IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 16, 2018, accepted November 27, 2018, date of publication December 12, 2018, date of current version January 7, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2886133

Indoor Scene Understanding in 2.5/3D for Autonomous Agents: A Survey

MUZAMMAL NASEER[1], SALMAN KHAN[2,3], (Member, IEEE), AND FATIH PORIKLI[1], (Fellow, IEEE)

[1]College of Engineering and Computer Science, Australian National University, Canberra, ACT 2600, Australia
[2]Data61, Commonwealth Scientific and Industrial Research Organization, Canberra, ACT 2601, Australia
[3]Inception Institute of Artificial Intelligence, Abu Dhabi, UAE

Corresponding author: Muzammal Naseer (muzammal.naseer@anu.edu.au)

This work was supported by the Australian Research Council's Discovery Projects Funding Scheme under Project DP150104645.

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 of indoor environments. This survey paper provides a comprehensive background to this research topic. We begin with a historical perspective, followed by a popular 3D data representation 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 this paper. 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 toward the open research problems requiring further investigation.

INDEX TERMS 3D scene understanding, semantic labeling, geometry estimation, deep networks, Markov random fields.

I. INTRODUCTION

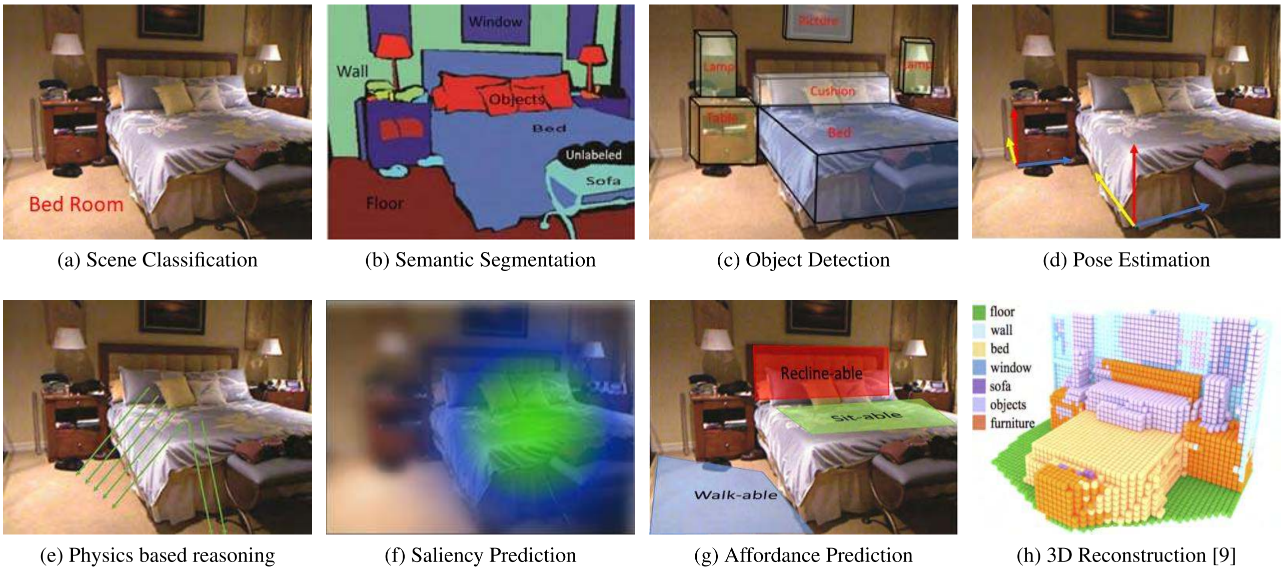
> It's not what you look at that matters, it's what you see.
>
> *H.D. Thoreau (1817-62)*

An image is simply a grid of numbers to a machine. In order to develop a comprehensive understanding of visual content, it is necessary to uncover the underlying geometric and semantic clues. As an example, given an RGB-D (2.5D) indoor scene, 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 comprehend 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.



(a) Scene Classification (b) Semantic Segmentation (c) Object Detection (d) Pose Estimation

(e) Physics based reasoning (f) Saliency Prediction (g) Affordance Prediction (h) 3D Reconstruction [9]

FIGURE 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]).

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.

II. 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., 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 Guzmán [21] as the primitives for representing generic

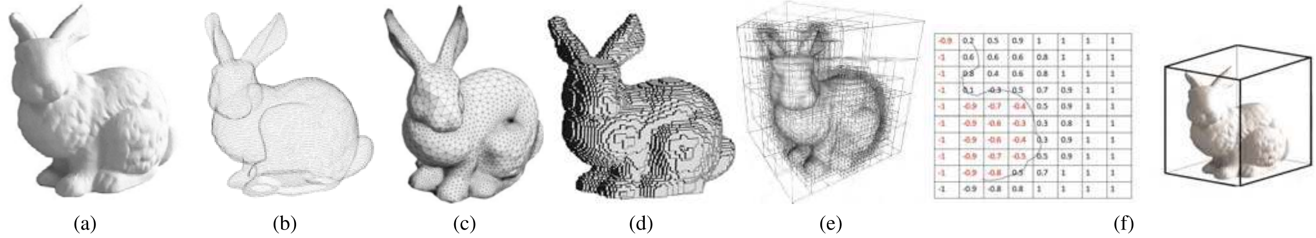


FIGURE 2. Visualization of different types of 3D data representations for Stanford bunny. (a) CAD Model. (b) Point Cloud. (c) Mesh. (d) Voxelized. (e) Octree. (f) TSDF.

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]–[27].

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. V and VI. 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.

III. 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.

A. 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.

B. 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.

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

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	⋆⋆⋆	⋆⋆⋆	⋆

C. 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 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 polygonal meshes, 'volumetric meshes' represent both the interior volume along with the object surface.

D. 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 [28]. Specifically, this geocentric embedding encodes depth image using height above the ground, horizontal disparity and angle with gravity for each pixel.

E. 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

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

Dataset	**NYUv2** [10]	**SUN3D** [29]	**SUN RGB-D** [30]	**Building Parser** [31]	**Matterport 3D** [32]	**ScanNet** [33]	**SUNCG** [34]	**RGBD Object** [35]	**SceneNN** [36]	**SceneNet RGB-D** [37]	**PiGraph** [38]	**TUM** [39]	**Pascal 3D+** [40]
Year	2012	2013	2015	2017	2017	2017	2016	2011	2016	2016	2016	2012	2014
Type	Real	Real	Real	Real	Real	Real	Synthetic	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	✗	-
Object Classes	894	-	800	13	40	50 - at least	84	51	50 - at least	255	5 subjects	✗	12
In/Outdoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	In+Out
Available Data Types													
RGB	✓	-	✓	✓	✓	-	-	✓	-	✓	✗	-	✓
Depth	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✗	-	✗
Video	✓	✓	✓	✗	✗	✓	✗	✓	✓	✓	✓	✓	✗
Point cloud	✗	✗	✓	✓	✓	✗	✗	✗	✗	✗	-	✗	✗
Mesh/CAD	✗	✗	✗	✓	✓	✓	✓	✗	✓	✗	-	✗	✓
Available Annotation Types													
Scene Classes	✓	✗	✓	✓	✓	✗	✗	✗	✓	✓	✗	✗	✗
Semantic Label	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✗	✗	✗
Object BB	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓	✗	✗	✓
Camera Poses	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓
Object Poses	✓	✗	✓	✗	✗	✗	✗	✓	✓	✗	✗	✗	✓
Trajectory	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✓	✗
Action	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗

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

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.

F. 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 [41]. 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.

G. 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, 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.

H. 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.

IV. DATASETS

High quality datasets play important role in development of machine vision algorithms. Here, we review important datasets, Table 2, for scene understanding available to researchers.

A. NYU-DEPTH

Silberman and Fergus [42] introduced NYU Depth v1 and v2 Silberman Silberman *et al.* [10] in 2011 and 2012, respectively. NYU Depth v1 [42] 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.

B. Sun3D

Sun3D [29] 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.

C. SUN RGB-D

Sun RGB-D [30] 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.

D. BUILDING PARSER

Armeni *et al.* [31] 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.

E. ScanNet

ScanNet [33] 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.

F. PiGraph

Savva *et al.* [38] 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.

G. SUNCG

SUNCG [34] 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 [43]. 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.

H. PASCAL3D+

Xiang *et al.* [40] 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 [44]) and performed 3D labeling. Further, they included additional images for each category from ImageNet dataset [45]. The resulting dataset has around 3000 object instances per category.

I. RGBD OBJECT

This dataset [35] 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.

J. TUM

Reference [39] 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.

K. SceneNN

SceneNN [36] 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.

L. Matterport3D

Matterport3D [32] 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.

M. 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.

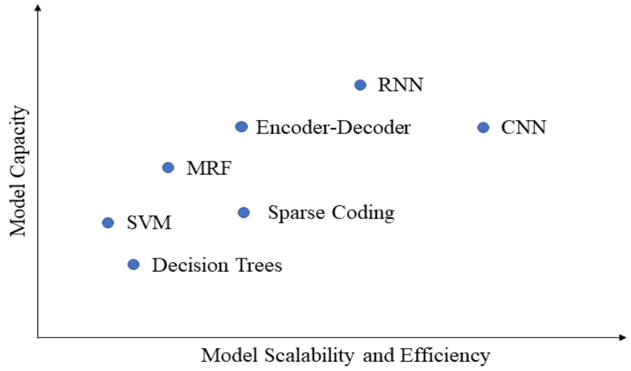


FIGURE 3. Core-techniques comparison.

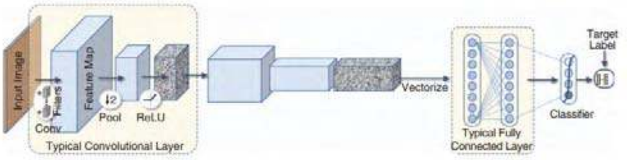


FIGURE 4. A basic CNN architecture with a convolution, pooling, activation along with a fully connected layer.

V. 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 (Fig. 3). 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., [46]–[48], which combine strengths of different core techniques to achieve better performances.

A. 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. 4, which shows the above-mentioned layers.

CNNs have shown excellent performance on many scene understanding tasks (e.g., [25]–[27], [49]–[52]). 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.

