MCGraph: Multi-criterion representation for scene understanding

Moos Hueting[*] Aron Monszpart[*] Nicolas Mellado
University College London

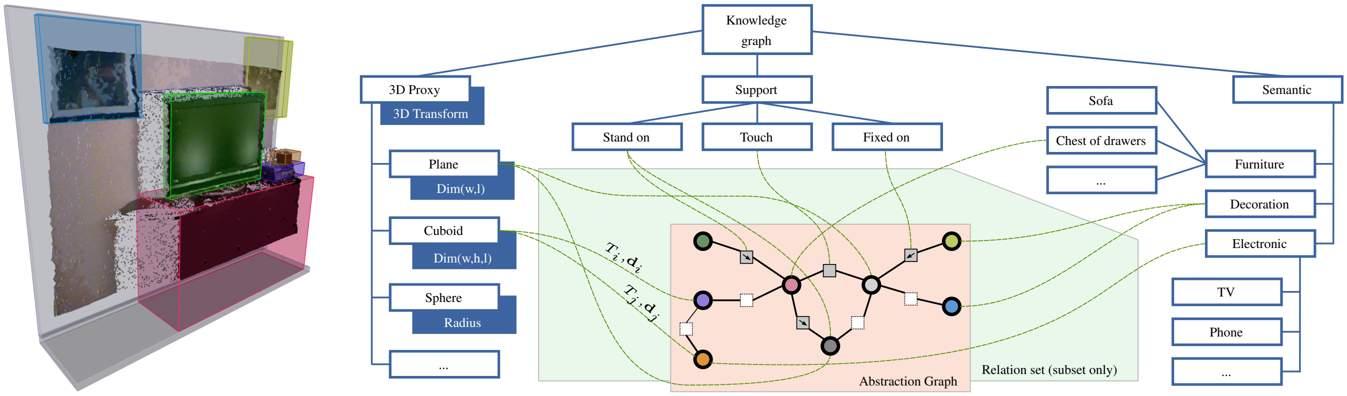


Figure 1: *Left: RGBD scene and its primitive abstraction. Right: Overview of the proposed MCGraph. It contains a hierarchical knowledge graph, the nodes of which describe multi-criteria prior knowledge units (rectangle nodes). An abstraction graph is built using processing or manual editing, where nodes represent objects (circle nodes, colours are consistent with the scene view) and (oriented) relations (square nodes). The nodes of the abstraction graph can be connected to the knowledge graph through relation edges (dashed), which form a relation set. Each edge characterizes, labels or abstracts objects of the scene using prior knowledge units, possibly by specifying parameters (e.g., cuboid dimensions and positions). For the sake of clarity, only a subset of the relation set is shown.*

Abstract

The field of scene understanding endeavours to extract a broad range of information from 3D scenes. Current approaches exploit one or at most a few different criteria (e.g., spatial, semantic, functional information) simultaneously for analysis. We argue that to take scene understanding to the next level of performance, we need to take into account many different, and possibly previously unconsidered types of knowledge simultaneously. A unified representation for this type of processing is as of yet missing. In this work we propose MCGraph: a unified multi-criterion data representation for understanding and processing of large-scale 3D scenes. Scene abstraction and prior knowledge are kept separated, but highly connected. For this purpose, primitives (i.e., proxies) and their relationships (e.g., contact, support, hierarchical) are stored in an *abstraction graph*, while the different categories of prior knowledge necessary for processing are stored separately in a *knowledge graph*. These graphs complement each other bidirectionally, and are processed concurrently. We illustrate our approach by expressing previous techniques using our formulation, and present promising avenues of research opened up by using such a representation. We also distribute a set of MCGraph annotations for a small number of NYU2 scenes, to be used as ground truth multi-criterion abstractions.

CR Categories: I.2.4 [Artifical Intelligence]: Knowledge Representation Formalisms and Methods—Representations I.3.3 [Computer Graphics]: Picture/Image Generation—Digitizing and scanning I.3.6 [Computer Graphics]: Methodology and Techniques—Graphics data structures and data types

Keywords: scene understanding, multi-criteria, scene abstraction

1 Introduction

The acquisition of 3D scene data is more popular than ever. With the increase in computational power and efficiency of algorithms (e.g. multi-view stereo), together with the advent of cheap consumer hardware (e.g. depth cameras, laser scans), data gets generated at an ever increasing rate. Concurrently, the complexity of analysing these data is growing. Besides the larger data volume, the variety of scenes we want to capture has grown from very constrained and specific scenarios to more unconstrained and uncontrolled environments, significantly pushing up the typical signal-to-noise ratio. Moreover, the expectations of the results we try to achieve with the data are higher than before. From simple object counting we have advanced to much more complex endeavours such as dense scene segmentation and labelling, scene simplification and geometric abstraction, and inference of inter-object relationships.

Many techniques have been researched across different fields for the purpose of scene analysis. The range of approaches proposed is highly diverse and input-dependent. In geometry processing, where input is often given as 3D data such as meshes and pointclouds, techniques range from local geometric descriptors and multiscale feature extraction to scene abstraction using either 3D primitives or shapes from a given collection. The resulting abstractions are used for inferring a multitude of types of information. Typical exam-

[*] Joint first authors.

ples include semantic information, which relates scene objects to categorical knowledge about the world, and functional information, which tells us the way in which objects can be interacted with. In computer vision, input and end-goals are often different. SLAM, for example, requires the analysis of still images and video frames with temporal information, as well as depth maps and pointclouds, for the purpose of robot localization and mapping. These differences notwithstanding, the steps necessary for reaching the goals are similar: scene abstraction, relationship inference, object classification, etc.

One overarching observation is that most methods consider both in processing and in inference one type of information at a time. We argue that for complex and complete scene understanding, it is necessary to consider the different types of information at the same time. Instead of building disjoint abstraction layers on top of one another, we suggest a joint representation where all different criteria can be generated and refined by each other. For illustration purposes, consider the following example.

