Wang-Zhou Dai, Qiuling Xu, Yang Yu, and Zhi-Hua Zhou. “Bridging Machine Learning and Logical Reasoning by Abductive Learning.” In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[DZ17] Wang-Zhou Dai and Zhi-Hua Zhou. “Combining logical abduction and statistical induction: Discovering written primitives with human knowledge.” In *AAAI Conference on Artificial Intelligence (AAAI)*, 2017.

[EG18] Richard Evans and Edward Grefenstette. “Learning explanatory rules from noisy data.” *Journal of Artificial Intelligence Research*, **61**:1–64, 2018.

[Elm93] Jeffrey L Elman. “Learning and development in neural networks: The importance of starting small.” *Cognition*, 1993.

[Eme00] Nathan J. Emery. “The eyes have it: the neuroethology, function and evolution of social gaze.” *Neuroscience & Biobehavioral Reviews*, **24**(6):581 – 604, 2000.

[EMS18] Kevin M Ellis, Lucas E Morales, Mathias Sablé-Meyer, Armando Solar Lezama, and Joshua B Tenenbaum. "Library learning for neurally-guided bayesian program induction." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[ER13] Susan E Embretson and Steven P Reise. *Item response theory*. Psychology Press, 2013.

[ERS18] Kevin Ellis, Daniel Ritchie, Armando Solar-Lezama, and Joshua B Tenenbaum. "Learning to infer graphics programs from hand-drawn images." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[EWN20] Kevin Ellis, Catherine Wong, Maxwell Nye, Mathias Sable-Meyer, Luc Cary, Lucas Morales, Luke Hewitt, Armando Solar-Lezama, and Joshua B Tenenbaum. "Dreamcoder: Growing generalizable, interpretable knowledge with wake-sleep bayesian program learning." *arXiv preprint arXiv:2006.08381*, 2020.

[Fan18] Yang Fan et al. "Learning to teach." *International Conference on Learning Representations (ICLR)*, 2018.

[FAS10] Afsaneh Fazly, Afra Alishahi, and Suzanne Stevenson. "A probabilistic computational model of cross-situational word learning." *Annual Conference of the Cognitive Science Society (CogSci)*, 2010.

[FC03] Lisa Feigenson and Susan Carey. "Tracking individuals via object-files: evidence from infants' manual search." *Developmental Science*, **6**(5):568–584, 2003.

[FCS09] Yasutaka Furukawa, Brian Curless, Steven M Seitz, and Richard Szeliski. "Manhattan-world stereo." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[FCT18] Hao-Shu Fang, Jinkun Cao, Yu-Wing Tai, and Cewu Lu. "Pairwise body-part attention for recognizing human-object interactions." In *European Conference on Computer Vision (ECCV)*, 2018.

[FCW18] Lifeng Fan, Yixin Chen, Ping Wei, Wenguan Wang, and Song-Chun Zhu. "Inferring Shared Attention in Social Scene Videos." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[FD18] Abram L Friesen and Pedro M Domingos. "Submodular field grammars: representation, inference, and application to image parsing." In *Advances in Neural Information Processing Systems*, 2018.

[FFG89] Brian Falkenhainer, Kenneth D Forbus, and Dedre Gentner. "The structure-mapping engine: Algorithm and examples." *Artificial intelligence*, **41**(1):1–63, 1989.

[FFM19] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. "Slowfast networks for video recognition." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[FHC18] Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. "Speaker-follower models for vision-and-language navigation." In *Advances in Neural Information Processing Systems*, pp. 3314–3325, 2018.

[FKE18] David F Fouhey, Wei-cheng Kuo, Alexei A Efros, and Jitendra Malik. "From lifestyle vlogs to everyday interactions." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[FL02] Jerry A Fodor and Ernest Lepore. *The compositionality papers.* Oxford University Press, 2002.

[Fod75] Jerry A Fodor. *The language of thought*, volume 5. Harvard university press, 1975.

[FP88] Jerry A Fodor, Zenon W Pylyshyn, et al. "Connectionism and cognitive architecture: A critical analysis." *Cognition*, **28**(1-2):3–71, 1988.

[Fri03] Arthur Fridman. "Mixed markov models." *Proceedings of the National Academy of Sciences (PNAS)*, 2003.

[FRS12] Matthew Fisher, Daniel Ritchie, Manolis Savva, Thomas Funkhouser, and Pat Hanrahan. "Example-based synthesis of 3D object arrangements." *ACM Transactions on Graphics (TOG)*, 2012.

[FS16] Chaz Firestone and Brian J Scholl. "Cognition does not affect perception: Evaluating the evidence for "top-down" effects." *Behavioral and Brain Sciences*, **39**, 2016.

[FWC19] Deng-Ping Fan, Wenguan Wang, Ming-Ming Cheng, and Jianbing Shen. "Shifting more attention to video salient object detection." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[FWH19] Lifeng Fan, Wenguan Wang, Siyuan Huang, Xinyu Tang, and Song-Chun Zhu. "Understanding human gaze communication by spatio-temporal graph reasoning." In *International Conference on Computer Vision (ICCV)*, 2019.

[FWS18] Yao Feng, Fan Wu, Xiaohu Shao, Yanfeng Wang, and Xi Zhou. "Joint 3D Face Reconstruction and Dense Alignment with Position Map Regression Network." In *European Conference on Computer Vision (ECCV)*, 2018.

[FXW18] Hao-Shu Fang, Yuanlu Xu, Wenguan Wang, Xiaobai Liu, and Song-Chun Zhu. "Learning pose grammar to encode human body configuration for 3D pose estimation." In *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[GBK02a] György Gergely, Harold Bekkering, and Ildikó Király. "Developmental psychology: Rational imitation in preverbal infants." *Nature*, **415**(6873):755, 2002.

[GBK02b] György Gergely, Harold Bekkering, and Ildikó Király. "Rational imitation in preverbal infants." *Nature*, **415**(6873):755–755, 2002.

[GBM17] Alex Graves, Marc G. Bellemare, Jacob Menick, Rémi Munos, and Koray Kavukcuoglu. "Automated Curriculum Learning for Neural Networks." In *International Conference on Machine Learning (ICML)*, 2017.

[GDG17] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. "Accurate, large minibatch sgd: Training imagenet in 1 hour." *arXiv preprint arXiv:1706.02677*, 2017.

[GEH10] Abhinav Gupta, Alexei A Efros, and Martial Hebert. "Blocks world revisited: Image understanding using qualitative geometry and mechanics." In *European Conference on Computer Vision (ECCV)*, 2010.

[GF16] Noah D Goodman and Michael C Frank. "Pragmatic language interpretation as probabilistic inference." *Trends in cognitive sciences*, **20**(11):818–829, 2016.

[GH13] Ruiqi Guo and Derek Hoiem. "Support surface prediction in indoor scenes." In *International Conference on Computer Vision (ICCV)*, 2013.

[Gha15] Zoubin Ghahramani. "Probabilistic machine learning and artificial intelligence." *Nature*, **521**(7553):452–459, 2015.

[GHK10] Abhinav Gupta, Martial Hebert, Takeo Kanade, and David M Blei. "Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2010.

[GHZ18] Sheng Guo, Weilin Huang, Haozhi Zhang, Chenfan Zhuang, Dengke Dong, Matthew R. Scott, and Dinglong Huang. "CurriculumNet: Weakly Supervised Learning from Large-Scale Web Images." *ArXiv*, **abs/1808.01097**, 2018.

[Gib79] James Jerome Gibson. *The ecological approach to visual perception.* Houghton, Mifflin and Company, 1979.

[Gir15] Ross Girshick. "Fast r-cnn." In *International Conference on Computer Vision (ICCV)*, 2015.

[GKD09] Abhinav Gupta, Aniruddha Kembhavi, and Larry S Davis. "Observing human-object interactions: Using spatial and functional compatibility for recognition." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **31**(10):1775–1789, 2009.

[GKM17] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. "The "Something Something" Video Database for Learning and Evaluating Visual Common Sense." In *International Conference on Computer Vision (ICCV)*, 2017.

[GKS17] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. "Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[GLB19] Jonathan Gordon, David Lopez-Paz, Marco Baroni, and Diane Bouchacourt. "Permutation equivariant models for compositional generalization in language." In *International Conference on Learning Representations (ICLR)*, 2019.

[GLG08] Artur SD'Avila Garcez, Luis C Lamb, and Dov M Gabbay. *Neural-symbolic cognitive reasoning.* Springer Science & Business Media, 2008.

[GLG20] Dileep George, Miguel Lázaro-Gredilla, and J Swaroop Guntupalli. "From CAPTCHA to Commonsense: How Brain Can Teach Us About Artificial Intelligence." *Frontiers in Computational Neuroscience*, 2020.

[GLL17] Chuang Gan, Yandong Li, Haoxiang Li, Chen Sun, and Boqing Gong. "Vqs: Linking segmentations to questions and answers for supervised attention in vqa and question-focused semantic segmentation." In *International Conference on Computer Vision (ICCV)*, 2017.

[GLS13] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. "Vision meets robotics: The KITTI dataset." *International Journal of Robotics Research (IJRR)*, **32**(11):1231–1237, 2013.

[GLT18] Jon Gauthier, Roger Levy, and Joshua B Tenenbaum. "Word learning and the acquisition of syntactic-semantic overhypotheses." *Annual Conference of the Cognitive Science Society (CogSci)*, 2018.

[GMH13] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. "Speech recognition with deep recurrent neural networks." In *2013 IEEE international conference on acoustics, speech and signal processing*, 2013.

[GPL17] Kelvin Guu, Panupong Pasupat, Evan Zheran Liu, and Percy Liang. "From language to programs: Bridging reinforcement learning and maximum marginal likelihood." *arXiv preprint arXiv:1704.07926*, 2017.

[GR20] Rohit Girdhar and Deva Ramanan. "CATER: A diagnostic dataset for Compositional Actions and TEmporal Reasoning." *International Conference on Learning Representations (ICLR)*, 2020.

