

Indoor Scene Understanding in 2.5/3D: A Survey

Muzammal Naseer, Salman H Khan, Fatih Porikli

Abstract—With the availability of low-cost and compact 2.5/3D visual sensing devices, computer vision community is experiencing a growing interest in visual scene understanding. This survey paper provides a comprehensive background to this research topic. We begin with a historical perspective, followed by popular 3D data representations and a comparative analysis of available datasets. Before delving into the application specific details, this survey provides a succinct introduction to the core technologies that are the underlying methods extensively used in the literature. Afterwards, we review the developed techniques according to a taxonomy based on the scene understanding tasks. This covers holistic indoor scene understanding as well as subtasks such as scene classification, object detection, pose estimation, semantic segmentation, 3D reconstruction, saliency detection, physics-based reasoning and affordance prediction. Later on, we summarize the performance metrics used for evaluation in different tasks and a quantitative comparison among the recent state-of-the-art techniques. We conclude this review with the current challenges and an outlook towards the open research problems requiring further investigation.

Index Terms—3D scene understanding, semantic labeling, geometry estimation, deep networks, Markov random fields.

1 INTRODUCTION

It's not what you look at that matters, it's what you see.

H.D. Thoreau (1817-62)

A n image is simply a grid of numbers to a machine. In order to develop a comprehensive understanding of the visual content, it is necessary to uncover the underlying geometric and semantic clues. As an example, given an RGB-D (2.5D) image, a vision-based AI agent should be able to understand the complete 3D spatial layout, functional attributes and semantic labels of the scene and its constituent objects. Furthermore, it is also required to unseal both the apparent and hidden relationships present between scene elements. These capabilities are fundamental to the way humans perceive and interpret images, thus imparting these astounding abilities in machines has been a long-standing goal in computer vision discipline. We can formally define visual scene understanding problem in machine vision as follows:

Scene Understanding: “*To analyze a scene by considering the geometric and semantic context of its contents and the intrinsic relationships between them.*”

Visual scene understanding can be broadly divided into two categories based on the input media: static (for an image) and dynamic (for a video) understanding. This survey specifically attends to static scene understanding of 2.5/3D visual data for indoor scenes. We focus on the 3D media since the 3D scene understanding capabilities are central to the development of general-purpose AI agents that can be deployed for emerging application areas as diverse as autonomous vehicles [1], domestic robotics [2], health-care

systems [3], education [4], environment preservation [5] and infotainment [6]. According to an estimate from WHO, there are 253 million people suffering from vision impairment [7]. 3D scene understanding can help them safely navigate by detecting obstacles and analyzing the terrain [8]. Domestic robots with cognitive abilities can be used to take care of elderly people, whose number is expected to reach 1.5 billion by the year 2050.

As much as being highly significant, 3D scene understanding is also remarkably challenging due to the complex interactions between objects, heavy occlusions, cluttered indoor environments, major appearance, viewpoint and scale changes across different scenes and the inherent ambiguity in the limited information provided by a static scene. Recent developments in large-scale data-driven models, fueled by the availability of big annotated datasets have sparked a renewed interest in addressing these challenges. This survey aims to provide an inclusive background to this field, with a review of the competing methods developed recently. Our intention is not only to explore the existing wealth of knowledge but also to identify the key areas lacking substantial interest from the community and the potential future directions crucial for the development of practical AI-based systems. To this end, we cover both the specific problem domains under the umbrella of scene understanding as well as the underlying computational tools that have been used to develop state-of-the-art solutions to various scene analysis problems (Fig. 1). To the best of our knowledge, this is the first review that broadly summarizes the progress and promising new directions in 2.5/3D indoor scene understanding. We believe this contribution will serve as a helpful reference to the community.

2 A BRIEF HISTORY OF 3D SCENE ANALYSIS

There exists a fundamental difference in the way a machine and a human would perceive the visual content. An image or a video is, in essence, a tensor with numeric values representing color (e.g., r, g and b channels) or location (e.g.,

- M. Naseer, S. H. Khan and F. Porikli are with the Australian National University (ANU), Canberra ACT.
E-mail: {muzammal.naseer, fatih.porikli}@anu.edu.au
- S. H. Khan is also with Data61, Commonwealth Scientific and Industrial Research Organization (CSIRO)
E-mail: salman.khan@data61.csiro.au

Manuscript received April 25, 2014; revised April 28, 2015.

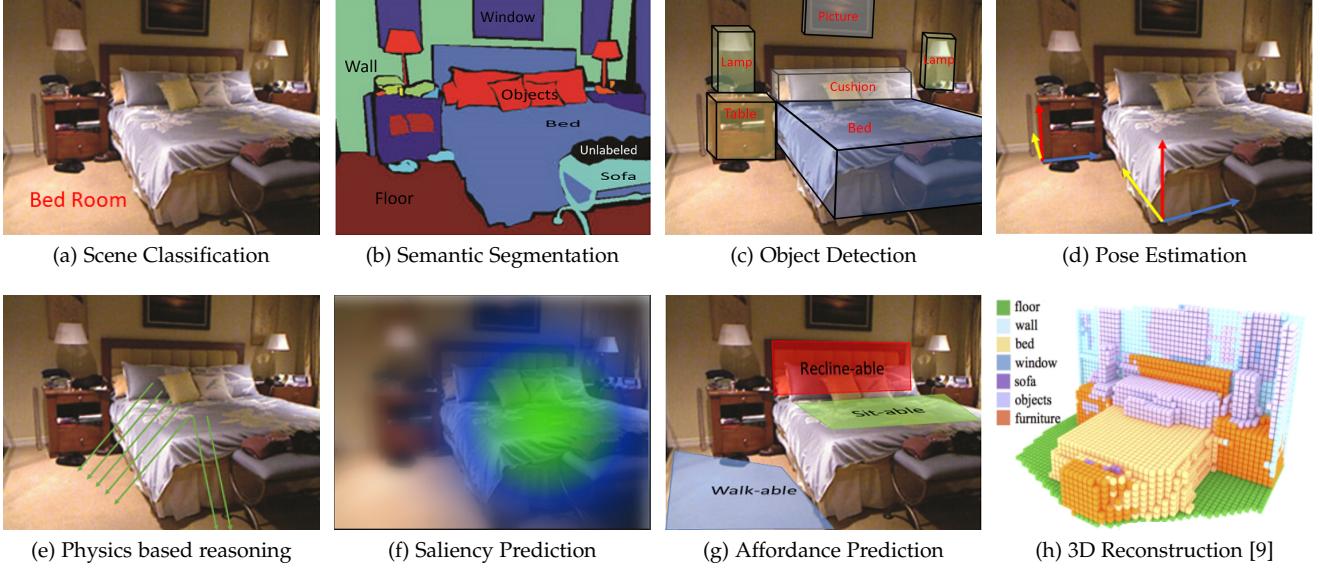


Fig. 1: Given a RGB-D image, visual scene understanding can involve the image and pixel level semantic labeling (a-b), 3D object detection and pose estimation (c-d), inferring physical relationships (e), identifying salient regions (f), predicting affordances (g), full 3D reconstruction (h) and holistic reasoning about multiple such tasks (sample image from the NYU-Depth dataset [10]).

x, y and z coordinates) information. An obvious way of processing such information is to compute local features representing color and texture characteristics. To this end, a number of local feature descriptors have been designed over the years to faithfully encode visual information. Some of these include e.g., SIFT [11], HOG [12], SURF [13], Region Covariance [14] and LBP [15] to name a few. The human visual system not only perceives the local visual details but also cognitively reasons about semantics and geometry in a scene, and can understand complex relationships between objects. Efforts have been made to replicate these remarkable visual capabilities in machine vision for advanced applications such as context-aware personal digital assistants, health-care and domestic robotic systems, content-driven retrieval and assistive devices for visually impaired.

Initial work on scene understanding was motivated by the human cognitive psychology and neuroscience. In this regard, several notable ideas were put forward to explain the working of the human visual system. In 1867, Helmholtz [16] explained his concept of ‘unconscious conclusion,’ which attributes the involuntary visual perception to our longstanding previous interactions with the 3D surroundings. In 1920s, Gestalt theory argued that the holistic interpretation of a scene developed by humans is due to eight main factors, the prominent ones being proximity, closure, and common motion [17]. Barrow and Tenenbaum [18] introduced the idea of ‘intrinsic images,’ which are layers of visual information a human can easily extract from a given scene. These include illumination, reflectance, depth, and orientation. Around half a century ago, Marr proposed his three-level vision theory, which transitions from a 2D primal sketch of a scene (consisting of edges and regions), first to a 2.5D sketch (consisting of texture and orientations) and finally to a 3D model which encodes complete shape of a scene [19].

Representation is a key element of understanding the 3D world around us. In the early days of computer vision, researchers favored parts-based representations for object description and scene understanding. One of the initial efforts in this regard was made by L.G. Roberts [20], who presented an approach to denote objects using a set of 3D polyhedral shapes. Afterwards, a set of object parts was identified by A. Guzman [21] as the primitives for representing generic 2D shapes in line-drawings. Another seminal idea was put forward by T. Binford, who demonstrated that several curved objects could be represented using generalized cylinders [22]. Based on the generalized cylinders, a pioneering contribution was made by I. Biederman, who introduced a set of basic primitives (termed as ‘geons’ meaning geometrical ions) and linked it with the object recognition in human cognitive system [23]. Recently, data-driven feature representations learned using deep neural networks have been shown to perform superior for describing visual data [24], [25], [26], [27].

Apart from geometric descriptions, understanding scene semantics has remained an important goal. Yakimovsky and Feldman [28] introduced a segmentation framework for road scenes and ventricular angiograms, where semantic information was incorporated into a Bayesian framework. Ohta et al. [29] combined bottom-up grouping and top-down decomposition to reach an improved partitioning of outdoor scenes. A distinguishing feature of their approach was a seamless integration of domain-specific semantic information regarding object categories and regions in both the bottom-up and top-down hierarchies. The ‘Visions’ system from Hanson extended this idea by considering several levels of scene decompositions such as regions, surfaces, and objects [30]. In the early 1980s, Brooks [31] proposed a modular system, which combines geometric reasoning along with the object class and part-based information to

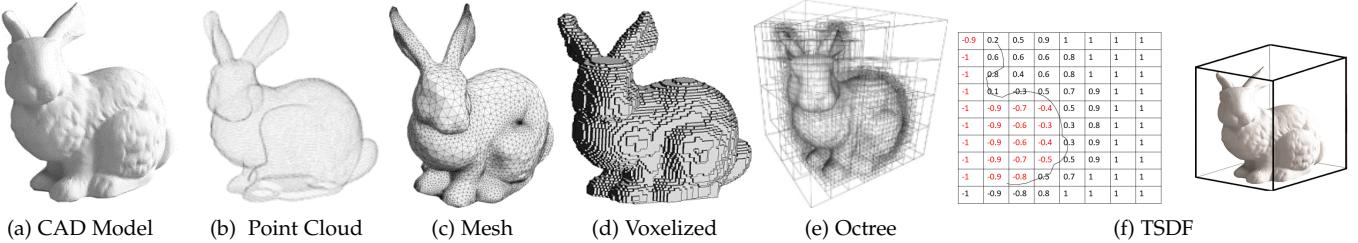


Fig. 2: Visualization of different types of 3D data representations for Stanford bunny.

understand images. Along similar lines, Douglass used depth cues along with semantic rationale over superpixel regions to build a 3D interpretation of an image [32]. An interesting direction was adopted by [33], where a combination of static long-term memory and dynamic short-term memory was used to provide a semantically meaningful interpretation of a scene.

While the initial systems developed for scene analysis bear notable ideas and insights, they lack generalizability to new scenes. This was mainly caused due to handcrafted rules and brittle logic-based pipelines. Recent advances in automated scene analysis seek to resolve these issues by devising more flexible, learning based approaches that offer rich expressiveness, efficient training, and inference in the designed models. We will systematically review the recent approaches and core tools in Sec. 5 and 6. However, before that, we provide an overview of the underlying data representations and datasets for RGB-D and 3D data in the next two sections.

3 DATA REPRESENTATIONS

In the following, we highlight the popular 2.5D and 3D data representations used to represent and analyze scenes. An illustration of different representations is provided in Fig. 2, while a comparative analysis is reported in Table 1.

- **Point Cloud:** A ‘point cloud’ is a collection of data points in 3D space. The combination of these points can be used to describe the geometry of the individual object or the complete scene. Every point in the point cloud is defined by x , y and z coordinates, which denote the physical location of the point in 3D. Range scanners (typically based on laser, e.g., LiDAR) are also used to capture 3D point clouds of objects or scenes.

- **Voxel Representation:** A voxel (volumetric element) is the 3D counterpart of a pixel (picture element) in a 2D image. Voxelization is a process of converting a continuous geometric object into a set of discrete voxels that best approximate the object. A voxel can be considered as a cubic volume representing a unit sample on a uniformly spaced 3D grid. Usually, a voxel value is mapped to either 0 or 1, where 0 indicates an empty voxel while 1 indicates the presence of range points inside the voxel.

- **3D Mesh:** The mesh representation encodes a 3D object geometry in terms of a combination of edges, vertices, and faces. A mesh that represents the surface of a 3D object using polygon (e.g., triangles or quadrilaterals) shaped faces is termed as the ‘polygon mesh.’ A mesh might contain arbitrary polygons but a ‘regular mesh’ is composed of

Representation	Data Dimension	Memory Efficiency	Shape Details	Computation Efficiency
Point cloud	3D	**	***	***
Voxel	3D	***	**	***
Mesh	3D	*	***	**
Depth	2.5D	***	*	***
Octree	3D	***	**	**
Stixel	2.5D	***	*	*
TSDF	3D	**	***	**
CSG	3D	***	***	*

TABLE 1: Comparison between data representations. The symbols *, ** and *** represent low, moderate and high respectively.

only a single type of polygons. A commonly used mesh is a triangular mesh that is composed entirely of triangle shaped faces. In contrast to polygon meshes, ‘volumetric meshes’ represent both the interior volume along with the object surface.

- **Depth Channel and Encodings:** A depth channel in a 2.5D representation shows the estimated distance of each pixel from the viewer. This raw data has been used to obtain more meaningful encodings such as HHA [34]. Specifically, this geocentric embedding encodes depth image using height above the ground, horizontal disparity and angle with gravity for each pixel.

- **Octree Representations:** An octree is a voxelized representation of a 3D shape that provides high compactness. The underlying data structure is a tree where each node has eight children. The idea is to divide 3D occupancy of an object recursively into smaller regions such that empty and similar voxels are represented with bigger voxels. An octree of an object is obtained by a hierarchical process as follows: start by considering 3D object occupancy as a single block, divide it into eight octants. Then, octants that partially contain an object part are further divided. This process continues until a minimum allowed size is reached. The octants can be labeled based on the object occupancy.

- **Stixels:** The idea of stixels is to reduce the gap between pixel and object level information, thus reducing the number of pixels in a scene to few hundreds [47]. In stixel representation, a 3D scene is represented by vertically oriented rectangles with a certain height. Such a representation is specifically useful for traffic scenes, but limited in its capability to encode generic 3D scenes.

- **Truncated Signed Distance Function:** Truncated signed distance function (TSDF) is another volumetric representation of a 3D scene. Instead of mapping a voxel to 0 or 1,

TABLE 2: Comparison between various publicly available 2.5/3D indoor datasets.

Dataset	NYUv2 [10]	SUN3D [35]	SUN RGB-D [36]	Building Parser [37]	Matterport 3D [38]	ScanNet [39]	SUNCG [40]	RGBD Object [41]	SceneNN [42]	SceneNet RGB-D [43]	PiGraph [44]	TUM [45]	Pascal 3D+ [46]
Year	2012	2013	2015	2017	2017	2017	2016	2011	2016	2016	2016	2012	2014
Type	Real	Real	Real	Real	Real	Synthetic	Real	Real	Real	Synthetic	Synthetic	Real	Real
Total Scans	464	415	-	270	-	1513	45,622	900	100	57	63	39	-
Labels	1449 images	8 scans	10k images	70k images	194k images	1513 scans	130k images	900 scans	100 scans	5M images	21 scans*	39 scans	24k images
Objects/Scenes	Scene	Scene	Scene	Scene	Scene	Scene	Scene	Object	Scene	Scene	Scene	Scene	Object
Scene Classes	26	254	47	11	61	707	24	-	-	5	30	X	-
Object Classes	894	-	800	13	40	50 - at least	84	51	50 - at least	255	5 subjects	X	12
In/Outdoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	In+Out
Available Data Types													
RGB	✓	-	✓	✓	✓	-	-	✓	-	✓	X	-	✓
Depth	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	-	X
Video	✓	✓	✓	X	X	✓	X	✓	✓	✓	✓	✓	X
Point cloud	X	X	✓	✓	✓	X	X	X	X	X	-	X	X
Mesh/CAD	X	X	X	✓	✓	✓	✓	X	✓	X	-	X	✓
Available Annotation Types													
Scene Classes	✓	X	✓	✓	X	X	X	X	X	X	X	X	X
Semantic Label	✓	✓	✓	✓	✓	✓	✓	X	X	✓	X	X	X
Object Label	X	X	✓	X	X	X	X	✓	✓	X	X	X	✓
Camera Poses	X	✓	X	X	✓	✓	X	X	X	X	X	✓	✓
Object Poses	X	X	✓	X	X	X	X	✓	✓	X	X	X	✓
Trajectory	X	X	X	X	X	X	X	X	X	X	X	✓	X
Action	X	X	X	X	X	X	X	X	X	X	✓	X	X

-: means information not available, *: Average reported; 4.9 actions annotated per scan and there are 298 actions with 8.4s length available.

each voxel in the 3D grid is mapped to the signed distance to the nearest surface. The signed distance is negative if the voxel lies with in the shape and positive otherwise. RGB-D camera (e.g., Kinect) representations are based on TSDF further fuse them to obtain a complete 3D model.

- **Constructive Solid Geometry:** Constructive solid geometry (CSG) is a building block technique in which simple objects such as cubes, spheres, cones, and cylinders are combined with a set of operations such as union, intersection, addition, and subtraction to model complex objects. CSG is represented as a binary tree with primitive shapes and the combination operations as its nodes. This representation is often used for CAD models in computer vision and graphics.