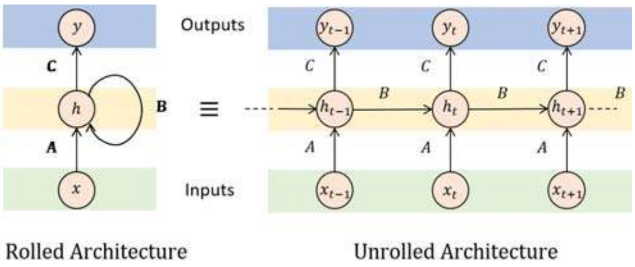


FIGURE 5. A basic RNN architecture in the rolled and unrolled form.

B. 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. 5). 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.

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) [53], Gated Recurrent Unit (GRU) [54], Bidirectional RNN (B-RNN) [55] and Neural Turing Machines (NTM) [56]. 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 [47], [57].

C. 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

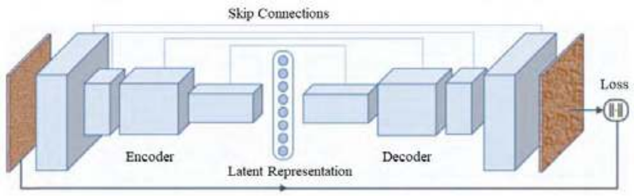


FIGURE 6. A basic autoencoder architecture.

either the original input or an output of another form (e.g., pixel labels for an image) using a 'decoder' (Fig. 6). 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.

Encoder-decoder style networks have been used in combination with both convolutional [46] and recurrent designs [58]. The applications of such designs for scene understanding tasks include [46], [59]–[63]. 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.

D. 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 interdependent patterns instead of, e.g., a single category label in the case of classification. Scene understanding tasks often involve structured prediction e.g., [64]–[69]. These models allow the incorporation of context while making local predictions. The context can be encoded in the model by pairwise 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.

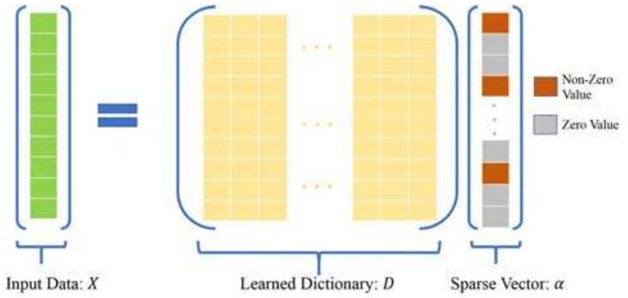


FIGURE 7. Dictionary learning for sparse coding.

E. 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 [70]. The set of basis vectors is called as a 'dictionary' (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 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 D and α can be computationally expensive. Examples of sparse coding based approaches in scene understanding literature include [71]–[73].

F. 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 [74]. 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 [71], [75]–[78].

G. 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) [79], [80] 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.

VI. A TAXONOMY OF PROBLEMS

A. IMAGE CLASSIFICATION

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 [81], automatic feature learning [47], [52], unsupervised learning [82] and work on different 3D representations such as voxels [83] and point clouds [49].

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.

3) METHODS OVERVIEW

A bottom-up approach to scene recognition was introduced in [81], 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 [84]) to RGB-D images by effectively incorporating the depth information. Note that the gPb-ucm approach produces hierarchical image segmentation by using contour information [84]. The predicted semantic segmentation maps were used as features for scene classification. They used a special pyramid formulation, similar to spatial pyramid matching approach [85], along with SVM as a classifier.

Sochar *et al.* [47] 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 [86]. 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, [87] extracted features from RGB and depth modalities via two stream networks, which were then fused together. Reference [88] 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.* [89] 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 translate 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. Reference [90] showed that DBN could learn the joint distributions of 2D image pixels and labels. Reference [89] 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, [91] advocates using 3D CNN on a voxel grid to extract meaningful representations while [51] proposes to approximate 3D spaces as volumetric fields to deal with the computational cost of directly applying 3D CNN to voxels.

Though it seems logical to build a model that can directly consume 3D shapes to recognize them (e.g., [89]), however the 3D resolution of a shape must be significantly reduced

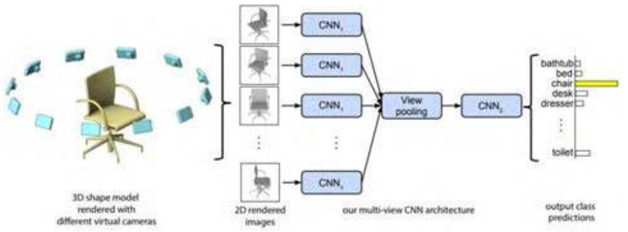


FIGURE 8. Multi-view CNN for 3D shape recognition [52]. Extracted features of different views from CNN$_1$ are pooled together before passing through CNN$_2$ for final score prediction. (Courtesy of [52]).

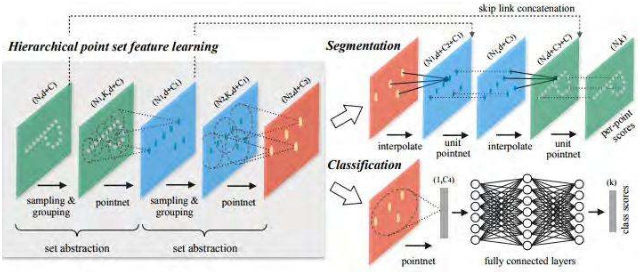


FIGURE 9. PointNet++ [96] architecture for point cloud classification and segmentation. PointNet architecture [49] is being used in a hierarchical fashion to extract local geometric features (Courtesy of [96]).

to allow feasible training of a deep neural network. As an example, 3D ShapeNets [89] used a 30 × 30 × 30 binary voxel grid to represent 3D shapes. Su *et al.* [52] 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 [92] to render 2D views of 3D shapes. Afterwards, a pre-trained VGG-M network [93] 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, [94] 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 [52] perform better than volumetric CNNs [89]. Qi *et al.* [83] argued that network architecture differences and input resolutions are the reasons for the gap in performance. Inspired from multi-view CNNs, [83] 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 [52] performance by introducing a multi-resolution scheme. Inspired by the performance efficiency of MVCNN and volumetric CNN, [95] fused both modalities to learn better features for classification.

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 [49] 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.* [96] 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 [97], 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 [89] and generative adversarial networks (GAN) [98], Wu *et al.* [82] 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. Reference [82] 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.

B. OBJECT DETECTION

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 [66], [99], [100] and the recent transition to deep neural networks based region proposal (object candidate) generation and object detection pipelines [24], [101], [102]. Apart from the supervised models, we also review unsupervised 3D object detection techniques [103].

2) CHALLENGES

Key challenges for object detection are as follows:

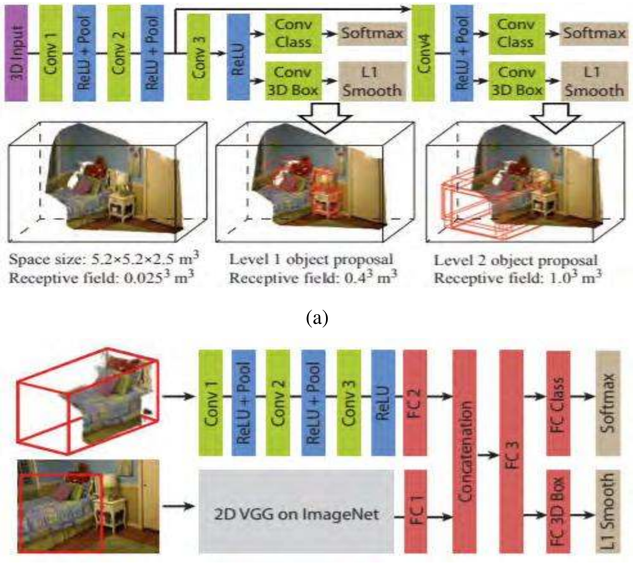
- Real world environments can be highly cluttered and object identification in such environments is very challenging.
- Detection algorithm should also be able to handle viewpoint and illuminations variations and deformations.
- In many scenarios, it is necessary to understand the scene context to successfully detect objects.
- Objects categories have a long-tail (imbalanced) distribution, which makes it challenging to model the infrequent classes.

3) METHODS OVERVIEW

Jiang *et al.* [104] 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, [99] 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.* [66] extended the technique presented in [104] 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 [24], [72], [100], [105] have been proposed to provide category information and location of each detected object instance. Reference [72] 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 [72] learns codebooks for RGB-D images via K-SVD algorithm [106], 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 [107] 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 and Xiao [100] 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 [108] 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 [100] represented objects with CAD models, other representations are also explored in literature such as [105] and [109] proposed 3D deformable wire-frame modeling and cloud of oriented gradients representation, respectively, and [110] build object detector based on 3D mesh representation of indoor scenes.



(a)

(b)

FIGURE 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]).