[Gre76] Ulf Grenander. "Lectures in pattern theory I, II and III: Pattern analysis, pattern synthesis and regular structures.", 1976.

[Gre93] Ulf Grenander. *General pattern theory-A mathematical study of regular structures.* Clarendon Press, 1993.

[Gri75] Herbert P Grice. "Logic and conversation." In *Speech acts*, pp. 41–58. Brill, 1975.

[GSR98] Arthur M. Glenberg, Jennifer L. Schroeder, and David A. Robertson. "Averting the gaze disengages the environment and facilitates remembering." *Memory & Cognition*, **26**(4):651–658, 1998.

[GSR17] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. "Neural Message Passing for Quantum Chemistry." In *International Conference on Machine Learning (ICML)*, 2017.

[GSR18] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. "Ava: A video dataset of spatio-temporally localized atomic visual actions." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[GSS09] Abhinav Gupta, Praveen Srinivasan, Jianbo Shi, and Larry S Davis. "Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[GTC15] Bernard Ghanem, Ali Thabet, Juan Carlos Niebles, and Fabian Caba Heilbron. "Robust manhattan frame estimation from a single rgb-d image." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[GXG18] Hong Ge, Kai Xu, and Zoubin Ghahramani. "Turing: A language for flexible probabilistic inference." In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2018.

[GXH19] Ruiqi Gao, Jianwen Xie, Siyuan Huang, Yufan Ren, Song-Chun Zhu, and Ying Nian Wu. "Learning vector representation of local content and matrix representation of local motion, with implications for V1." *arXiv preprint arXiv:1902.03871*, 2019.

[GZW03] Cheng-en Guo, Song-Chun Zhu, and Ying Nian Wu. "Towards a mathematical theory of primal sketch and sketchability." In *International Conference on Computer Vision (ICCV)*, 2003.

[GZW07] Cheng-en Guo, Song-Chun Zhu, and Ying Nian Wu. "Primal sketch: Integrating structure and texture." *Computer Vision and Image Understanding (CVIU)*, **106**(1):5–19, 2007.

[HAR17] Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Kate Saenko. "Learning to Reason: End-to-End Module Networks for Visual Question Answering." *International Conference on Computer Vision (ICCV)*, pp. 804–813, 2017.

[Has70] W Keith Hastings. "Monte Carlo sampling methods using Markov chains and their applications." *Biometrika*, **57**(1):97–109, 1970.

[HBM77] Marshall M. Haith, Terry Bergman, and Michael J. Moore. "Eye contact and face scanning in early infancy." *Science*, **198**(4319):853–855, 1977.

[HBM20] Steven Holtzen, Guy Van den Broeck, and Todd Millstein. "Scaling exact inference for discrete probabilistic programs." *Proceedings of the ACM on Programming Languages*, **4**(OOPSLA):1–31, 2020.

[HCY19] Siyuan Huang, Yixin Chen, Tao Yuan, Siyuan Qi, Yixin Zhu, and Song-Chun Zhu. "Perspectivenet: 3d object detection from a single rgb image via perspective points." *NIPS*, 2019.

[HDY12] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. "Deep neural networks for acoustic modeling in speech recognition." *Signal Processing Magazine*, **29**(6):82–97, 2012.

[HEH05] Derek Hoiem, Alexei A Efros, and Martial Hebert. "Automatic photo pop-up." *ACM Transactions on Graphics (TOG)*, **24**(3):577–584, 2005.

[HGD17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. "Mask r-cnn." In *International Conference on Computer Vision (ICCV)*, 2017.

[HHF09] Varsha Hedau, Derek Hoiem, and David Forsyth. "Recovering the spatial layout of cluttered rooms." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[HHF10] Varsha Hedau, Derek Hoiem, and David Forsyth. "Thinking inside the box: Using appearance models and context based on room geometry." In *European Conference on Computer Vision (ECCV)*, 2010.

[Hin02] Geoffrey E Hinton. "Training products of experts by minimizing contrastive divergence." *Neural Computation*, 2002.

[HLC20] Yining Hong, Qing Li, Daniel Ciao, Siyuan Haung, and Song-Chun Zhu. "Learning by Fixing: Solving Math Word Problems with Weak Supervision." *AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

[HLG20] Yining Hong, Qing Li, Ran Gong, Daniel Ciao, Siyuan Huang, and Song-Chun Zhu. "SMART: A Situation Model for Algebra Story Problems via Attributed Grammar." *AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

[HLZ21] Yining Hong, Qing Li, Song-Chun Zhu, and Siyuan Huang. "VLGrammar: Grounded Grammar Induction of Vision and Language." *arXiv preprint arXiv:2103.12975*, 2021.

[HM18] Drew A Hudson and Christopher D Manning. "Compositional Attention Networks for Machine Reasoning." In *International Conference on Learning Representations (ICLR)*, 2018.

[HM19] Drew A Hudson and Christopher D Manning. “GQA: A New Dataset for Real-World Visual Reasoning and Compositional Question Answering.” *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[HMV13] Mark Hendrikx, Sebastiaan Meijer, Joeri Van Der Velden, and Alexandru Iosup. “Procedural content generation for games: A survey.” *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2013.

[HOT06] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. “A fast learning algorithm for deep belief nets.” *Neural computation*, **18**(7):1527–1554, 2006.

[HPB16] Ankur Handa, Viorica Pătrăucean, Vijay Badrinarayanan, Simon Stent, and Roberto Cipolla. “Understanding real world indoor scenes with synthetic data.” In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[HPS16] Ankur Handa, Viorica Patraucean, Simon Stent, and Roberto Cipolla. “SceneNet: an Annotated Model Generator for Indoor Scene Understanding.” In *International Conference on Robotics and Automation (ICRA)*, 2016.

[HQX18] Siyuan Huang, Siyuan Qi, Yinxue Xiao, Yixin Zhu, Ying Nian Wu, and Song-Chun Zhu. “Cooperative Holistic Scene Understanding: Unifying 3D Object, Layout and Camera Pose Estimation.” In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[HQZ18] Siyuan Huang, Siyuan Qi, Yixin Zhu, Yinxue Xiao, Yuanlu Xu, and Song-Chun Zhu. “Holistic 3D Scene Parsing and Reconstruction from a Single RGB Image.” In *European Conference on Computer Vision (ECCV)*, 2018.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. “Long short-term memory.” *Neural computation*, **9**(8):1735–1780, 1997.

[HS06] Geoffrey E Hinton and Ruslan R Salakhutdinov. “Reducing the dimensionality of data with neural networks.” *Science*, 2006.

[HS19] Tong He and Stefano Soatto. “Mono3D++: Monocular 3D Vehicle Detection with Two-Scale 3D Hypotheses and Task Priors.” *arXiv preprint arXiv:1901.03446*, 2019.

[HT11] Chien-Ming Huang and Andrea L. Thomaz. “Effects of responding to, initiating and ensuring joint attention in human-robot interaction.” In *2011 Ro-Man*, 2011.

[HWK15] Qixing Huang, Hai Wang, and Vladlen Koltun. “Single-view reconstruction via joint analysis of image and shape collections.” *ACM Transactions on Graphics (TOG)*, 2015.

[HWM14] Ankur Handa, Thomas Whelan, John McDonald, and Andrew J Davison. “A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM.” In *International Conference on Robotics and Automation (ICRA)*, 2014.

[HYL17] Will Hamilton, Zhitao Ying, and Jure Leskovec. "Inductive representation learning on large graphs." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[HZ05] Feng Han and Song-Chun Zhu. "Bottom-up/top-down image parsing by attribute graph grammar." In *International Conference on Computer Vision (ICCV)*, 2005.

[HZR16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[IB09] Roxane J. Itier and Magali Batty. "Neural bases of eye and gaze processing: the core of social cognition." *Neuroscience & Biobehavioral Reviews*, **33**(6):843–863, 2009.

[IKN98] Laurent Itti, Christof Koch, and Ernst Niebur. "A model of saliency-based visual attention for rapid scene analysis." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **20**(11):1254–1259, 1998.

[IMD16] Mostafa S Ibrahim, Srikanth Muralidharan, Zhiwei Deng, Arash Vahdat, and Greg Mori. "A hierarchical deep temporal model for group activity recognition." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[IPO13] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. "Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **36**(7):1325–1339, 2013.

[ISS17a] Hamid Izadinia, Qi Shan, and Steven M Seitz. "IM2CAD." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[ISS17b] Hamid Izadinia, Qi Shan, and Steven M Seitz. "Im2cad." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[Jac02] Robert A Jacobs. "What determines visual cue reliability?" *Trends in cognitive sciences*, **6**(8):345–350, 2002.

[JCH20] Baoxiong Jia, Yixin Chen, Siyuan Huang, Yixin Zhu, and Song-Chun Zhu. "LEMMA: A Multi-view Dataset for LEarning Multi-agent Multi-task Activities." In *European Conference on Computer Vision (ECCV)*, 2020.

[JHB18] Mathis Jording, Arne Hartz, Gary Bente, Martin Schulte-Rüther, and Kai Vogeley. "The "Social Gaze Space": A Taxonomy for Gaze-Based Communication in Triadic Interactions." *Frontiers in Psychology*, **9**:226, 2018.

[JHM17a] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. "Clevr: A diagnostic dataset for compositional language and elementary visual reasoning." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[JHM17b] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. "CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning." *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[JHM17c] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Judy Hoffman, Fei Fei Li, C. Zitnick, and Ross Girshick. "Inferring and Executing Programs for Visual Reasoning." In *International Conference on Computer Vision (ICCV)*, 2017.

[JHV17] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Judy Hoffman, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. "Inferring and executing programs for visual reasoning." In *International Conference on Computer Vision (ICCV)*, 2017.

[Jia14] Lu Jiang et al. "Self-paced learning with diversity." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

[Jia15] Lu Jiang et al. "Self-paced curriculum learning." In *AAAI Conference on Artificial Intelligence (AAAI)*, 2015.