4 DATASETS

- **NYU-Depth:** Silberman et al. introduced NYU Depth v1 [48] and v2 [10] in 2011 and 2012, respectively. NYU Depth v1 [48] consists of 64 different indoor scenes with 7 scene types. There are 2347 RGBD images available. The dataset is roughly divided into 60%/40% for train/test respectively. NYU Depth v2 [10] consists of 1449 RGBD images representing 464 different indoor scenes with 26 scene types. Pixel level labeling is provided for each image. There are 795 images in train set and 654 images in the test set. Both versions were collected using Microsoft Kinect.

- **Sun3D:** Sun3D [35] dataset provides videos of indoor scenes that are registered into point clouds. The semantic class and instance labels are automatically propagated through the video from the seed frames. Dataset provides 8 annotated sequences, and there are in total of 415 sequences available for 254 different spaces in 41 different buildings.

- **SUN RGB-D:** Sun RGB-D [36] contains 10335 indoor images with dense annotations in 2D and 3D for both objects and indoor scenes. It includes 146617 2D polygons and 64595 3D bounding boxes for object orientation as well as scene category and 3D room layout for each image. There are 47 scene categories, 800 object categories, and each image contains on average 14.2 objects. This data set is captured

by four different kinds of RGB-D sensors and designed to evaluate scene classification, semantic segmentation, 3D object detection, object orientation, room layout estimation and total scene understanding. The data is divided into training and test sets such that each sensor data has half allocation for training and the other half for testing.

- **Building Parser:** Armeni et al.[37] provide a dataset with instance level semantic and geometric annotations. The dataset was collected from 6 different areas, and it contains 70496 RGB and 1412 equirectangular RGB with their corresponding depths, semantic annotations, surface normal, global XYZ openEXR format and camera metadata. These 6 different areas are divided into training and test splits with a 3-fold cross-validation scheme, i.e., training with 5 areas, training with 4 areas and training with 3 areas while testing with the rest of scans in each case.

- **ScanNet:** ScanNet [39] is the 3D reconstructed dataset with 2.5 million data frames obtained from 1513 RGB scans. These 1513 annotated scans represent 707 different spaces including small ones like closets, bathrooms, and utility rooms, and large spaces like classrooms, apartments, and libraries. These scans are annotated with instance level semantic category labels. There are 1205 scans in the training set and another 312 scans in the test set.

- **PiGraph:** Savva et al. [44] proposed the PiGraph representation to link human poses with object arrangements in indoor environment. Their dataset contains 30 scenes and 63 video recordings for five human subjects obtained by Kinect v2. There are 298 actions available in approximately 2-hour of recordings. Each recording is about 2 minute long with on average 4.9 action annotations. They link 13 common human actions like sitting, reading to 19 object categories such as couch and computer monitor.

- **SUNCG:** SUNCG [40] is a densely annotated, large scale dataset of 3D scenes. It contains 45622 different scenes that are semantically annotated at object level. These scenes are manually created using the Planner5D platform [49]. Planner5D is an interior design tool that can be used to generate novel scene layouts. This dataset contains around 49K floor maps, 404K rooms and 5697K object instances

covering 84 object categories. All the objects are manually assigned to a category label.

- **PASCAL3D+:** Xiang et al. [46] introduced Pascal3D+ for 3D object detection and pose estimation tasks. They picked 12 categories of objects including airplane, bicycle, boat, bottle, bus, car, chair, motorbike, dining table, sofa, tv monitor, and train (from Pascal VOC dataset [50]) and performed 3D labeling. Further, they included additional images for each category from ImageNet dataset [51]. The resulting dataset has around 3000 object instances per category.

- **RGBD Object:** This dataset [41] provides video recordings of 300 household objects assigned to 51 different categories. The objects are categorized using WordNet hypernym-hyponym relationships. There are 3 video sequences for each object category recorded by mounting Kinect camera at different heights. The videos are recorded at a frame rate of 30Hz with 640×480 resolution for RGB and depth images. This dataset also contains 8 annotated video sequences of indoor scene environments.

- **TUM:** [45] provides a large-scale dataset for tasks like visual odometry and SLAM (simultaneous localization and mapping). The dataset contains RGB and depth images obtained using Kinect sensor along with the groundtruth sensor trajectory (poses and positions). The dataset is recorded at a frame rate of 30Hz with 640×480 resolution for RGB and depth images. The groundtruth trajectory was obtained using high speed cameras working at 100Hz. There are in total 39 sequences of indoor environments.

- **SceneNN:** SceneNN [42] is a fine-grain annotated RGBD dataset of indoor environments. It consists of 100 scenes where each scene is represented as a triangular mesh having per vertex and per pixel annotations. The dataset is further enriched by providing information such as oriented bounding boxes, axis-aligned bounding boxes and object poses.

- **Matterport3D:** Matterport3D [38] provides a diverse and large-scale RGBD dataset for indoor environments. This dataset provides 10800 panoramic images covering 360° views captured by Matterport camera. Matterport camera comes with three color and three depth cameras. To get a panoramic view, they rotated the Matterport camera by 360° , stopping at six locations and capturing three RGB images at each location. The depth cameras continuously acquired depth information during rotation, which was then aligned with each color image. Each panoramic image contains 18 RGB images. In total, there are 194400 color and depth images representing indoor scenes of 90 buildings. This dataset is annotated for 2D and 3D semantic segmentations, camera poses and surface reconstructions.

- **SceneNet RGB-D:** SceneNet is a synthetic video dataset which provides pixel level annotations for nearly 5M frames. The dataset is divided into training set with 5M images while validation and test set contains 300K images. This dataset can be used for multiple scene understanding tasks including semantic segmentation, instance segmentation and object detection.

5 CORE TECHNIQUES

We begin with an overview of the core techniques employed in the literature for various scene understanding problems. For each respective technique, we discuss its pros and cons

in comparison to other competing methods 7. Later in this survey, we provide a detailed description of recent methods that built on the strengths of these core techniques or attempt to resolve some of their weaknesses. In this regard, several hybrid approaches have also been proposed in the literature e.g., [52], [53], [54], which combine strengths of different core techniques to achieve better performances.

5.1 Convolutional Neural Networks

An artificial neural network (ANN) consists of a number of computational units, which are arranged in multiple interconnected layers. Convolutional Neural Network (CNN) is a special type of ANN whose main building blocks consist of filters that are spatially convolved with the inputs to generate output feature maps. A building block is called as the ‘convolutional layer,’ which usually repeats several times in a CNN architecture. A convolution layer drastically reduces the network parameters through weight-sharing. Further, it makes the network invariant to translations in the input domain. The convolution layers are interleaved with other layers such as pooling (to subsampling inputs), normalization (to rescale activations) and fully connected layers (to reduce the feature dimensions or to densely connect input and output units). A simple CNN architecture is illustrated in Fig. 3, which shows the above-mentioned layers.

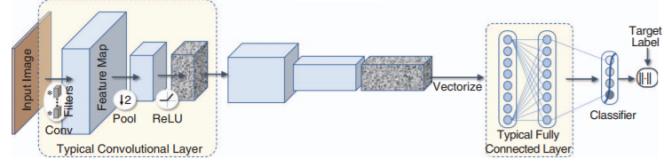


Fig. 3: A basic CNN architecture with a convolution, pooling, activation along with a fully connected layer.

CNNs have shown excellent performance on many scene understanding tasks (e.g., [26], [27], [55], [56], [25], [57], [58]). Some distinguishing features that permit CNNs to achieve superior results can be considered as end-to-end learning of network weights, scalability to large problem sets and computationally efficiency in their derivation of large-scale models. However, it is nontrivial to incorporate prior knowledge and rich relationships between variables in a traditional CNN. Besides, conventional CNNs do not operate on arbitrarily shaped inputs such as point clouds, meshes, and variable length sequences.

5.2 Recurrent Neural Networks

While CNNs are feedforward networks (i.e., they do not have cycles or loops), Recurrent Neural Network (RNN) has a feedback architecture where information flow happens along directed cycles. This capability allows them to work with arbitrary sized inputs and outputs. RNNs exhibit memorization ability and can store information and sequence relationships in their internal memory states. A prediction at a specific time instance ' t ' can then be made while considering the current input as well as the previous hidden states (Fig. 4). Similar to the case of convolution

layer in CNNs where weights are shared along the spatial dimensions of the inputs, the RNN weights are shared along the temporal domain, i.e., same weights are applied to inputs at each time instance. Compared to CNNs, RNNs have considerably less number of parameters due to such weight sharing mechanism.

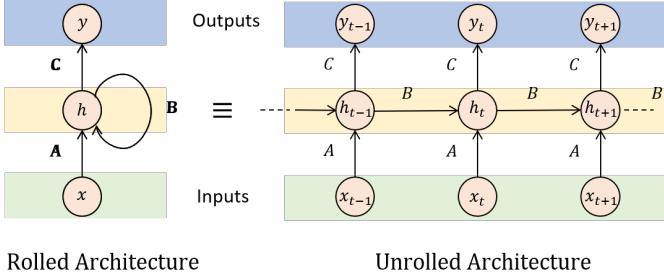


Fig. 4: A basic RNN architecture in the rolled and unrolled form.

As discussed above, the hidden state of the RNN provides a memory mechanism, but it is not effective when the goal is to remember long-term relationships in sequential data. Therefore, RNN only accommodates short-term memory and faces with difficulties in ‘remembering’ (a few time-steps away) old information processed through it. To overcome this limitation, improved versions of recurrent networks have been introduced in the literature which include the Long Short-Term Memory (LSTM) [59], Gated Recurrent Unit (GRU) [60], Bidirectional RNN (B-RNN) [61] and Neural Turing Machines (NTM) [62]. These architectures introduce additional gates and recurrent connections to improve the storage ability. Some representative works in 3D scene understanding that leverage the strengths of RNNs include [53], [63].

5.3 Encoder-Decoder Architectures

The encoder-decoder networks are a type of ANNs, which can be used for both supervised and unsupervised learning tasks. Given an input, an ‘encoder’ module learns a compact representation of the data which is then used to reconstruct either the original input or an output of another form (e.g., pixel labels for an image) using a ‘decoder’ (Fig. 5). This type of network is called an autoencoder when the input to the encoder is reconstructed back using the decoder. Autoencoders are typically used for unsupervised learning tasks. A closely related variant of an autoencoder is a variational autoencoder (VAE) that introduces constraints on the latent representation learned by the encoder.

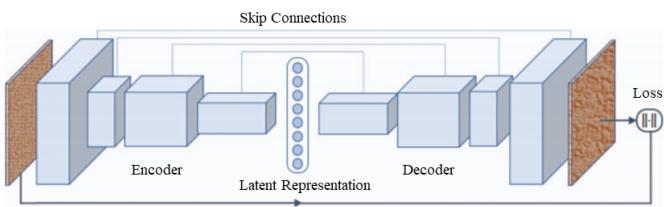


Fig. 5: A basic autoencoder architecture.

Encoder-decoder style networks have been used in combination with both convolutional [52] and recurrent designs

[64]. The applications of such designs for scene understanding tasks include [52], [65], [66], [67], [68], [69]. The strength of these approaches is to learn a highly compact latent representation from the data, which is useful for dimensionality reduction and can be directly employed as discriminative features or transformed using a decoder to generate desired outputs. In some cases, the encoding step leads to irreversible loss of information which makes it challenging to reach the desired output.

5.4 Markov Random Field

Markov Random Field (MRF) is a class of undirected probabilistic models that are defined over arbitrary graphs. The graph structure is composed of nodes (e.g., individual pixels or super-pixels in an image) interconnected by a set of edges (connections between pixels in an image). Each node represents a random variable which satisfies Markovian property, i.e., conditional independence from all variables if the neighboring variables are known. The learning process for a MRF involves estimating a generative model i.e., the joint probability distribution over input (data; \mathbf{X}) and output (prediction; \mathbf{Y}) variables i.e., $P(\mathbf{X}, \mathbf{Y})$. For several problems, such as classification and regression, it is more convenient to directly model the conditional distribution $P(\mathbf{Y}|\mathbf{X})$ using the training data. The resulting discriminative Conditional Random Field (CRF) models often provide more accurate predictions.

Both MRF and CRF models are ideally suited for structured prediction tasks where the predicted outputs have inter-dependent patterns instead of, e.g., a single category label in the case of classification. Scene understanding tasks often involve structured prediction e.g., [70], [71], [72], [73], [74], [75]. These models allow the incorporation of context while making local predictions. The context can be encoded in the model by pair-wise potentials and clique potentials (defined over groups of random variables). This results in more informed and coherent predictions which respect the mutual relationships between labels in the output prediction space. However, training and inference in several of such model instantiations are not tractable, which makes their application challenging.

5.5 Sparse Coding

Sparse coding is an unsupervised method used to find a set of basis vectors such that an input vector ‘ x ’ can be represented by their linear sparse combination [76]. The set of basis vectors is called as a ‘dictionary’ (\mathbf{D}), which is typically learned over the training data. Given the dictionary, a sparse vector α is calculated such that the input x can be accurately reconstructed back using \mathbf{D} and α . Sparse coding can be seen as decomposing a non-linear input into sparse combination of linear vectors. If the basis vectors are large in number or when the dimension of feature vectors is high, optimization process required to calculate \mathbf{D} and α can be computationally expensive. Examples of sparse coding based approaches in scene understanding literature include [77], [78], [79].

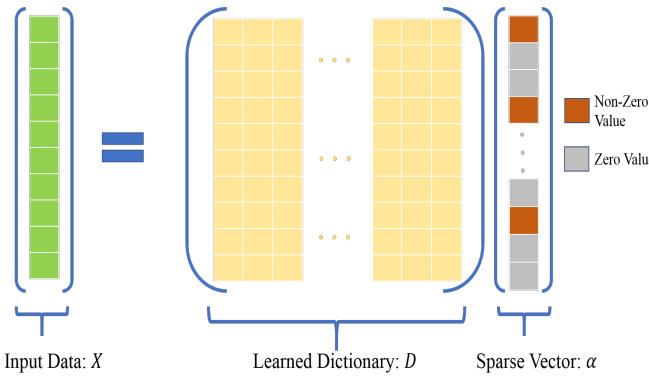


Fig. 6: Dictionary learning for sparse coding.

5.6 Decision Forests

A decision tree is a supervised algorithm that classifies data based on a graph based hierarchy of rules learned over the training set. Each internal node in a decision tree represents a test or attribute (true or false question) while each leaf node represents the decision on a class label. To build a decision tree, we start with a root node that receives all the training data and based on the test question we split the data into subsets. These subsets then become the inputs for the next two child nodes. This process continues until we produce the best possible distribution of the labels at each node, i.e., a total unmixing of data is achieved. One can quantify the mixing or uncertainty at a single node by a metric called ‘Gini impurity’ which can be minimized by devising rules based on information gain. We can use these measures to ask the best question at each node and continue to build the decision tree recursively until there are no more questions to ask. Decision trees can quickly overfit the training data which can be rectified by using random forests [80]. Random forest builds an ensemble of decision trees using a random selection of data and produces class labels based on many decisions trees. Representative works using random forests include [81], [82], [83], [77], [84].

5.7 Support Vector Machines

When it comes to classifying the n-dimensional data points, the goal is not only to separate the data into a certain number of categories but to find such a dividing boundary that offers maximum possible separation between classes. Support vector machine (SVM) [85], [86] offers such a solution. SVM is a supervised method that separates the data with a linear hyperplane, also called maximum-margin hyperplane, that offers maximum separation between any combination of two classes. SVM can also be used to learn nonlinear classification boundaries with the kernel trick. The idea of kernel trick is to project the nonlinearly separable low dimensional data into a high dimensional space where the data is linearly separable. After applying SVM into high dimensional linearly separable space, project the solution back to low-dimensional, nonlinearly separable space to get a nonlinear hyperplane boundary.

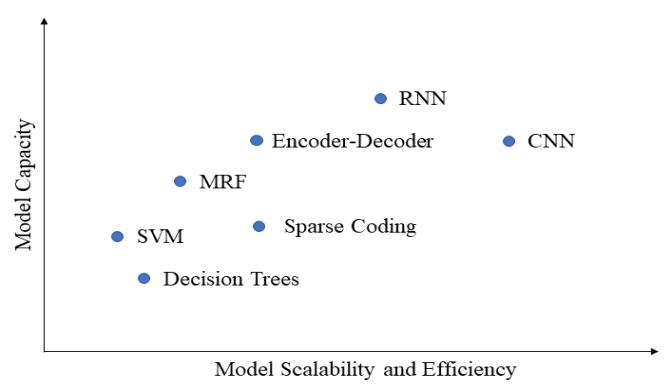


Fig. 7: Core-Techniques Comparison.

6 A TAXONOMY OF PROBLEMS

6.1 Image Classification

6.1.1 Prologue and Significance

Image recognition is a basic, yet one of the fundamental tasks for visual scene understanding. Information about the scene or object category can help in more sophisticated tasks such as scene segmentation and object detection. Classification algorithms are being used in diverse areas such as medical imaging, self-driving cars and context-aware devices. In this section, we will provide an overview of some of the most important methods for 2.5D/3D scene classification. These approaches employ a diverse set strategies including handcrafted features [87], automatic feature learning [53], [58], unsupervised learning [88] and work on different 3D representations such as voxels [89] and point clouds [55].

6.1.2 Challenges

Important challenges for image classification include:

- 2.5/3D data can be represented in multiple ways as discussed above. Challenge then is to choose the data representation that provides maximum information with minimum computational complexity.
- A key challenge is to distinguish between fine-grained categories and appropriately model intra-class variations.
- Designing algorithms that can handle illuminations, background clutter and 3D deformations.
- Designing algorithm that can learn from limited data.

6.1.3 Methods overview

A bottom-up approach to scene recognition was introduced in [87], where the constituent objects were first identified to improve the scene classification accuracy. In this regard, they first extended a contour detection method (gPb-ucm [90]) to RGB-D images by effectively incorporating the depth information. Note that the gPb-ucm approach produces hierarchical image segmentation by using contour information [90]. The predicted semantic segmentation maps were used as features for scene classification. They used a special pyramid formulation, similar to spatial pyramid matching approach [91], along with SVM as a classifier.