Say we're trying to perform object detection and classification. If a spoon has been detected by virtue of local geometric descriptors, and we have a second blob of points close to the spoon which we haven't classified yet, it could be useful to concurrently use our prior knowledge that spoons and knives often co-occur at close distance. This could either be modeled as prior knowledge, or be inferred from the fact that other spoon-knife pairs have been found in the scene, or indeed in other scenes. In another task, we might try to understand the workings of a mechanical apparatus. Considering motion information (gear A influences the motion of gear B in a counter-clockwise fashion) together with functional information (crank C can be turned) might enable inference of more complex properties. In general, considering different types of information concurrently is more useful than considering them in turn [Chai et al. 2012; Wu et al. 2014]. Indeed, with different information sources, the whole is often greater than the sum of its parts.

Considering different types of information for concurrent processing needs a powerful representation. A common solution for representing complex forms of data is to use a graph structure. In this paper, we suggest the use of graphs to represent different types of scene information in the same place. We present MCGraph, a multi-criterion representation, in which structure and meaning are modelled separately, but connected fully. All its parts are presented in Section 3. In Section 4 we show how the MCGraph can be harnessed for achieving common goals in scene understanding. In Section 4.2 and 4.3 we discuss two specific works in scene understanding and propose how they can be beneficially adapted to the use of MCGraph. After, in Section 5, we discuss topics of research which we believe will benefit from using the MCGraph for the modeling of both knowledge and data. We discuss limitations and possible future research in Section 6.

Concretely, the contributions of this paper are twofold:

- A data representation for multi-criteria scene understanding, e.g. geometric abstractions, functional and semantic analysis of arbitrary scenes. Our representation is graph-based, agnostic to the type of data (e.g. RGB or RGBD images, point-clouds) and compatible with state-of-the-art algorithms for scene understanding and abstraction.

- A set of 3 MCGraph annotations for 3 NYU2 [Silberman et al. 2012] datasets, defining the first dataset with ground-truth for multi-criteria scene understanding.

2 Related work

The aim of this paper is to present a unified representation for scene understanding that can be used to store and analyse acquired data such as images, RGBD frames, or point clouds. Presenting a large amount of works published in this field is out of the scope of this paper, and we refer to the recent and extensive survey by Guo et al. [2014] presenting descriptor-based 3D object recognition techniques.

2.1 Image based analysis

We focus on analysis of the data representations used in state-of-the-art scene understanding techniques, which have thus proven efficient in aiding the solution of concrete problems. The methods are usually inspired by success in the image analysis domain, therefore we will also review recent success in that field. The involved problems in this field have been studied for a long time and their results can inspire, and indeed have inspired, pertinent ideas for scene understanding. For instance, analysis of arrangements of feature-abstractions has proven itself effective in image based object recognition [Felzenszwalb et al. 2010], and has more recently appeared in the analysis of recurring part arrangements of 3D shape collections [Zheng et al. 2014].

Abstraction Extracting objects from raw data is one of the first steps of scene understanding: the result provides an abstraction that can be used to infer object properties and relations between them. An effective way to extract objects is to over-segment the data into small homogeneous patches [Zitnick et al. 2004], or larger regions potentially corresponding to objects or their parts [Gould et al. 2009; Kumar and Koller 2010]. Another common technique is to abstract objects via invariant feature descriptors and their relative locations to perform robust object recognition [Wu et al. 2009; Hoiem et al. 2008; Felzenszwalb et al. 2010].

Graph based segmentation Graph-based image segmentation was proposed by Felzenszwalb and Huttenlocher [2004], and defines a robust and versatile representation of the data segments, with abstractions as nodes and their relations as edges. Because of this strong potential and its predominance, such representation is now ubiquitous [Peng et al. 2013] to segment 2D images [Rother et al. 2004; Kim et al. 2012a; Gould and Zhang 2012], RGBD frames [Silberman et al. 2012; Kim et al. 2013; Zheng et al. 2013a] or acquired 3D data [Huang et al. 2011].

2.2 3D analysis

The proposed graph based representations are all slightly different, and a generic formulation is today missing for scene understanding.

Primitive abstraction For instance, there is a large range of concepts that have been proposed to abstract scenes: abstraction using sets of elementary primitives [Schnabel et al. 2007], planes [Gallup et al. 2007; Arikan et al. 2013; Lafarge and Alliez 2013; Cabral and Furukawa 2014] or cuboids [Fidler et al. 2012; Jia et al. 2013; Shao* et al. 2014].

Inter-shape analysis Facilitating the different forms of abstractions, many works have been analyzing 3D shapes and their collections, as assessed by Mitra et al. [2014]. The power of processing shapes in groups receives special emphasis when designing multi-criteria representations. According to [Xu et al. 2014] co-analysis of semantically homogeneous shape collections can be performed by using local feature descriptors and their quantitative distances [Nguyen et al. 2011], or qualitative distances using heat kernels [Fisher et al. 2011], part-abstractions [Zheng et al. 2014; Fish* et al. 2014], similarities of hierarchical part-topologies [van

Kaick et al. 2013] and pairwise dissimilarities of similar shape pairs [Huang et al. 2013b].

2.3 Single and multi-criteria analysis

Single criterion It is important to note that most of the works doing 3D scene understanding are focusing on few criteria to analyse a scene, e.g. appearance co-occurrence and relative spatial layout [Hedau et al. 2010], primitive abstraction and physics [Gupta et al. 2010] or appearance and semantic labelling [Huang et al. 2013a].

Multi-criteria However, multi-criteria analysis has been proven efficient for image-based understanding. Jointly modelling appearance, shape and context [Shotton et al. 2009] or connecting descriptions of situations involving actors and actions occurring in images [Zitnick et al. 2013], has been shown to aid successful inference about object properties and their relations. Recently new domains are included in image analysis, as material, function and spatial envelope used by Patterson et al. [2014].
Similar approaches have more recently been successfully applied to process 3D data. The intra and inter-object symmetries of shapes coupled with the laws of physics and assumptions about the function of man-made objects can aid the understanding of involved machinery [Mitra et al. 2010]. Discovering regularities over shape-collections enable segmentation and inference of function of shapes [Laga et al. 2013], whilst further assumptions about the objects' users allow the complexity of the analyzed shapes to further increase [Jiang et al. 2013; Kim et al. 2014]. We can expect the potential of multi-criteria analysis to facilitate further work in this direction in the next years.