[JKS13] Yun Jiang, Hema Koppula, and Ashutosh Saxena. "Hallucinated humans as the hidden context for labeling 3d scenes." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[JKS16] Yun Jiang, Hema S Koppula, and Ashutosh Saxena. "Modeling 3D environments through hidden human context." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2016.

[Joh73] Gunnar Johansson. "Visual perception of biological motion and a model for its analysis." *Perception & psychophysics*, **14**(2):201–211, 1973.

[JQZ18] Chenfanfu Jiang, Siyuan Qi, Yixin Zhu, Siyuan Huang, Jenny Lin, Lap-Fai Yu, Demetri Terzopoulos, and Song-Chun Zhu. "Configurable 3D Scene Synthesis and 2D Image Rendering with Per-pixel Ground Truth Using Stochastic Grammars." *International Journal of Computer Vision (IJCV)*, **126**(9):920–941, 2018.

[JS14] Yun Jiang and Ashutosh Saxena. "Modeling High-Dimensional Humans for Activity Anticipation using Gaussian Process Latent CRFs." In *Robotics: Science and Systems*, 2014.

[JSL17] Hanbyul Joo, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Scott Godisart, Bart Nabbe, Iain Matthews, et al. "Panoptic studio: A massively multiview system for social interaction capture." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2017.

[Jul62] Bela Julesz. "Visual pattern discrimination." *IRE transactions on Information Theory*, **8**(2):84–92, 1962.

[Jul81] Bela Julesz. “Textons, the elements of texture perception, and their interactions.” *Nature*, **290**(5802):91, 1981.

[JZS16] Ashesh Jain, Amir R. Zamir, Silvio Savarese, and Ashutosh Saxena. “Structural-RNN: Deep learning on spatio-temporal graphs.” In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[Kan81] Takeo Kanade. “Recovery of the three-dimensional shape of an object from a single view.” *Artificial intelligence*, **17**(1-3):409–460, 1981.

[KAS14] Hilde Kuehne, Ali Arslan, and Thomas Serre. “The language of actions: Recovering the syntax and semantics of goal-directed human activities.” In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[KB14] Diederik Kingma and Jimmy Ba. “Adam: A method for stochastic optimization.” *arXiv preprint arXiv:1412.6980*, 2014.

[KB15] Diederik Kingma and Jimmy Ba. “Adam: A method for stochastic optimization.” In *International Conference on Learning Representations (ICLR)*, 2015.

[KD09] Kai A Krueger and Peter Dayan. “Flexible shaping: How learning in small steps helps.” *Cognition*, 2009.

[KDV15] Till Kroeger, Dengxin Dai, and Luc Van Gool. “Joint vanishing point extraction and tracking.” In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[KFW18] Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. “Neural Relational Inference for Interacting Systems.” In *International Conference on Machine Learning (ICML)*, 2018.

[KGS13] Hema Swetha Koppula, Rudhir Gupta, and Ashutosh Saxena. “Learning human activities and object affordances from rgb-d videos.” *International Journal of Robotics Research (IJRR)*, **32**(8):951–970, 2013.

[KHA16] Max Kleiman-Weiner, Mark K Ho, Joseph L Austerweil, Michael L Littman, and Joshua B Tenenbaum. “Coordinate to cooperate or compete: abstract goals and joint intentions in social interaction.” In *Annual Conference of the Cognitive Science Society (CogSci)*, 2016.

[KHL17] James R Kubricht, Keith J Holyoak, and Hongjing Lu. “Intuitive physics: Current research and controversies.” *Trends in cognitive sciences*, **21**(10):749–759, 2017.

[KK97] Hiromi Kobayashi and Shiro Kohshima. “Unique morphology of the human eye.” *Nature*, **387**(6635):767, 1997.

[KK18] Nikita Kitaev and Dan Klein. “Constituency parsing with a self-attentive encoder.” In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018.

[KKT15] Tejas D Kulkarni, Pushmeet Kohli, Joshua B Tenenbaum, and Vikash Mansinghka. "Picture: A probabilistic programming language for scene perception." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[Kle86] Chris L. Kleinke. "Gaze and eye contact: a research review." *Psychological Bulletin*, **100**(1):78, 1986.

[KLR18] Abhijit Kundu, Yin Li, and James M Rehg. "3D-RCNN: Instance-Level 3D Object Reconstruction via Render-and-Compare." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[KMP18] Ashwin Kalyan, Abhishek Mohta, Oleksandr Polozov, Dhruv Batra, Prateek Jain, and Sumit Gulwani. "Neural-guided deductive search for real-time program synthesis from examples." *arXiv preprint arXiv:1804.01186*, 2018.

[KMT17] Wadim Kehl, Fabian Manhardt, Federico Tombari, Slobodan Ilic, and Nassir Navab. "SSD-6D: Making RGB-based 3D detection and 6D pose estimation great again." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[Kof13] Kurt Koffka. *Principles of Gestalt psychology.* Routledge, 2013.

[Koh20] Wolfgang Köhler. *Die physischen Gestalten in Ruhe und im stationärenZustand. Eine natur-philosophische Untersuchung [The physical Gestalten at rest and in steady state].* Braunschweig, Germany: Vieweg und Sohn., 1920.

[Koh38] Wolfgang Köhler. "Physical Gestalten." In *A source book of Gestalt psychology*, pp. 17–54. London, England: Routledge & Kegan Paul, 1938.

[KRK11] Hedvig Kjellström, Javier Romero, and Danica Kragić. "Visual object-action recognition: Inferring object affordances from human demonstration." *Computer Vision and Image Understanding (CVIU)*, **115**(1):81–90, 2011.

[KS83] Philip J Kellman and Elizabeth S Spelke. "Perception of partly occluded objects in infancy." *Cognitive psychology*, **15**(4):483–524, 1983.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2012.

[KSS19] Daniel Keysers, Nathanael Schärli, Nathan Scales, Hylke Buisman, Daniel Furrer, Sergii Kashubin, Nikola Momchev, Danila Sinopalnikov, Lukasz Stafiniak, Tibor Tihon, et al. "Measuring compositional generalization: A comprehensive method on realistic data." In *International Conference on Learning Representations (ICLR)*, 2019.

[KTS14] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. "Large-scale video classification with convolutional neural networks." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[Kum10] M Pawan Kumar et al. "Self-paced learning for latent variable models." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2010.

[KW17] Thomas N. Kipf and Max Welling. "Semi-supervised classification with graph convolutional networks." In *International Conference on Learning Representations (ICLR)*, 2017.

[LaV98] Steven M LaValle. *Rapidly-Exploring Random Trees: A New Tool for Path Planning*. Ames, IA, USA, 1998.

[LB95] Yann LeCun, Yoshua Bengio, et al. "Convolutional networks for images, speech, and time series." *The handbook of brain theory and neural networks*, **3361**(10):1995, 1995.

[LB14] Matthew M Loper and Michael J Black. "OpenDR: An approximate differentiable renderer." In *European Conference on Computer Vision (ECCV)*, 2014.

[LB18] Brenden M. Lake and Marco Baroni. "Generalization without Systematicity: On the Compositional Skills of Sequence-to-Sequence Recurrent Networks." In *International Conference on Machine Learning (ICML)*, 2018.

[LBH15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." *nature*, **521**(7553):436–444, 2015.

[LBL16a] Chen Liang, Jonathan Berant, Quoc Le, Kenneth D Forbus, and Ni Lao. "Neural Symbolic Machines: Learning Semantic Parsers on Freebase with Weak Supervision." In *ACL*, 2016.

[LBL16b] Chen Liang, Jonathan Berant, Quoc Le, Kenneth D Forbus, and Ni Lao. "Neural symbolic machines: Learning semantic parsers on freebase with weak supervision." *arXiv preprint arXiv:1611.00020*, 2016.

[LBM17] Chen-Yu Lee, Vijay Badrinarayanan, Tomasz Malisiewicz, and Andrew Rabinovich. "RoomNet: End-to-End Room Layout Estimation." In *International Conference on Computer Vision (ICCV)*, 2017.

[LC14] Sijin Li and Antoni B Chan. "3D human pose estimation from monocular images with deep convolutional neural network." In *Asian Conference on Computer Vision (ACCV)*, 2014.

[LC19] Guillaume Lample and François Charton. "Deep learning for symbolic mathematics." *arXiv preprint arXiv:1912.01412*, 2019.

[LC20] Guillaume Lample and François Charton. "Deep learning for symbolic mathematics." In *International Conference on Learning Representations (ICLR)*, 2020.

[LCL07] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. "Crowds by example." In *Computer Graphics Forum*, 2007.

[LDG17] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. "Feature pyramid networks for object detection." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[LDH17] Weiyang Liu, Bo Dai, Ahmad Humayun, Charlene Tay, Chen Yu, Linda B Smith, James M Rehg, and Le Song. "Iterative machine teaching." In *International Conference on Machine Learning (ICML)*, 2017.

[LeC15] Yann LeCun et al. "LeNet-5, convolutional neural networks." *URL: http://yann. lecun. com/exdb/lenet*, **20**:5, 2015.

[LFU13] Dahua Lin, Sanja Fidler, and Raquel Urtasun. "Holistic scene understanding for 3d object detection with rgbd cameras." In *International Conference on Computer Vision (ICCV)*, 2013.

[LFY18] Qing Li, Jianlong Fu, Dongfei Yu, Tao Mei, and Jiebo Luo. "Tell-and-answer: Towards explainable visual question answering using attributes and captions." *Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

[LG17] Jean Lahoud and Bernard Ghanem. "2d-driven 3d object detection in rgb-d images." In *International Conference on Computer Vision (ICCV)*, 2017.

[LHH20a] Qing Li, Siyuan Huang, Yining Hong, Yixin Chen, Ying Nian Wu, and Song-Chun Zhu. "Closed Loop Neural-Symbolic Learning via Integrating Neural Perception, Grammar Parsing, and Symbolic Reasoning." In *International Conference on Machine Learning (ICML)*, 2020.