Sochar et al. [53] introduced a method to learn features from RGB-D images using RNN. They used a convolutional

layer to learn low-level features which were then passed through multiple RNNs to learn high-level feature representations before feeding to a classifier. At the CNN stage, RGB and depth patches were clustered using k-means to obtain the convolutional filters in an unsupervised manner [92]. These filters were then convolved with images to get low-level features. After performing dimensionality reduction via pooling process, these features were then fed to multiple RNNs which recursively operate in a tree-like structure to learn high-level feature representations. The outputs of these multiple RNNs were concatenated to form a final vector which is forwarded to a SoftMax classifier for the final decision. An interesting insight of their work is that weights of RNNs were not learned through back propagation rather set to random values. Increasing the number of RNNs resulted in an improved model classification accuracy. Another important insight of their work is that RGB and depth images produce independent complimentary features and their combination improves the model accuracy. Similar to this work, [93] extracted features from RGB and depth modalities via two stream networks, which were then fused together. [94] extended the same pattern by learning features from modalities like RGB, depth and surface normals. They also proposed to encode local CNN features with fisher vector embedding and then combine them with global CNN features to obtain better representations.

In an effort to build a 3D shape classifier, Wu et al. [95] introduced a convolutional deep belief network (DBN) trained on 3D voxelized representations. Note that different from restricted Boltzmann machines (RBM), a DBN is a directed model that can detect patterns from unlabeled data. An RBM is a two-way translator that takes input in a forward pass and translates it to latent representation that encodes the input, while in the backward pass it takes the latent representation and translates it back to reconstruct the input. [96] showed that DBN could learn the joint distributions of 2D image pixels and labels. [95] extended this idea to learn joint probabilistic distributions of 3D voxels and object categories. The novelty in their architecture is to introduce convolutional layers which, in contrast to fully connected layers, allow weight sharing and significantly reduce the number of parameters in the DBN. On similar lines, [97] advocates using 3D CNN on a voxel grid to extract meaningful representations while [57] proposes to approximate 3D spaces as volumetric fields to deal with the computational cost of directly applying 3D CNN to voxels.

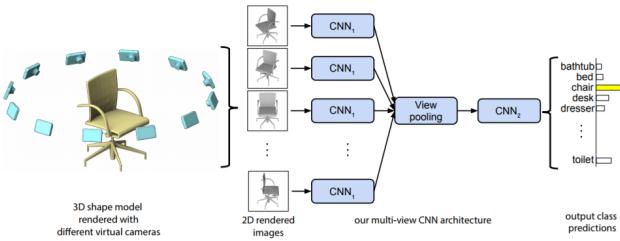


Fig. 8: Multi-view CNN for 3D shape recognition [58]. Extracted features of different views from CNN_1 are pooled together before passing through CNN_2 for final score prediction. (Courtesy of [58])

Though it seems logical to build a model that can directly consume 3D shapes to recognize them (e.g., [95]), however the 3D resolution of a shape must be significantly reduced to allow feasible training of a deep neural network. As an example, 3D ShapeNets [95] used a $30 \times 30 \times 30$ binary voxel grid to represent 3D shapes. Su et al. [58] provided evidence that 3D shapes can be recognized by their 2D views and presented a multi-view CNN (MVCNN) architecture to recognize 3D shapes that can be trained on 2D rendered views. They used the Phong reflection method [98] to render 2D views of 3D shapes. Afterwards, a pre-trained VGG-M network [99] was fine-tuned on these rendered views. To aggregate the complementary information across different views, each rendered view was passed through the first part of the network (CNN_1) separately, and the results across views were combined using element-wise maximum operation at the pooling layer before passing them through the rest of network (CNN_2 , see Figure 8). MVCNN thus combines the multiple view information to better recognize 3D shapes. While MVCNN represented 3D shapes with multiple 2D images, [100] proposes to convert 3D shapes into a panoramic view.

We have observed so far that the existing models utilize different 3D shape representations (i.e., volumetric, multi-view and panoramic) to extract useful features. Intuitively volumetric representations should contain more information about the 3D shape, but multi-view CNNs [58] perform better than volumetric CNNs [95]. Qi et al. [89] argued that network architecture differences and input resolutions are the reasons for the gap in performance. Inspired from multi-view CNNs, [89] introduced a multi-orientation network architecture that takes various orientations of input voxel grid, extract features for each orientation using a shared network CNN_1 , pooled the features before passing through CNN_2 . To take benefit of well trained 2D CNN, they introduced 3D-to-2D projection using anisotropic probing kernels to classify the 2D projection of the 3D shape. They also improved multi-view CNNs [58] performance by introducing a multi-resolution scheme. Inspired by the performance efficiency of MVCNN and volumetric CNN, [101] fused both modalities to learn better features for classification.

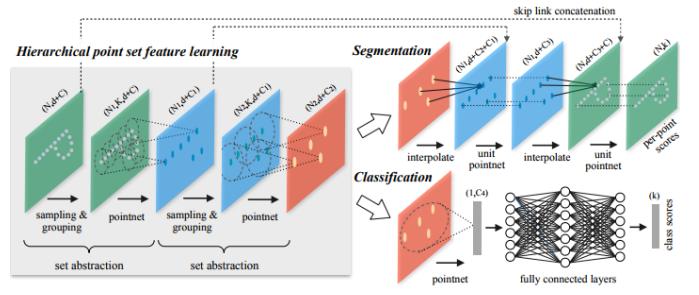


Fig. 9: PointNet++ [102] architecture for point cloud classification and segmentation. PointNet architecture [55] is being used in a hierarchical fashion to extract local geometric features (Courtesy of [102]).

A point cloud is a primary geometric representation captured by 3D scanners. However, due to its variable number of points from one shape to another, it needs to be transformed to a regular input data format, e.g., voxel

grid or multi-view images. This transformation, however, can increase the data size and result in undesired artifacts. PointNet [55] is a deep net architecture that can consume point clouds directly and output the class label. PointNet takes a set of points as input, performs feature transformations for each point, assemble feature across points via max-pooling and output the classification score. Even though PointNet process unordered point clouds but by design, it lacks the ability to capture local contextual features due to metric space of the points. Just like a CNN architecture learns hierarchical features mapped from local patterns to more abstract motifs, Qi et al. [102] applied PointNet on point sets recursively to learn local geometric features and then grouped these features to produce high-level features for the whole point set (see Figure 9). A similar idea was adopted in [103], which performed a hierarchical feature learning over a k-d tree structured partitioning of 3D point clouds.

Finally, we would like to describe an unsupervised method for 3D object recognition. By combining the power of volumetric CNN [95] and generative adversarial networks (GAN) [104], Wu et al. [88] presented a novel framework called 3D-GAN for 3D object generation and recognition. An adversarial discriminator in GAN learns to classify whether an object is real or synthesized. [88] showed that representations learned by an adversarial discriminator without supervision could be used as features for linear SVM to obtain classification scores of 3D objects.

6.2 Object Detection

6.2.1 Prologue and Significance

Object detection deals with recognizing object instances and categories. Usually, an object detection algorithm outputs both the location (defined by a 2/3D bounding box around the visible parts of an object instance) and the class of an object, e.g., sofa, chair. This task has high significance for applications such as self-driving cars, augmented and virtual reality. However, in applications such as robot navigation, we need so-called ‘amodal object detection’ that tries to find an object’s location as well as its complete shape and orientation in 3D space when only a part of it is visible. In this section, we review 2.5/3D object detection methods mainly focused on indoor scenes. We observe the role of handcrafted features in object recognition [72], [105], [106] and the recent transition to deep neural networks based region proposal (object candidate) generation and object detection pipelines [107], [24], [108]. Apart from the supervised models, we also review unsupervised 3D object detection techniques [109].

6.2.2 Challenges

Key challenges for object detection are as follows:

- Real world environments can be highly cluttered and object identification in such environments is very challenging.
- Detection algorithm should also be able to handle view-point and illuminations variations and deformations.
- In many scenarios, it is necessary to understand the scene context to successfully detect objects.

- Objects categories have a long-tail (imbalanced) distribution, which makes it challenging to model the infrequent classes.

6.2.3 Methods Overview

Jiang et al. [110] proposed a bottom-up approach to detect 3D object bounding boxes using RGB-D images. Starting from a large number of object proposals, physically plausible boxes were identified by using volumetric properties such as solidness, 3D overlap, and occlusion relationships. Later, [105] argued that convex shapes are more descriptive than cuboids and can be used to represent generic objects. A limitation of these techniques is that they ignore semantics in a scene and are limited to finding object shape. In a real-world scenario, a scene can contain regular objects (e.g., furniture) as well as cluttered regions (e.g., clothes pile on a bed). Khan et al. [72] extended the technique presented in [110] to jointly detect 3D object cuboids and indoor structures (e.g., floor, walls) along with pixel level labeling of cluttered regions in RGB-D images. A CRF model was used to model the relationships between objects and cluttered regions in indoor scenes. However, these approaches do not provide object-level semantic information apart from a broad categorization into regular objects, clutter, and background.

Several object detection approaches [106], [111], [24], [78] have been proposed to provide category information and location of each detected object instance. [78] proposed a sparse coding network to learn hierarchical features for object recognition from RGB-D images. Sparse coding models data as a linear combination of atoms belonging to a codebook subject to sparsity constraints. The multi-layer network [78] learns codebooks for RGB-D images via K-SVD algorithm [112], using the grayscale, color, depth, and surface normal information. The feature hierarchy is built as the receptive field size increases along the network depth which helps to learn more abstract representations of RGB-D images. They used orthogonal matching pursuit [113] algorithm for sparse coding, feature pooling to reduce dimensionality and contrast normalization at each layer of the network.

The performance of an object detection algorithm can suffer due to variations in object shapes, viewpoints, illumination, texture, and occlusion. Song et al. [106] proposed a method to deal with these variations by exploiting synthetic depth data. They take a collection of 3D CAD models of an object category and render it from different viewpoints to obtain depth maps. The feature vectors corresponding to depth maps of an object category are then used as positives to train exemplar SVM [114] against negatives obtain from RGB-D datasets [10]. At test time, a 3D window is slid on the scene to be classified by the learned SVMs. While [106] represented objects with CAD models, other representations are also explored in literature such as [111], [115] proposed 3D deformable wire-frame modeling and cloud of oriented gradients representation, respectively, and [116] build object detector based on 3D mesh representation of indoor scenes.

Typically, an object detection algorithm produces a bounding box on visible parts of the object on an image plane, but for practical reasons, it is desirable to capture the full extent of the object regardless of occlusion or truncation.

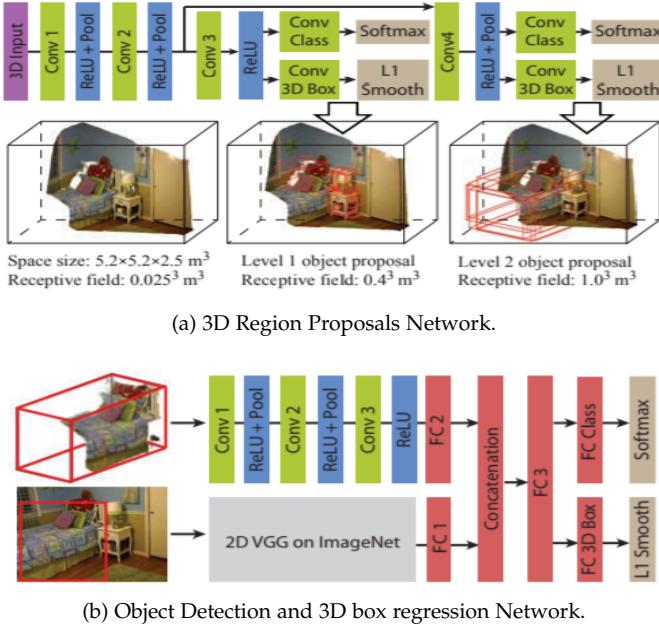


Fig. 10: (a) 3D region proposals extraction using using CNNs operating on 3D volumes. (b) A combination of 2D and 3D CNN jointly used to predict object category and location through regression. (Courtesy of [24])

Song et al. [24] introduced a deep learning framework for amodal object detection. They used three deep network architectures to produce object category labels along with 3D bounding boxes. First, a 3D network called Region Proposal Network (RPN) takes a 3D volume generated from depth map and produces 3D regional proposals for the whole object. Each region proposal is feed into another 3D convolutional net, and its 2D projection is fed to a 2D convolutional network to jointly learn color and depth features. The final output is the object category along with the 3D bounding box (see Figure 10). A limitation of this work is that the object orientation is not explicitly considered. As [117] demonstrated with their oriented-boosted 3D CNN (Vox-net), this can adversely affect the detection performance and joint reasoning about the object category, location, and 3D pose leads to a better performance.

Deng et al. [108] introduced a novel neural network architecture based on Fast-RCNN [26] for 3D amodal object detection. Given RGB-D images, they first computed the 2D bounding boxes using multiscale combinatorial grouping (MCG) [118] over superpixel segmentations. For each 2D bounding box, they initialized the location of the 3D box. The goal is then to predict the class label and adjust the location, orientation, and dimension of the initialized 3D box. In doing so, they successfully showed the correlation between 2.5D features and 3D object detections. Novotny et al. [109] proposed to learn 3D object categories from videos in an unsupervised manner. They used Siamese factorization network architecture to align videos of 3D objects to estimate viewpoint, then produce depth maps using the estimated viewpoints, and finally, the 3D object model is constructed using the estimated depth map (see Figure 11).

Finally, we would like to mention 3D object detection with attention mechanism. To understand a specific aspect

of an image, humans can selectively focus their attention on a specific part of the image to gain information. Inspired by this, [119] proposed 3D attention model that scan a scene to select best views and focus on most informative regions for object recognition task. It further combines the 3D CAD models to replace the actual objects, such that a full 3D scene can be reconstructed. This demonstrates how object detection can help in other tasks such as scene completion.

6.3 Semantic Segmentation

6.3.1 Prologue and Significance

This task relates to the labeling of each pixel in an image with its corresponding semantically meaningful category. Applications of semantic segmentation include domestic robots, content-based retrieval, self driving cars and medical imaging. Efforts to address the semantic segmentation problem have come a long way from using hand crafted and data specific features to automatic feature learning techniques. Here, we summarize the important challenges for the problem and some of the most important methods for semantic segmentation that have had significant impact and inspired a great deal of research in this area.

6.3.2 Challenges

Despite being an important task, segmentation is highly challenging because:

- Pixel level labeling requires both local and global information and challenge then is to design such algorithms that can incorporate the wide contextual information together.
- The difficulty level increases a lot for the case of instance segmentation, where the same class is segmented into different instances.
- Obtaining dense pixel level predictions, especially close to object boundaries, is challenging due to occlusions and confusing back-grounds.
- Segmentation is also affected by appearance, view-point and scale changes.

6.3.3 Methods Overview

Traditionally, CRFs have been the default choice in the context of semantic segmentation [120], [121], [122], [123]. This is due to the reason that CRFs provide a flexible framework to model contextual information. As an example, [122] exploit this property of CRF for the case of semantic segmentation of RGB-D images. They first developed a 2D semantic segmentation method based on decision forests [80] and then transferred the 2D labels to 3D using a 3D CRF model to improve the RGB-D segmentation results. Other efforts to formulate semantic segmentation task into CRF framework include [120], [121], [123]. More recently, CNNs have been used to extract rich local features for 2.5D/3D image segmentation tasks [124], [34], [125], [126]. A dominant trend in the deep learning based methods for semantic segmentation tasks has been to use encoder-decoder networks in an end-to-end learnable pipeline, which enable high resolution segmentation maps [127], [52], [66], [128].

The work by Couprie et al. [124] is among the pioneering efforts to use depth information along with RGB images

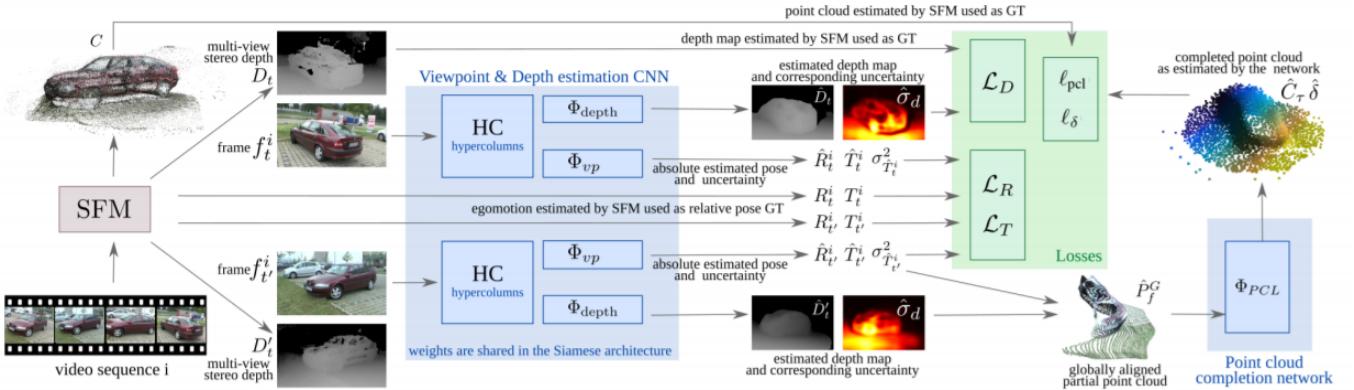


Fig. 11: Siamese factorization network [109] that takes pair of frames and estimate view point, depth and finally produce point cloud of estimated 3D geometry. Once the network is trained, it can produce viewpoint, depth and 3D geometry from single image at test time. (Courtesy of [109])

for feature learning in semantic segmentation task. They used multi-scale convolutional neural networks (MCNN) for feature extraction that can be efficiently implemented using GPUs to operate in real-time during inference. Their work-flow involves fusing the RGB image with depth image using a Laplacian pyramid scheme which was then fed into the MCNN for feature extraction. The resulting features had a spatially low resolution, this was overcome using an up-sampling step. In parallel, RGB image was segmented into super-pixels. Final scene labeling was produced by aggregating the classifier predictions into the super-pixels. Note that although this approach was applied for video segmentation, it does not leverage temporal relationships and independently segments each frame. The real-time scene labeling of video sequences was achieved by using a computational efficient graph based scheme [129] to compute temporal consistent super-pixels. This technique was able to compute super-pixels in quasi-linear time, thereby making it possible to use for real-time video segmentation.