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. Song and Xiao [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 [111] demonstrated with their oriented-boosted

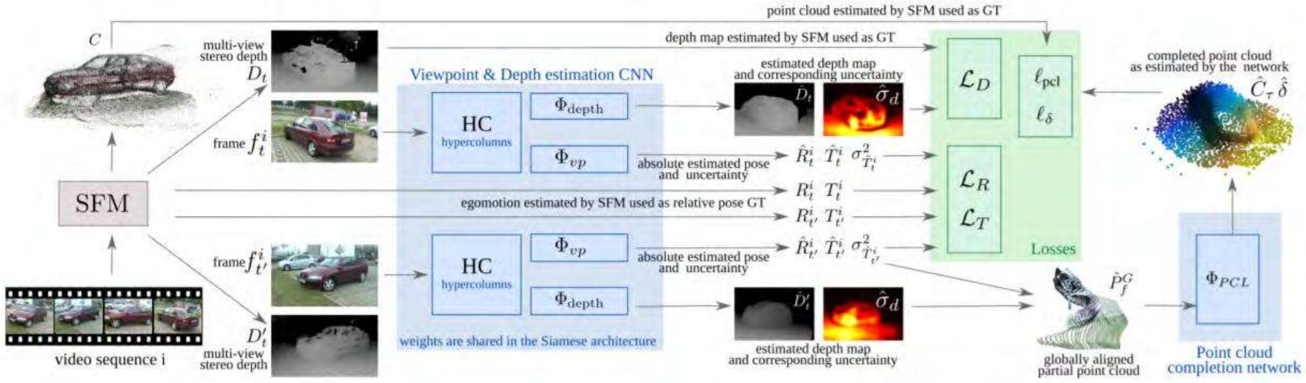

point cloud estimated by SFM used as GT
depth map estimated by SFM used as GT
C
multi-view stereo depth D_t
frame f_t^i
SFM
video sequence i
frame $f_{t'}^i$
D'_t multi-view stereo depth
Viewpoint & Depth estimation CNN
HC hypercolumns
Φ_{depth}
Φ_{vp}
weights are shared in the Siamese architecture
estimated depth map and corresponding uncertainty
$\hat{D}_t$
$\hat{\sigma}_d$
absolute estimated pose and uncertainty
$\hat{R}_t^i\ \hat{T}_t^i\ \sigma^2_{\hat{T}_t^i}$
$R_t^i\ T_t^i$
egomotion estimated by SFM used as relative pose GT
$R_{t'}^i\ T_{t'}^i$
$\hat{R}_{t'}^i\ \hat{T}_{t'}^i\ \sigma^2_{\hat{T}_{t'}^i}$
$\hat{D}'_t$
$\mathcal{L}_D$
ℓ_{pcl}
ℓ_δ
$\mathcal{L}_R$
$\mathcal{L}_T$
Losses
completed point cloud as estimated by the network
$\hat{C}_\tau\ \hat{\delta}$
$\hat{P}_f^G$
globally aligned partial point cloud
Φ_{PCL}
Point cloud completion network

FIGURE 11. Siamese factorization network [103] 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 [103]).

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.* [102] 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) [112] 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.* [103] 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, [113] 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.

C. SEMANTIC SEGMENTATION

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.

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, viewpoint and scale changes.

3) METHODS OVERVIEW

Traditionally, CRFs have been the default choice in the context of semantic segmentation [114]–[117]. This is due to the reason that CRFs provide a flexible framework to model contextual information. As an example, [116] 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 [74] and then transfered 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 [114], [115], [117]. More recently, CNNs have been used to extract rich local features for 2.5D/3D image segmentation tasks [28], [118]–[120]. 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 [46], [60], [121], [122].

The work by Couprie *et al.* [118] is among the pioneering efforts to use depth information along with RGB images 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 workflow 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 upsampling 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 [123] to compute temporal consistent super-pixels. This technique was able to compute super-pixels in quasi-linear time, there by making it possible to use for real-time video segmentation.

Girshick *et al.* [124] presented a region CNN (R-CNN) method for detection and segmentation of RGB images which was later extended to RGB-D images [28]. The R-CNN [124] 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.* [28] then extended the R-CNN method to RGB-D case by using a novel embedding for depth images. They purposed 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 [28] first uses multiscale combinatorial grouping (MCG) [112] to obtain region proposals from RGB-D images, followed by feature extraction using a CNN [125] pre-trained on Imagenet [45] 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 [81] on the output of the object detectors for semantic scene segmentation.

Long *et al.* [121] 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. Reference [121] first redefined the pre-trained classification networks (AlexNet [125], VGG net [27], GoogLeNet [126]) 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, [121] 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 [27], [125], [126] was removed and the fully connected layers were replaced with 1x1 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, [121] trained two networks: one for RGB images and a second for depth images represented by three dimensional HHA depth encoding introduced in [28]. The predictions from both nets are then summed at the final layer. After the successful application of FCNs [121] 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 [127], contour detection [128] and depth regression [129]. In a follow up paper Shelhamer *et al.* [130] revisited the FCNs for semantic segmentation to further analyze, tune and improve results.

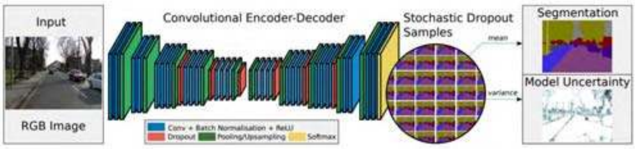

Input
RGB Image
Convolutional Encoder-Decoder
Conv + Batch Normalisation + ReLU
Dropout
Pooling/Upsampling
Softmax
Stochastic Dropout Samples
mean
variance
Segmentation
Model Uncertainty

FIGURE 12. A schematic of Bayesian encoder-decoder architecture for semantic segmentation with a measure of model uncertainty. (Courtesy of [60]).

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.* [60] 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 [46] to Bayesian convolutional network that can produce probabilistic segmentation [131] (see figure 12). This technique can not only be used to convert many state of the art architecture like FCN [121], Segnet [46] and Dilation Network [132] to output probabilistic semantic segmentation but also improves the segmentation results by 2-3% [60]. The work of Kendall *et al.* [60] is inspired by Gal and Ghahramani [131] and [133] show that dropout [134] can be used to approximate inference in a Bayesian neural network. Reference [133] shows that dropout [134] used at test time impose a Bernoulli distribution over the network's 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. Reference [60] 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

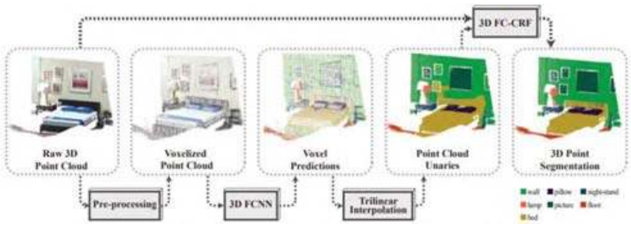


FIGURE 13. **SegCloud [135] framework that takes a 3D point cloud as input, that is voxelized before feeding to 3D CNN. The voxelized representation is projected to the point cloud representation using trilinear interpolation. (Courtesy of [135]).**

distribution from the Monte Carlo sampling [60], [131], [133] is an overall measure of the model's 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.* [49] 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 [96], [97]. 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 [135]. To enable a fully learnable system, the CRF is implemented as a differentiable recurrent network [48]. 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 [28] can help improve model accuracy. Finally, a measure of model uncertainty can be highly useful for practical applications which demand high safety.

D. PHYSICS-BASED REASONING

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 [136], [137], understanding the support relationships and stability of objects [10], [138], [139], volumetric and occlusion reasoning [73], [78], [140]. Applications of such algorithms include task and motion planning for robots, surveillance and monitoring.

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.

3) METHODS OVERVIEW

a: DYNAMICS PREDICTION

Mottaghi *et al.* [136] 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 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. [137] 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 [141]. 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.

b: 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

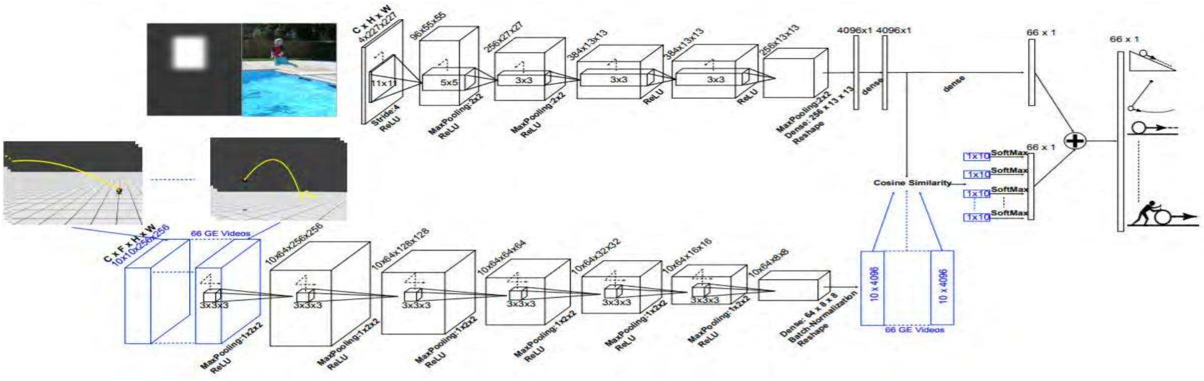


FIGURE 14. Newtonian Neural Network (N^3) [136]. The top stream of the architecture takes RGB image augmented with localization map of the targeted object and the bottom stream processes inputs from a game engine. Features from both streams are combined using cosine similarity and maximum response is used to find the scenario that best describes object motion in an image. (Courtesy of [136]).

are important for robotic manipulation and interaction in man-made environments. Silberman *et al.* [10] 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 [142], [143].

c: 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 objects and scene parses. Note that the support relationships predicted in [10] does not ensure the physical stability of objects. Zheng et al. [138] reasoned about the stability of 3D volumetric shapes, which were recovered from a 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 [139]. 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 [138] performed physical reasoning utilizing mainly depth information, Jia et al. [140] incorporate both color and depth data for such analysis. Similar to [138] and [140] 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 [144] 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.

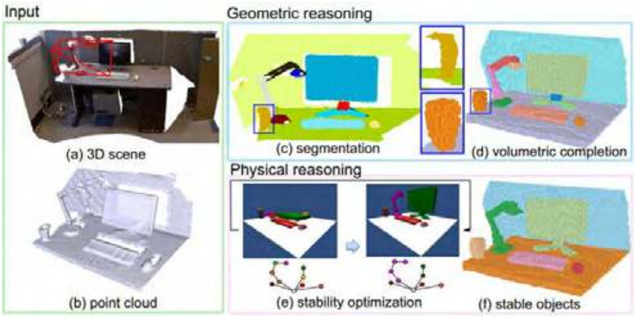


FIGURE 15. A 3D scene converted into point cloud before feeding to geometric and physical reasoning module [138]. (Courtesy of [138]).

d: 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.* [145] 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.* [146] have proposed to use CNN to perform automatic risk assessment in scenes.

e: OCCLUSION REASONING

Occlusion relationship is another important physical and contextual cue, that commonly appears in cluttered scenes. Wang *et al.* [73] 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) [147] and Textons [148]. 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.* [78] 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.

E. OBJECT POSE ESTIMATION

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 [149] or in a joint framework [76], [111], [150]. Direct feature matching techniques (e.g., between images and models) have also been explored for pose estimation [75], [151].

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.

3) METHODS OVERVIEW

Lim *et al.* [151] 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.* [75] 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 [152] and trained on only positive data samples of 3D synthetic model renderings. They used LINEMOD [153], 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.