[LHH20b] Qing Li, Siyuan Huang, Yining Hong, and Song-Chun Zhu. "A Competence-aware Curriculum for Visual Concepts Learning via Question Answering." In *European Conference on Computer Vision (ECCV)*, 2020.

[LHH21] Qing Li, Siyuan Huang, Yining Hong, Yixin Zhu, Ying Nian Wu, and Song-Chun Zhu. "A HINT from Arithmetic: On Systematic Generalization of Perception, Syntax, and Semantics." *arXiv preprint arXiv:2103.01403*, 2021.

[LHK09] David C Lee, Martial Hebert, and Takeo Kanade. "Geometric reasoning for single image structure recovery." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[LII12] Chaoran Liu, Carlos T. Ishi, Hiroshi Ishiguro, and Norihiro Hagita. "Generation of nodding, head tilting and eye gazing for human-robot dialogue interaction." In *HRI*, 2012.

[LKT14] Joseph J Lim, Aditya Khosla, and Antonio Torralba. "Fpm: Fine pose parts-based model with 3d cad models." In *European Conference on Computer Vision (ECCV)*, 2014.

[Llo12] John W Lloyd. *Foundations of logic programming.* Springer Science &amp; Business Media, 2012.

[LLR18] Yin Li, Miao Liu, and James M Rehg. "In the eye of beholder: Joint learning of gaze and actions in first person video." In *European Conference on Computer Vision (ECCV)*, 2018.

[LMJ95] Michael S Landy, Laurence T Maloney, Elizabeth B Johnston, and Mark Young. "Measurement and modeling of depth cue combination: In defense of weak fusion." *Vision research*, **35**(3):389–412, 1995.

[LMR99] Michael Land, Neil Mennie, and Jennifer Rusted. "The roles of vision and eye movements in the control of activities of daily living." *Perception*, **28**(11):1311–1328, 1999.

[LMS17] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. "Refinenet: Multi-path refinement networks for high-resolution semantic segmentation." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[LNB18a] Chen Liang, Mohammad Norouzi, Jonathan Berant, Quoc Le, and Ni Lao. "Memory Augmented Policy Optimization for Program Synthesis and Semantic Parsing." In *NIPS*, July 2018.

[LNB18b] Chen Liang, Mohammad Norouzi, Jonathan Berant, Quoc V. Le, and Ni Lao. "Memory Augmented Policy Optimization for Program Synthesis and Semantic Parsing." In *NeurIPS*, 2018.

[Low87] David G Lowe. "Three-dimensional object recognition from single two-dimensional images." *Artificial Intelligence*, **31**(3):355–395, 1987.

[Low04] David G Lowe. "Distinctive image features from scale-invariant keypoints." *International journal of computer vision*, **60**(2):91–110, 2004.

[Low12] David Lowe. *Perceptual organization and visual recognition*, volume 5. Springer Science & Business Media, 2012.

[LRB16] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. "Deeper depth prediction with fully convolutional residual networks." In *International Conference on 3D Vision (3DV)*, 2016.

[LSD15] Jonathan Long, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[LSR19] Dennis Lee, Christian Szegedy, Markus N Rabe, Sarah M Loos, and Kshitij Bansal. "Mathematical Reasoning in Latent Space." *arXiv preprint arXiv:1909.11851*, 2019.

[LSR20] Dennis Lee, Christian Szegedy, Markus N Rabe, Sarah M Loos, and Kshitij Bansal. "Mathematical Reasoning in Latent Space." In *International Conference on Learning Representations (ICLR)*, 2020.

[LST15] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. "Human-level concept learning through probabilistic program induction." *Science*, **350**(6266):1332–1338, 2015.

[LTB16] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. "Gated graph sequence neural networks." In *International Conference on Machine Learning (ICML)*, 2016.

[LTJ18] Qing Li, Qingyi Tao, Shafiq Joty, Jianfei Cai, and Jiebo Luo. "VQA-E: Explaining, elaborating, and enhancing your answers for visual questions." In *European Conference on Computer Vision (ECCV)*, 2018.

[LTL17] Ruiyu Li, Makarand Tapaswi, Renjie Liao, Jiaya Jia, Raquel Urtasun, and Sanja Fidler. "Situation Recognition With Graph Neural Networks." In *International Conference on Computer Vision (ICCV)*, 2017.

[LUT17] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. "Building machines that learn and think like people." *Behavioral and Brain Sciences*, **40**, 2017.

[LWY16] John P. Lalor, Hao Wu, and Hong Yu. "Building an Evaluation Scale using Item Response Theory." *Empirical Methods in Natural Language Processing (EMNLP)*, 2016.

[LWY19] John P Lalor, Hao Wu, and Hong Yu. "Learning Latent Parameters without Human Response Patterns: Item Response Theory with Artificial Crowds." *Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

[LYC18] Chen Liu, Jimei Yang, Duygu Ceylan, Ersin Yumer, and Yasutaka Furukawa. "PlaneNet: Piece-wise Planar Reconstruction from a Single RGB Image." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[LZL10] Wanqing Li, Zhengyou Zhang, and Zicheng Liu. "Action recognition based on a bag of 3d points." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[LZZ14] Xiaobai Liu, Yibiao Zhao, and Song-Chun Zhu. "Single-view 3d scene parsing by attributed grammar." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[LZZ17] Xiaobai Liu, Yibiao Zhao, and Song-Chun Zhu. "Single-view 3d scene reconstruction and parsing by attribute grammar." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **40**(3):710–725, 2017.

[MAF17] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Košecká. "3d bounding box estimation using deep learning and geometry." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[Mag09] Lorenzo Magnani. *Abductive cognition: The epistemological and eco-cognitive dimensions of hypothetical reasoning*, volume 3. Springer Science & Business Media, 2009.

[Mar82] David Marr. *Vision: A computational investigation into the human representation and processing of visual information.* WH Freeman, 1982.

[Mar98] Gary F Marcus. "Rethinking eliminative connectionism." *Cognitive psychology*, **37**(3):243–282, 1998.

[Mar18] Gary F Marcus. *The algebraic mind: Integrating connectionism and cognitive science.* MIT press, 2018.

[MAZ19] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, Dan Gutfruend, Carl Vondrick, et al. "Moments in Time Dataset: one million videos for event understanding." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2019.

[MBR16] Roozbeh Mottaghi, Hessam Bagherinezhad, Mohammad Rastegari, and Ali Farhadi. "Newtonian scene understanding: Unfolding the dynamics of objects in static images." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[MBS10] Andrew N. Meltzoff, Rechele Brooks, Aaron P. Shon, and Rajesh P.N. Rao. ""Social" robots are psychological agents for infants: A test of gaze following." *Neural networks*, **23**(8-9):966–972, 2010.

[MCV19] Farnam Mansouri, Yuxin Chen, Ara Vartanian, Xiaojin Zhu, and Adish Singla. "Preference-Based Batch and Sequential Teaching: Towards a Unified View of Models." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[MD94] Stephen Muggleton and Luc De Raedt. "Inductive logic programming: Theory and methods." *The Journal of Logic Programming*, **19**:629–679, 1994.

[MDD14] Chris Moore, Philip J. Dunham, and Phil Dunham. *Joint attention: Its origins and role in development.* Psychology Press, 2014.

[MDK18] Robin Manhaeve, Sebastijan Dumancic, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. "Deepproblog: Neural probabilistic logic programming." In *Advances in Neural Information Processing Systems*, pp. 3749–3759, 2018.

[MF14] Mateusz Malinowski and Mario Fritz. "A Multi-world Approach to Question Answering About Real-world Scenes Based on Uncertain Input." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

[MFH06] Bilge Mutlu, Jodi Forlizzi, and Jessica Hodgins. "A storytelling robot: Modeling and evaluation of human-like gaze behavior." In *IEEE-RAS ICHR*, 2006.

[MG18] Francisco Massa and Ross Girshick. "maskrcnn-benchmark: Fast, modular reference implementation of Instance Segmentation and Object Detection algorithms in PyTorch." `https://github.com/facebookresearch/maskrcnn-benchmark`, 2018.

[MGF17] Ishan Misra, Ross B. Girshick, Rob Fergus, Martial Hebert, Abhinav Gupta, and Laurens van der Maaten. "Learning by Asking Questions." *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[MGK19] Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B Tenenbaum, and Jiajun Wu. "The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision." *International Conference on Learning Representations (ICLR)*, 2019.

[MHL17] John McCormac, Ankur Handa, Stefan Leutenegger, and Andrew J Davison. "SceneNet RGB-D: Can 5M Synthetic Images Beat Generic ImageNet Pretraining on Indoor Segmentation?" In *International Conference on Computer Vision (ICCV)*, 2017.

[MKF12] Bilge Mutlu, Takayuki Kanda, Jodi Forlizzi, Jessica Hodgins, and Hiroshi Ishiguro. "Conversational gaze mechanisms for humanlike robots." *ACM TIIS*, **1**(2):12, 2012.

[ML15] Arun Mallya and Svetlana Lazebnik. "Learning informative edge maps for indoor scene layout prediction." In *International Conference on Computer Vision (ICCV)*, 2015.

[ML16] Arun Mallya and Svetlana Lazebnik. "Learning models for actions and person-object interactions with transfer to question answering." In *European Conference on Computer Vision (ECCV)*, 2016.

[MLZ16] Rui Ma, Honghua Li, Changqing Zou, Zicheng Liao, Xin Tong, and Hao Zhang. "Action-driven 3D indoor scene evolution." *ACM Transactions on Graphics (TOG)*, **35**(6):173–1, 2016.

[MN78] David Marr and Herbert Keith Nishihara. "Representation and recognition of the spatial organization of three-dimensional shapes." *Proceedings of the Royal Society of London. Series B. Biological Sciences*, **200**(1140):269–294, 1978.