Girshick et al. [130] presented a region CNN (R-CNN) method for detection and segmentation of RGB images which was later extended to RGB-D images [34]. The R-CNN [130] method extracts regions of interest from an input image, compute the features for each of the extracted regions using a CNN and then classify each region using class-specific linear SVMs. Gupta et al. [34] then extended the R-CNN method to RGB-D case by using a novel embedding for depth images. They proposed a geo-centric embedding called HHA to encode depth images using height above ground, horizontal disparity and angle with gravity for each pixel. They demonstrated that CNN can learn better features using the HHA embedding compared to raw depth images. Their proposed method [34] first uses multiscale combinatorial grouping (MCG) [118] to obtain region proposals from RGB-D images, followed by feature extraction using a CNN [131] pre-trained on Imagenet [51] and fin-tuned on HHA encoded depth images. Finally, they pass these learned features of RGB and depth images through SVM classifier to perform object detection. They used superpixel classification framework [87] on the output of the object detectors for semantic scene segmentation.

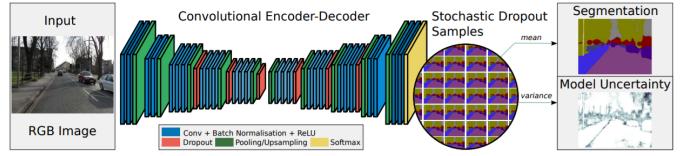


Fig. 12: A schematic of Bayesian encoder-decoder architecture for semantic segmentation with a measure of model uncertainty. (Courtesy of [66])

Long et al. [127] built an encoder-decoder architecture using a Fully Convolutional Network (FCN) for pixel-wise semantic label prediction. The network can take arbitrary sized input and produce corresponding sized outputs due to the fully convolutional architecture. [127] first redefined the pre-trained classification networks (AlexNet [131], VGG net [27], GoogLeNet [132]) into their equivalent FCNs and therefore transferred their learned representation to the segmentation task. As one can expect, the FCNs based on classification nets downgrade the spatial resolution of visual information through consecutive sub-sampling operations. To improve the spatial resolution, [127] augments the FCN with a convolution transpose block for upsampling while keeping the end-to-end learning intact. Further the final classification layer of each classification net [131], [27], [132] was removed and the fully connected layers were replaced with 1×1 convolution followed by deconvolutional layer to upsample the output. To refine the predictions and for detailed segmentation, they introduced a skip architecture which combined deep semantic information and shallow appearance information by fusing the intermediate activations. To extend their method to RGB-D images, [127] trained two networks: one for RGB images and a second for depth images represented by three dimensional HHA depth encoding introduced in [34]. The predictions from both nets are then summed at the final layer. After the successful application of FCNs [127] to semantic segmentation, FCN based architectures since attracted a lot of attention from the research community and are extended to number of new tasks like region proposal [133], contour detection [134] and

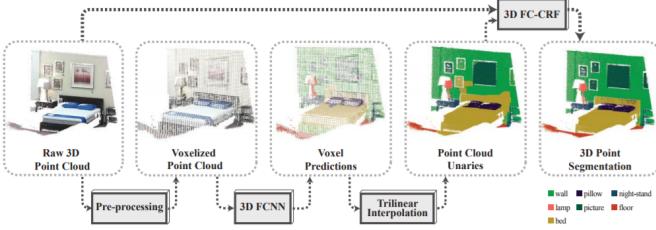


Fig. 13: SegCloud [141] framework that takes a 3D point cloud as input, that is voxelized before feeding to 3D CNN. The voxelized representation is projected to the point cloud representation using trilinear interpolation. (Courtesy of [141])

depth regression [135]. In a follow up paper [136], authors revisited the FCNs for semantic segmentation to further analyze, tune and improve results.

A measure of confidence based on which we can trust the semantic segmentation output of our model can be important in many important applications such as autonomous driving. None of the methods we discussed so far can produce probabilistic segmentation with a measure of model uncertainty. Kendall et al. [66] came up with a framework to assign class labels to pixels with a measure of model uncertainty. Their method converts a convolutional encoder decoder network [52] to Bayesian convolutional network that can produce probabilistic segmentation [137] (see figure 12). This technique can not only be used to convert many state of the art architecture like FCN [127], Segnet [52] and Dilation Network [138] to output probabilistic semantic segmentation but also improves the segmentation results by 2-3% [66]. Their work [66] is inspired by [139], [137] where authors show that dropout [140] can be used to approximate inference in a Bayesian neural network. [139] shows that dropout [140] used at test time impose a Bernoulli distribution over the networks filter weights by sampling the network with randomly dropped out units at the test time. This can be considered as obtaining Monte Carlo samples from the posterior distributions over the model. [66] used this method to perform probabilistic inference over the segmentation model. It is important to note that softmax classifier produces relative probabilities between the class labels while the probability distribution from the Monte Carlo sampling [66], [139], [137] is an overall measure of the models uncertainty.

Finally, we would like to mention that deep learning based models can learn to segment from irregular data representations e.g., consuming raw point clouds (with variable number of points) without the need of any voxelization or rendering. Qi et al. [55] developed a novel deep learning architecture called PointNet that can directly take point clouds as inputs and outputs segment labels for each point in the input. Subsequent works based on deep networks which directly operates on point clouds have also demonstrated excellent performance on the semantic segmentation task [103], [102]. Instead of solely using deep networks for context modeling, some recent efforts combine both CNN and CRFs for improved segmentations. As an example, a recent work on 3D point cloud segmentation combines the

FCN and a fully connected CRF model which helps in better contextual modeling at each point in 3D [141]. To enable a fully learnable system, the CRF is implemented as a differentiable recurrent network [54]. Local context is incorporate in the proposed scheme by obtaining a voxelized representation at a coarse scale, and the predictions over voxels are used as the unary potentials in the CRF model (see Figure 13).

We note that the encoder-decoder and dilation based architectures provide a natural solution to resolve the low resolution segmentation maps in RGBD based CNN architectures. Geometrically motivated encodings of raw depth information e.g., HHA encoding [34] can help improve model accuracy. Finally, a measure of model uncertainty can be highly useful for practical applications which demand high safety.

6.4 Physics-based Reasoning

6.4.1 Prologue and Significance

A scene is a static picture of the visual world. However, when humans look at the static image, they can infer hidden dynamics in a scene. As an example, from a still picture of a football field with players and a ball, we can understand the pre-existing motion patterns and guess the future events which are likely to happen in a scene. As a result, we can plan our moves and take well-informed decisions. In line with this human cognitive ability, efforts have been made in computer vision to develop an insight into the underlying physical properties of a scene. These include estimating both current and future dynamics from a static scene [142], [143], understanding the support relationships and stability of objects [144], [145], [146], volumetric and occlusion reasoning [147], [79], [84]. Applications of such algorithms include task and motion planning for robots, surveillance and monitoring.

6.4.2 Challenges

Key challenges for physics-based reasoning include:

- This task requires starting with very limited information (e.g., a still image) and performing extrapolation to predict rich information about scene dynamics.
- A desirable characteristic is to adequately model prior information about the physical world.
- Physics based reasoning requires algorithms to reason about the contextual informations.

6.4.3 Methods Overview

Dynamics Prediction: Mottaghi et al. [142] predicted the forces acting on an object and its future motion patterns to develop a deep physical understanding of a scene. To this end, they mapped a real world scenario to a set of 3D physical abstractions which model the motion of an object and the forces acting on it in the simplest terms e.g., a ball that is rolling, falling, bouncing or moving along a projectile. This mapping is performed using a neural network with two branches, the first one processes a 2D real image while the other one processes 3d abstractions. The 3D abstractions were obtained from game rendering engines and their corresponding RGB, depth, surface normal and optical flow

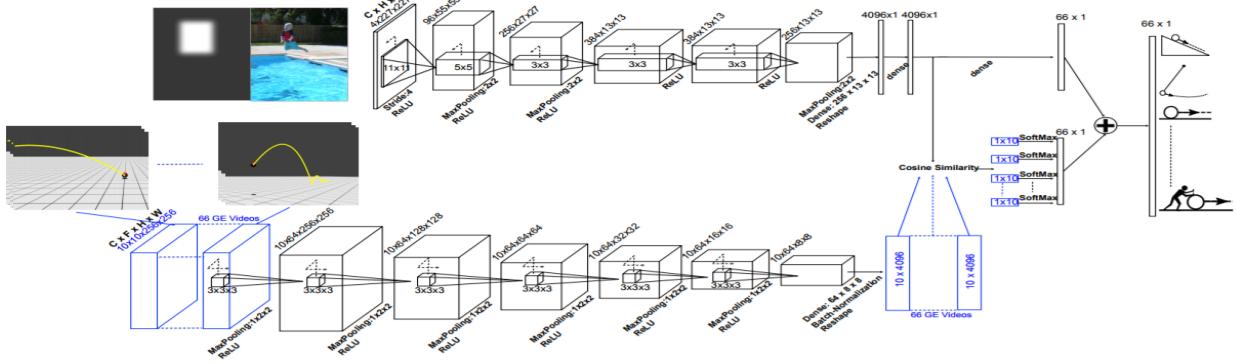


Fig. 14: Newtonian Neural Network (N^3) [142]. The top stream of the architecture takes RGB image augmented with localization map of the targeted object and the bottom stream processes inputs from a game engine. Features from both streams are combined using cosine similarity and maximum response is used to find the scenario that best describes object motion in an image. (Courtesy of [142])

data was fed to the deep network as input. Based on the mapped 3D abstraction, long term motion patterns from a static image were predicted (see Figure 14).

Wu et al. [143] proposed a generative model based on behavioral studies with the argument that physical scene understanding developed by human brain is a simulation of a mental physics engine [148]. The mental engine carries physical information about the world objects and the Newtonian laws they obey, and performs simulations to understand and infer scene dynamics. The proposed generative model, called '*Galileo*', predicts physical attributes of objects (i.e., 3D shape position, mass and friction) by taking the feedback from a physics engine which estimates the future scene dynamics by performing simulations. An interesting aspect of this work is that a deep network was trained using the predictions from the Galileo, resulting in a model which can efficiently predict physical properties of objects and future scene dynamics in static images.

Support Relationships: Alongside the hidden dynamics in a scene, there exist rich physical relationships in a scene which are important for scene understanding. As an example, a book on a table will be supported by the table surface and the table will be supported by the floor or the wall. These support relationships are important for robotic manipulation and interaction in man-made environments. Silberman et al. [145] proposed a CRF model to segment cluttered indoor environments and identify support relationships between the objects in RGB-D imagery. They categorized a scene into four geometric classes, namely *ground*, *fixed structures* (e.g., walls, ceiling), *furniture* (e.g., cabinet, tables) and *props* (small moveable objects). The overall energy function incorporated both local features (e.g., appearance cues) and pairwise interactions between objects (e.g., physical proximity). An integer programming formulation was introduced to efficiently minimize the energy function. The incorporation of support relationship information has been shown to improve the performance on other relevant tasks such as the scene parsing [149], [150].

Stability Analysis: In static scenes, it is highly unlikely to find objects which are unstable with respect to gravity. This physical concept has been employed in scene understanding to recover geometrically and physically stable

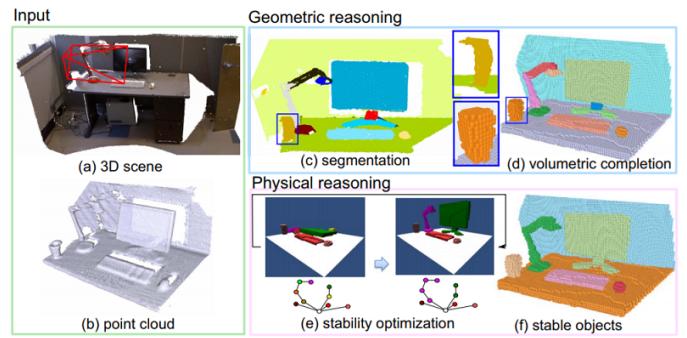


Fig. 15: A 3D scene converted into point cloud before feeding to geometric and physical reasoning module [144]. (Courtesy of [144])

objects and scene parses. Note that the support relationships predicted in [145] does not ensure the physical stability of objects. Zheng et al. [144] reasoned about the stability of 3D volumetric shapes, which were recovered from either a sparse or a dense 3D point cloud of indoor scenes (available from range sensors). A parse graph was built such that each primitive was constrained to be stable under gravity, and the falling primitives were grouped together to form stable candidates. The graph labeling problem was solved using the Swendsen-Wang Cut partitioning algorithm [146]. They noted that such a reasoning helps in achieving better performance on linked tasks such as object segmentation and 3D volume completion (see Figure 15).

While [144] performed physical reasoning utilizing mainly depth information, Jia et al. [147] incorporate both color and depth data for such analysis. Similar to [144], [147] also fits 3D volumetric shapes on RGB-D images and performs physics-based reasoning by considering their 3D intersections, support relationships and stability. As an example, a plausible explanation of a scene is the one, where 3D shapes cannot overlap each other, are supported by each other and will not fall under gravity. An energy function was defined over the over-segmented image and a number of unary and pairwise features were used to account for stability, support, appearance and volumetric

characteristics. The energy function was minimized using a randomized sampling approach [151] which either splits or merges the individual segments to obtain improved segmentations. They used the physical information for semantic scene segmentation, where it was shown to improve the performance.

Hazard Detection: An interesting direction from the previous works is to predict which objects can potentially fall in a scene. This can be highly useful to ensure safety and avoid accidents in work places (e.g., a construction site), domestic environments (e.g., child care) and due to natural disasters (e.g., earth quake). Zheng et al. [152] first estimated potential causes of disturbance (i.e. human activity and natural disasters) and then predicted the potentially unstable objects which can fall as a result of disturbance. Given a 3D point cloud, a '*disturbance field*' is predicted for a possible type of disturbance (e.g. using motion capture data for human movement) and its effect is estimated using the mechanics principles (e.g., conservation of energy and momentum after collision). In terms of predicting scene dynamics, this approach goes beyond inferring motions from the given static image, rather it considers "what if?" scenarios and predicts associated dynamics. More recently, Dupre et al.[153] have proposed to use CNN to perform automatic risk assessment in scenes.

Occlusion Reasoning: Occlusion relationship is another important physical and contextual cue, that commonly appears in cluttered scenes. Wang et al. [79] showed that occlusion reasoning helps in object detection. They introduced a hough voting scheme, which uses depth context at multiple levels (e.g., object relationship with near-by, far-away and occlusion patches) in the model to jointly predict the object centroid and its visibility mask. They used a dictionary learning approach based on local features such as Histogram of Gradients (HOG) [154] and Textons [155]. An interesting result was that the occlusion relationships are important contextual cues which can be useful for object detection and segmentation. In another subsequent work, Bonde et al. [84] used occlusion information computed from depth data to recognize individual object instances. They used a random decision forest classifier trained using a max-margin objective to improve the recognition performance.

6.5 Object Pose Estimation

6.5.1 Prologue and Significance

The pose estimation task deals with finding object's position and orientation with respect to a specific coordinate system. Information about an object's pose is crucial for object manipulation by robotic platforms and scene reconstruction e.g., by fitting 3D CAD models. Note that the pose estimation task is highly related to the object detection task, therefore existing works address both problems sequentially [156] or in a joint framework [82], [117], [157]. Direct feature matching techniques (e.g., between images and models) have also been explored for pose estimation [158], [81].

6.5.2 Challenges

Important difficulties that pose estimation algorithms encounter are:

- The requirement of detecting objects and estimating their orientation at the same makes this task particularly challenging.
- Object's pose can vary significantly from one scene to another, therefore algorithm should be invariant to these changes.
- Occlusions and deformations make the pose estimation task difficult especially when multiple objects are simultaneously present.

6.5.3 Methods Overview

Lim et al. [158] used 3D object models to estimate object pose in a given image. Object appearances can change from one scene to another due to number of factors including geometric deformation and occlusions. The challenge then is not only to retrieve the relevant model for an object but to accurately fit it to real images. Their proposed algorithm takes key detectors like geometric distances, their local correspondence and global alignment to find candidate poses. Tejani et al. [81] proposed a method for 3D object detection and pose estimation which is robust to foreground occlusion and background clutter. The presented framework called Latent-Class Hough Forests (LCHF) is based on a patch based detector called Hough Forests [159] and trained on only positive data samples of 3D synthetic model renderings. They used LINEMOD [160], a 3D holistic template descriptor, for patch representation and integrate it into random forest framework using template-based splitting function. At test time, class distributions are iteratively inferred to jointly estimate 3D object detection, pose and pixel-wise visibility map.

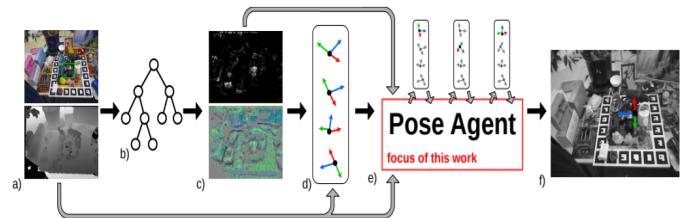


Fig. 16: Pose estimation pipeline as proposed by [161]. RGB-D input is first processed by random forest. These predictions are used to find pose candidates then a reinforcement agent refines these candidates to find the best pose. (Courtesy of [161])

Mottaghi et al. [82] argued that object detection, 3D pose estimation and sub-category recognition are correlated tasks. One task can provide complimentary information to better understand the others, therefore, they introduced a hierachal method based on a hybrid random field model that can handle both continuous and discrete variables to jointly tackle these three tasks. The main idea here is to represent objects in a hierachal fashion such that the top-layer captures high level coarse information e.g., discrete view point and object rough location and layers below capture more accurate and refined information e.g., continuous view point and object category (e.g., a car) and sub-category (e.g., specific type of a car). Similar to [82], Brannchman et al. [157] proposed a method to jointly estimate object class and its 6D pose (3D rotation, 3D translation) in a

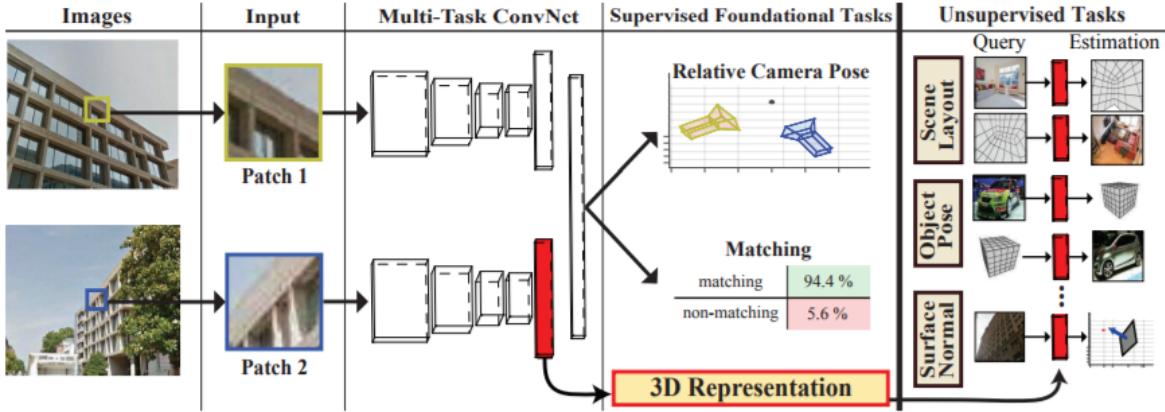


Fig. 17: CNN trained to learn generic features by matching image patches with their corresponding camera angle. These generic feature representation can be used for multiple tasks at test time including object pose estimation. (Courtesy of [162])

given RGB-D image. They trained a decision forest with 20 objects under two different lighting conditions and a set of background images. The forest consists of three trees that use color and depth difference features to jointly learn 3D object coordinates and object instances probabilities. A distinguishing feature of their approach is the ability to scale to both textured and texture-less objects.