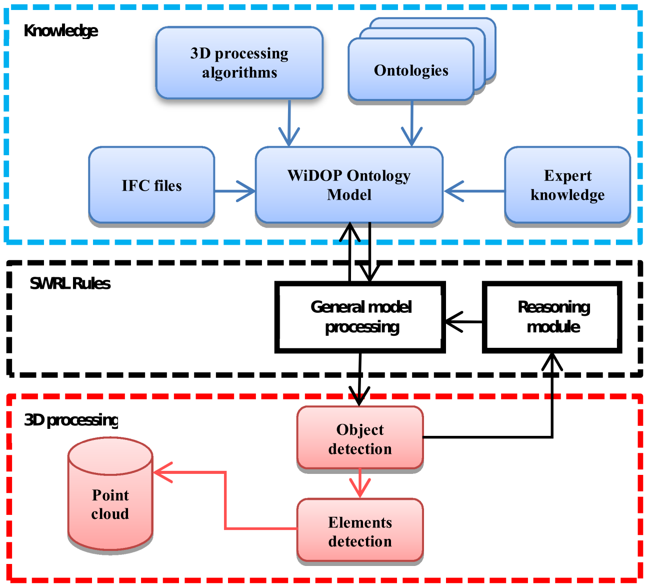


Figure 2: *Hmida et al. [2013] designed a representation to abstract knowledge from abstraction, but merges processing and representation in its concept. Our proposed, graph based representation abstracts processing from knowledge and gives greater freedom in connecting the knowledge graphs with the abstraction layer.*

Graph based information fusion Some attempts have been made in image and 3D based analysis to formalize multi-criteria abstractions. The power of graph representation is embraced by Zhang et al. [2014] performing feature analysis using hypergraphs, but is restricted to low-level information. Pan et al. [2004] claim to use multi-modal data for semantic analysis, but are similarly only performing image-processing operations to label satellite imagery. Hmida et al. [2013] represent prior knowledge applied to 3D processing in a knowledge graph, that is connected through rules with a 3D abstraction layer. The variety of representable information is however restricted to a few domains only in their setup. The research field of information fusion and its sub-field Multi-Criteria Decision Analysis target to develop effective methods for aggregating knowledge from different domains [Michael Doumpos 2013]. Using our representation we aim for the same multi-domain integration, but adapted to be able to accommodate common processing methods in indoor scene understanding.

3 MCGraph representation

3.1 Overview

The focus of this paper is to present a new representation for scene understanding, where data is abstracted along different criteria, e.g. spatial relations or semantic categories. Building such an abstraction requires *a priori* knowledge during the analysis stage, for instance to describe which objects and relations to look for in the data, and how to characterize them. A common solution is to abstract the analysed scene by a graph structure, where nodes are the discovered objects and edges their relations (see Figure 3-a). The resulting graph is then a mix between *a priori* knowledge, e.g. node and edge types, and a learned abstraction, e.g. the graph arrangement. Our claim here is that such a mixed representation restrains the use of the scene abstraction, by merging *a priori* and *discovered* information. Disambiguating between the two, as well as comparing graphs built from different kinds of prior knowledge, can be difficult after the fact.

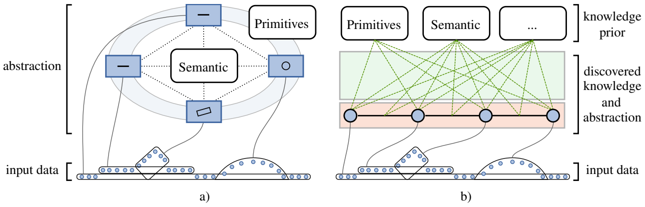


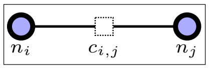
Figure 3: *a) Standard graph-based scene abstraction, where objects are nodes and edges their relations. b) Our representation, where prior knowledge is represented apart from the scene abstraction, facilitating multi-criteria description.*

The representation we present and describe in this section aims at defining a general and unified formulation to represent data for scene understanding and processing. In contrast to common methods, it is designed to both keep the discovered scene abstraction and the prior knowledge separated at all times (as illustrated in Figure 3-b) as well as to inherently enable multi-criteria processing. To do so, the scene is abstracted by an *abstraction graph* (see Section 3.2), whereas prior knowledge is represented independently of the scene as a *knowledge graph* (see Section 3.3). This approach is inspired by modern large scale graph representations [Neo4j 2012] that store labels in tables, and represent labelling operations as links between nodes and their labels.

3.2 Abstraction graph and relation set

The goal of the *abstraction graph* is to represent the scene as a set of objects and relations (see Figure 1, graph with light red background). As stated earlier, this graph must be as independent as possible from the knowledge priors. The key idea here is to represent both objects and their relations as nodes, and abstract their properties through a *relation set*. Note that here we focus on the data representation, and do not assume a specific algorithm to generate such graphs; see Section 4 for case studies and Section 6 for discussion.

Let $\mathcal{G}_a$ be the *abstraction graph* of the scene, composed of two sets of vertices. First, the set $\mathcal{X}$ is the set of nodes n_i representing objects discovered in the input data (see inset below). The second set $\mathcal{C}$ is formed by the connection nodes noted $c_{i,j}$. We attach to each node n_i a list of input samples from the input data, to represent the fact a node is *abstracting* those input samples. This relation is non-exclusive, and a sample can be owned by multiple nodes. On the other hand, nodes do not have to be connected to input samples if they are not related directly to original data. Such situations usually arise when new abstractions are generated in the analysis stage, and thus do not necessary rely on a set of samples.