[Mon03] Stephen Monsell. "Task switching." *Trends in cognitive sciences*, **7**(3):134–140, 2003.

[MSG17] Kenneth Marino, Ruslan Salakhutdinov, and Abhinav Gupta. "The More You Know: Using Knowledge Graphs for Image Classification." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[MSL11] Paul Merrell, Eric Schkufza, Zeyang Li, Maneesh Agrawala, and Vladlen Koltun. "Interactive furniture layout using interior design guidelines." *ACM Transactions on Graphics (TOG)*, 2011.

[MSS17] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. “Vnect: Real-time 3d human pose estimation with a single rgb camera.” *ACM Transactions on Graphics (TOG)*, **36**(4):44, 2017.

[MTS18] David Mascharka, Philip Tran, Ryan Soklaski, and Arjun Majumdar. “Transparency by design: Closing the gap between performance and interpretability in visual reasoning.” In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[MWF83] Michael McCloskey, Allyson Washburn, and Linda Felch. “Intuitive physics: the straight-down belief and its origin.” *Journal of Experimental Psychology: Learning, Memory, and Cognition*, **9**(4):636, 1983.

[NAK16] Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkov. “Learning convolutional neural networks for graphs.” In *International Conference on Machine Learning (ICML)*, 2016.

[Nee97] Amy Needham. “Factors affecting infants’ use of featural information in object segregation.” *Current Directions in Psychological Science*, **6**(2):26–33, 1997.

[NHH15] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. “Learning deconvolution network for semantic segmentation.” In *International Conference on Computer Vision (ICCV)*, 2015.

[NM90] Mark Nitzberg and David Mumford. “The 2.1-D sketch.” In *International Conference on Computer Vision (ICCV)*, 1990.

[NNM16] Prathiba Natesan, Ratna Nandakumar, Tom Minka, and Jonathan D Rubright. “Bayesian prior choice in IRT estimation using MCMC and variational Bayes.” *Frontiers in psychology*, 2016.

[NYD16] Alejandro Newell, Kaiyu Yang, and Jia Deng. “Stacked hourglass networks for human pose estimation.” In *European Conference on Computer Vision (ECCV)*, 2016.

[OF96] Bruno A Olshausen and David J Field. “Emergence of simple-cell receptive field properties by learning a sparse code for natural images.” *Nature*, **381**(6583):607, 1996.

[OHP11] Sangmin Oh, Anthony Hoogs, Amitha Perera, Naresh Cuntoor, Chia-Chih Chen, Jong Taek Lee, Saurajit Mukherjee, JK Aggarwal, Hyungtae Lee, Larry Davis, et al. “A large-scale benchmark dataset for event recognition in surveillance video.” In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[Oli05] Aude Oliva. “Gist of the scene.” In *Neurobiology of attention*, pp. 251–256. Elsevier, 2005.

[OPO10] Jan Ondřej, Julien Pettré, Anne-Hélène Olivier, and Stéphane Donikian. "A synthetic-vision based steering approach for crowd simulation." *ACM Transactions on Graphics (TOG)*, 2010.

[OT06a] Sanae Okamoto-Barth and Masaki Tomonaga. "Development of joint attention in infant chimpanzees." In *Cognitive development in chimpanzees*, pp. 155–171. Springer, 2006.

[OT06b] Aude Oliva and Antonio Torralba. "Building the gist of a scene: The role of global image features in recognition." *Progress in brain research*, **155**:23–36, 2006.

[PCZ19] Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. "Specaugment: A simple data augmentation method for automatic speech recognition." In *Interspeech*, 2019.

[Pea89] Giuseppe Peano. *Arithmetices principia: Nova methodo exposita.* Fratres Bocca, 1889.

[Pen87] Alex P Pentland. "Perceptual organization and the representation of natural form." In *Readings in Computer Vision*, pp. 680–699. Elsevier, 1987.

[PES09] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. "You'll never walk alone: Modeling social behavior for multi-target tracking." In *International Conference on Computer Vision (ICCV)*, 2009.

[Pet04] Gail B Peterson. "A day of great illumination: BF Skinner's discovery of shaping." *Journal of the experimental analysis of behavior*, 2004.

[PGC17] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. "Automatic differentiation in PyTorch." In *NIPS-W*, 2017.

[PJS12] Hyun S. Park, Eakta Jain, and Yaser Sheikh. "3D social saliency from head-mounted cameras." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2012.

[PMS16] Emilio Parisotto, Abdel-rahman Mohamed, Rishabh Singh, Lihong Li, Dengyong Zhou, and Pushmeet Kohli. "Neuro-symbolic program synthesis." In *International Conference on Learning Representations (ICLR)*, 2016.

[Pot75] Mary C Potter. "Meaning in visual search." *Science*, **187**(4180):965–966, 1975.

[Pot76] Mary C Potter. "Short-term conceptual memory for pictures." *Journal of experimental psychology: human learning and memory*, **2**(5):509, 1976.

[PR12] Hamed Pirsiavash and Deva Ramanan. "Detecting activities of daily living in first-person camera views." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[PS15] Hyun Soo Park and Jianbo Shi. "Social saliency prediction." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[PSL14] Anastasia Pentina, Viktoriia Sharmanska, and Christoph H. Lampert. "Curriculum learning of multiple tasks." *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5492–5500, 2014.

[PSN19] Emmanouil Antonios Platanios, Otilia Stretcu, Graham Neubig, Barnabás Póczos, and Tom M. Mitchell. "Competence-based Curriculum Learning for Neural Machine Translation." In *North American Chapter of the Association for Computational Linguistics(NAACL-HLT)*, 2019.

[PSV17] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron C. Courville. "FiLM: Visual Reasoning with a General Conditioning Layer." In *AAAI Conference on Artificial Intelligence (AAAI)*, 2017.

[Pyl84] Zenon W Pylyshyn. "Computation and cognition: Towards a foundation for cognitive science.", 1984.

[QHW17] Siyuan Qi, Siyuan Huang, Ping Wei, and Song-Chun Zhu. "Predicting Human Activities Using Stochastic Grammar." In *International Conference on Computer Vision (ICCV)*, 2017.

[QJH20] Siyuan Qi, Baoxiong Jia, Siyuan Huang, Ping Wei, and Song-Chun Zhu. "A Generalized Earley Parser for Human Activity Parsing and Prediction." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

[QJZ18] Siyuan Qi, Baoxiong Jia, and Song-Chun Zhu. "Generalized Earley Parser: Bridging Symbolic Grammars and Sequence Data for Future Prediction." In *International Conference on Machine Learning (ICML)*, 2018.

[QLW18] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. "Frustum PointNets for 3D Object Detection from RGB-D Data." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[QSN16] Charles R Qi, Hao Su, Matthias Niessner, Angela Dai, Mengyuan Yan, and Leonidas J Guibas. "Volumetric and Multi-View CNNs for Object Classification on 3D Data." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[QWJ18a] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. "Learning human-object interactions by graph parsing neural networks." In *European Conference on Computer Vision (ECCV)*, 2018.

[QWJ18b] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. "Learning Human-Object Interactions by Graph Parsing Neural Networks." In *European Conference on Computer Vision (ECCV)*, 2018.

[QWL15] Hang Qi, Tianfu Wu, Mun-Wai Lee, and Song-Chun Zhu. "A restricted visual turing test for deep scene and event understanding." *arXiv preprint arXiv:1512.01715*, 2015.

[QY16] Weichao Qiu and Alan Yuille. "UnrealCV: Connecting Computer Vision to Unreal Engine." *ACM Multimedia Open Source Software Competition*, 2016.

[QZ18] Siyuan Qi and Song-Chun Zhu. "Intent-aware Multi-agent Reinforcement Learning." In *International Conference on Robotics and Automation (ICRA)*, 2018.

[QZH18] Siyuan Qi, Yixin Zhu, Siyuan Huang, Chenfanfu Jiang, and Song-Chun Zhu. "Human-centric indoor scene synthesis using stochastic grammar." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[RAA12] Marcus Rohrbach, Sikandar Amin, Mykhaylo Andriluka, and Bernt Schiele. "A database for fine grained activity detection of cooking activities." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[Rec85] Mark D Reckase. "The difficulty of test items that measure more than one ability." *Applied psychological measurement*, 1985.

[Rec09] Mark D Reckase. "Multidimensional item response theory models." In *Multidimensional item response theory*. Springer, 2009.

[REM16] Danilo Jimenez Rezende, SM Ali Eslami, Shakir Mohamed, Peter Battaglia, Max Jaderberg, and Nicolas Heess. "Unsupervised learning of 3d structure from images." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[RHA16] Vignesh Ramanathan, Jonathan Huang, Sami Abu-El-Haija, Alexander Gorban, Kevin Murphy, and Li Fei-Fei. "Detecting events and key actors in multi-person videos." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[RHG15] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

[Rie49] Morris D. Riemer. "The averted gaze." *Psychiatric Quarterly*, **23**(1):108–115, 1949.

[RKS12] Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. "Reconstructing 3d human pose from 2d image landmarks." In *European Conference on Computer Vision (ECCV)*, 2012.

[RLC16] Yuzhuo Ren, Shangwen Li, Chen Chen, and C-C Jay Kuo. "A coarse-to-fine indoor layout estimation (CFILE) method." In *Asian Conference on Computer Vision (ACCV)*, 2016.

[RME01] Joshua S Rubinstein, David E Meyer, and Jeffrey E Evans. "Executive control of cognitive processes in task switching." *Journal of experimental psychology: human perception and performance*, **27**(4):763, 2001.

[RMG15] Daniel Ritchie, Ben Mildenhall, Noah D Goodman, and Pat Hanrahan. "Controlling procedural modeling programs with stochastically-ordered sequential monte carlo." *ACM Transactions on Graphics (TOG)*, 2015.

[RRR16] Marcus Rohrbach, Anna Rohrbach, Michaela Regneri, Sikandar Amin, Mykhaylo Andriluka, Manfred Pinkal, and Bernt Schiele. "Recognizing fine-grained and composite activities using hand-centric features and script data." *International Journal of Computer Vision (IJCV)*, **119**(3):346–373, 2016.