The basic idea behind methods like [157] is to generate number of interpretations or sample pose hypothesis and then find the one that best describe an object pose. [157] achieves this by minimizing an energy function using RANSAC. [163] built upon the idea in [157], but used a CNN trained with a probabilistic approach to find the best pose hypothesis. [163] generated pose hypothesis, scored it based on their quality and decided which hypothesis to explore next. This sort of decision making is non-differentiable and does not allow an end-to-end learning framework. Krull et al. [161] improved upon the work presented in [163] by introducing a reinforcement learning to incorporate non-differentiable decision making into an end-to-end learning framework (see Figure 16).

To benefit from representation power of CNN, Schwarz et al. [156] used AlexNet[131], a large-scale CNN model trained on ImageNet visual recognition dataset. This model was used to extract features for object detection and pose estimation in RGB-D images. The novelty in their work is the pre-processing of color and depth images. Their algorithm segments objects from a given RGB image and removes the background. They colorized the depth image based on the distance from the object center. Both processed images are then fed to the AlexNet to extract features which are then concatenated before passing through a SVM classifier for object detection and a support vector regressor (SVR) for pose estimation.

One major challenge for pose estimation is that it can vary significantly from one image to another. To address this issue, Wohlhart et al. [164] proposed to create clusters of CNN based features, that are indicative of object category and their poses. The idea is to generate multiple views of each object in the database, then each object view is represented by a learned descriptor that stores information about object identity and its pose. The CNN is trained

under euclidean distance constraints such that the distance between descriptors of different objects is large and the distance between descriptors of same object is small but still provides a measure of differences in pose. In this manner, clusters of object labels and poses are formed in the descriptor space. At test time, a nearest neighbor search was used to find similar descriptor for a given object. To further improve and tackle the the pose variation issue in an end-to-end fashion, [165] formulated the pose estimation problem as regression task and introduced an end-to-end Siamese learning framework. Angles variations of the same object in multiple images are tackled by using Siamese network architecture with novel loss function to enforce similarity between the features of given training images and their corresponding poses.

CNN models have an extraordinary ability to learn generic representations that are transferable across tasks. Zamir et al. [162] validated this by training a CNN to learn 3D generic representations to simultaneously address multiple tasks. In this regard, they trained a multi-task CNN to jointly learn camera pose estimation and key point matching across extreme poses. They showed with extensive experimentation that internal representation of such a trained CNN can be used for other predictions tasks such as object pose, scene layout and surface normal estimation (see Figure 17). Another important approach on pose estimation was introduced in [65], where instead of full images, a network was trained on image patches. Kehl et al. [65] proposed a framework that consists of sampling the scene at discrete steps and extracting local and scale invariant RGB-D patches. For each patch, they compute its deep-regressed feature using a trained auto-encoder network and perform K-NN search with a codebook of local object patches. Each codebook entry holds a local 6D vote and is cast into Hough space only to survive a confident threshold. The codebook entries are coming from densely sampled synthetic views. Each entry stores its deep-regression feature and a 6D local vote. They employ a convolutional auto-encoder (CAE) that has been trained on 1.5M local RGB-D patches.

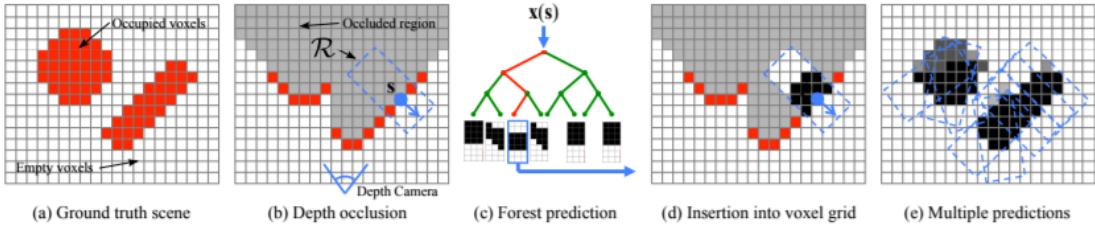


Fig. 18: A Random forest based framework to predict 3D geometry [83]. (Courtesy of [83])

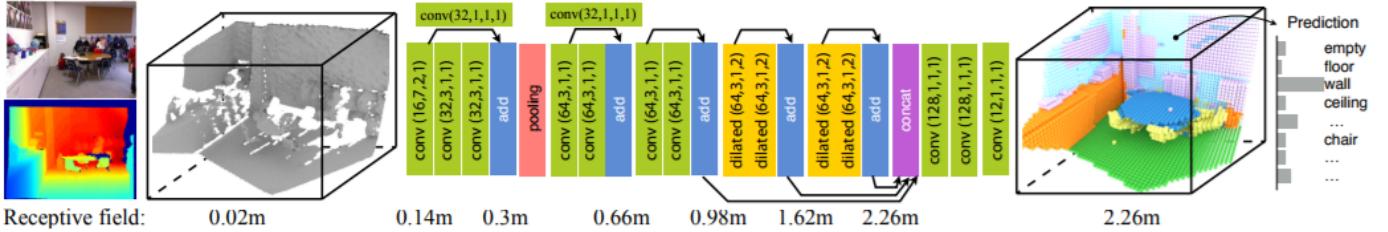


Fig. 19: SSCNet architecture [9] trained to reconstruct complete 3D scene from a single depth image. (Courtesy of [9])

6.6 3D Reconstruction from RGB-D

6.6.1 Prologue and Significance

Humans visualize and interpret surrounding environments in 3D. The 3D reasoning about an object or a scene allows a deeper understanding of the mechanics, shape and 3D texture characteristics. For this purpose, it is often desirable to recover the full 3D shape from a single or multiple RGB-D images. 3D reconstruction is useful in many applications areas including medical imaging, virtual reality and computer graphics. Since, the 3D reconstruction from densely overlapping RGB-D views of an object [166], [167] is relatively a simpler problem, here we focus on scene reconstruction from either a single or a set of RGB-D images with partial occlusions leading to incomplete information. .

6.6.2 Challenges

3D reconstruction is highly challenging problem because:

- Complete 3D reconstruction from incomplete information is an ill-posed problem with no unique solution.
- This problem poses a significant challenge due to sensor noise, low depth resolution, missing data and quantization errors.
- It requires appropriately incorporating external information about the scene or object geometry for a successful reconstruction.

6.6.3 Methods Overview

3D reconstruction from a single RGB-D image has recently gained popularity due to the availability of cheap depth sensors and powerful representation learning networks. This task is also called as the '*shape or volumetric completion*' task, since a RGB-D image provides a sparse and incomplete point cloud which is completed to produce a full 3D output. For this task, CRF models have been a natural choice because of their flexibility to encode geometric and stability relationships to generate physically viable outputs [73], [144]. Specifically, Kim et al. [73] proposed a CRF model

defined over voxels to jointly reconstruct 3D volumetric output along with the semantic category labels for each voxel. Such a joint formulation helps in modeling the complex interplay between semantic and geometric information in a scene. Firman et al. [83] proposed a structured prediction framework developed using a Random Forest to predict the 3D geometry given the observed incomplete shapes. Unlike [73], a shortcoming of their model is that it does not uses semantic details of voxels alongside the geometric information (see Figure 18).

With the success of deep learning, the above mentioned ideas have recently been formulated as end-to-end trainable networks with several interesting extensions. As an example, Song et al. [9] proposed a 3D CNN to jointly perform semantic voxel labeling and scene completion from a single RGB-D image. The CNN architecture makes use of successful ideas in deep learning such as skip connections [168] and dilated convolutions [138] to aggregate scene context and the use of a large-scale dataset (SUNCG) (see Figure 19). A convolutional LSTM based recurrent network has been proposed in [63] for 3D reconstruction of individual objects (in contrast to complete scenes as in [9]). First, an object view is encoded followed by learning a representation using LSTM which is then used for decoding. The benefit of this approach is that the latent representations can be stored in the memory (LSTM) and updated if more views of an object become available. Another similar approach for shape completion uses first an encoder-decoder architecture to obtain a coarse 3D output which is refined using similar high-resolution shapes available as prior knowledge [69]. This incorporates both bottom-up and top-down knowledge transfer (i.e. using shape category information along with the incomplete input) to recover better quality 3D outputs. Gupta et al. [169] investigated a similar data-driven approach by first identifying individual object instances in a scene and using a library of common indoor objects to retrieve and align the 3D model with the given RGB-D object instance. This approach, however, does not implement an end-to-end learnable pipeline and focuses only on object

reconstruction instead of a full scene reconstruction.

Early work on 3D reconstruction from multiple overlapping RGB-D images used the concept of averaging TSDF obtained from each of the RGB-D views [170], [171]. However, the TSDF based reconstruction techniques require a large number of highly overlapping images due to their inability to complete occluded shapes. More recently, OctNet based representations have been used in [172], which allow scaling 3D CNNs to work on considerably high dimensional volumetric inputs compared to the regular voxel based models [173]. The octree based representations take account of empty spaces in 3D environments and use low spatial resolution voxels for the unoccupied regions, thus leading to faster processing with high resolutions in 3D deep networks [67]. In contrast to the regular octree based methods [67], the scene completion task requires the prediction of reconstructed scene along with a suitable 3D partitioning for the octree representation. These two outputs are predicted using a u-shaped 3D encoder-decoder network in [172], where the short-cut connections exist between the corresponding coarse-to-fine layers in the encoder and the decoder modules.

6.7 Saliency Prediction

6.7.1 Prologue and Significance

The human visual system selectively attends to salient parts of a scene and performs a detailed understanding for the most salient regions. The detection of salient regions corresponds to important objects and events in a scene and their mutual relationships. In this section, we will review saliency estimation approaches which use a variety of 2.5/3D sensing modalities including RGB-D [175], [176], stereopsis [177], [178], light-field imaging [179] and point-clouds [180]. Saliency prediction is valuable in several applications e.g., user experience analysis, scene summarization, automatic image/video tagging, preferential processing on resource constrained devices, object tracking and novelty detection.

6.7.2 Challenges

Important problems for the saliency prediction task are:

- Saliency is a complex function of different factors including appearance, texture, background properties, location, depth etc. It is a challenge to model these intricate relationships.
- It requires both top-down and bottom-up cues to accurately model objects saliency.
- A key requisite is to adequately encode the local and global context.

6.7.3 Methods Overview

Lang et al. [175] were the first to introduce a RGB-D and 3D dataset (NUS-3DSaliency) with corresponding eye-fixation data from human viewers. They analyzed the differences between the human attention maps for 2D and 3D data and found depth to be an important cue for visual attention (e.g., attention is focused more on nearby depth ranges). To learn the relationships between visual saliency and depth, a generative model was trained to learn their joint distributions.

They showed that the incorporation of depth resulted in a consistent improvement for previous saliency detection methods designed for 2D images.

Based on the insight that salient objects are likely to appear at different depths, Peng et al. [176] proposed a multi-stage model where local, global and background contrast-based cues were used to predict a rough estimate of saliency. This initial saliency estimate was used to calculate a foreground probability map which was combined with an object prior to generate final saliency predictions. The proposed method was evaluated on a newly-introduced large-scale benchmark dataset. In another similar approach, Coptadi et al. [181] calculated local 3D shape and layout features (e.g., plane and normal cues) using the depth information to improve object saliency. Feng et al. [182] advocated that simple depth-contrast based features create confusions for rich backgrounds. They proposed a new descriptor which measures the enclosure provided by the background to foreground salient objects.

More recently, [183] proposed to use a CNN for RGB-D saliency prediction. However, their approach is not end-to-end trainable as they first extract several hand-crafted features and fuse them together followed by an off-line smoothing and saliency prediction stage with in a CNN. Shigematsu et al. [174] extended a RGB based saliency detection network (ELD-Net [184]) for the case of RGBD saliency detection (see Figure 20). They augmented the high-level feature description from a pre-trained CNN with a number of low-level feature descriptions such as the depth contrast, angular disparity and background enclosure [182]. Due to the limited size of available RGB-D datasets, this technique relies on the weights learned on color image based saliency datasets.

Stereoscopic images provide approximate depth information based on the disparity map between a pair of images. This additional information has been shown to assist in visual saliency detection especially when the salient objects do not carry significant color and texture cues. Niu et al. [177] introduced an approach based on the disparity contrast to incorporate depth information in saliency detection. Further, a prior was introduced based on domain knowledge in stereoscopic photography, which prefers regions inside the viewing comfort zone to be more salient. Fang et al. [178] developed on similar lines and used appearance as well as depth feature contrast from stereo images. A Gaussian model was used to weight the distance between the patches, such that both the global and local contrast can be accounted for saliency detection.

Plenoptic camera technology can capture the light field of a scene, which provides both the intensity and direction of the light rays. Light field cameras provide the flexibility to refocus after photo capture and can provide a depth map in both indoor and outdoor environments. Li et al. [179] used the focusness and depth information available from light field cameras to improve saliency detection. Specifically, frequency domain analysis was performed to measure focusness in each image. This information was used alongside depth to estimate foreground and background regions, which were subsequently improved using contrast and objectness measures.

In contrast to the above techniques which mainly aug-

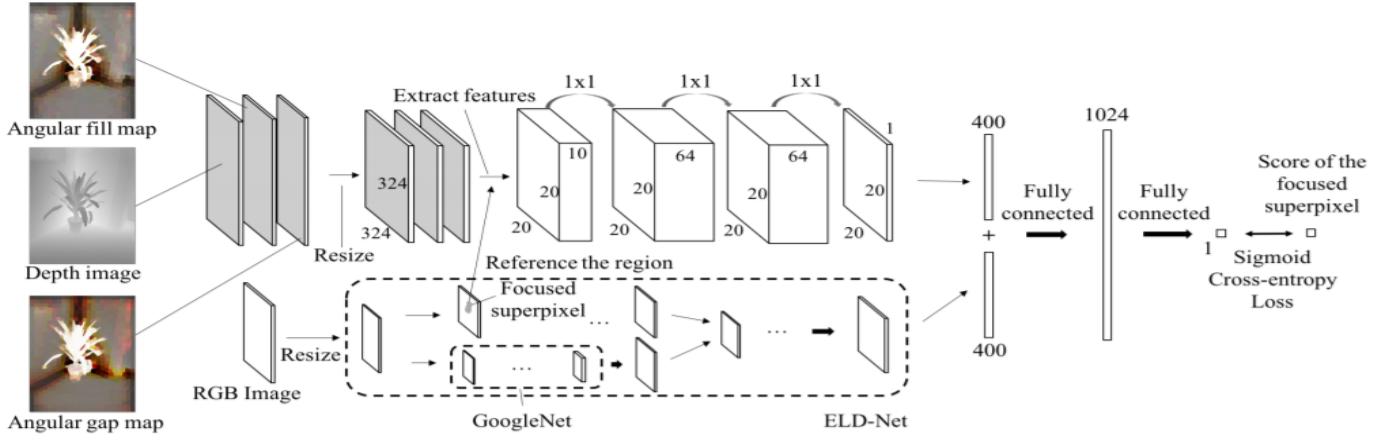


Fig. 20: Network architecture for saliency prediction [174]. RGB saliency features are fused with super-pixel based handcrafted features to get the overall saliency score. (Courtesy of [174])

ment depth information for saliency prediction, [180] detected salient patterns in 3D city-scale point clouds to identify land-mark buildings. To this end, they introduced a distance measure which quantifies the uniqueness of a landmark by considering its distinctiveness compared to the neighborhood. While this work is focused on outdoor man-made structures, to the best of our knowledge, the problem of finding salient objects in indoor 3D scans is not investigated in the literature.

6.8 Affordance Prediction

6.8.1 Prologue and Significance

Object based relationships (e.g., chairs are close to desk) have been used with success in scene understanding tasks such as semantic segmentation and holistic reasoning [25]. However, an interesting direction to interpret indoor scenes is by understanding the functionality or affordances of objects [185] i.e., what actions can be performed on a particular object (e.g., one can sit on a chair, place a coffee cup on a table). These characteristics of objects can be used as attributes, which have been found to be useful to transfer knowledge across categories [186]. Such a capability is important in application domains such as assistive, domestic and industrial robotics, where the robots need to actively interact with the surrounding environments.

6.8.2 Challenges

Affordance detection is challenging because:

- This task requires information from multiple sources and reasons about the content to discover relationships.
- It often requires modeling the hidden context (e.g., humans not present in the scene) to predict the correct affordances of objects.
- Reasoning about physical and material properties is crucial for this affordance detection.