We note $\mathcal{R}$ the *relation set*, composed of all the edges e_k between nodes of the abstraction graph and nodes of the knowledge graph (see Figure 1, graph with light green background). Like in [Neo4j 2012], these edges can represent the assigning of a label in the knowledge graph to a node of the abstraction graph $\mathcal{G}_a$. Edges can both be constructed between object nodes $n_i \in \mathcal{X}$ and nodes in the knowledge graph, as well as between relation nodes $c_{i,j} \in \mathcal{C}$ and nodes in the knowledge graph. As explained in the next section, the knowledge graph can contain more complex nodes than just labels. A specific knowledge node can potentially require of incident edges for parameters to be instantiated, in which case for a given node in $\mathcal{G}_a$ the set of necessary parameters is stored on the edge, as illustrated in Figure 1 for the connections "cuboid".

3.3 Knowledge graph

The purpose of a knowledge graph is to encode prior knowledge used to process (e.g. segment, approximate with primitives, infer inter-object relations) a given scene. As mentioned before, we do not focus on data processing algorithms, but on how to represent such prior knowledge. Note that a knowledge graph is defined *a priori*, and can be applied to different scenes. We take our inspiration from the large scale graph representation Neo4j [2012], where labels applied on nodes are stored in arrays and connected by edges. Our goal here is to extend this formalism to more complex operations than labelling, so we propose two major changes.

Firstly, instead of labels the graph stores *knowledge units*. They can be simple labels, but also more complex concepts such as primitive proxies, or spatial relations. In some cases, parameters are required to specialize a generic notion for the input data. For instance in Figure 1, a cuboid proxy is instantiated multiples time with different parameters (e.g. dimensions) to approximate input geometry.

Secondly, knowledge units are stored in a graph, instead of a simple array. According to Wu et al. [2014], hierarchical label graphs improve significantly the performance of abstraction algorithms in inferring relations between objects. Applied to knowledge units instead of labels, this means that required parameters for low-level knowledge units are inherited from their ancestors. Employing this concept of hierarchy, we store different types of knowledge in different knowledge sub-graphs, connected to the main knowledge graph through its root node. This allows us to store multicriterion knowledge separately.

Aside from hierarchical relations, other types of relations between knowledge units can also be stored. An example would be to have a relationship node (just as in the abstraction graph) connected to a node "appearTogether" in a topological knowledge subgraph, which is connected to a node "fork" and a node "spoon" in a classification knowledge subgraph (see Figure 4). Like in the abstraction graph, parameters required by the node representing the relationship ("appearTogether" in the example) are stored on the edge between itself and the relationship node, and could reference parameters required by the two members of the relationship ("fork" and "spoon"). This way we can not only model relationships that exist between specific instances in a data set, but also relationships that exist between knowledge units. Note that these relationships could either be defined as prior knowledge, or learned from the data seen by the system – the representation does not exclude either possibility.

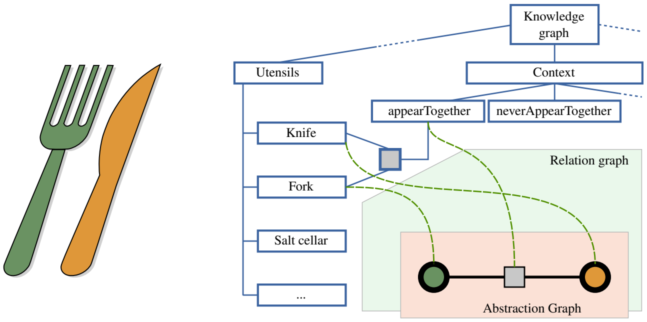


Figure 4: *Relationships can exist in the abstraction graph as well as in the knowledge graph*

4 Common Applications

In this section we illustrate how to use our representation when performing typical operations related to scene understanding to scenes and their abstractions. In Section 4.1 we present how to abstract typical scene and object properties, such as primitive abstractions, and object relationships and labels, using the concept of *knowledge graphs*. We will then look at two specific instances of past research in the popular sub-fields of *primitive abstraction* (Section 4.2) and *scene annotation*, specifically of RGBD data sets (Section 4.3).

4.1 Typical knowledge sub-graphs

Primitive proxies In the case of scene understanding, the nodes in the abstraction graph represent segments of the scene. It can be useful to represent these nodes using primitive proxies that approximate the data covered by these segments, for instance by using primitive arrangements [Li et al. 2011] or cuboid approximations [Jia et al. 2013]. The knowledge sub-graph pertaining to this type of knowledge contains a different knowledge unit (node) for each type of primitive. The nodes in the abstraction graphs are then connected to these knowledge units, storing the parameters of the primitive instance on the edge. Knowledge units can be hierarchically stored. For example, all primitives can be connected to a main unit "primitive". More specifically, a "polyhedron" node can exist, e.g. with the children "box", "pyramid". This type of hierarchical knowledge within the knowledge graph adds to the inference power of the system: knowing a node in the abstraction graph is a pyramid means it is a polyhedron as well. The opposite might impose a prior on classifying the node: knowing the object is a polyhedron can trigger a more specific method for disambiguating between the different types of polyhedra, as shown by Wu et al. [2014].

Objects labelling A semantic knowledge sub-graph can also be built to describe specific concepts in the world. For example, it can hold a "furniture"-knowledge unit, containing generic information about furniture (or indeed how to classify objects as furniture). More specific units, such as "sofa" and "tv bench" represent more