[RS16] Zhile Ren and Erik B Sudderth. "Three-dimensional object detection and layout prediction using clouds of oriented gradients." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[RVR16] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. "Playing for data: Ground truth from computer games." In *European Conference on Computer Vision (ECCV)*, 2016.

[RZL16] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. "Squad: 100,000+ questions for machine comprehension of text." In *Empirical Methods in Natural Language Processing (EMNLP)*, 2016.

[Sac16] Mrinmaya Sachan et al. "Easy questions first? a case study on curriculum learning for question answering." In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2016.

[SAJ10] Valentin I Spitkovsky, Hiyan Alshawi, and Daniel Jurafsky. "From baby steps to leapfrog: How less is more in unsupervised dependency parsing." In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 751–759. Association for Computational Linguistics, 2010.

[SBL15] Julian Straub, Nishchal Bhandari, John J Leonard, and John W Fisher. "Real-time Manhattan world rotation estimation in 3D." In *International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[SC09] Maria Staudte and Matthew W. Crocker. "Visual attention in spoken human-robot interaction." In *HRI*, 2009.

[SC11] Maria Staudte and Matthew W. Crocker. "Investigating joint attention mechanisms through spoken human-robot interaction." *Cognition*, **120**(2):268–291, 2011.

[SCH14] Manolis Savva, Angel X Chang, Pat Hanrahan, Matthew Fisher, and Matthias Nießner. "SceneGrok: Inferring action maps in 3D environments." *ACM Transactions on Graphics (TOG)*, **33**(6):212, 2014.

[SCH15] Manolis Savva, Angel X Chang, and Pat Hanrahan. "Semantically-enriched 3D models for common-sense knowledge." In *CVPR Workshop*, 2015.

[SCH16] Manolis Savva, Angel X. Chang, Pat Hanrahan, Matthew Fisher, and Matthias Nießner. "PiGraphs: Learning Interaction Snapshots from Observations." *ACM Transactions on Graphics (TOG)*, **35**(4), 2016.

[SCN06] Ashutosh Saxena, Sung H Chung, and Andrew Y Ng. "Learning depth from single monocular images." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2006.

[SCS13] Amy E Skerry, Susan E Carey, and Elizabeth S Spelke. "First-person action experience reveals sensitivity to action efficiency in prereaching infants." *Proceedings of the National Academy of Sciences (PNAS)*, 2013.

[SD04] Grant Schindler and Frank Dellaert. "Atlanta world: An expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[SDL17] Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. "Aerial Informatics and Robotics Platform." Technical report, Microsoft Research, 2017.

[SF15] Aimee E Stahl and Lisa Feigenson. "Observing the unexpected enhances infants' learning and exploration." *Science*, **348**(6230):91–94, 2015.

[SFP13] Alexander G Schwing, Sanja Fidler, Marc Pollefeys, and Raquel Urtasun. "Box in the box: Joint 3d layout and object reasoning from single images." In *International Conference on Computer Vision (ICCV)*, 2013.

[SG09] Dave Shreiner, Bill The Khronos OpenGL ARB Working Group, et al. *OpenGL programming guide: the official guide to learning OpenGL, versions 3.0 and 3.1*. Pearson Education, 2009.

[SGS18] Gunnar A Sigurdsson, Abhinav Gupta, Cordelia Schmid, Ali Farhadi, and Karteek Alahari. "Charades-ego: A large-scale dataset of paired third and first person videos." *arXiv preprint arXiv:1804.09626*, 2018.

[SGT09] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. "The graph neural network model." *IEEE TNNLS*, **20**(1):61–80, 2009.

[She10] Stephen Shepherd. "Following Gaze: Gaze-Following Behavior as a Window into Social Cognition." *Frontiers in integrative neuroscience*, **4**:5, 2010.

[SHK12] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. "Indoor segmentation and support inference from rgbd images." In *European Conference on Computer Vision (ECCV)*, 2012.

[SHM14] Hao Su, Qixing Huang, Niloy J Mitra, Yangyan Li, and Leonidas Guibas. "Estimating image depth using shape collections." *ACM Transactions on Graphics (TOG)*, 2014.

[SHP12] Alexander G Schwing, Tamir Hazan, Marc Pollefeys, and Raquel Urtasun. "Efficient structured prediction for 3d indoor scene understanding." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[SJC06] Atsushi Senju, Mark H. Johnson, and Gergely Csibra. "The development and neural basis of referential gaze perception." *Social neuroscience*, **1**(3-4):220–234, 2006.

[SK07] Elizabeth S Spelke and Katherine D Kinzler. "Core knowledge." *Developmental Science*, **10**(1):89–96, 2007.

[SK17] Martin Simonovsky and Nikos Komodakis. "Dynamic edgeconditioned filters in convolutional neural networks on graphs." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[Ski58] Burrhus F Skinner. "Reinforcement today." *American Psychologist*, 1958.

[SKM19] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. "Habitat: A platform for embodied ai research." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9339–9347, 2019.

[SLJ15] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[SLX15] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. "Sun RGB-D: A RGB-D scene understanding benchmark suite." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[SM11] Vasant Srinivasan and Robin R. Murphy. "A survey of social gaze." In *HRI*, 2011.

[SM13] Sebastian Stein and Stephen J McKenna. "Combining embedded accelerometers with computer vision for recognizing food preparation activities." In *ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2013.

[SNS13] Renato F Salas-Moreno, Richard A Newcombe, Hauke Strasdat, Paul HJ Kelly, and Andrew J Davison. "Slam++: Simultaneous localisation and mapping at the level of objects." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[SO94] Philippe G Schyns and Aude Oliva. "From blobs to boundary edges: Evidence for time-and spatial-scale-dependent scene recognition." *Psychological science*, **5**(4):195–200, 1994.

[Soa13] Stefano Soatto. “Actionable information in vision.” In *Machine learning for computer vision*, pp. 17–48. Springer, 2013.

[SQL15] Hao Su, Charles R Qi, Yangyan Li, and Leonidas J Guibas. “Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views.” In *International Conference on Computer Vision (ICCV)*, 2015.

[SRA12] Edgar Simo-Serra, Arnau Ramisa, Guillem Alenyà, Carme Torras, and Francesc Moreno-Noguer. “Single image 3D human pose estimation from noisy observations.” In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[SRB17] Adam Santoro, David Raposo, David G. Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Tim Lillicrap. “A simple neural network module for relational reasoning.” In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[SRF14] Julian Straub, Guy Rosman, Oren Freifeld, John J Leonard, and John W Fisher. “A mixture of manhattan frames: Beyond the manhattan world.” In *International Conference on Computer Vision (ICCV)*, 2014.

[SS14] Baochen Sun and Kate Saenko. “From Virtual to Reality: Fast Adaptation of Virtual Object Detectors to Real Domains.” In *British Machine Vision Conference (BMVC)*, 2014.

[SSF16] Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. “Learning multiagent communication with backpropagation.” In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[SST18] Supasorn Suwajanakorn, Noah Snavely, Jonathan J Tompson, and Mohammad Norouzi. “Discovery of latent 3d keypoints via end-to-end geometric reasoning.” In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[ST05] Wei Shao and Demetri Terzopoulos. “Autonomous pedestrians.” In *ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA)*, 2005.

[STN16] Noor Shaker, Julian Togelius, and Mark J Nelson. *Procedural Content Generation in Games.* Springer, 2016.

[Sun94] Ron Sun. *Integrating rules and connectionism for robust commonsense reasoning.* John Wiley & Sons, Inc., 1994.

[SVD03] Gregory Shakhnarovich, Paul Viola, and Trevor Darrell. “Fast pose estimation with parameter-sensitive hashing.” In *International Conference on Computer Vision (ICCV)*, 2003.

[SVL14] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. “Sequence to sequence learning with neural networks.” In *Advances in neural information processing systems*, 2014.

[SVW16] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. "Hollywood in homes: Crowdsourcing data collection for activity understanding." In *European Conference on Computer Vision (ECCV)*, 2016.

[SX14] Shuran Song and Jianxiong Xiao. "Sliding shapes for 3d object detection in depth images." In *European Conference on Computer Vision (ECCV)*, 2014.

[SX16] Shuran Song and Jianxiong Xiao. "Deep sliding shapes for amodal 3D object detection in RGB-D images." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[SXR15] Tianmin Shu, Dan Xie, Brandon Rothrock, Sinisa Todorovic, and Song Chun Zhu. "Joint inference of groups, events and human roles in aerial videos." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[SYZ17a] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. "Semantic scene completion from a single depth image." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[SYZ17b] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. "Semantic Scene Completion from a Single Depth Image." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[SZ14] Karen Simonyan and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556*, 2014.

[SZS12] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. "UCF101: A dataset of 101 human actions classes from videos in the wild." *arXiv preprint arXiv:1212.0402*, 2012.

[TCH17] Sam Toyer, Anoop Cherian, Tengda Han, and Stephen Gould. "Human pose forecasting via deep markov models." In *International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, 2017.

[TCS08] Pavan Turaga, Rama Chellappa, Venkatramana S Subrahmanian, and Octavian Udrea. "Machine recognition of human activities: A survey." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **18**(11):1473–1488, 2008.

[TCY05] Zhuowen Tu, Xiangrong Chen, Alan L Yuille, and Song-Chun Zhu. "Image parsing: Unifying segmentation, detection, and recognition." *International Journal of computer vision*, 2005.

[TDR19] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. "COIN: A large-scale dataset for comprehensive instructional video analysis." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[TFL16] Yulia Tsvetkov, Manaal Faruqui, Wang Ling, Brian MacWhinney, and Chris Dyer. "Learning the curriculum with bayesian optimization for task-specific word representation learning." *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2016.

[TFM96] Simon Thorpe, Denis Fize, and Catherine Marlot. "Speed of processing in the human visual system." *Nature*, **381**(6582):520, 1996.