6.8.3 Methods overview

A seminal work on affordance reasoning by Grabner et al. [187] estimated places where a person can ‘sit’ in an indoor

3D scene. Their key idea was to predict affordance attributes by assuming the presence of an interacting entity i.e., a human. The functional attributes proved to be a complementary source of information which in turn improved the ‘chair’ detection performance. On similar lines, Jiang et al. [188], [74] hallucinated humans in indoor environments to predict the human-object relationships. To this end, a latent CRF model was introduced, which jointly infers the human pose and object affordances. The proposed probabilistic graphical model was composed of objects as nodes and their relationships were encoded as graph edges. Alongside these, latent variables were used to represent hidden human context. The relationships between object and humans were used to perform 3D semantic labeling of point clouds.

Koppula and Saxena [75] used object affordances in a RGB-D image based CRF model to forecast the future human actions so that an assistive robot can generate a response in time. [189] suggested affordance descriptors which model the way an object is operated by a human in a RGB-D video. The above mentioned approaches deal with the affordance prediction for generic objects. Myers et al. [77] introduced a new dataset comprising of everyday use tools (e.g., hammer, knife). A given image was first divided into super-pixels, followed by computation of a number of geometric features such as normals and curvedness. A sparse coding based dictionary learning approach was used to identify parts and predict the corresponding affordances. For large dictionary sizes, such an approach can be quite computationally expensive, therefore a random forest based classifier was proposed for real-time applications. Note that all of the approaches mentioned so far, including [77], used hand-crafted features for affordance prediction.

More recently, automatic feature learning mechanisms such as CNNs have been used for object affordance prediction [190], [191], [68]. Nquyen et al. [68] proposed a convolutional encoder-decoder architecture to predict grasp affordances for tools using RGB-D images. The network input was encoded as a HHA encoding [34] of depth along with the color image. A more generic affordance prediction framework was presented in [191], which used a multi-scale CNN to provide affordance segmentations for indoor scenes

(see Figure 21). Their architecture explicitly used mid-level geometric and semantic representations such as labels, surface normals and depth maps at coarse and fine levels to effectively aggregate information. Ye et al. [190] framed the affordance prediction problem as a region detection task and used the VGGnet CNN to categorize a region into a rich set of functional classes e.g., whether a region can afford an open, move, sit or manipulate operation.

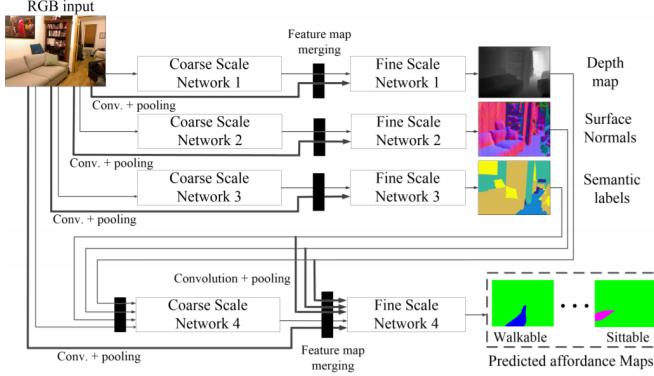


Fig. 21: A multi-scale CNN proposed in [191]. The coarse scale network extracts global representations encoding wide context while the fine-scale network extracts local representations such as object boundaries. Affordance labels are predicted by combining both representations. (Courtesy of [191])

6.9 Holistic/Hybrid Approaches

6.9.1 Prologue and Significance

Up till now, we have covered individual tasks that are important to develop an understanding about e.g., scene semantics, constituent objects and their locations, object functionalities and their saliency. In holistic scene understanding, a model aims to simultaneously reason about multiple complimentary aspects of a scene to provide a detailed scene understanding. Such an integration of individual tasks can lead to practical systems which require joint reasoning such as robotic platforms interacting with the real world (e.g., automated systems for hazard detection and quick response and rescue). In this section, we will review some of the significant efforts for holistic 2.5D/3D scene understanding. We will outline the important challenges and explore different ways the information from multiple sources is integrated in the literature for the specific case of indoor scenes [192], [70], [193], [150], [194].

6.9.2 Challenges

Important obstacles for holistic scene understanding are:

- Accurately modeling relationships between objects and background is a hard task in real-world environments due to the complexity of inter-object interactions.
- Efficient training and inference is difficult due to the requirement of reasoning at multiple levels of scene decomposition.
- Integration of multiple individual tasks and complementing one source of information with another is a key challenge.

6.9.3 Methods Overview

Li et al. [192] proposed a Feedback Enabled Cascaded Classification Model (FE-CCM), which combines individual classifiers trained for a specific task e.g., object, event detection, scene classification and saliency prediction. This combination is performed in a cascaded fashion with a feedback mechanism to jointly learn all task specific models for scene understanding and robot grasping. They argued that with the feedback mechanism, FE-CCM learns meaningful relationships between sub-tasks. An important benefit of FE-CCM [192] is that it can be trained on heterogeneous datasets meaning it does not require data points to have labels for all the tasks. Similar to [192], a two-layer generic model with a feedback mechanism was presented in [195]. In contrast to the above methods, [70] presented a holistic graphical model (a CRF) to integrate scene geometry, relations between objects, interaction of objects with scene environment for 3D object recognition (see Figure 23). They extended Constrained Parametric Min-Cuts (CPMC) [196] method to generate cuboids from RGB-D images. These cuboids contain information about scene geometry, appearance and help in modeling contextual information for objects.

To understand complex scenes, it is desirable to learn interactions between scene elements e.g., scene structures-to-object interaction and object-to object interaction. Choi et al. [193] proposed a method that can learn these scene interactions and integrate information at multiple levels to estimate scene composition. Their hierachal scene model learns to reason about complex scenes by fusing together scene classification, layout estimation and object detection tasks. The model takes a single image as an input and generates a parse graph that best fit the image observations. The graph root represents the scene category and layout, while the graph leaves represent objects detections. In between, they introduced novel 3D Geometric Phrases (3DGP) that encode semantic and geometric relations between objects. A Reversible Jump Markov Chain Monte Carlo (RJMCMC) sampling technique was used to search for the best fit graph for the given image.

The wide contextual information available from human eyes plays a critical role in human scene understanding. Field of view (FOV) of a typical camera is only 15% compared to human vision system. Zhang et al. [56] argued that due to limited FOV, a typical camera cannot capture full details presented in a scene e.g., number of objects or occurrences of an object. Therefore, a model built on single images with limited FOV cannot exploit full contextual information in a scene. Their proposed method takes a 360-degree panorama view and generates 3D box representation of the room layout along with all of the major objects. Multiple 3D representations are generated using variety of image characteristics and then a SVM classifier is used to find the best one.

Zhang et. al [25] developed a 3D deep learning architecture to jointly learn furniture category and location from a single depth image (see Figure 22). They introduced a template representation for 3D scene to be consumed by a deep net for learning. A scene template encodes a set of objects and their contextual information in the scene.

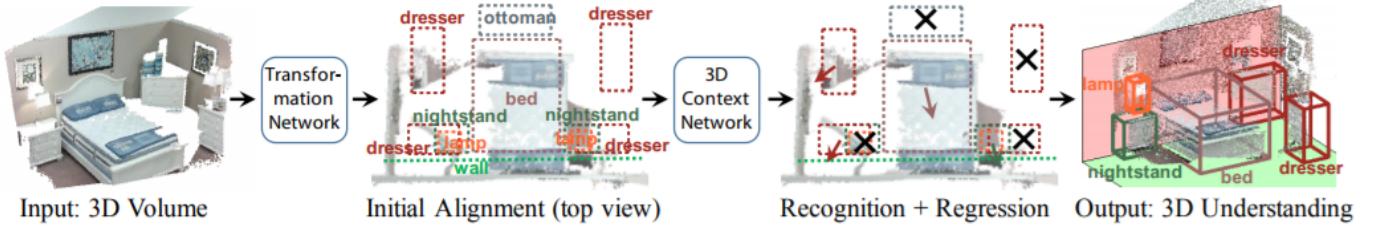


Fig. 22: 3D scene understanding framework as proposed by [25]. A volumetric representation is first derived from depth image and aligned with the input data, next a 3D CNN estimates the objects presence and adjust them based on holistic scene features for full 3D scene understanding (Courtesy of [25]).

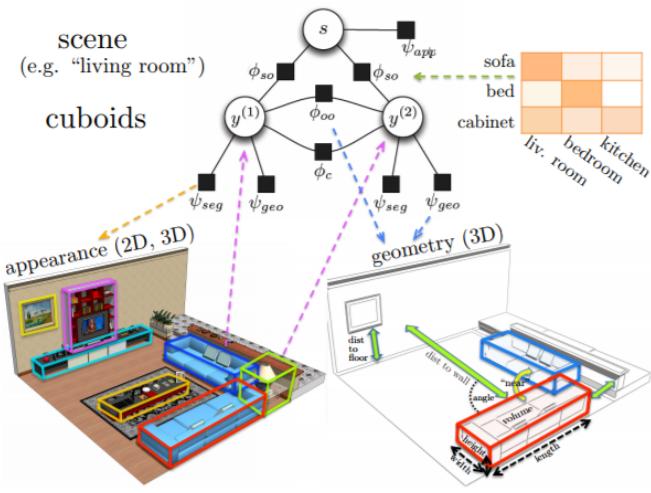


Fig. 23: Appearance and geometric properties are represented using object cuboids which are then used to define scene to object and object to object relations. This information is integrated by a CRF model for holistic scene understanding. (Courtesy of [70])

After training, their so called DeepContext net learns to recognize multiple objects and their locations based on both local object and contextual features. Zhou et al. [150] introduced a method to jointly learn instance segmentation, semantic labeling and support relationships by exploiting hierarchical segmentation using a Markov Random Field for indoor RGB-D images. The inference in the MRF model is performed using an integer linear program that can be efficiently solved.

Note that some of the approaches discussed previously under the individual sub-tasks also perform holistic reasoning. For example, [145] jointly models the semantic labels and physical relationships between objects, [9] jointly reconstructs 3D scene and provides voxel labels, [72] concurrently performs segmentation and cuboid detection while [81] detects the objects and their 3D pose in a unified framework. Such a task integration helps in incorporating wider context and results performance improvements across the tasks, however the model complexity significantly increases and efficient learning and inference algorithms are therefore required for a feasible solution.

7 EVALUATION AND DISCUSSION

7.1 Evaluation Metrics

7.1.1 Metric for classification

Classification is the task of categorizing a scene or an object in to its relevant class. Classifier performance can be measured by classification accuracy as follows:

$$\text{Accuracy} = \frac{\text{Number of samples correctly classified}}{\text{Total number of samples}}. \quad (1)$$

7.1.2 Metric for object detection

Object detection is the task of recognizing each object instance and its category. Average precision (AP) is a commonly used metric to measure an object detector's performance:

$$\text{AP} = \frac{1}{\text{classes}} \sum_{i \in \text{classes}} \frac{TP(i)}{TP(i) + FP(i)}, \quad (2)$$

where

- $TP(i)$ represent the number of true positives i.e., predictions for class i that match with the ground-truth.
- $FP(i)$ represent the number of false positives i.e., predictions for class i that do not match with the ground-truth.

7.1.3 Metric for pose estimation

Object pose estimation task deals with finding object's position and orientation with respect to a certain coordinate system. The percentage of correctly predicted poses is the efficiency measure of a pose estimator. A pose estimation is consider correct if average distance between the estimated pose and ground truth is less than a specific threshold (e.g., 10% of the object diameter).

7.1.4 Saliency prediction evaluation metric

Saliency prediction deals with the detection of important objects and events in a scene. There are many evaluation metrics for saliency prediction including similarity, area under the curve, normalized scanpath saliency (NSS) and F-measure (F_β).

$$\bullet \text{ Similarity} = \text{SM} - \text{FM}, \quad (3)$$

where

- SM is the predicted saliency map.

- FM is the human eye fixation map or ground truth.

$$\bullet \quad NSS(p) = \frac{SM(p) - \mu_{SM}}{\delta_{SM}}, \quad (4)$$

where

- SM is the predicted saliency map.
- p is the location of one fixation.
- μ_{SM} is the mean value of predicted saliency map.
- δ_{SM} is the standard deviation of the predicted saliency map.
- Final NSS score is the average of all $NSS(p)$ for all fixation.

$$\bullet \quad F_\beta = \frac{(1 + \beta^2) * (precision) * (recall)}{\beta^2 * (precision) + (recall)}, \quad (5)$$

where β^2 is a hyper parameter normally set to 0.3.

7.1.5 Segmentation evaluation metrics and results

Semantic segmentation is the task that involves labeling each pixel in a given image by its corresponding class. Following evaluation metrics are used to evaluate model efficiency:

$$\begin{aligned} PA &= \frac{\sum_i n_{ii}}{\sum_i t_i} \\ MA &= \left(\frac{1}{n_{cl}} \right) \sum_i \frac{n_{ii}}{t_i} \\ MIoU &= \left(\frac{1}{n_{cl}} \right) \sum_i \frac{n_{ii}}{\left(t_i + \sum_j n_{ji} - n_{ii} \right)} \\ FWIoU &= \left(\sum_k t_k \right)^{-1} \sum_i \frac{t_i n_{ii}}{\left(t_i + \sum_j n_{ji} - n_{ii} \right)}, \end{aligned} \quad (6)$$

where

- PA = Pixel Accuracy
- MA = Mean Accuracy
- MIoU = Mean Intersection over Union
- FWIoU = Frequency weighted Intersection over Union
- n_{cl} are the number of different classes.
- n_{ii} is the number of pixels of class i predicted to belong to class i .
- n_{ji} is the number of pixels of class i predicted to belong to class j .
- $t_i = \sum_j n_{ji}$ is the total number of pixels belong to class i .

7.1.6 Affordance prediction evaluation metric

Affordance is the ability of a robot to predict possible actions that can be performed on or with an object. The common evaluation metric for affordance is the accuracy:

$$\text{Accuracy} = \frac{\text{Number of affordances correctly classified}}{\text{Total number of affordances}} \quad (7)$$

7.1.7 3D Reconstruction

3D reconstruction is a task of recovering full 3D shape from a single or multiple RGB-D images. Intersection over union can be used as evaluation metric for the 3D reconstruction task.

$$\text{IoU} = \sum_i \frac{v_{ii}}{\left(T_i + \sum_j v_{ji} - v_{ii} \right)} \quad (8)$$

where

- v_{ii} is the number of voxels of class i predicted to belong to class i .
- v_{ji} is the number of voxels of class i predicted to belong to class j .
- $T_i = \sum_j v_{ji}$ is the total number of voxels belong to class i .

TABLE 3: Performance comparison between prominent 3D object classification methods on ModelNet40 dataset [95].

Method	Key Point	Accuracy
Wu et al. [95]	Feature learning from 3D voxelized input	77.3
Wu et al. [88]	Unsupervised feature learning using 3D generative adversarial modeling	83.3
Qi et al. [55]	Point-cloud based representation learning	86.2
Su et al. [58]	Multi-view 2D images for feature learning	90.1
Qi et al. [89]	Volumetric and multi-view feature learning	91.4
Brock et al. [197]	Generative and discriminative voxel modeling	95.5

TABLE 4: Performance comparison between state-of-the-art 3D object detection methods

Method	Key Point	Dataset	mAP
Song et al. [24]	3D CNN for amodal object detection	SUN	42.1
Lahoud and Ghanem [198]	Using 2D detection algorithms for 3D object detection	RGB-D	45.1
Ren et al. [115]	Based on oriented gradient descriptor	[36]	47.6
Qi et al. [199]	Processing raw point cloud using CNN		54.0
Song et al. [24]	3D CNN for amodal object detection	NYUv2	36.3
Deng et al. [108]	Two-stream CNN with 2D object proposals for 3D amodal detection.	[10]	40.9

7.2 Discussion on Results

In this section, we present quantitative comparisons on a set of key sub-tasks including shape classification, object detection, segmentation, saliency prediction and 3D reconstruction. Wu et al. [95] created a 3D dataset, ModelNet40, for shape classification which is publicly available. Since then a number of algorithms have been proposed and tested on this dataset. Performance comparison is shown in Table 3. It can be seen that most successful methods use CNN to extract features from 3D voxelized or 2D multi-view representations and the method based on generative and discriminative modeling [197] outperformed other competitors on this dataset. Object detection results are shown in Table 4. Note that point cloud is normally a difficult to process data representation and therefore, it is usually converted to other representations such as voxel or octree before feeding to an algorithm. However, it is evident from

TABLE 5: Performance of state-of-art segmentation methods on NYU v2 [10] dataset

Method	Key Point	No. of classes	Pixel Accuracy	Mean Class Accuracy	Mean IOU
Silberman et al. [145]	Hand-crafted features (SIFT)	4	58.6	-	-
Couprise et al. [124]	CNN as feature extractor used with super-pixels	4	64.5	63.5	-
Couprise et al. [124]	CNN as feature extractor used with super-pixels	13	52.4	36.2	-
Hermans et al. [122]	2D to 3D label transfer and use of 3D CRF	13	54.2	48.0	-
Wolf et al. [200]	3D decision forests	13	64.9	55.6	39.5
Tchapmi et al. [141]	CNN with fully connected CRF	13	66.8	56.4	43.5
Gupta et al. [34]	CNN as feature extractor with novel depth embedding	40	60.3	-	28.6
Long et al. [127]	Encoder decoder architecture with FCN	40	65.4	46.1	34.0
Badrinarayanan et al. [52]	Encoder decoder architecture with reduced parameters	40	66.1	36.0	23.6
Kendall et al. [66]	Encoder decoder architecture with probability estimates	40	68.0	45.8	32.4

TABLE 6: Performance of state-of-the-art methods for 3D reconstruction through scene completion on Rendered NYU [83].