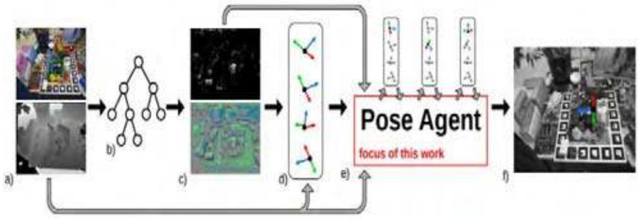


FIGURE 16. Pose estimation pipeline as proposed by [154]. RGB-D input is first processed by random forest. These predictions are used to find pose candidates then a reinforcement agent refines these candidates to find the best pose. (Courtesy of [154]).

Mottaghi *et al.* [76] 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 hierarchal 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 hierarchal 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 [76], Brannchman et al. [150] proposed a method to jointly estimate object class and its 6D pose (3D rotation, 3D translation) in a 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 [150] is to generate number of interpretations or sample pose hypothesis and then find the one that best describe an object pose. Reference [150] achieves this by minimizing an energy function using RANSAC. Reference [156] built upon the idea in [150], but used a CNN trained with a probabilistic approach to find

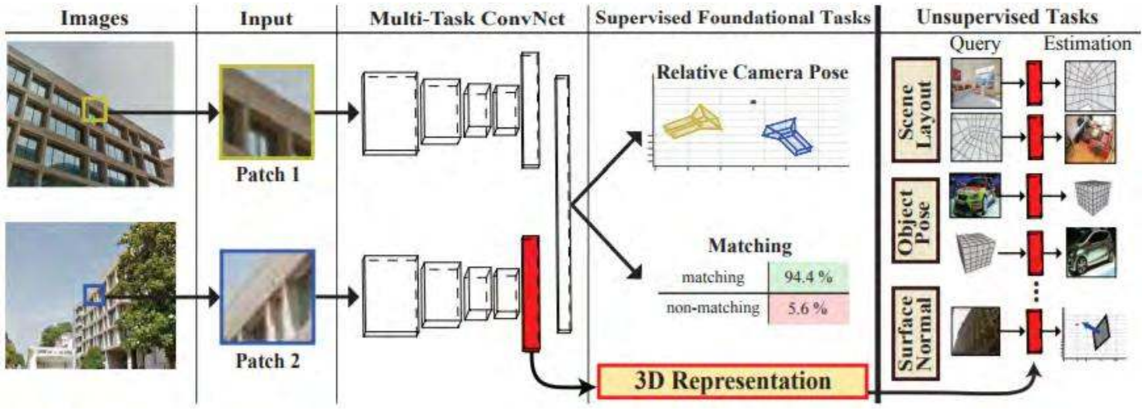


FIGURE 17. CNN trained to learn generic features by matching image patches with their corresponding camera angle. These generic feature representation can be used for multiple tasks at test time including object pose estimation. (Courtesy of [155]).

the best pose hypothesis. [156] 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.* [154] improved upon the work presented in [156] 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.* [149] used AlexNet [125], 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 form 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 and Lepetit [157] 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 pose variation issue in an end-to-end fashion, [158] 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.* [155] 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 [59], where instead of full images, a network was trained on image patches. Kehl *et al.* [59] 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.

F. 3D RECONSTRUCTION FROM RGB-D

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

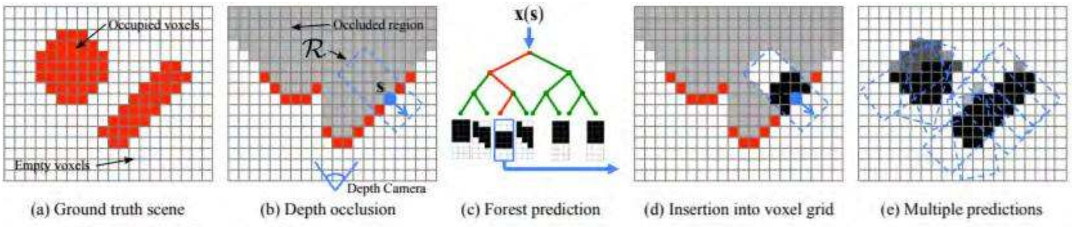


FIGURE 18. A Random forest based framework to predict 3D geometry [77]. (Courtesy of [77]).

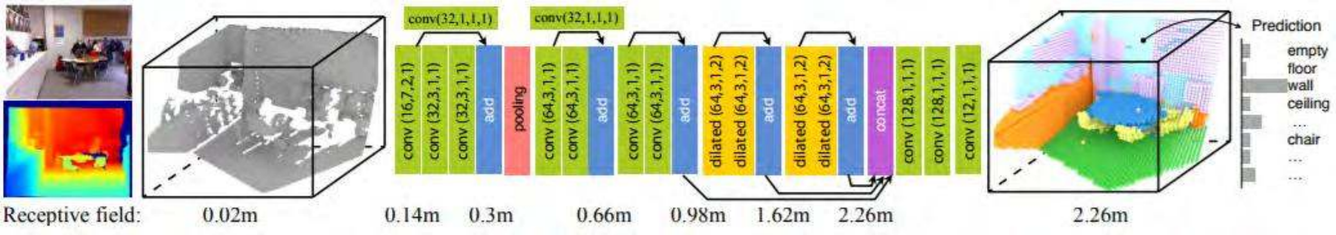


FIGURE 19. SSCNet architecture [9] trained to reconstruct complete 3D scene from a single depth image. (Courtesy of [9]).

densely overlapping RGB-D views of an object [159], [160] 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. .

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.

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 [67], [138]. Specifically, Kim et al. [67] 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.* [77] proposed a structured prediction framework developed using a Random Forest to predict the 3D geometry given the observed incomplete shapes. Unlike [67], 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 [161] and dilated convolutions [132] 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 [57] 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 [63]. 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.* [162] 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 [163], [164].

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 [165], which allow scaling 3D CNNs to work on considerably high dimensional volumetric inputs compared to the regular voxel based models [166]. 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 [61]. In contrast to the regular octree based methods [61], 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 [165], where the short-cut connections exist between the corresponding coarse-to-fine layers in the encoder and the decoder modules.

G. SALIENCY PREDICTION

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 [167], [168], stereopsis [169], [170], light-field imaging [171] and point-clouds [172]. 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.

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.
- An key requisite is to adequately encode the local and global context.

3) METHODS OVERVIEW

Lang *et al.* [167] 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.* [168] 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.* [174] calculated local 3D shape and layout features (e.g., plane and normal cues) using the depth information to improve object saliency. Feng *et al.* [175] 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 fore-ground salient objects.

More recently, [176] 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.* [173] extended a RGB based saliency detection network (ELD-Net [177]) 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 [175]. 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.* [169] 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.* [170] 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.* [171] 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

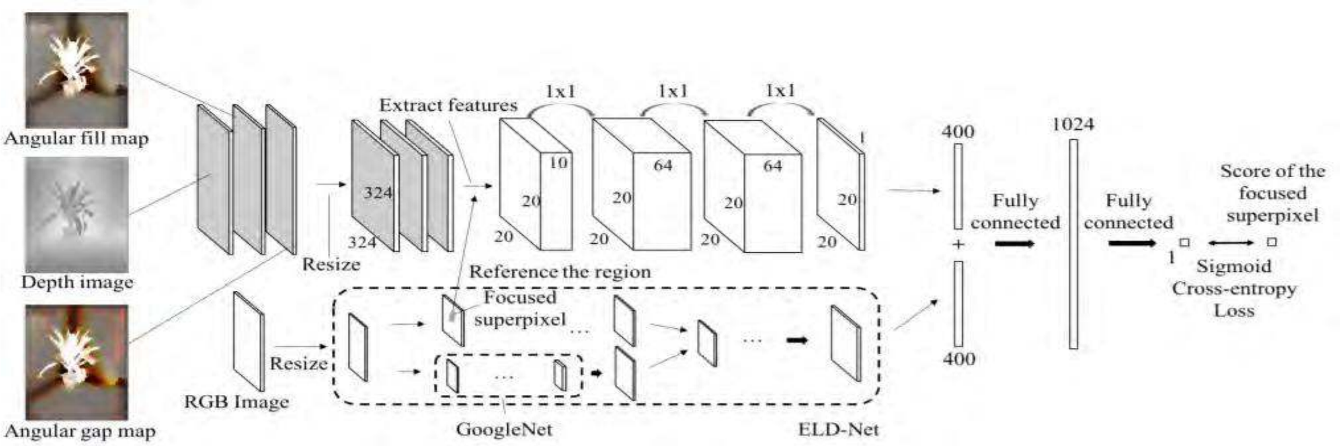


FIGURE 20. Network architecture for saliency prediction [173]. RGB saliency features are fused with super-pixel based handcrafted features to get the overall saliency score. (Courtesy of [173]).

depth to estimate foreground and background regions, which were subsequently improved using contrast and objectness measures.

In contrast to the above techniques which mainly augment depth information for saliency prediction, [172] 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.

H. AFFORDANCE PREDICTION

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 [178] 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 [179]. 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.

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.

3) METHODS OVERVIEW

A seminal work on affordance reasoning by Grabner *et al.* [180] 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.* [68], [181] 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 [69] 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. [182] 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.* [71] 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 [71], used hand-crafted features for affordance prediction.

More recently, automatic feature learning mechanisms such as CNNs have been used for object affordance

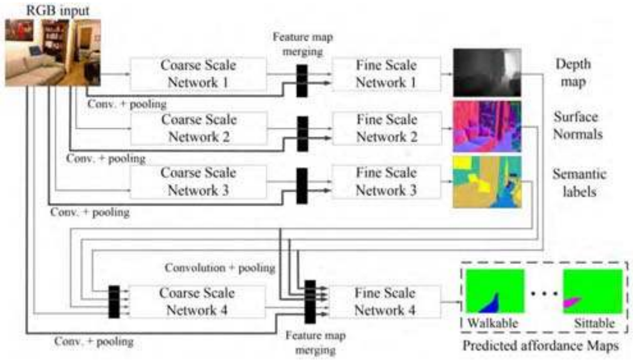


FIGURE 21. A multi-scale CNN proposed in [184]. The coarse scale network extracts global representations encoding wide context while the fine-scale network extracts local representations such as object boundaries. Affordance labels are predicted by combining both representations. (Courtesy of [184]).

prediction [62], [183], [184]. Nquyen *et al.* [62] 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 [28] of depth along with the color image. A more generic affordance prediction framework was presented in [184], 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.* [183] 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.