[TGF18] Shubham Tulsiani, Saurabh Gupta, David Fouhey, Alexei A Efros, and Jitendra Malik. "Factoring shape, pose, and layout from the 2D image of a 3D scene." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[THK87] Nancy Termine, Timothy Hrynick, Roberta Kestenbaum, Henry Gleitman, and Elizabeth S Spelke. "Perceptual completion of surfaces in infancy." *Journal of Experimental Psychology: Human Perception and Performance*, **13**(4):524, 1987.

[TLL11] Jerry O Talton, Yu Lou, Steve Lesser, Jared Duke, Radomír Měch, and Vladlen Koltun. "Metropolis procedural modeling." *ACM Transactions on Graphics (TOG)*, 2011.

[TM15] Shubham Tulsiani and Jitendra Malik. "Viewpoints and keypoints." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[TML14] Kewei Tu, Meng Meng, Mun Wai Lee, Tae Eun Choe, and Song-Chun Zhu. "Joint video and text parsing for understanding events and answering queries." *IEEE MultiMedia*, 2014.

[TMS07] Adriana Tapus, Maja J. Mataric, and Brian Scassellati. "The Grand Challenges in Socially Assistive Robotics." *IEEE Robotics & Automation Magazine*, **14**(1):35–42, 2007.

[Tom10] Michael Tomasello. *Origins of human communication.* MIT Press, 2010.

[TRA17] Denis Tome, Christopher Russell, and Lourdes Agapito. "Lifting from the deep: Convolutional 3d pose estimation from a single image." *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[TSF18] Bugra Tekin, Sudipta N Sinha, and Pascal Fua. "Real-time seamless single shot 6d object pose prediction." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[TZS16] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. "Movieqa: Understanding stories in movies through question-answering." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[VBC19] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. "Grandmaster level in StarCraft II using multi-agent reinforcement learning." *Nature*, **575**(7782):350–354, 2019.

[VCC18] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. "Graph attention networks." In *International Conference on Learning Representations (ICLR)*, 2018.

[VDL19] Ramakrishna Vedantam, Karan Desai, Stefan Lee, Marcus Rohrbach, Dhruv Batra, and Devi Parikh. "Probabilistic neural-symbolic models for interpretable visual question answering." *arXiv preprint arXiv:1902.07864*, 2019.

[VFJ15] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. "Pointer Networks." In *NIPS*, 2015.

[VPR13] Carl Vondrick, Donald Patterson, and Deva Ramanan. "Efficiently scaling up crowdsourced video annotation." *International Journal of Computer Vision (IJCV)*, **101**(1):184–204, 2013.

[VSP17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. "Attention is all you need." *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[VVG20] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. "Scan: Learning to classify images without labels." In *European Conference on Computer Vision (ECCV)*, 2020.

[WA93] John YA Wang and Edward H Adelson. "Layered representation for motion analysis." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 1993.

[WA94] John YA Wang and Edward H Adelson. "Representing moving images with layers." *Transactions on Image Processing (TIP)*, **3**(5):625–638, 1994.

[Wal75] David Waltz. "Understanding line drawings of scenes with shadows." In *The psychology of computer vision*, 1975.

[WEK12] Johan Wagemans, James H Elder, Michael Kubovy, Stephen E Palmer, Mary A Peterson, Manish Singh, and Rüdiger von der Heydt. "A century of Gestalt psychology in visual perception: I. Perceptual grouping and figure–ground organization." *Psychological bulletin*, **138**(6):1172, 2012.

[Wer12] Max Wertheimer. "Experimentelle studien uber das sehen von bewegung [Experimental studies on the seeing of motion]." *Zeitschrift fur Psychologie*, **61**:161–265, 1912.

[Wer23] Max Wertheimer. "Untersuchungen zur Lehre von der Gestalt, II. [Investigations in Gestalt Theory: II. Laws of organization in perceptual forms]." *Psychologische Forschung*, **4**:301–350, 1923.

[Wer38] Max Wertheimer. "Laws of organization in perceptual forms." In *A source book of Gestalt psychology*, pp. 71–94. London, England: Routledge & Kegan Paul, 1938.

[WFF19] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. "Long-term feature banks for detailed video understanding." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[WFG12] Johan Wagemans, Jacob Feldman, Sergei Gepshtein, Ruth Kimchi, James R Pomerantz, Peter A Van der Helm, and Cees Van Leeuwen. "A century of Gestalt psychology in visual perception: II. Conceptual and theoretical foundations." *Psychological bulletin*, **138**(6):1218, 2012.

[Wil92] Ronald J Williams. "Simple statistical gradient-following algorithms for connectionist reinforcement learning." *Machine learning*, 1992.

[WLF19] Wenguan Wang, Qiuxia Lai, Huazhu Fu, Jianbing Shen, and Haibin Ling. "Salient Object Detection in the Deep Learning Era: An In-Depth Survey." *arXiv preprint arXiv:1904.09146*, 2019.

[Woo98] Amanda L Woodward. "Infants selectively encode the goal object of an actor's reach." *Cognition*, **69**(1):1–34, 1998.

[Woo99] Amanda L Woodward. "infants' ability to distinguish between purposeful and non-purposeful behaviors." *Infant Behavior and Development*, **22**(2):145–160, 1999.

[WS18] Wenguan Wang and Jianbing Shen. "Deep visual attention prediction." *Transactions on Image Processing (TIP)*, **27**(5):2368–2378, 2018.

[WSG10] Ying Nian Wu, Zhangzhang Si, Haifeng Gong, and Song-Chun Zhu. "Learning active basis model for object detection and recognition." *International Journal of Computer Vision (IJCV)*, **90**(2):198–235, 2010.

[WT11] Max Welling and Yee W Teh. "Bayesian learning via stochastic gradient Langevin dynamics." In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 681–688, 2011.

[Wu18] Lijun Wu et al. "Learning to teach with dynamic loss functions." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[WWX17] Jiajun Wu, Yifan Wang, Tianfan Xue, Xingyuan Sun, Bill Freeman, and Josh Tenenbaum. "Marrnet: 3d shape reconstruction via 2.5 d sketches." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[WXL16] Jiajun Wu, Tianfan Xue, Joseph J Lim, Yuandong Tian, Joshua B Tenenbaum, Antonio Torralba, and William T Freeman. "Single image 3d interpreter network." In *European Conference on Computer Vision (ECCV)*, 2016.

[WXS18] Wenguan Wang, Yuanlu Xu, Jianbing Shen, and Song-Chun Zhu. "Attentive Fashion Grammar Network for Fashion Landmark Detection and Clothing Category Classification." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[WYL15] Jiajun Wu, Ilker Yildirim, Joseph J Lim, Bill Freeman, and Josh Tenenbaum. "Galileo: Perceiving physical object properties by integrating a physics engine with deep learning." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

[WZG18] Lei Wang, Dongxiang Zhang, Lianli Gao, Jingkuan Song, Long Guo, and Heng Tao Shen. "Mathdqn: Solving arithmetic word problems via deep reinforcement learning." In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[WZS15] Chenxia Wu, Jiemi Zhang, Silvio Savarese, and Ashutosh Saxena. "Watch-n-patch: Unsupervised understanding of actions and relations." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[WZZ13] Ping Wei, Yibiao Zhao, Nanning Zheng, and Song-Chun Zhu. "Modeling 4d human-object interactions for event and object recognition." In *International Conference on Computer Vision (ICCV)*, 2013.

[XC18] Bin Xu and Zhenzhong Chen. "Multi-level fusion based 3d object detection from monocular images." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[XF14] Jianxiong Xiao and Yasutaka Furukawa. "Reconstructing the world's museums." *International Journal of Computer Vision (IJCV)*, 2014.

[XHR13] Jianxiong Xiao, James Hays, Bryan C Russell, Genevieve Patterson, Krista Ehinger, Antonio Torralba, and Aude Oliva. "Basic level scene understanding: categories, attributes and structures." *Frontiers in psychology*, **4**:506, 2013.

[XLE18] Xu Xie, Hangxin Liu, Mark Edmonds, Feng Gao, Siyuan Qi, Yixin Zhu, Brandon Rothrock, and Song-Chun Zhu. "Unsupervised Learning of Hierarchical Models for Hand-Object Interactions." In *ICRA*, 2018.

[XLZ20a] Keyulu Xu, Jingling Li, Mozhi Zhang, Simon S Du, Ken-ichi Kawarabayashi, and Stefanie Jegelka. "How neural networks extrapolate: From feedforward to graph neural networks." *arXiv preprint arXiv:2009.11848*, 2020.

[XLZ20b] Keyulu Xu, Jingling Li, Mozhi Zhang, Simon S Du, Ken-ichi Kawarabayashi, and Stefanie Jegelka. "What can neural networks reason about?" In *International Conference on Learning Representations (ICLR)*, 2020.

[XMY21] Sirui Xie, Xiaojian Ma, Peiyu Yu, Yixin Zhu, Ying Nian Wu, and Song-Chun Zhu. "HALMA: Humanlike Abstraction Learning Meets Affordance in Rapid Problem Solving." *arXiv preprint arXiv:2102.11344*, 2021.

[XTZ13] Dan Xie, Sinisa Todorovic, and Song-Chun Zhu. "Inferring "dark matter" and "dark energy" from videos." In *International Conference on Computer Vision (ICCV)*, 2013.

[XZH18] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. “Gibson env: Real-world perception for embodied agents.” In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[YCX09] Qiang Yang, Yuqiang Chen, Gui-Rong Xue, Wenyuan Dai, and Yong Yu. “Heterogeneous transfer learning for image clustering via the social web.” In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th international joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pp. 1–9. Association for Computational Linguistics, 2009.

[YGL20] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. “Clevrer: Collision events for video representation and reasoning.” *ICLR*, 2020.