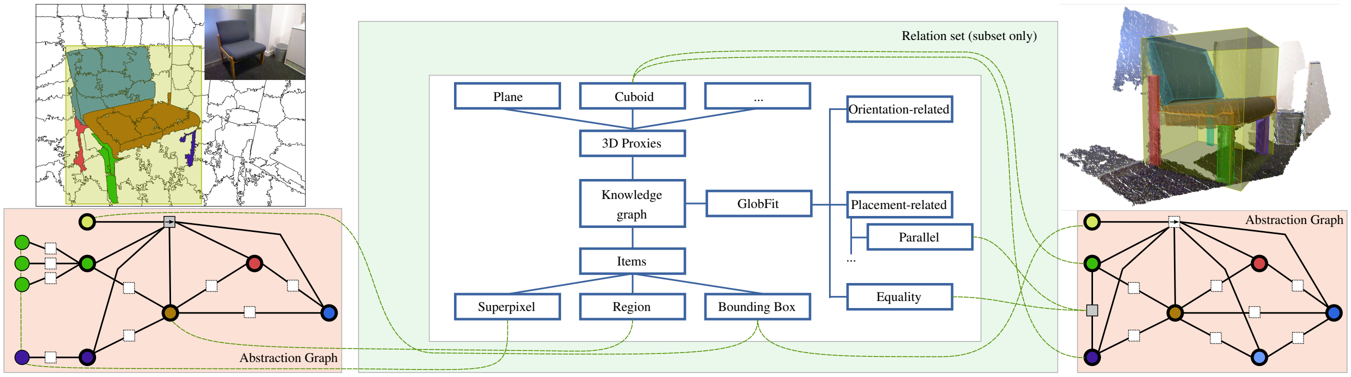


Figure 5: *The representation is agnostic to the type of input data – as long as the part of the abstraction graph which is reasoned with is the same, two different types of input data can be processed equally*

detailed knowledge pertaining to their specific properties. A node in the abstraction graph can be connected to any of these units, representing the specificity of our knowledge or classification. By extension, describing functions of objects can also be useful, for instance in robotic task solving. Light objects can be labelled as "canBeLifted", or switches can be labelled "canBePushed". We refer interested readers to the recent survey of Patterson et al. [2014], presenting an exhaustive taxonomy of 102 attributes for large-scale scenes, grouped by functions/affordances, materials, surface properties and spatial envelopes.

Objects relations Binary relationships between objects are also important to be considered. Recently, Zheng et al. [2013a] proposed to hallucinate invisible parts of objects by building a contact graph between them (fixed joint, support), estimating the stability of objects and extending their volume to get a physically stable solution. More formal definitions can be found to model contact relations between objects, for instance the Dimensionally Extended nine-Intersection Model [Clementini et al. 1993; Cle 1994] (DE-9IM) , a standard used in Geographic Information Systems to model topological relations between 2D polygons with: equals, disjoint, touches, contains, covers, intersects, within (inside), coveredby, crosses, overlap. An extension of this model to 3D has been proposed by Billen and Zlatanova [2003], and can thus be applied to 3D scene understanding.
Functional relations can also be characterized, for instance by considering simple arrangements such as "coaxial" and "coplanar" [Li et al. 2011], or a more complex Symmetry Functional Arrangement like defined in [Zheng et al. 2013b]. Another example is to model contextual relations, for instance by labelling "belongTogether" to characterize the edge between a fork and a knife on a table, or between a wall and a balcony [Schnabel et al. 2008].

Arbitrary knowledge sub-graphs can be proposed and used, and we do not aim to define an exhaustive set of criteria for scene understanding. The efficacy of different types of combinations of information types is an interesting subject for further research.

The proposed MCGraph is a versatile and generic representation, which does not impose any restrictions on the processing techniques used in conjunction with it. We will now present how existing research and their processing techniques can be adapted to the use of MCGraphs, without any modification or loss of generality.

4.2 Primitive abstraction

A common goal for scene understanding is primitive abstraction: representing (noisy) pointclouds using simplified primitives, such as cuboids, spheres and cylinders. In GlobFit [Li et al. 2011] the authors propose an iterative procedure in which primitive objects are fitted and global spatial relations between parts of the input are inferred. These inferred relations are then used to optimize the primitive representation.

They focus on three common spatial relationships: *orientation-related*, such as parallelism and orthogonality, *placement-related* such as coplanarity and coaxial alignment, and *equality* relations between equal primitive instances. In our representation, these three types of relationships can be modelled as separate subgraphs in the knowledge graph. The fitted primitives with their parameters can be represented as links between nodes in the abstraction graph and primitive nodes in the knowledge graph. The parameters of the primitive instances can be stored on the edges. All other operations the authors propose can then still be applied.

By itself, this reformulation is essentially equal to the original. Using our representation, however, comes with benefits – especially when considering possible extensions. If, for example, we would like to perform object classification on top of the primitive approximation, we could add another type of information to the knowledge graph. The added nodes could represent the types of object to detect. To perform the actual classification, the processing algorithm could either draw from knowledge already present in the graph, or supply a method itself. In both cases, the new knowledge nodes would be algorithmically related to the data, and could by extension be related to the existing knowledge in the graph.

Another useful addition would be to abstract the scene on multiple levels. In the original formulation, the spatial relationships are extracted between primitive abstractions. More relationships can be captured by allowing relationships to be formed between different groups of primitives, which are abstracted by one higher-level primitive. Consider, for example, a pointcloud of a bicycle. The spokes of a single wheel might have a rotational spatial relationship with respect to one another, whereas the two wheels have a relationship of parallelism. Our representation directly allows for this type of hierarchical abstraction, by having a primitive node in the abstraction graph connected to all of its constituent sub-parts.

In short, for GlobFit, our representation is capable of capturing all information and abstraction necessary for the original formulation, and is by design easily extended with new information without loss of generality.

4.3 RGBD annotation

Another often-discussed goal is the automatic annotation and labelling of RGBD data. Many approaches have been proposed [Silberman et al. 2012; Kim et al. 2013; Zheng et al. 2013a]. As an example, in [Wong et al. 2014] the authors propose a pipeline in which learned priors over RGBD images and their segmentations and labelling are used for classification of new RGBD images. Colours and estimated normals are used to over-segment the input data. Floors and walls are inferred automatically using a simple rule-based technique. The rest of the scene is approximated using cuboids. Priors learned from manually annotated scenes are then used to create an initial prediction of the classification.

Their approach can be modelled in our representation as follows. Taking the RGBD image as input, the segments resulting from the initial over-segmentation based on the predicted normals and colours could be seen as initial nodes in the abstraction graph. A knowledge node *superpixel* could be connected to each of these. Applying rules to detect the floor and walls would yield new abstraction nodes, connected to their respective nodes in a knowledge subgraph *primitives*. This subgraph would also contain the node *cuboid*, connected to the cuboids detected from the remainder of the data. The original pixels are owned by the superpixel nodes. Through the hierarchy, it is then possible to go from a single pixel to its superpixel and the primitive (or primitives) it belongs to. Please refer to Figure 5 for a simple example.