I. HOLISTIC/HYBRID APPROACHES

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 [64], [143], [185]–[187].

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.

3) METHODS OVERVIEW

Li *et al.* [185] 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 [185] 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 [185], a two-layer generic model with a feedback mechanism was presented in [188]. In contrast to the above methods, [64] 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) [189] 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.* [186] proposed a method that can learn these scene interactions and integrate information at multiple levels to estimate scene composition. Their hierarchal 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.* [50] 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

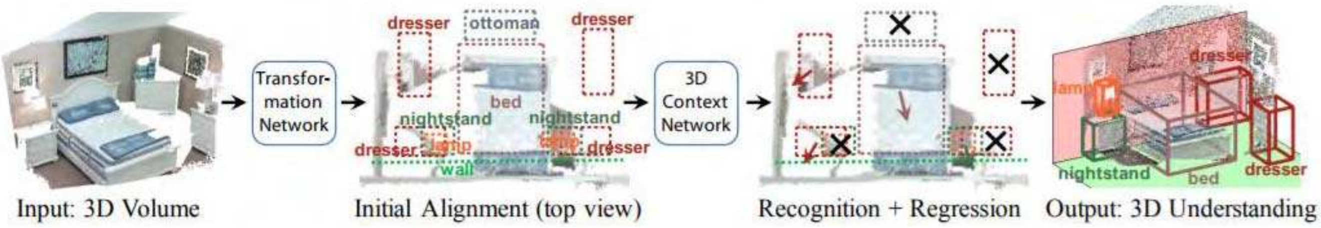


FIGURE 22. 3D scene understanding framework as proposed by [25]. A volumetric representation is first derived from depth image and aligned with the input data, next a 3D CNN estimates the objects presence and adjust them based on holistic scene features for full 3D scene understanding (Courtesy of [25]).

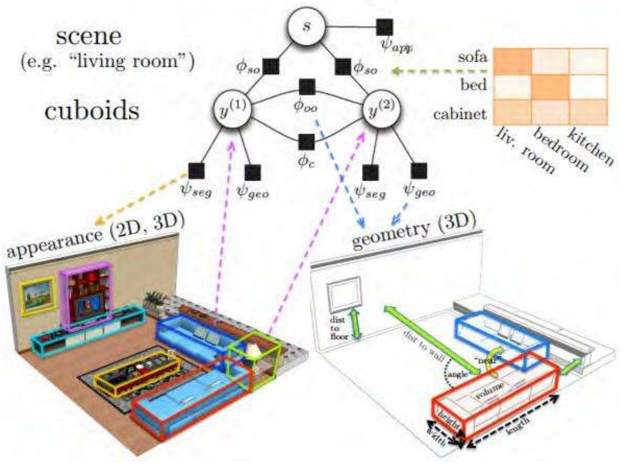


FIGURE 23. Appearance and geometric properties are represented using object cuboids which are then used to define scene to object and object to object relations. This information is integrated by a CRF model for holistic scene understanding. (Courtesy of [64].

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. 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.* [143] 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, [10] jointly models the semantic labels and physical relationships between objects, [9] jointly reconstructs 3D scene and provides voxel labels, [66] concurrently performs segmentation and cuboid detection while [75] 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.

VII. EVALUATION AND DISCUSSION

A. EVALUATION METRICS

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:

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

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.

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).

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, Normalized Scanpath Saliency (NSS) and F-measure (F_β).

- $$\text{Similarity} = SM - FM, \tag{3}$$

where SM is the predicted saliency map and FM is the human eye fixation map or ground truth.

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

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

- $$F_\beta = \frac{(1+\beta^2) * (percision) * (recall)}{\beta^2 * (percision) + (recall)}, \tag{5}$$

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

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 a model's accuracy:

$$\text{Pixel Accuracy} = \frac{\sum_i n_{ii}}{\sum_i t_i}$$

$$\text{Mean Accuracy} = \left(\frac{1}{n_{cl}}\right) \sum_i \frac{n_{ii}}{t_i}$$

$$\text{MIoU} = \left(\frac{1}{n_{cl}}\right) \sum_i \frac{n_{ii}}{\left(t_i + \sum_j n_{ji} - n_{ii}\right)}$$

$$\text{FIoU} = \left(\sum_k t_k\right)^{-1} \sum_i \frac{t_i n_{ii}}{\left(t_i + \sum_j n_{ji} - n_{ii}\right)}, \tag{6}$$

where, MIoU stands for Mean Intersection over Union, FIoU denotes 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 and $t_i = \sum_j n_{ji}$ is the total number of pixels belong to class i.

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:

$$\text{Accuracy} = \frac{\text{Number of affordances correctly classified}}{\text{Total number of affordances}}. \tag{7}$$

7) 3D RECONSTRUCTION

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

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

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

TABLE 3. Performance comparison between prominent 3D object classification methods on ModelNet40 dataset [89].

Method	Key Point	Accuracy
Wu et al. [89]	Feature learning from 3D voxelized input	77.3
Wu et al. [82]	Unsupervised feature learning using 3D generative adversarial modeling	83.3
Qi et al. [49]	Point-cloud based representation learning	86.2
Su et al. [52]	Multi-view 2D images for feature learning	90.1
Qi et al. [83]	Volumetric and multi-view feature learning	91.4
Brock et al. [190]	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 RGB-D [30]	42.1
Lahoud and Ghanem [191]	Using 2D detection algorithms for 3D object detection		45.1
Ren et al. [109]	Based on oriented gradient descriptor		47.6
Qi et al. [192]	Processing raw point cloud using CNN		54.0
Song et al. [24]	3D CNN for amodal object detection	NYUv2 [10]	36. 3
Deng et al. [102]	Two-stream CNN with 2D object proposals for 3D amodal detection.		40.9

B. 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. [89] 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 [190] outperformed other competitors on this dataset. Object detection results are shown in Table 4. 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 further processing. However, an interesting result from Table 4 is that processing point clouds directly using CNNs 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 [60] outperforms for RGBD semantic segmentation task. Saliency prediction algorithms are compared against LSFD [171] in Table 7 where the most promising results are delivered when hand-crafted features are combined with CNN feature representation [176]. 3D reconstruction algorithms are compared in Table 6 where again best performing method is based on CNN that

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. [10]	Hand-crafted features (SIFT)	4	58.6	-	-
Couprie et al. [118]	CNN as feature extractor used with super-pixels	4	64.5	63.5	-
Couprie et al. [118]	CNN as feature extractor used with super-pixels	13	52.4	36.2	-
Hermans et al. [116]	2D to 3D label transfer and use of 3D CRF	13	54.2	48.0	-
Wolf et al. [193]	3D decision forests	13	64.9	55.6	39.5
Tchapmi et al. [135]	CNN with fully connected CRF	13	66.8	56.4	43.5
Gupta et al. [28]	CNN as feature extractor with novel depth embedding	40	60.3	-	28.6
Long et al. [121]	Encoder decoder architecture with FCN	40	65.4	46.1	34.0
Badrinarayanan et al. [46]	Encoder decoder architecture with reduced parameters	40	66.1	36.0	23.6
Kendall et al. [60]	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 [77].

Method	Key Point	Train set	Precision	Recall	IoU
Zheng et al. [138]	Based on geometric and physical reasoning	Rendered NYU [77]	60.1	46.7	34.6
Firman et al. [77]	Estimating occluded voxels in a scene by comparison with a similar scene		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 [77] + SUNCG [34]	75.0	96.0	73.0

effectively incorporates contextual information using dilated convolutions. As a general trend, we note that context plays a key role in individual scene understanding tasks as well as holistic scene understanding. Several approaches to incorporate scene context have been proposed e.g., skip connections in encoder-decoder frameworks, dilated convolutions, combination of global and local features. Still, the encoding of useful scene context is an open research problem.

VIII. CHALLENGES AND FUTURE DIRECTIONS

A. LIGHT-WEIGHT MODELS

In the last few years, we have seen a dramatic growth in the capabilities of computational resources for machine vision applications [200]. 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 parallelizeable algorithms and compact representations for 3D data.

B. TRANSFER LEARNING

Scene understanding involves prediction about several interrelated 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 transfered to the new domain.

TABLE 7. Performance of state-of-art saliency detection methods on LSFD [171] dataset.

Method	Key Point	F-measure
He et al. [194]	Super-pixels with CNN	0.698
Zhang et al. [195]	Using min. barrier distance transform	0.730
Qin et al. [196]	Dynamic evolution modeling	0.731
Wang et al. [197]	Based on local and global features	0.738
Peng et al. [168]	Fusion of depth modality with RGB	0.704
Ju et al. [198]	Based on anisotropic center-surround diff.	0.757
Ren et al. [199]	Based on depth and normal priors	0.788
Feng et al. [175]	Local background enclosure based feature	0.712
Qu et al. [176]	Fusing engineered features with CNN	0.844

C. 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.

D. 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.

E. MULTI-TASK LEARNING

Given the multi-task nature of complete scene understanding, a suitable but less investigated 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.

F. 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 [37], [201].

G. 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 [202]), 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 [31].

H. 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 Frosst *et al.* [203] 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 [204]. 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 3354–3361.

[2] T. Breuer *et al.*, "Johnny: An autonomous service robot for domestic environments," *J. Intell. Robot. Syst.*, vol. 66, nos. 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," *Proc. SPIE*, vol. 5029, pp. 512–520, May 2003.

[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 *Proc. IEEE Intell. Vehicles Symp. (IV)*, Oct. 2000, pp. 382–386.

[6] *Qualcomm Announces 3D Camera Technology for Android Ecosystem*. Accessed: Dec. 10, 2017. [Online]. Available: https://goo.gl/JkApyZ

[7] *Who: Vision Impairment and Blindness*. Accessed: Dec. 8, 2017. [Online]. Available: http://www.who.int/mediacentre/factsheets/fs282/en/

[8] A. Rodríguez, L. M. Bergasa, P. F. Alcantarilla, J. Yebes, and A. Cela, "Obstacle avoidance system for assisting visually impaired people," in *Proc. IEEE Intell. Vehicles Symp. Workshops*, Madrid, Spain, vol. 35, Jun. 2012, p. 16.