[YHZ18] Shunyu Yao, Tzu Ming Hsu, Jun-Yan Zhu, Jiajun Wu, Antonio Torralba, Bill Freeman, and Josh Tenenbaum. “3D-aware scene manipulation via inverse graphics.” In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[YIK16] Hashim Yasin, Umar Iqbal, Björn Krüger, Andreas Weber, and Juergen Gall. “A Dual-Source Approach for 3D Pose Estimation from a Single Image.” In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[YJK11] Bangpeng Yao, Xiaoye Jiang, Aditya Khosla, Andy Lai Lin, Leonidas Guibas, and Li Fei-Fei. “Human action recognition by learning bases of action attributes and parts.” In *International Conference on Computer Vision (ICCV)*, 2011.

[YK06] Alan Yuille and Daniel Kersten. “Vision as Bayesian inference: analysis by synthesis?” *Trends in cognitive sciences*, **10**(7):301–308, 2006.

[YLF20] Tao Yuan, Hangxin Liu, Lifeng Fan, Zilong Zheng, Tao Gao, Yixin Zhu, and Song-Chun Zhu. “Joint Inference of States, Robot Knowledge, and Human (False-)Beliefs.” In *International Conference on Robotics and Automation (ICRA)*, 2020.

[YQK17] Tian Ye, Siyuan Qi, James Kubricht, Yixin Zhu, Hongjing Lu, and Song-Chun Zhu. “The Martian: Examining Human Physical Judgments across Virtual Gravity Fields.” *IEEE Transactions on Visualization and Computer Graph (TVCG)*, 2017.

[YRJ18] Serena Yeung, Olga Russakovsky, Ning Jin, Mykhaylo Andriluka, Greg Mori, and Li Fei-Fei. “Every moment counts: Dense detailed labeling of actions in complex videos.” *International Journal of Computer Vision (IJCV)*, **126**(2-4):375–389, 2018.

[YWG18] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. “Neural-symbolic vqa: Disentangling reasoning from vision and language understanding.” In *Advances in Neural Information Processing Systems*, 2018.

[YYT11] Lap Fai Yu, Sai Kit Yeung, Chi Keung Tang, Demetri Terzopoulos, Tony F Chan, and Stanley J Osher. "Make it home: automatic optimization of furniture arrangement." *ACM Transactions on Graphics (TOG)*, 2011.

[YYW12] Yi-Ting Yeh, Lingfeng Yang, Matthew Watson, Noah D Goodman, and Pat Hanrahan. "Synthesizing open worlds with constraints using locally annealed reversible jump mcmc." *ACM Transactions on Graphics (TOG)*, 2012.

[YYY16] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. "Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[YZH18] Pengcheng Yin, Chunting Zhou, Junxian He, and Graham Neubig. "Struct-VAE: Tree-structured latent variable models for semi-supervised semantic parsing." *arXiv preprint arXiv:1806.07832*, 2018.

[ZBK17] Yinda Zhang, Mingru Bai, Pushmeet Kohli, Shahram Izadi, and Jianxiong Xiao. "Deepcontext: Context-encoding neural pathways for 3d holistic scene understanding." In *International Conference on Computer Vision (ICCV)*, 2017.

[ZCN17] Quanshi Zhang, Ruiming Cao, Ying Nian Wu, and Song-Chun Zhu. "Mining object parts from cnns via active question-answering." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[ZCS18] Chuhang Zou, Alex Colburn, Qi Shan, and Derek Hoiem. "LayoutNet: Reconstructing the 3D Room Layout from a Single RGB Image." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[ZGF20] Yixin Zhu, Tao Gao, Lifeng Fan, Siyuan Huang, Mark Edmonds, Hangxin Liu, Feng Gao, Chi Zhang, Siyuan Qi, Ying Nian Wu, et al. "Dark, beyond deep: A paradigm shift to cognitive ai with humanlike common sense." *Engineering*, **6**(3):310–345, 2020.

[ZGH21] Yaxuan Zhu, Ruiqi Gao, Siyuan Huang, Song-Chun Zhu, and Ying Nian Wu. "Learning Neural Representation of Camera Pose with Matrix Representation of Pose Shift via View Synthesis." *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[ZGW05] Song-Chun Zhu, Cheng-En Guo, Yizhou Wang, and Zijian Xu. "What are textons?" *International Journal of Computer Vision (IJCV)*, **62**(1-2):121–143, 2005.

[Zho19] Zhi-Hua Zhou. "Abductive learning: towards bridging machine learning and logical reasoning." *Science China Information Sciences*, **62**:1–3, 2019.

[Zhu15] Xiaojin Zhu. "Machine teaching: An inverse problem to machine learning and an approach toward optimal education." In *AAAI Conference on Artificial Intelligence (AAAI)*, 2015.

[ZJR15] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. "Conditional random fields as recurrent neural networks." In *International Conference on Computer Vision (ICCV)*, 2015.

[ZJZ16] Yixin Zhu, Chenfanfu Jiang, Yibiao Zhao, Demetri Terzopoulos, and Song-Chun Zhu. "Inferring forces and learning human utilities from videos." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[ZKA16] Tinghui Zhou, Philipp Krähenbühl, Mathieu Aubry, Qixing Huang, and Alexei A. Efros. "Learning Dense Correspondence via 3D-guided Cycle Consistency." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[ZLH17] Chuhang Zou, Zhizhong Li, and Derek Hoiem. "Complete 3D Scene Parsing from Single RGBD Image." *arXiv preprint arXiv:1710.09490*, 2017.

[ZLX14] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. "Learning deep features for scene recognition using places database." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

[ZLY17] Hao Zhao, Ming Lu, Anbang Yao, Yiwen Guo, Yurong Chen, and Li Zhang. "Physics Inspired Optimization on Semantic Transfer Features: An Alternative Method for Room Layout Estimation." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[ZM07] Song-Chun Zhu, David Mumford, et al. "A stochastic grammar of images." *Foundations and Trends in Computer Graphics and Vision*, 2007.

[ZMS18] Andrei Zanfir, Elisabeta Marinoiu, and Cristian Sminchisescu. "Monocular 3D Pose and Shape Estimation of Multiple People in Natural Scenes–The Importance of Multiple Scene Constraints." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[ZRH20] Quanshi Zhang, Jie Ren, Ge Huang, Ruiming Cao, Ying Nian Wu, and Song-Chun Zhu. "Mining Interpretable AOG Representations from Convolutional Networks via Active Question Answering." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

[ZSQ17] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. "Pyramid scene parsing network." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[ZST14] Yinda Zhang, Shuran Song, Ping Tan, and Jianxiong Xiao. "Panocontext: A whole-room 3d context model for panoramic scene understanding." In *European Conference on Computer Vision (ECCV)*, 2014.

[ZSY17a] Yinda Zhang, Shuran Song, Ersin Yumer, Manolis Savva, Joon-Young Lee, Hailin Jin, and Thomas Funkhouser. "Physically-based rendering for indoor scene understanding using convolutional neural networks." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[ZSY17b] Yinda Zhang, Shuran Song, Ersin Yumer, Manolis Savva, Joon-Young Lee, Hailin Jin, and Thomas Funkhouser. "Physically-Based Rendering for Indoor Scene Understanding Using Convolutional Neural Networks." *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[ZSZ18] Xiaojin Zhu, Adish Singla, Sandra Zilles, and Anna N Rafferty. "An overview of machine teaching." *arXiv preprint arXiv:1801.05927*, 2018.

[Zub08] Klaus Zuberbühler. "Gaze following." *Current Biology*, **18**(11):R453–R455, 2008.

[ZWM98] Song Chun Zhu, Yingnian Wu, and David Mumford. "Filters, random fields and maximum entropy (FRAME): Towards a unified theory for texture modeling." *International Journal of Computer Vision (IJCV)*, **27**(2):107–126, 1998.

[ZWM17] Ruiqi Zhao, Yan Wang, and AM Martinez. "A Simple, Fast and Highly-Accurate Algorithm to Recover 3D Shape from 2D Landmarks on a Single Image." *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **40**(12):3059–3066, 2017.

[ZXC18] Luowei Zhou, Chenliang Xu, and Jason J Corso. "Towards automatic learning of procedures from web instructional videos." In *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[ZYS15] Yinda Zhang, Fisher Yu, Shuran Song, Pingmei Xu, Ari Seff, and Jianxiong Xiao. "Large-scale scene understanding challenge: Room layout estimation." In *CVPR Workshop*, 2015.

[ZZ11] Yibiao Zhao and Song-Chun Zhu. "Image parsing with stochastic scene grammar." In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2011.

[ZZ13] Yibiao Zhao and Song-Chun Zhu. "Scene parsing by integrating function, geometry and appearance models." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[ZZC15] Yixin Zhu, Yibiao Zhao, and Song Chun Zhu. "Understanding tools: Task-oriented object modeling, learning and recognition." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[ZZJ14] Bo Zheng, Yibiao Zhao, C Yu Joey, Katsushi Ikeuchi, and Song-Chun Zhu. "Detecting potential falling objects by inferring human action and natural disturbance." In *International Conference on Robotics and Automation (ICRA)*, 2014.

[ZZY13] Bo Zheng, Yibiao Zhao, Joey C Yu, Katsushi Ikeuchi, and Song-Chun Zhu. "Beyond point clouds: Scene understanding by reasoning geometry and physics." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[ZZY15] Bo Zheng, Yibiao Zhao, Joey Yu, Katsushi Ikeuchi, and Song-Chun Zhu. “Scene understanding by reasoning stability and safety.” *International Journal of Computer Vision (IJCV)*, 2015.

[ZZZ18a] Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Joshua B Tenenbaum, William T Freeman, and Jiajun Wu. “Learning to Reconstruct Shapes from Unseen Classes.” In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[ZZZ18b] Jun-Yan Zhu, Zhoutong Zhang, Chengkai Zhang, Jiajun Wu, Antonio Torralba, Joshua B. Tenenbaum, and William T. Freeman. “Visual Object Networks: Image Generation with Disentangled 3D Representations.” In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2018.