The *structure graph* mentioned in the paper are modelled as relations between nodes in the abstraction graph and nodes in a knowledge subgraph pertaining to the two different types of structural relationships mentioned (spatial and support). Relationships between nodes in the abstraction graph and nodes in the structure knowledge graph then follow as specified by the authors.

The specific classes are stored in their own knowledge subgraph. They are connected to the structure knowledge graph to model the *learned priors*, which are essentially relationships between different types of information.

We can once more model all information and data pertaining to the original task in our representation. This again simplifies the addition of new knowledge and the inference of yet unknown relationships. Adding co-occurrence information (e.g., bed and nightstand often occur together), for example, could improve classification results. We can easily add a knowledge sub-graph pertaining to this type of knowledge. As an added benefit, this type of knowledge could be useful for a large range of goals, for which the knowledge sub-graph can be reused.

5 Extensions

In the previous section we discussed the application of the MCGraph representation to existing techniques. We will now look at directions of research for which we believe it highly beneficial to use the MCGraph representation. In all of these suggestions, the high connectivity of the knowledge and data in the MCGraph is used to improve upon results of current representations.

5.1 Scene collections for assisted living

In connection with our ageing community a lot of focus is put on the research of automated assistance for the elderly at home. An ideal robotic assistant is capable of proactively mapping the environment to avoid accidents and to ensure that it is capable of physically approaching its subject at all times. The system therefore needs to continuously infer the potential hazards of its configuration from the ever changing state of the indoor environment.

One way to build and train such a system can be to perform indoor scene-understanding on a large set of scans of usual and less usual room configurations of elderly people, possibly over a longer time period. Event recognition of image streams is a well researched subject, and can give a good starting point to path and intervention planning for robots, but can be limited by its simplistic modelling of objects in the scenery. For proactive accident avoidance however, exact knowledge about function, affordance and possible deformations of complex objects is required.

The current state of the art analysis of shape and scene collections is equipped to perform inference over shape and scene context. Frequent configurations of object parts, objects and scenes can be discovered, as we discussed in Section 2. These methods are currently limited to typical configurations. We need to explore and understand all types of home environments, emphasizing anomalous configurations with special semantic interpretations, i.e. situations considered dangerous or capable of causing distress to its inhabitants.

Using the MCGraph representation, multi-criteria top-down queries become possible. In an existing scene collection setup, a query can already be formulated in the spatial or primitive approximation domain, i.e. searching for a special part pattern or object arrangement is possible. For example, Huang et al. [2013a] showed, that given some labels in a shape collection, the semantic class of most shapes can be inferred, rendering look-ups in the semantic domain possible. However, given our example, we would often like to look for ”objects that easily fall over”, or ”appliances to be kept away from rooms with water”. Our representation enables the easy addition of the knowledge domains necessary for these queries (stability, function, affordance, etc.) to the existing systems. Additionally cross-domain queries can now be formulated, i.e. the combination of the above two queries: ”objects, that easily fall, and incur danger when in contact with water”. A robot actively looking out for these combinations could prevent accidents from happening by putting objects back to their original places.

5.2 Multi-criteria multi-scale

It is well established to apply multi-scale analysis to scene understanding problems as well assessed by Mitra et al. [2014], for instance using Heat Kernel Signatures [Sun et al. 2009] or Growing Least Squares [Mellado et al. 2012]. A specific branch of these works facilitate the power of graphs when targeting 3D matching over scale-space. Berner et al. [2008] find symmetric structures across scales by matching graph abstractions of local structures. Hou et al. [2012] employ feature graphs to minimize description length and without losing the ability to create multi-scale similarity queries.

The success of these methods is restricted by the fact that they operate only in the spatial domain. We consider them as single-criterion approaches, since they use geometry or primitive approximation to establish intra- and inter-object relationships. They only work well under controlled conditions, where we have quite concrete information about the extents of the search space.

The analysis of shapes, their heterogeneous collections and occurrences in scenes in the real world is usually much more diverse. This so called *open-world problem* emphasizes the complications triggered by continuously growing model sets with unknown or unreliable object categories and size information, as expressed by Savva et al. [2014]. They successfully applied semantic knowledge, such as size measurements coupled to the description of the models to perform general scale estimation of large object databases.

Their solution can however be limited by the fact that only a single modality of semantic knowledge is available, and only for a small subset of the models.

In our representation, one can seamlessly include processing in other domains, turn to function or affordance analysis for man-made objects, or include topological analysis over the shape collections. One can also imagine, that the methods used for intra-domain analysis can be modified or extended for inter-domain knowledge retrieval. This can now be done, since the domain specific graphs are all defined in the same place, over the same abstraction graph.

Considering the previous example, we now assume that we have a trained assistance system using multi-criteria scene collections. Due to the open-world problem of objects appearing in the scenes with increasing diversity, it is important to be able too recognize and categorize them accurately over several domains (category, function, affordance, etc.). Multi-criteria multi-scale look-ups can be used to narrow down the search space for object classification based on the time of the year, geographic location, or function. In other words, a large, scissor-like object in New Jersey in July is likely to be a hedge trimmer, that was forgotten on the kitchen table, and should be placed back in the garden shed. A small one in Kyoto in December is more probable to be a Bonsai leaf trimmer and might have been put there by the owner, as a reminder to perform the weekly trimming.

5.3 Prior knowledge for object registration

Scene reconstruction from noisy pointcloud data is a popular area of research. One line of attack is to inform the reconstruction system using object priors: instead of reconstructing meshes from pointcloud data directly, the system looks for occurrences of previously known shapes in a prior shape collection. In Kim et al. [2012b], for example, the authors reconstruct scenes from noisy pointcloud data by informing the system about the types of furniture to expect, including the types of articulation these shapes are capable of. In the 2D realm, in Su et al. [2014], the authors infer depth from single photographs by employing a database of shapes to which the shape in the image is assumed to be similar.