[9] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser. (2016). "Semantic scene completion from a single depth image." [Online]. Available: https://arxiv.org/abs/1611.08974

[10] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Computer Vision—ECCV*. Berlin, Germany: Springer, 2012, pp. 746–760.

[11] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2, Sep. 1999, pp. 1150–1157.

[12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, vol. 1, no. 1, pp. 886–893.

[13] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Computer Vision—ECCV*. Berlin, Germany: Springer, 2006, pp. 404–417.

[14] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2006, pp. 589–600.

[15] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.

[16] M. Charlton, "Helmholtz on perception: Its physiology and development," *Arch. Neurol.*, vol. 19, no. 3, p. 349, 1968.

[17] K. Koffka, *Principles of Gestalt Psychology*, vol. 44. Evanston, IL, USA: Routledge, 2013.

[18] H. Barrow and J. Tenenbaum, *Computer Vision Systems*, vol. 2. New York, NY, USA: Academic, 1978.

[19] D. Marr, *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. San Francisco, CA, USA: Freeman, 1982.

[20] L. G. Roberts, "Machine perception of three-dimensional solids," Ph.D. dissertation, Dept. Elect. Eng., Massachusetts Inst. Technol., Cambridge, MA, USA, 1963.

[21] A. Guzmán, "Decomposition of a visual scene into three-dimensional bodies," in *Proc. Fall Joint Comput. Conf.*, 1968, pp. 291–304.

[22] T. O. Binford, "Visual perception by computer," in *Proc. IEEE Conf. Syst. Control*, 1971.

[23] I. Biederman, "Human image understanding: Recent research and a theory," *Comput. Vis., Graph., Image Process.*, vol. 32, no. 1, pp. 29–73, 1985.

[24] S. Song and J. Xiao, "Deep sliding shapes for amodal 3D object detection in RGB-D images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 808–816.

[25] Y. Zhang, M. Bai, P. Kohli, S. Izadi, and J. Xiao. (2016). "DeepContext: Context-encoding neural pathways for 3D holistic scene understanding." [Online]. Available: https://arxiv.org/abs/1603.04922

[26] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1440–1448.

[27] K. Simonyan and A. Zisserman. (2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556

[28] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning rich features from RGB-D images for object detection and segmentation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 345–360.

[29] J. Xiao, A. Owens, and A. Torralba, "SUN3D: A database of big spaces reconstructed using SfM and object labels," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1625–1632.

[30] S. Song, S. P. Lichtenberg, and J. Xiao, "SUN RGB-D: A RGB-D scene understanding benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 567–576.

[31] I. Armeni, A. Sax, A. R. Zamir, and S. Savarese. (Feb. 2017). "Joint 2D-3D-semantic data for indoor scene understanding." [Online]. Available: https://arxiv.org/abs/1702.01105

[32] A. Chang *et al.* (2017). "Matterport3D: Learning from RGB-D data in indoor environments." [Online]. Available: https://arxiv.org/abs/1709.06158

[33] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "ScanNet: Richly-annotated 3D reconstructions of indoor scenes," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1–10.

[34] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, "Semantic scene completion from a single depth image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 190–198.

[35] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view RGB-D object dataset," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2011, pp. 1817–1824.

[36] 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 *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 92–101.

[37] J. McCormac, A. Handa, S. Leutenegger, and A. J. Davison. (2016). "SceneNet RGB-D: 5 m photorealistic images of synthetic indoor trajectories with ground truth." [Online]. Available: https://arxiv.org/abs/1612.05079

[38] M. Savva, A. X. Chang, P. Hanrahan, M. Fisher, and M. Nießner, "PiGraphs: Learning interaction snapshots from observations," *ACM Trans. Graph.*, vol. 35, no. 4, p. 139, 2016.

[39] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2012, pp. 573–580.

[40] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond PASCAL: A benchmark for 3D object detection in the wild," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2014, pp. 75–82.

[41] H. Badino, U. Franke, and D. Pfeiffer, "The Stixel world—A compact medium level representation of the 3d-world," in *Proc. DAGM-Symp.* Berlin, Germany: Springer, 2009, pp. 51–60.

[42] N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor," in *Proc. Int. Conf. Comput. Vis. Workshop 3D Represent. Recognit.*, Nov. 2011, pp. 601–608.

[43] *Planner 5D*. Accessed: Jan. 1, 2018. [Online]. Available: https://planner5d.com/

[44] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Sep. 2009.

[45] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 248–255.

[46] V. Badrinarayanan, A. Kendall, and R. Cipolla. (2015). "SegNet: A deep convolutional encoder-decoder architecture for image segmentation." [Online]. Available: https://arxiv.org/abs/1511.00561

[47] R. Socher, B. Huval, B. Bath, C. D. Manning, and A. Y. Ng, "Convolutional-recursive deep learning for 3D object classification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 656–664.

[48] S. Zheng *et al.*, "Conditional random fields as recurrent neural networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1529–1537.

[49] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. (2016). "PointNet: Deep learning on point sets for 3D classification and segmentation." [Online]. Available: https://arxiv.org/abs/1612.00593

[50] Y. Zhang, S. Song, P. Tan, and J. Xiao, "PanoContext: A whole-room 3D context model for panoramic scene understanding," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 668–686.

[51] Y. Li, S. Pirk, H. Su, C. R. Qi, and L. J. Guibas, "FPNN: Field probing neural networks for 3D data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 307–315.

[52] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3D shape recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 945–953.

[53] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[54] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio. (2014). "On the properties of neural machine translation: Encoder-decoder approaches." [Online]. Available: https://arxiv.org/abs/1409.1259

[55] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Netw.*, vol. 18, no. 5, pp. 602–610, 2005.

[56] A. Graves, G. Wayne, and I. Danihelka. (2014). "Neural turing machines." [Online]. Available: https://arxiv.org/abs/1410.5401

[57] 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 *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 628–644.

[58] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2048–2057.

[59] 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 *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 205–220.

[60] A. Kendall, V. Badrinarayanan, and R. Cipolla. (2015). "Bayesian SegNet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding." [Online]. Available: https://arxiv.org/abs/1511.02680

[61] G. Riegler, A. O. Ulusoy, and A. Geiger, "OctNet: Learning deep 3D representations at high resolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 6620–6629.

[62] A. Nguyen, D. Kanoulas, D. G. Caldwell, and N. G. Tsagarakis, "Detecting object affordances with convolutional neural networks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 2765–2770.

[63] A. Dai, C. R. Qi, and M. Nießner. (2016). "Shape completion using 3D-encoder-predictor CNNs and shape synthesis." [Online]. Available: https://arxiv.org/abs/1612.00101

[64] D. Lin, S. Fidler, and R. Urtasun, "Holistic scene understanding for 3D object detection with RGBD cameras," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1417–1424.

[65] S. Wang, S. Fidler, and R. Urtasun, "Holistic 3D scene understanding from a single geo-tagged image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3964–3972.

[66] S. H. Khan, X. He, M. Bennamoun, F. Sohel, and R. Togneri, "Separating objects and clutter in indoor scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4603–4611.

[67] B.-S. Kim, P. Kohli, and S. Savarese, "3D scene understanding by voxel-CRF," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1425–1432.

[68] Y. Jiang, H. S. Koppula, and A. Saxena, "Modeling 3D environments through hidden human context," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2040–2053, Oct. 2016.

[69] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 14–29, Jan. 2016.

[70] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 801–808.

[71] A. Myers, C. L. Teo, C. Fermüller, and Y. Aloimonos, "Affordance detection of tool parts from geometric features," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 1374–1381.

[72] L. Bo, X. Ren, and D. Fox, "Learning hierarchical sparse features for RGB-(D) object recognition," *Int. J. Robot. Res.*, vol. 33, no. 4, pp. 581–599, 2014.

[73] T. Wang, X. He, and N. Barnes, "Learning structured hough voting for joint object detection and occlusion reasoning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1790–1797.

[74] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[75] A. Tejani, D. Tang, R. Kouskouridas, and T.-K. Kim, "Latent-class Hough forests for 3D object detection and pose estimation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 462–477.

[76] R. Mottaghi, Y. Xiang, and S. Savarese, "A coarse-to-fine model for 3D pose estimation and sub-category recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 418–426.

[77] M. Firman, O. M. Aodha, S. Julier, and G. J. Brostow, "Structured prediction of unobserved voxels from a single depth image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 5431–5440.

[78] U. Bonde, V. Badrinarayanan, and R. Cipolla, "Robust instance recognition in presence of occlusion and clutter," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 520–535.

[79] M. A. Hearst, S. T. Dumais, E. Osman, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intell. Syst. Appl.*, vol. 13, no. 4, pp. 18–28, Jul./Aug. 2008.

[80] S. R. Gunn *et al.*, "Support vector machines for classification and regression," ISIS Lab., Univ. Notre Dame, Notre Dame, IN, USA, ISIS Tech. Rep. 14, 1998, pp. 85–86, vol. 14, no. 1.

[81] S. Gupta, P. Arbelaez, and J. Malik, "Perceptual organization and recognition of indoor scenes from RGB-D images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 564–571.

[82] 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 *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 82–90.

[83] 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 5648–5656.

[84] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.

[85] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, Jun. 2006, pp. 2169–2178.

[86] A. Coates, A. Y. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 215–223.

[87] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard, "Multimodal deep learning for robust RGB-D object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 681–687.

[88] A. Wang, J. Cai, J. Lu, and T.-J. Cham, "Modality and component aware feature fusion for RGB-D scene classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 5995–6004.

[89] Z. Wu *et al.*, "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1912–1920.

[90] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[91] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 922–928.

[92] B. T. Phong, "Illumination for computer generated pictures," *Commun. ACM*, vol. 18, no. 6, pp. 311–317, Jun. 1975.