Method	Key Point	Train set	Precision	Recall	IoU
Zheng et al. [144]	Based on geometric and physical reasoning	Rendered NYU	60.1	46.7	34.6
Firman et al. [83]	Estimating occluded voxels in a scene by comparison with a similar scene	[83]	66.5	69.7	50.8
Song et al. [9]	Dilation CNN with joint semantic labeling for scene completion		75.0	92.3	70.3
Song et al. [9]	Dilation CNN with joint semantic labeling for scene completion	Rendered NYU [83] + SUNCG [40]	75.0	96.0	73.0

TABLE 7: Performance of state-of-art saliency detection methods on LSFD [179] dataset

Method	Key Point	F-measure
He et al. [201]	Super-pixels with CNN	0.698
Zhang et al. [202]	Using min. barrier distance transform	0.730
Qin et al. [203]	Dynamic evolution modeling	0.731
Wang et al. [204]	Based on local and global features	0.738
Peng et al. [176]	Fusion of depth modality with RGB	0.704
Ju et al. [205]	Based on anisotropic center-surround diff.	0.757
Ren et al. [206]	Based on depth and normal priors	0.788
Feng et al. [182]	Local background enclosure based feature	0.712
Qu et al. [183]	Fusing engineered features with CNN	0.844

result comparison 4 that processing point cloud directly by CNN can boost the performance. Next comparison is shown in table 5 for semantic segmentation task on NYU [10] dataset. It is evident that encoder-decoder network architecture with a measure of uncertainty [66] outperforms for RGBD semantic segmentation task. Saliency prediction algorithms are compared against LSFD [179] in Table 7 where the most promising results are delivered when hand-crafted features are combined with CNN feature representation [183]. 3D reconstruction algorithms are compared in table 6 where again best performing method is based on CNN that effectively incorporates contextual information using dilations.

8 CHALLENGES AND FUTURE DIRECTIONS

Light-weight Models: In the last few years, we have seen a dramatic growth in the capabilities of computational resources for machine vision applications [207]. However, the deployment of large-scale models on hand-held devices still faces several challenges such as limited processing capability, low memory and power resources. The design of practical systems require a careful consideration about model complexity and desired performance. It also demands development of novel light-weight deep learning

models, highly parallelize-able algorithms and compact representations for 3D data.

Transfer Learning: Scene understanding involves prediction about several inter-related tasks, e.g., semantic labeling can benefit from scene categorization and object detection and vice versa. The basic tasks such as scene classification, scene parsing and object detection have normally large quantities of annotated examples available for training, however, other tasks such as affordance prediction, support prediction and saliency prediction do not have huge datasets available. A natural choice for these problems is to use the knowledge acquired from pre-training performed on a large-scale 2D, 2.5D or 3D annotated dataset. However, the two domains are not always closely related and it is an open problem to optimally adapt an existing model to the desired task such that the knowledge is adequately transferred to the new domain.

Emergence of Hybrid Models: Holistic scene understanding requires high flexibility in the learned model to incorporate domain knowledge and priors based on previous history of experiences and interactions with the physical world. Furthermore, it is often required to model wide contextual relationships between super-pixels, objects, labels or among similar type scenes to be able to reason about more complex tasks. Deep networks have turned out to be an excellent resource for automatic feature learning, however they only allow limited flexibility. We foresee a growth in the development of hybrid models, which take advantage from complementary strengths of model classes to better learn the contextual relationships.

Data Imbalance: In several scene understanding tasks such as semantic labeling, some class representations are scarce while others have abundant examples. Learning a model which respects both type of categories and equally performs well on frequent as well as less frequent ones remains a challenge and needs further investigation.

Multi-task Learning: Given the multi-task nature of complete scene understanding, a suitable but less investi-

gated paradigm is to jointly train models on a number of end-tasks. As an example, for the case of semantic instance segmentation, one approach could be jointly regress the object instance bounding box, its fore-ground mask and the category label for each box. Such a formulation can allow learning robust models without undermining the performance on any single task.

Learning from Synthetic Data: The availability of large-scale CAD model libraries and impressive rendering engines have provided huge quantities of synthetic data (esp. for indoor environments). Such data eliminates the extensive labeling requirements required for real data, which is a bottleneck for training large-scale data-hungry deep learning models [9]. Recent studies show that models trained on synthetic data can achieve strong performance on real data [43], [208].

Multi-modal Feature Learning: Joint feature learning across different sensing modalities has been investigated in the context of outdoor scenes (e.g., using LIDAR and stereo cameras [209]), but not for the case of indoor scenes. Recent sensing devices such as Matterport allow collection of multiple modalities (e.g., point clouds, mesh and depth data) in the indoor environments. Among these modalities, some have existing large-scale pre-trained models which are unavailable for other modalities. An open research problem is to leverage the frequently available data modalities and perform cross-modality knowledge transfer [37].

Robust and Explainable Models: With the adaptability of deep learning models in safety critical applications including self-driving cars, visual surveillance and medical field, there comes a responsibility to evaluate and explain the decision-making process of these models. We need to develop easy to interpret frameworks to better understand decision-making of deep learning systems like in [210], where Frosst et al. explained CNN decision-making using a decision tree. Furthermore, deep learning models have shown vulnerability to adversarial attacks. In these attacks, carefully-perturbed inputs are designed to mislead the model at inference time [211]. There is not only a need to develop methods that can actively detect and alarm against adversarial attacks, but better adversarial training mechanisms are also required to make model robust against these vulnerabilities.

REFERENCES

- [1] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 3354–3361.
- [2] T. Breuer, G. R. G. Macedo, R. Hartanto, N. Hochgeschwender, D. Holz, F. Hegger, Z. Jin, C. Müller, J. Paulus, M. Reckhaus *et al.*, "Johnny: An autonomous service robot for domestic environments," *Journal of intelligent & robotic systems*, vol. 66, no. 1-2, pp. 245–272, 2012.
- [3] M. Teistler, O. J. Bott, J. Dormeier, and D. P. Pretschner, "Virtual tomography: a new approach to efficient human-computer interaction for medical imaging," in *Medical Imaging 2003: Visualization, Image-Guided Procedures, and Display*, vol. 5029. International Society for Optics and Photonics, 2003, pp. 512–520.
- [4] M. Billinghurst and A. Duenser, "Augmented reality in the classroom," *Computer*, vol. 45, no. 7, pp. 56–63, 2012.
- [5] S. Widodo, T. Hasegawa, and S. Tsugawa, "Vehicle fuel consumption and emission estimation in environment-adaptive driving with or without inter-vehicle communications," in *Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE*. IEEE, 2000, pp. 382–386.
- [6] "Qualcomm announces 3d camera technology for android ecosystem," <https://www.qualcomm.com/news/releases/2017/08/15/qualcomm-first-announce-depth-sensing-camera-technology-designed-android> accessed: 2017-12-10.
- [7] "Who: Vision impairment and blindness," <http://www.who.int/mediacentre/factsheets/fs282/en/>, accessed: 2017-12-08.
- [8] A. Rodríguez, L. M. Bergasa, P. F. Alcantarilla, J. Yebes, and A. Cela, "Obstacle avoidance system for assisting visually impaired people," in *Proceedings of the IEEE Intelligent Vehicles Symposium Workshops, Madrid, Spain*, vol. 35, 2012, p. 16.
- [9] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, "Semantic scene completion from a single depth image," *arXiv preprint arXiv:1611.08974*, 2016.
- [10] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," *Computer Vision-ECCV 2012*, pp. 746–760, 2012.
- [11] D. G. Lowe, "Object recognition from local scale-invariant features," in *IEEE international conference on Computer vision*, vol. 2. Ieee, 1999, pp. 1150–1157.
- [12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.
- [13] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," *Computer vision-ECCV 2006*, pp. 404–417, 2006.
- [14] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," in *European Conference on Computer Vision (ECCV)*, 2006.
- [15] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 12, pp. 2037–2041, 2006.
- [16] M. Charlton, "Helmholtz on perception: Its physiology and development." *Archives of Neurology*, vol. 19, no. 3, pp. 349–349, 1968.
- [17] K. Koffka, *Principles of Gestalt psychology*. Routledge, 2013, vol. 44.
- [18] H. Barrow and J. Tenenbaum, "Computer vision systems," *Computer vision systems*, vol. 2, 1978.
- [19] D. Marr, "Vision: A computational investigation into the human representation and processing of visual information," *WH San Francisco: Freeman and Company*, 1982.
- [20] L. G. Roberts, "Machine perception of three-dimensional solids," Ph.D. dissertation, Massachusetts Institute of Technology, 1963.
- [21] A. Guzmán, "Decomposition of a visual scene into three-dimensional bodies," in *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*. ACM, 1968, pp. 291–304.
- [22] T. O. Binford, "Visual perception by computer," in *Proceeding, IEEE Conf. on Systems and Control*, 1971.
- [23] I. Biederman, "Human image understanding: Recent research and a theory," *Computer vision, graphics, and image processing*, vol. 32, no. 1, pp. 29–73, 1985.
- [24] S. Song and J. Xiao, "Deep sliding shapes for amodal 3d object detection in rgbd images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 808–816.
- [25] Y. Zhang, M. Bai, P. Kohli, S. Izadi, and J. Xiao, "Deepcontext: context-encoding neural pathways for 3d holistic scene understanding," *arXiv preprint arXiv:1603.04922*, 2016.
- [26] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.
- [27] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
- [28] Y. Yakimovsky and J. A. Feldman, "A semantics-based decision theory region analyser."
- [29] Y.-i. Ohta, T. Kanade, and T. Sakai, "An analysis system for scenes containing objects with substructures."
- [30] A. R. Hanson and E. M. Riseman, "VISIONS: A computer system for interpreting scenes," in *Computer Vision Systems*, A. R. Hanson and E. M. Riseman, Eds. New York: Academic Press, 1978.

- [31] R. A. Brooks, "Model-based three-dimensional interpretations of two-dimensional images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 2, pp. 140–150, 1983.
- [32] R. J. Douglass, "Interpreting three-dimensional scenes: A model building approach," *Computer Graphics and Image Processing*, vol. 17, no. 2, pp. 91–113, 1981.
- [33] M. D. Levine and S. I. Shaheen, "A modular computer vision system for picture segmentation and interpretation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 5, pp. 540–556, 1981.
- [34] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning rich features from rgb-d images for object detection and segmentation," in *European Conference on Computer Vision*. Springer, 2014, pp. 345–360.
- [35] J. Xiao, A. Owens, and A. Torralba, "Sun3d: A database of big spaces reconstructed using sfm and object labels," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1625–1632.
- [36] S. Song, S. P. Lichtenberg, and J. Xiao, "Sun rgb-d: A rgb-d scene understanding benchmark suite," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 567–576.
- [37] I. Armeni, A. Sax, A. R. Zamir, and S. Savarese, "Joint 2D-3D-Semantic Data for Indoor Scene Understanding," *ArXiv e-prints*, Feb. 2017.
- [38] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Nießner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3d: Learning from rgb-d data in indoor environments," *arXiv preprint arXiv:1709.06158*, 2017.
- [39] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017.
- [40] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, "Semantic scene completion from a single depth image," *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
- [41] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view rgb-d object dataset," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1817–1824.
- [42] B.-S. Hua, Q.-H. Pham, D. T. Nguyen, M.-K. Tran, L.-F. Yu, and S.-K. Yeung, "Scenenn: A scene meshes dataset with annotations," in *3D Vision (3DV), 2016 Fourth International Conference on*. IEEE, 2016, pp. 92–101.
- [43] J. McCormac, A. Handa, S. Leutenegger, and A. J. Davison, "Scenenet rgb-d: 5m photorealistic images of synthetic indoor trajectories with ground truth," *arXiv preprint arXiv:1612.05079*, 2016.
- [44] M. Savva, A. X. Chang, P. Hanrahan, M. Fisher, and M. Nießner, "Pigraphs: Learning interaction snapshots from observations," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, p. 139, 2016.
- [45] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 573–580.
- [46] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond pascal: A benchmark for 3d object detection in the wild," in *Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on*. IEEE, 2014, pp. 75–82.
- [47] H. Badino, U. Franke, and D. Pfeiffer, "The stixel world-a compact medium level representation of the 3d-world." in *DAGM Symposium*. Springer, 2009, pp. 51–60.
- [48] N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor," in *Proceedings of the International Conference on Computer Vision - Workshop on 3D Representation and Recognition*, 2011.
- [49] Planner5d. [Online]. Available: <https://planner5d.com/>.
- [50] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.
- [51] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.
- [52] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *arXiv preprint arXiv:1511.00561*, 2015.
- [53] R. Socher, B. Huval, B. Bath, C. D. Manning, and A. Y. Ng, "Convolutional-recursive deep learning for 3d object classification," in *Advances in Neural Information Processing Systems*, 2012, pp. 656–664.
- [54] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, "Conditional random fields as recurrent neural networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1529–1537.
- [55] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," *arXiv preprint arXiv:1612.00593*, 2016.
- [56] Y. Zhang, S. Song, P. Tan, and J. Xiao, "Panocontext: A whole-room 3d context model for panoramic scene understanding," in *European Conference on Computer Vision*. Springer, 2014, pp. 668–686.
- [57] Y. Li, S. Pirk, H. Su, C. R. Qi, and L. J. Guibas, "Fpnn: Field probing neural networks for 3d data," in *Advances in Neural Information Processing Systems*, 2016, pp. 307–315.
- [58] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3d shape recognition," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 945–953.
- [59] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [60] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.
- [61] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural Networks*, vol. 18, no. 5, pp. 602–610, 2005.
- [62] A. Graves, G. Wayne, and I. Danihelka, "Neural turing machines," *arXiv preprint arXiv:1410.5401*, 2014.
- [63] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3d-r2n2: A unified approach for single and multi-view 3d object reconstruction," in *European Conference on Computer Vision*. Springer, 2016, pp. 628–644.
- [64] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International Conference on Machine Learning*, 2015, pp. 2048–2057.
- [65] W. Kehl, F. Milletari, F. Tombari, S. Ilic, and N. Navab, "Deep learning of local rgb-d patches for 3d object detection and 6d pose estimation," in *European Conference on Computer Vision*. Springer, 2016, pp. 205–220.
- [66] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," *arXiv preprint arXiv:1511.02680*, 2015.
- [67] G. Riegler, A. O. Ulusoy, and A. Geiger, "Octnet: Learning deep 3d representations at high resolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
- [68] A. Nguyen, D. Kanoulas, D. G. Caldwell, and N. G. Tsagarakis, "Detecting object affordances with convolutional neural networks," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016, pp. 2765–2770.
- [69] A. Dai, C. R. Qi, and M. Nießner, "Shape completion using 3d-encoder-predictor cnns and shape synthesis," *arXiv preprint arXiv:1612.00101*, 2016.
- [70] D. Lin, S. Fidler, and R. Urtasun, "Holistic scene understanding for 3d object detection with rgbd cameras," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1417–1424.
- [71] S. Wang, S. Fidler, and R. Urtasun, "Holistic 3d scene understanding from a single geo-tagged image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3964–3972.
- [72] S. H. Khan, X. He, M. Bennamoun, F. Sohel, and R. Togneri, "Separating objects and clutter in indoor scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4603–4611.
- [73] B.-s. Kim, P. Kohli, and S. Savarese, "3d scene understanding by voxel-crf," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1425–1432.

- [74] Y. Jiang, H. S. Koppula, and A. Saxena, "Modeling 3d environments through hidden human context," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 10, pp. 2040–2053, 2016.
- [75] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 1, pp. 14–29, 2016.
- [76] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Advances in neural information processing systems*, 2007, pp. 801–808.
- [77] A. Myers, C. L. Teo, C. Fermüller, and Y. Aloimonos, "Affordance detection of tool parts from geometric features," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1374–1381.
- [78] L. Bo, X. Ren, and D. Fox, "Learning hierarchical sparse features for rgb-(d) object recognition," *The International Journal of Robotics Research*, vol. 33, no. 4, pp. 581–599, 2014.
- [79] T. Wang, X. He, and N. Barnes, "Learning structured hough voting for joint object detection and occlusion reasoning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1790–1797.
- [80] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [81] A. Tejani, D. Tang, R. Kouskouridas, and T.-K. Kim, "Latent-class hough forests for 3d object detection and pose estimation," in *European Conference on Computer Vision*. Springer, 2014, pp. 462–477.
- [82] R. Mottaghi, Y. Xiang, and S. Savarese, "A coarse-to-fine model for 3d pose estimation and sub-category recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 418–426.
- [83] M. Firman, O. Mac Aodha, S. Julier, and G. J. Brostow, "Structured prediction of unobserved voxels from a single depth image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5431–5440.
- [84] U. Bonde, V. Badrinarayanan, and R. Cipolla, "Robust instance recognition in presence of occlusion and clutter," in *European Conference on Computer Vision*. Springer, 2014, pp. 520–535.
- [85] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.
- [86] S. R. Gunn *et al.*, "Support vector machines for classification and regression," *ISIS technical report*, vol. 14, pp. 85–86, 1998.
- [87] S. Gupta, P. Arbelaez, and J. Malik, "Perceptual organization and recognition of indoor scenes from rgbd images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 564–571.
- [88] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling," in *Advances in Neural Information Processing Systems*, 2016, pp. 82–90.
- [89] C. R. Qi, H. Su, M. Nießner, A. Dai, M. Yan, and L. J. Guibas, "Volumetric and multi-view cnns for object classification on 3d data," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5648–5656.
- [90] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 5, pp. 898–916, 2011.
- [91] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Computer vision and pattern recognition, 2006 IEEE computer society conference on*, vol. 2. IEEE, 2006, pp. 2169–2178.
- [92] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 2011, pp. 215–223.
- [93] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard, "Multimodal deep learning for robust rgbd object recognition," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 681–687.
- [94] A. Wang, J. Cai, J. Lu, and T.-J. Cham, "Modality and component aware feature fusion for rgbd scene classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5995–6004.
- [95] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: A deep representation for volumetric shapes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1912–1920.
- [96] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.
- [97] D. Maturana and S. Scherer, "Voxnet: A 3d convolutional neural network for real-time object recognition," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 922–928.
- [98] B. T. Phong, "Illumination for computer generated pictures," *Communications of the ACM*, vol. 18, no. 6, pp. 311–317, 1975.
- [99] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *arXiv preprint arXiv:1405.3531*, 2014.
- [100] B. Shi, S. Bai, Z. Zhou, and X. Bai, "Deeppano: Deep panoramic representation for 3-d shape recognition," *IEEE Signal Processing Letters*, vol. 22, no. 12, pp. 2339–2343, 2015.
- [101] V. Hegde and R. Zadeh, "Fusionnet: 3d object classification using multiple data representations," *arXiv preprint arXiv:1607.05695*, 2016.
- [102] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *arXiv preprint arXiv:1706.02413*, 2017.
- [103] W. Zeng and T. Gevers, "3dcontextnet: Kd tree guided hierarchical learning of point clouds using local contextual cues," *arXiv preprint arXiv:1711.11379*, 2017.
- [104] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.
- [105] H. Jiang, "Finding approximate convex shapes in rgbd images," in *European Conference on Computer Vision*. Springer, 2014, pp. 582–596.
- [106] S. Song and J. Xiao, "Sliding shapes for 3d object detection in depth images," in *European conference on computer vision*. Springer, 2014, pp. 634–651.
- [107] X. Chen, K. Kundu, Y. Zhu, A. G. Berneshawi, H. Ma, S. Fidler, and R. Urtasun, "3d object proposals for accurate object class detection," in *Advances in Neural Information Processing Systems*, 2015, pp. 424–432.
- [108] Z. Deng and L. J. Latecki, "Amodal detection of 3d objects: Inferring 3d bounding boxes from 2d ones in rgbd-depth images."
- [109] D. Novotny, D. Larlus, and A. Vedaldi, "Learning 3d object categories by looking around them," *arXiv preprint arXiv:1705.03951*, 2017.
- [110] H. Jiang and J. Xiao, "A linear approach to matching cuboids in rgbd images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2171–2178.
- [111] M. Z. Zia, M. Stark, and K. Schindler, "Towards scene understanding with detailed 3d object representations," *International Journal of Computer Vision*, vol. 112, no. 2, pp. 188–203, 2015.
- [112] M. Aharon, M. Elad, and A. Bruckstein, "rmk-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4311–4322, 2006.
- [113] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*. IEEE, 1993, pp. 40–44.
- [114] T. Malisiewicz, A. Gupta, and A. A. Efros, "Ensemble of exemplar-svms for object detection and beyond," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 89–96.
- [115] Z. Ren and E. B. Sudderth, "Three-dimensional object detection and layout prediction using clouds of oriented gradients," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1525–1533.
- [116] A. Karpathy, S. Miller, and L. Fei-Fei, "Object discovery in 3d scenes via shape analysis," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2088–2095.
- [117] N. Sedaghat, M. Zolfaghari, and T. Brox, "Orientation-boosted voxel nets for 3d object recognition," *arXiv preprint arXiv:1604.03351*, 2016.