Using geometric priors about expected objects for 3D detection and reconstruction is thus useful. This kind of information is easy to store in the knowledge graph of our representation. However, the type of additional information used to inform the reconstruction system need not be limited to a specific type. Moreover, the relationships between different objects and object parts in typical scenes could be useful to guide the reconstruction process.

Consider, for example, a case where we want to register a model of an engine to a noisy pointcloud captured of a running instance of this engine. Using just the geometric information in the model is not enough: the model of the engine is not static. On the other hand, matching the different parts of the model separately is sub-optimal, as the relationships between the parts imposed by the structure of the engine are very strong priors on the possible distribution of the data.

Zooming out, if we do not know the specific type of engine, but aside from the engine itself the car is also captured by the data, we could use any information we have inferred from the car to help us decide what engine type we are dealing with, using knowledge of what engines are used in which type of car. To decide on the type of car, we could look at the type of wheels, and the dimensions of the primitives abstracting the car.

By now we are modelling quite a broad scala of types of knowledge and relationships between knowledge:

- Geometry of data informing pose of engine
- Motion relationships of specific engine informing pose of engine
- Types of car informing types of engine
- Geometry of data informing type of car

In summary, modelling prior information on objects for model reconstruction and pose recovery is useful, and using our representation we can model much more intricate knowledge and its connections than just simple geometric priors.

6 Discussion

In this paper, we proposed the MCGraph representation for scene understanding. Its fundamental strength lies in the collocated representation of prior knowledge and scene abstraction within the same graph, which allows for intricate relationships between both knowledge and scene abstractions to be made and reasoned with. We discussed both its applicability to current techniques, as well as suggestions for directions of research for which the MCGraph seems particularly suited.

We do not seek to propose an all-replacing representation for every problem in scene understanding. For some specific techniques, a special way of structuring both data and prior knowledge might be necessary. We argue, however, that the MCGraph is a generic enough concept to be applied to a vast array of subjects within the field.

Moreover, as we have mentioned before, we expect graph-based understanding to become more prevalent in scene understanding in the near future, as it has happened before in other fields such as image processing (see Section 2). Having a common representation for the different techniques that will surface both simplifies understanding of and comparison between them. In addition, merging knowledge graphs from different techniques can be used for combining efforts for either the same or for some novel goal.

Although the theoretical description of the MCGraph as proposed in this paper is complete, we have not discussed practical considerations related to implementation. This is a topic for future work. Specifically, we are currently developing a tool for annotating pointclouds using the MCGraph, for which executables and the code will be released.

We release 3 complete MCGraphs (abstraction graph, relation set and knowledge graph) of 3 different NYU2 [Silberman et al. 2012] scenes on our project page. As of yet, these scenes have been hand-annotated. We are currently developing a tool for annotating pointclouds manually using the MCGraph, for which the code will be released.

The foundation has now been laid for the application of our work to new methods in scene understanding. Case studies on how to use MCGraphs for specific directions of research is an interesting next step. Further, we could now dive into how to apply popular graph processing techniques such as graph cuts, spectral analysis and Bayesian network inference, to the MCGraph.

In summary, we believe that to take scene understanding to the next level, we need to look at combining as much knowledge as possible at both processing and inference time. Graph-based understanding, and by extension the MCGraph, are perfectly suited for this, and we are looking forward to future research taking full advantage of its implications.

References

Arikan, M., Schwärzler, M., Flöry, S., Wimmer, M., and Maierhofer, S. 2013. O-snap: Optimization-based snapping for modeling architecture. *ACM Transactions on Graphics 32* (Jan.), 6:1–6:15.

Berner, A., Bokeloh, M., Wand, M., Schilling, A., and Seidel, H.-P. 2008. A graph-based approach to symmetry detection. In *Symposium on Volume and Point-Based Graphics*, Eurographics Association, Los Angeles, CA, 1–8.

Billen, R., and Zlatanova, S. 2003. 3d spatial relationships model: a useful concept for 3d cadastre? *Computers, Environment and Urban Systems 27*, 4, 411 – 425. 3D Cadastres.

Cabral, R., and Furukawa, Y. 2014. Piecewise planar and compact floorplan reconstruction from images.

Chai, Y., Rahtu, E., Lempitsky, V., Van Gool, L., and Zisserman, A. 2012. Tricos: A tri-level class-discriminative co-segmentation method for image classification. In *European Conference on Computer Vision*.

1994. Modelling topological spatial relations: Strategies for query processing. *Computers & Graphics 18*, 6, 815 – 822.

Clementini, E., Di Felice, P., and van Oosterom, P. 1993. A small set of formal topological relationships suitable for end-user interaction. In *Advances in Spatial Databases*, D. Abel and B. Chin Ooi, Eds., vol. 692 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 277–295.

Felzenszwalb, P. F., and Huttenlocher, D. P. 2004. Efficient graph-based image segmentation. *Int. J. Comput. Vision 59*, 2 (Sept.), 167–181.

Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D. 2010. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 32*, 9 (Sept), 1627–1645.

Fidler, S., Dickinson, S., and Urtasun, R. 2012. 3d object detection and viewpoint estimation with a deformable 3d cuboid model. 620–628.

Fish*, N., Averkiou*, M., van Kaick, O., Sorkine-Hornung, O., Cohen-Or, D., and Mitra, N. J. 2014. Meta-representation of shape families. *Transactions on Graphics (Special issue of SIGGRAPH 2014)*. * joint first authors.

Fisher, M., Savva, M., and Hanrahan, P. 2011. Characterizing structural relationships in scenes using graph kernels. In *ACM SIGGRAPH 2011 Papers*, ACM, New York, NY, USA, SIGGRAPH '11, 34:1–34:12.

Gallup, D., Frahm, J.-M., Mordohai, P., Yang, Q., and Pollefeys, M. 2007. Real-time plane-sweeping stereo with multiple sweeping directions. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, 1–8.

Gould, S., and Zhang, Y. 2012. Patchmatchgraph: Building a graph of dense patch correspondences for label transfer. In *ECCV (5)*, 439–452.