[93] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. (2014). "Return of the devil in the details: Delving deep into convolutional nets." [Online]. Available: https://arxiv.org/abs/1405.3531

[94] B. Shi, S. Bai, Z. Zhou, and X. Bai, "DeepPano: Deep panoramic representation for 3-D shape recognition," *IEEE Signal Process. Lett.*, vol. 22, no. 12, pp. 2339–2343, Dec. 2015.

[95] V. Hegde and R. Zadeh. (2016). "FusionNet: 3D object classification using multiple data representations." [Online]. Available: https://arxiv.org/abs/1607.05695

[96] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. (2017). "Pointnet++: Deep hierarchical feature learning on point sets in a metric space." [Online]. Available: https://arxiv.org/abs/1706.02413

[97] W. Zeng and T. Gevers. (2017). "3DContextNet: K-d tree guided hierarchical learning of point clouds using local and global contextual cues." [Online]. Available: https://arxiv.org/abs/1711.11379

[98] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[99] H. Jiang, "Finding approximate convex shapes in RGBD images," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 582–596.

[100] S. Song and J. Xiao, "Sliding shapes for 3D object detection in depth images," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 634–651.

[101] X. Chen *et al.*, "3D object proposals for accurate object class detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 424–432.

[102] Z. Deng and L. J. Latecki, "Amodal detection of 3D objects: Inferring 3D bounding boxes from 2D ones in RGB-depth images," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1–2.

[103] D. Novotny, D. Larlus, and A. Vedaldi. (2017). "Learning 3D object categories by looking around them." [Online]. Available: https://arxiv.org/abs/1705.03951

[104] H. Jiang and J. Xiao, "A linear approach to matching cuboids in RGBD images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2171–2178.

[105] M. Z. Zia, M. Stark, and K. Schindler, "Towards scene understanding with detailed 3D object representations," *Int. J. Comput. Vis.*, vol. 112, no. 2, pp. 188–203, 2015.

[106] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[107] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. Conf. Rec. 27th Asilomar Conf. Signals, Syst. Comput.*, Nov. 1993, pp. 40–44.

[108] T. Malisiewicz, A. Gupta, and A. A. Efros, "Ensemble of exemplar-SVMs for object detection and beyond," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 89–96.

[109] Z. Ren and E. B. Sudderth, "Three-dimensional object detection and layout prediction using clouds of oriented gradients," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1525–1533.

[110] A. Karpathy, S. Miller, and L. Fei-Fei, "Object discovery in 3D scenes via shape analysis," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2013, pp. 2088–2095.

[111] N. Sedaghat, M. Zolfaghari, and T. Brox. (2016). "Orientation-boosted voxel nets for 3D object recognition." [Online]. Available: https://arxiv.org/abs/1604.03351

[112] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 328–335.

[113] K. Xu *et al.*, "3D attention-driven depth acquisition for object identification," *ACM Trans. Graph.*, vol. 35, no. 6, p. 238, 2016.

[114] C. Cadena and J. Košecka, "Semantic parsing for priming object detection in RGB-D scenes," in *Proc. 3rd Workshop Semantic Perception, Mapping Exploration*, 2013, pp. 1–7.

[115] A. C. Müller and S. Behnke, "Learning depth-sensitive conditional random fields for semantic segmentation of RGB-D images," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 6232–6237.

[116] A. Hermans, G. Floros, and B. Leibe, "Dense 3D semantic mapping of indoor scenes from RGB-D images," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 2631–2638.

[117] Z. Deng, S. Todorovic, and L. J. Latecki, "Semantic segmentation of RGBD images with mutex constraints," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1733–1741.

[118] C. Couprie, C. Farabet, L. Najman, and Y. LeCun, "Toward real-time indoor semantic segmentation using depth information," *J. Mach. Learn. Res.*, vol. 1, pp. 1–48, 2014.

[119] E. Kalogerakis, M. Averkiou, S. Maji, and S. Chaudhuri. (2016). "3D shape segmentation with projective convolutional networks." [Online]. Available: https://arxiv.org/abs/1612.02808

[120] L. Schneider *et al.*, "Multimodal neural networks: RGB-D for semantic segmentation and object detection," in *Proc. Scand. Conf. Image Anal.* Cham, Switzerland: Springer, 2017, pp. 98–109.

[121] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3431–3440.

[122] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers, "FuseNet: Incorporating depth into semantic segmentation via fusion-based CNN architecture," in *Proc. Asian Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 213–228.

[123] C. Couprie, C. Farabet, Y. LeCun, and L. Najman, "Causal graph-based video segmentation," in *Proc. 20th IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2013, pp. 4249–4253.

[124] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[125] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[126] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.

[127] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2015, pp. 91–99.

[128] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1395–1403.

[129] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning depth from single monocular images using deep convolutional neural fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2024–2039, Oct. 2016.

[130] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[131] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1050–1059.

[132] F. Yu and V. Koltun. (2015). "Multi-scale context aggregation by dilated convolutions." [Online]. Available: https://arxiv.org/abs/1511.07122

[133] Y. Gal and Z. Ghahramani. (2015). "Bayesian convolutional neural networks with Bernoulli approximate variational inference." [Online]. Available: https://arxiv.org/abs/1506.02158

[134] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[135] L. P. Tchapmi, C. B. Choy, I. Armeni, J. Gwak, and S. Savarese. (2017). "SEGCloud: Semantic segmentation of 3D point clouds." [Online]. Available: https://arxiv.org/abs/1710.07563

[136] R. Mottaghi, H. Bagherinezhad, M. Rastegari, and A. Farhadi, "Newtonian scene understanding: Unfolding the dynamics of objects in static images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 3521–3529.

[137] 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 *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 127–135.

[138] B. Zheng, Y. Zhao, J. C. Yu, K. Ikeuchi, and S.-C. Zhu, "Beyond point clouds: Scene understanding by reasoning geometry and physics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3127–3134.

[139] A. Barbu and S.-C. Zhu, "Generalizing Swendsen-Wang to sampling arbitrary posterior probabilities," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1239–1253, Aug. 2005.

[140] Z. Jia, A. C. Gallagher, A. Saxena, and T. Chen, "3D reasoning from blocks to stability," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 5, pp. 905–918, May 2015.

[141] P. W. Battaglia, J. B. Hamrick, and J. B. Tenenbaum, "Simulation as an engine of physical scene understanding," *Proc. Nat. Acad. Sci. USA*, vol. 110, no. 45, pp. 18327–18332, 2013.

[142] Z. Jia, A. Gallagher, A. Saxena, and T. Chen, "3D-based reasoning with blocks, support, and stability," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1–8.

[143] W. Zhuo, M. Salzmann, X. He, and M. Liu, "Indoor scene parsing with instance segmentation, semantic labeling and support relationship inference," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, p. 1.

[144] J. Chang and J. W. Fisher, "Efficient MCMC sampling with implicit shape representations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 2081–2088.

[145] 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 *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 3417–3424.

[146] R. Dupre, G. Tzimiropoulos, and V. Argyriou, "Automated risk assessment for scene understanding and domestic robots using RGB-D data and 2.5 D CNNs at a patch level," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jul. 2017, pp. 5–6.

[147] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.

[148] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "*TextonBoost*: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2006, pp. 1–15.

[149] M. Schwarz, H. Schulz, and S. Behnke, "RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 1329–1335.

[150] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother, "Learning 6D object pose estimation using 3D object coordinates," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 536–551.

[151] J. J. Lim, H. Pirsiavash, and A. Torralba, "Parsing IKEA objects: Fine pose estimation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2992–2999.

[152] J. Gall, A. Yao, N. Razavi, L. Van Gool, and V. Lempitsky, "Hough forests for object detection, tracking, and action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2188–2202, Nov. 2011.

[153] S. Hinterstoisser *et al.*, "Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 858–865.

[154] A. Krull, E. Brachmann, S. Nowozin, F. Michel, J. Shotton, and C. Rother. (2016). "Poseagent: Budget-constrained 6D object pose estimation via reinforcement learning." [Online]. Available: https://arxiv.org/abs/1612.03779

[155] A. R. Zamir, T. Wekel, P. Agrawal, C. Wei, J. Malik, and S. Savarese, "Generic 3D representation via pose estimation and matching," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 535–553.

[156] A. Krull, E. Brachmann, F. Michel, M. Y. Yang, S. Gumhold, and C. Rother, "Learning analysis-by-synthesis for 6D pose estimation in RGB-D images," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 954–962.

[157] P. Wohlhart and V. Lepetit, "Learning descriptors for object recognition and 3D pose estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3109–3118.

[158] A. Doumanoglou, V. Balntas, R. Kouskouridas, and T.-K. Kim. (2016). "Siamese regression networks with efficient mid-level feature extraction for 3D object pose estimation." [Online]. Available: https://arxiv.org/abs/1607.02257

[159] T. Whelan, M. Kaess, M. Fallon, H. Johannsson, J. Leonard, and J. McDonald, "Kintinuous: Spatially extended KinectFusion," in *RSS Workshop on RGB-D: Advanced Reasoning With Depth Cameras*, 2012.

[160] R. A. Newcombe *et al.*, "KinectFusion: Real-time dense surface mapping and tracking," in *Proc. 10th IEEE Int. Symp. Mixed Augmented Reality (ISMAR)*, Oct. 2011, pp. 127–136.

[161] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[162] S. Gupta, P. Arbeláez, R. Girshick, and J. Malik, "Aligning 3D models to RGB-D images of cluttered scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4731–4740.

[163] H. Hoppe, "Progressive meshes," in *Proc. 23rd Annu. Conf. Comput. Graph. Interact. Techn.*, 1996, pp. 99–108.

[164] S. Choi, Q.-Y. Zhou, and V. Koltun, "Robust reconstruction of indoor scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5556–5565.

[165] G. Riegler, A. O. Ulusoy, H. Bischof, and A. Geiger. (2017). "OctNet-Fusion: Learning depth fusion from data." [Online]. Available: https://arxiv.org/abs/1704.01047

[166] J. Wu *et al.*, "Single image 3D interpreter network," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 365–382.

[167] 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*. Cham, Switzerland: Springer, 2012, pp. 101–115.

[168] H. Peng, B. Li, W. Xiong, W. Hu, and R. Ji, "RGBD salient object detection: A benchmark and algorithms," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 92–109.