- [118] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 328–335.
- [119] K. Xu, Y. Shi, L. Zheng, J. Zhang, M. Liu, H. Huang, H. Su, D. Cohen-Or, and B. Chen, "3d attention-driven depth acquisition for object identification," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, p. 238, 2016.
- [120] C. Cadena and J. Košecka, "Semantic parsing for priming object detection in rgb-d scenes," in *3rd Workshop on Semantic Perception, Mapping and Exploration*, 2013.
- [121] A. C. Müller and S. Behnke, "Learning depth-sensitive conditional random fields for semantic segmentation of rgb-d images," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 6232–6237.
- [122] A. Hermans, G. Floros, and B. Leibe, "Dense 3d semantic mapping of indoor scenes from rgb-d images," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2631–2638.
- [123] Z. Deng, S. Todorovic, and L. Jan Latecki, "Semantic segmentation of rgbd images with mutex constraints," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1733–1741.
- [124] C. Couprise, C. Farabet, L. Najman, and Y. LeCun, "Toward real-time indoor semantic segmentation using depth information," *Journal of Machine Learning Research*, 2014.
- [125] E. Kalogerakis, M. Averkiou, S. Maji, and S. Chaudhuri, "3d shape segmentation with projective convolutional networks," *arXiv preprint arXiv:1612.02808*, 2016.
- [126] L. Schneider, M. Jasch, B. Fröhlich, T. Weber, U. Franke, M. Pollefeys, and M. Rätsch, "Multimodal neural networks: Rgb-d for semantic segmentation and object detection," in *Scandinavian Conference on Image Analysis*. Springer, 2017, pp. 98–109.
- [127] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.
- [128] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers, "Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture," in *Asian Conference on Computer Vision*. Springer, 2016, pp. 213–228.
- [129] C. Couprise, C. Farabet, Y. LeCun, and L. Najman, "Causal graph-based video segmentation," in *Image Processing (ICIP), 2013 20th IEEE International Conference on*. IEEE, 2013, pp. 4249–4253.
- [130] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.
- [131] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [132] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.
- [133] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.
- [134] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1395–1403.
- [135] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning depth from single monocular images using deep convolutional neural fields," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 10, pp. 2024–2039, 2016.
- [136] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 4, pp. 640–651, 2017.
- [137] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*, 2016, pp. 1050–1059.
- [138] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.
- [139] Y. Gal and Z. Ghahramani, "Bayesian convolutional neural networks with bernoulli approximate variational inference," *arXiv preprint arXiv:1506.02158*, 2015.
- [140] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [141] L. P. Tchapmi, C. B. Choy, I. Armeni, J. Gwak, and S. Savarese, "Segcloud: Semantic segmentation of 3d point clouds," *arXiv preprint arXiv:1710.07563*, 2017.
- [142] R. Mottaghi, H. Bagherinezhad, M. Rastegari, and A. Farhadi, "Newtonian scene understanding: Unfolding the dynamics of objects in static images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3521–3529.
- [143] J. Wu, I. Yildirim, J. J. Lim, B. Freeman, and J. Tenenbaum, "Galileo: Perceiving physical object properties by integrating a physics engine with deep learning," in *Advances in neural information processing systems*, 2015, pp. 127–135.
- [144] B. Zheng, Y. Zhao, J. C. Yu, K. Ikeuchi, and S.-C. Zhu, "Beyond point clouds: Scene understanding by reasoning geometry and physics," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3127–3134.
- [145] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," *Computer Vision-ECCV 2012*, pp. 746–760, 2012.
- [146] A. Barbu and S.-C. Zhu, "Generalizing swendsen-wang to sampling arbitrary posterior probabilities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1239–1253, 2005.
- [147] Z. Jia, A. C. Gallagher, A. Saxena, and T. Chen, "3d reasoning from blocks to stability," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 5, pp. 905–918, 2015.
- [148] P. W. Battaglia, J. B. Hamrick, and J. B. Tenenbaum, "Simulation as an engine of physical scene understanding," *Proceedings of the National Academy of Sciences*, vol. 110, no. 45, pp. 18 327–18 332, 2013.
- [149] Z. Jia, A. Gallagher, A. Saxena, and T. Chen, "3d-based reasoning with blocks, support, and stability," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1–8.
- [150] W. Zhuo, M. Salzmann, X. He, and M. Liu, "Indoor scene parsing with instance segmentation, semantic labeling and support relationship inference," in *Conference on Computer Vision and Pattern Recognition*, no. EPFL-CONF-227441, 2017.
- [151] J. Chang and J. W. Fisher, "Efficient mcmc sampling with implicit shape representations," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 2081–2088.
- [152] B. Zheng, Y. Zhao, C. Y. Joey, K. Ikeuchi, and S.-C. Zhu, "Detecting potential falling objects by inferring human action and natural disturbance," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 3417–3424.
- [153] R. Dupre, G. Tzimiropoulos, and V. Argyriou, "Automated risk assessment for scene understanding and domestic robots using rgbd data and 2.5 d cnns at a patch level," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 5–6.
- [154] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.
- [155] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," in *European conference on computer vision*. Springer, 2006, pp. 1–15.
- [156] M. Schwarz, H. Schulz, and S. Behnke, "Rgbd object recognition and pose estimation based on pre-trained convolutional neural network features," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1329–1335.
- [157] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother, "Learning 6d object pose estimation using 3d object coordinates," in *European conference on computer vision*. Springer, 2014, pp. 536–551.
- [158] J. J. Lim, H. Pirsiavash, and A. Torralba, "Parsing ikea objects: Fine pose estimation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2992–2999.
- [159] J. Gall, A. Yao, N. Razavi, L. Van Gool, and V. Lempitsky, "Hough forests for object detection, tracking, and action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 11, pp. 2188–2202, 2011.

- [160] S. Hinterstoisser, S. Holzer, C. Cagniart, S. Ilic, K. Konolige, N. Navab, and V. Lepetit, "Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 858–865.
- [161] A. Krull, E. Brachmann, S. Nowozin, F. Michel, J. Shotton, and C. Rother, "Poseagent: Budget-constrained 6d object pose estimation via reinforcement learning," *arXiv preprint arXiv:1612.03779*, 2016.
- [162] A. R. Zamir, T. Wekel, P. Agrawal, C. Wei, J. Malik, and S. Savarese, "Generic 3d representation via pose estimation and matching," in *European Conference on Computer Vision*. Springer, 2016, pp. 535–553.
- [163] A. Krull, E. Brachmann, F. Michel, M. Ying Yang, S. Gumhold, and C. Rother, "Learning analysis-by-synthesis for 6d pose estimation in rgb-d images," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 954–962.
- [164] P. Wohlhart and V. Lepetit, "Learning descriptors for object recognition and 3d pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3109–3118.
- [165] A. Doumanoglou, V. Balntas, R. Kouskouridas, and T.-K. Kim, "Siamese regression networks with efficient mid-level feature extraction for 3d object pose estimation," *arXiv preprint arXiv:1607.02257*, 2016.
- [166] T. Whelan, M. Kaess, M. Fallon, H. Johannsson, J. Leonard, and J. McDonald, "Kintinuous: Spatially extended kinectfusion," 2012.
- [167] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*. IEEE, 2011, pp. 127–136.
- [168] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [169] S. Gupta, P. Arbeláez, R. Girshick, and J. Malik, "Aligning 3d models to rgb-d images of cluttered scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4731–4740.
- [170] H. Hoppe, "Progressive meshes," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. ACM, 1996, pp. 99–108.
- [171] S. Choi, Q.-Y. Zhou, and V. Koltun, "Robust reconstruction of indoor scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5556–5565.
- [172] G. Riegler, A. O. Ulusoy, H. Bischof, and A. Geiger, "Octo-nfusion: Learning depth fusion from data," *arXiv preprint arXiv:1704.01047*, 2017.
- [173] J. Wu, T. Xue, J. J. Lim, Y. Tian, J. B. Tenenbaum, A. Torralba, and W. T. Freeman, "Single image 3d interpreter network," in *European Conference on Computer Vision*. Springer, 2016, pp. 365–382.
- [174] R. Shigematsu, D. Feng, S. You, and N. Barnes, "Learning rgbd salient object detection using background enclosure, depth contrast, and top-down features," *arXiv preprint arXiv:1705.03607*, 2017.
- [175] C. Lang, T. V. Nguyen, H. Katti, K. Yadati, M. Kankanhalli, and S. Yan, "Depth matters: Influence of depth cues on visual saliency," in *Computer vision–ECCV 2012*. Springer, 2012, pp. 101–115.
- [176] H. Peng, B. Li, W. Xiong, W. Hu, and R. Ji, "Rgbd salient object detection: a benchmark and algorithms," in *European conference on computer vision*. Springer, 2014, pp. 92–109.
- [177] Y. Niu, Y. Geng, X. Li, and F. Liu, "Leveraging stereopsis for saliency analysis," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 454–461.
- [178] Y. Fang, J. Wang, M. Narwaria, P. Le Callet, and W. Lin, "Saliency detection for stereoscopic images," *IEEE Transactions on Image Processing*, vol. 23, no. 6, pp. 2625–2636, 2014.
- [179] N. Li, J. Ye, Y. Ji, H. Ling, and J. Yu, "Saliency detection on light field," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2806–2813.
- [180] N. Kobyshev, H. Riemenschneider, A. Bódó-Szomorú, and L. Van Gool, "3d saliency for finding landmark buildings," in *3D Vision (3DV), 2016 Fourth International Conference on*. IEEE, 2016, pp. 267–275.
- [181] A. Ciptadi, T. Hermans, and J. M. Rehg, "An in depth view of saliency." Georgia Institute of Technology, 2013.
- [182] D. Feng, N. Barnes, S. You, and C. McCarthy, "Local background enclosure for rgbd salient object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2343–2350.
- [183] L. Qu, S. He, J. Zhang, J. Tian, Y. Tang, and Q. Yang, "Rgbd salient object detection via deep fusion," *IEEE Transactions on Image Processing*, vol. 26, no. 5, pp. 2274–2285, 2017.
- [184] G. Lee, Y.-W. Tai, and J. Kim, "Deep saliency with encoded low level distance map and high level features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 660–668.
- [185] J. J. Gibson, *The ecological approach to visual perception: classic edition*. Psychology Press, 2014.
- [186] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1778–1785.
- [187] H. Grabner, J. Gall, and L. Van Gool, "What makes a chair a chair?" in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 1529–1536.
- [188] Y. Jiang, H. Koppula, and A. Saxena, "Hallucinated humans as the hidden context for labeling 3d scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2993–3000.
- [189] A. Pieropan, C. H. Ek, and H. Kjellström, "Functional descriptors for object affordances," in *IROS 2015 Workshop*, 2015.
- [190] C. Ye, Y. Yang, R. Mao, C. Fermüller, and Y. Aloimonos, "What can i do around here? deep functional scene understanding for cognitive robots," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4604–4611.
- [191] A. Roy and S. Todorovic, "A multi-scale cnn for affordance segmentation in rgbd images," in *European Conference on Computer Vision*. Springer, 2016, pp. 186–201.
- [192] C. Li, A. Kowdle, A. Saxena, and T. Chen, "Towards holistic scene understanding: Feedback enabled cascaded classification models," in *Advances in Neural Information Processing Systems*, 2010, pp. 1351–1359.
- [193] W. Choi, Y.-W. Chao, C. Pantofaru, and S. Savarese, "Understanding indoor scenes using 3d geometric phrases," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 33–40.
- [194] S. Gupta, P. Arbeláez, R. Girshick, and J. Malik, "Indoor scene understanding with rgbd images: Bottom-up segmentation, object detection and semantic segmentation," *International Journal of Computer Vision*, vol. 112, no. 2, pp. 133–149, 2015.
- [195] C. Li, A. Kowdle, A. Saxena, and T. Chen, "A generic model to compose vision modules for holistic scene understanding," in *European Conference on Computer Vision*. Springer, 2010, pp. 70–85.
- [196] J. Carreira and C. Sminchisescu, "Cpmc: Automatic object segmentation using constrained parametric min-cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1312–1328, 2012.
- [197] A. Brock, T. Lim, J. M. Ritchie, and N. Weston, "Generative and discriminative voxel modeling with convolutional neural networks," *arXiv preprint arXiv:1608.04236*, 2016.
- [198] J. Lahoud and B. Ghanem, "2d-driven 3d object detection in rgbd images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4622–4630.
- [199] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum pointnets for 3d object detection from rgbd data," *arXiv preprint arXiv:1711.08488*, 2017.
- [200] D. Wolf, J. Prankl, and M. Vincze, "Enhancing semantic segmentation for robotics: the power of 3-d entangled forests," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 49–56, 2016.
- [201] S. He, R. W. Lau, W. Liu, Z. Huang, and Q. Yang, "Supercnn: A superpixelwise convolutional neural network for salient object detection," *International journal of computer vision*, vol. 115, no. 3, pp. 330–344, 2015.
- [202] J. Zhang, S. Sclaroff, Z. Lin, X. Shen, B. Price, and R. Mech, "Minimum barrier salient object detection at 80 fps," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1404–1412.

- [203] Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency detection via cellular automata," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 110–119.
- [204] L. Wang, H. Lu, X. Ruan, and M.-H. Yang, "Deep networks for saliency detection via local estimation and global search," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3183–3192.
- [205] R. Ju, L. Ge, W. Geng, T. Ren, and G. Wu, "Depth saliency based on anisotropic center-surround difference," in *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1115–1119.
- [206] J. Ren, X. Gong, L. Yu, W. Zhou, and M. Ying Yang, "Exploiting global priors for rgb-d saliency detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 25–32.
- [207] "Accelerating ai with gpus," <https://blogs.nvidia.com/blog/2016/01/12/accelerating-ai-artificial-intelligence-gpus/>, accessed: 2017-12-08.
- [208] J. McCormac, A. Handa, S. Leutenegger, and A. J. Davison, "Scenenet rgb-d: Can 5m synthetic images beat generic imagenet pre-training on indoor segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2678–2687.
- [209] C. Cadena, A. R. Dick, and I. D. Reid, "Multi-modal auto-encoders as joint estimators for robotics scene understanding," in *Robotics: Science and Systems*, 2016.
- [210] N. Frosst and G. Hinton, "Distilling a neural network into a soft decision tree," *arXiv preprint arXiv:1711.09784*, 2017.
- [211] X. Yuan, P. He, Q. Zhu, R. R. Bhat, and X. Li, "Adversarial examples: Attacks and defenses for deep learning," *arXiv preprint arXiv:1712.07107*, 2017.
- [212] datasets. [Online]. Available: <http://www0.cs.ucl.ac.uk/staff/M.Firman/RGDBdatasets/>
- [213] T. Hackel, N. Savinov, L. Ladicky, J. D. Wegner, K. Schindler, and M. Pollefeys, "Semantic3d. net: A new large-scale point cloud classification benchmark," *arXiv preprint arXiv:1704.03847*, 2017.
- [214] Oms2world. [Online]. Available: <http://oms2world.org>.
- [215] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky, "Depth from familiar objects: A hierarchical model for 3d scenes," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 2. IEEE, 2006, pp. 2410–2417.
- [216] Y. Zhao and S.-C. Zhu, "Scene parsing by integrating function, geometry and appearance models," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3119–3126.
- [217] W. Liu, R. Ji, and S. Li, "Towards 3d object detection with bimodal deep boltzmann machines over rgbd imagery," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3013–3021.
- [218] K. Yamaguchi, D. McAllester, and R. Urtasun, "Efficient joint segmentation, occlusion labeling, stereo and flow estimation," in *European Conference on Computer Vision*. Springer, 2014, pp. 756–771.
- [219] M. Zhu, K. G. Derpanis, Y. Yang, S. Brahmbhatt, M. Zhang, C. Phillips, M. Lecce, and K. Daniilidis, "Single image 3d object detection and pose estimation for grasping," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 3936–3943.
- [220] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [221] "Who: Number of road traffic deaths," http://www.who.int/gho/road_safety/mortality/traffic_deaths_number/en/, accessed: 2017-12-08.
- [222] Y. Ohta, *Knowledge-based interpretation of outdoor natural color scenes*. Morgan Kaufmann, 1985, vol. 4.