[169] Y. Niu, Y. Geng, X. Li, and F. Liu, "Leveraging stereopsis for saliency analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 454–461.

[170] Y. Fang, J. Wang, M. Narwaria, P. Le Callet, and W. Lin, "Saliency detection for stereoscopic images," *IEEE Trans. Image Process.*, vol. 23, no. 6, pp. 2625–2636, Jun. 2014.

[171] N. Li, J. Ye, Y. Ji, H. Ling, and J. Yu, "Saliency detection on light field," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2806–2813.

[172] N. Kobyshev, H. Riemenschneider, A. Bódis-Szomorú, and L. van Gool, "3D saliency for finding landmark buildings," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 267–275.

[173] R. Shigematsu, D. Feng, S. You, and N. Barnes. (2017). "Learning RGB-D salient object detection using background enclosure, depth contrast, and top-down features." [Online]. Available: https://arxiv.org/abs/1705.03607

[174] A. Ciptadi, T. Hermans, and J. M. Rehg, "An in depth view of saliency," in *Proc. Brit. Mach. Vis. Conf. (BMVC)* Bristol, U.K., Sep. 2013.

[175] D. Feng, N. Barnes, S. You, and C. McCarthy, "Local background enclosure for RGB-D salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2343–2350.

[176] L. Qu *et al.*, "RGBD salient object detection via deep fusion," *IEEE Trans. Image Process.*, vol. 26, no. 5, pp. 2274–2285, May 2017.

[177] G. Lee, Y.-W. Tai, and J. Kim, "Deep saliency with encoded low level distance map and high level features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 660–668.

[178] J. J. Gibson, *The Ecological Approach to Visual Perception: Classic Edition*. Psychology Press, 2014.

[179] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 1778–1785.

[180] H. Grabner, J. Gall, and L. Van Gool, "What makes a chair a chair?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 1529–1536.

[181] Y. Jiang, H. Koppula, and A. Saxena, "Hallucinated humans as the hidden context for labeling 3D scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2993–3000.

[182] A. Pieropan, C. H. Ek, and H. Kjellström, "Functional descriptors for object affordances," in *Proc. IROS Workshop*, 2015, pp. 1–2.

[183] 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 *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 4604–4611.

[184] A. Roy and S. Todorovic, "A multi-scale CNN for affordance segmentation in RGB images," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 186–201.

[185] C. Li, A. Kowdle, A. Saxena, and T. Chen, "Towards holistic scene understanding: Feedback enabled cascaded classification models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 1351–1359.

[186] W. Choi, Y.-W. Chao, C. Pantofaru, and S. Savarese, "Understanding indoor scenes using 3D geometric phrases," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 33–40.

[187] S. Gupta, P. Arbeláez, R. Girshick, and J. Malik, "Indoor scene understanding with RGB-D images: Bottom-up segmentation, object detection and semantic segmentation," *Int. J. Comput. Vis.*, vol. 112, no. 2, pp. 133–149, 2015.

[188] C. Li, A. Kowdle, A. Saxena, and T. Chen, "A generic model to compose vision modules for holistic scene understanding," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2010, pp. 70–85.

[189] J. Carreira and C. Sminchisescu, "CPMC: Automatic object segmentation using constrained parametric min-cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1312–1328, Jul. 2012.

[190] A. Brock, T. Lim, J. M. Ritchie, and N. Weston. (2016). "Generative and discriminative voxel modeling with convolutional neural networks." [Online]. Available: https://arxiv.org/abs/1608.04236

[191] J. Lahoud and B. Ghanem, "2D-driven 3D object detection in RGB-D images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Oct. 2017, pp. 4622–4630.

[192] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas. (2017). "Frustum pointnets for 3D object detection from RGB-D data." [Online]. Available: https://arxiv.org/abs/1711.08488

[193] D. Wolf, J. Prankl, and M. Vincze, "Enhancing semantic segmentation for robotics: The power of 3-D entangled forests," *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 49–56, Jan. 2016.

[194] S. He, R. W. H. Lau, W. Liu, Z. Huang, and Q. Yang, "SuperCNN: A superpixelwise convolutional neural network for salient object detection," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 330–344, 2015.

[195] J. Zhang, S. Sclaroff, Z. Lin, X. Shen, B. Price, and R. Mech, "Minimum barrier salient object detection at 80 fps," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1404–1412.

[196] Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency detection via cellular automata," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 110–119.

[197] L. Wang, H. Lu, X. Ruan, and M.-H. Yang, "Deep networks for saliency detection via local estimation and global search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3183–3192.

[198] R. Ju, L. Ge, W. Geng, T. Ren, and G. Wu, "Depth saliency based on anisotropic center-surround difference," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 1115–1119.

[199] J. Ren, X. Gong, L. Yu, W. Zhou, and M. Y. Yang, "Exploiting global priors for RGB-D saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2015, pp. 25–32.

[200] *Accelerating AI With GPUS*. Accessed: Dec. 8, 2017. [Online]. Available: https://blogs.nvidia.com/blog/2016/01/12/accelerating-ai-artificial-intelligence-gpus/

[201] 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Oct. 2017, pp. 2678–2687.

[202] 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*. Ann Arbor, MI, USA: Univ. Michigan, 2016.

[203] N. Frosst and G. Hinton. (2017). "Distilling a neural network into a soft decision tree." [Online]. Available: https://arxiv.org/abs/1711.09784

[204] X. Yuan, P. He, Q. Zhu, R. R. Bhat, and X. Li. (2017). "Adversarial examples: Attacks and defenses for deep learning." [Online]. Available: https://arxiv.org/abs/1712.07107

[205] *Datasets*. [Online]. Available: http://www.michaelfirman.co.uk/RGBDdatasets/

[206] T. Hackel, N. Savinov, L. Ladicky, J. D. Wegner, K. Schindler, and M. Pollefeys. (2017). "Semantic3D. Net: A new large-scale point cloud classification benchmark." [Online]. Available: https://arxiv.org/abs/1704.03847

[207] *Oms2world*. Accessed: Jan. 1, 2018. [Online]. Available: http://osm2world.org

[208] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky, "Depth from Familiar Objects: A Hierarchical Model for 3D Scenes," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, Jun. 2006, pp. 2410–2417.

[209] Y. Zhao and S.-C. Zhu, "Scene parsing by integrating function, geometry and appearance models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3119–3126.

[210] W. Liu, R. Ji, and S. Li, "Towards 3D object detection with bimodal deep Boltzmann machines over RGBD imagery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3013–3021.

[211] K. Yamaguchi, D. McAllester, and R. Urtasun, "Efficient joint segmentation, occlusion labeling, stereo and flow estimation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 756–771.

[212] M. Zhu *et al.*, "Single image 3D object detection and pose estimation for grasping," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 3936–3943.

[213] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[214] *Who: Number of Road Traffic Deaths*. Accessed: Dec. 8, 2017. [Online]. Available: http://www.who.int/gho/road_safety/mortality/traffic_deaths_number/en/

[215] Y. Yakimovsky and J. A. Feldman, "A semantics-based decision theory region analyser," in *Proc. IJCAI*, 1973, pp. 580–588.

[216] Y. Ohta, *Knowledge-Based Interpretation of Outdoor Natural Color Scenes*, vol. 4. San Mateo, CA, USA: Morgan Kaufmann, 1985.

[217] Y.-I. Ohta, T. Kanade, and T. Sakai, "An analysis system for scenes containing objects with substructures," in *Proc. 4th Int. Joint Conf. Pattern Recognit.*, Nov. 1978, pp. 752–754.

[218] R. A. Brooks, "Model-based three-dimensional interpretations of two-dimensional images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-5, no. 2, pp. 140–150, Mar. 1983.

[219] R. J. Douglass, "Interpreting three-dimensional scenes: A model building approach," *Comput. Graph. Image Process.*, vol. 17, no. 2, pp. 91–113, 1981.

[220] M. D. Levine and S. I. Shaheen, "A modular computer vision system for picture segmentation and interpretation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-3, no. 5, pp. 540–556, Sep. 1981.

[221] 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, NY, USA: Academic, 1978.

MUZAMMAL NASEER received the B.Sc. degree (Hons.) in electrical engineering from the University of Punjab and the master's degree in electrical engineering from the King Fahd University of Petroleum and Minerals. He is currently pursuing the Ph.D. degree with Australian National University, where he is a recipient of a competitive postgraduate scholarship. His current research interests are computer vision and machine learning. He received the M.Sc. degree Scholarship from the Ministry of Higher Education, Saudi Arabia, and also the Gold Medal for outstanding performance in the B.Sc. degree.

SALMAN KHAN (M'15) received the B.E. degree in electrical engineering from the National University of Sciences and Technology, Pakistan, in 2012, and the Ph.D. degree from The University of Western Australia, in 2016. His Ph.D. thesis received an honorable mention on the Dean's List Award. He was a Visiting Researcher with National ICT Australia, CRL, in 2015. He has been a Research Scientist with Data61, Commonwealth Scientific and Industrial Research Organization, and an Adjunct Lecturer with Australian National University, since 2016. His research has been published in top venues, including TPAMI, TIP, TNNLS, IJCV, TIP, and TGRS. His research interests include computer vision, pattern recognition, and machine learning. He was a recipient of several prestigious scholarships, including Fulbright and IPRS. He has served as a program committee member for several premier conferences, including CVPR, ICCV, and ECCV.

FATIH PORIKLI (M'96–SM'04–F'14) received the Ph.D. degree from New York University, in 2002. He was a Distinguished Research Scientist with Mitsubishi Electric Research Laboratories. He is currently a Professor with the Research School of Engineering, Australian National University, and a Chief Scientist with the Global Media Technologies Lab, Huawei, Santa Clara. He has authored over 300 publications, co-edited two books, and invented 66 patents. His research interests include computer vision, pattern recognition, manifold learning, image enhancement, robust and sparse optimization, and online learning with commercial applications in video surveillance, car navigation, robotics, satellite, and medical systems. He was a recipient of the Research and Development 100 Scientist of the Year Award, in 2006. He received five best paper awards at premier IEEE conferences and five other professional prizes. He is serving as an associate editor for several journals for the past 12 years. He has also served in the organizing committees of several flagship conferences, including ICCV, ECCV, and CVPR.