Gould, S., Gao, T., and Koller, D. 2009. Region-based segmentation and object detection. In *Advances in Neural Information Processing Systems (NIPS 2009)*.

Guo, Y., Bennamoun, M., Sohel, F., Lu, M., and Wan, J. 2014. 3d object recognition in cluttered scenes with local surface features: A survey. vol. PP, 1–1.

Gupta, A., Efros, A. A., and Hebert, M. 2010. Blocks world revisited: Image understanding using qualitative geometry and mechanics.

Hedau, V., Hoiem, D., and Forsyth, D. 2010. Thinking inside the box: Using appearance models and context based on room geometry. In *Proceedings of the 11th European Conference on Computer Vision: Part VI*, Springer-Verlag, Berlin, Heidelberg, ECCV'10, 224–237.

Hmida, H. B., Cruz, C., Boochs, F., and Nicolle, C. 2013. Knowledge base approach for 3d objects detection in point clouds using 3d processing and specialists knowledge. *CoRR abs/1301.4991*.

Hoiem, D., Efros, A. A., and Hebert, M. 2008. Putting objects in perspective. *International Journal of Computer Vision 80*, 1, 3–15.

Hou, T., Hou, X., Zhong, M., and Qin, H. 2012. Bag-of-feature-graphs: A new paradigm for non-rigid shape retrieval. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, 1513–1516.

Huang, Q., Koltun, V., and Guibas, L. 2011. Joint-shape segmentation with linear programming. *ACM Transactions on Graphics 30*, 125:1–125:11.

Huang, Q., Su, H., and Guibas, L. 2013. Fine-grained semi-supervised labeling of large shape collections. *ACM Transactions on Graphics 32*, 190:1–190:10.

Huang, S.-S., Shamir, A., Shen, C.-H., Zhang, H., Sheffer, A., Hu, S.-M., and Cohen-Or, D. 2013. Qualitative organization of collections of shapes via quartet analysis. *ACM Trans. Graph. 32*, 4 (July), 71:1–71:10.

Jia, Z., Gallagher, A., Saxena, A., and Chen, T. 2013. 3d-based reasoning with blocks, support, and stability. 1–8.

Jiang, Y., Koppula, H., and Saxena, A. 2013. Hallucinated humans as the hidden context for labeling 3d scenes. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2993–3000.

Kim, B., Sun, M., Kohli, P., and Savarese, S. 2012. Relating things and stuff by high-order potential modeling. In *in ECCV'12 Workshop on Higher-Order Models and Global Constraints in Computer Vision*.

Kim, Y. M., Mitra, N. J., Yan, D.-M., and Guibas, L. 2012. Acquiring 3d indoor environments with variability and repetition. *ACM Transactions on Graphics (TOG) 31*, 6, 138.

Kim, B.-s., Kohli, P., and Savarese, S. 2013. 3D scene understanding by Voxel-CRF. In *Proceedings of the International Conference on Computer Vision*.

Kim, V. G., Chaudhuri, S., Guibas, L., and Funkhouser, T. 2014. Shape2pose: Human-centric shape analysis. *Trans. on Graphics (Proc. of SIGGRAPH 2014)*.

Kumar, M. P., and Koller, D. 2010. Efficiently selecting regions for scene understanding. In *CVPR*, 3217–3224.

Lafarge, F., and Alliez, P. 2013. Surface reconstruction through point set structuring. In *Proc. of Eurographics*.

Laga, H., Mortara, M., and Spagnuolo, M. 2013. Geometry and context for semantic correspondence and functionality recognition in manmade 3d shapes. *ACM Transactions on Graphics (Presented at SIGGRAPH 2014) 32*, 5, Article no. 150.

Li, Y., Wu, X., Chrysanthou, Y., Sharf, A., Cohen-Or, D., and Mitra, N. J. 2011. Globfit: Consistently fitting primitives by discovering global relations. *ACM Transactions on Graphics 30*, 4, 52:1–52:12.

Mellado, N., Guennebaud, G., Barla, P., Reuter, P., and Schlick, C. 2012. Growing least squares for the analysis of manifolds in scale-space. *Comp. Graph. Forum 31*, 5 (Aug.), 1691–1701.

Michael Doumpos, E. G. 2013. *Multicriteria Decision Aid and Artificial Intelligence: Links, Theory and Applications*. John Wiley & Sons, Ltd.

Mitra, N. J., Yang, Y.-L., Yan, D.-M., Li, W., and Agrawala, M. 2010. Illustrating how mechanical assemblies work. *ACM Transactions on Graphics 29*, 3, 58:1–58:12.

Mitra, N. J., Wand, M., Zhang, H., Cohen-Or, D., Kim, V., and Huang, Q.-X. 2014. Structure-aware shape processing. In *ACM SIGGRAPH 2014 Courses*, ACM, New York, NY, USA, SIGGRAPH '14, 13:1–13:21.

Neo4j, 2012. Neo4j - the worlds leading graph database.

Nguyen, A., Ben-Chen, M., Welnicka, K., Ye, Y., and Guibas, L. 2011. An optimization approach to improving collections of shape maps. In *Eurographics Symposium on Geometry Processing (SGP)*, 1481–1491.

Pan, J.-Y., Yang, H.-J., Faloutsos, C., and Duygulu, P. 2004. Gcap: Graph-based automatic image captioning. In *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW '04. Conference on*, 146–146.

Patterson, G., Xu, C., Su, H., and Hays, J. 2014. The sun attribute database: Beyond categories for deeper scene understanding. *Int. J. Comput. Vision 108*, 1-2 (May), 59–81.

Peng, B., Zhang, L., and Zhang, D. 2013. A survey of graph theoretical approaches to image segmentation. *Pattern Recogn. 46*, 3 (Mar.), 1020–1038.

Rother, C., Kolmogorov, V., and Blake, A. 2004. "grabcut": Interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph. 23*, 3 (Aug.), 309–314.

Savva, M., Chang, A. X., Bernstein, G., Manning, C. D., and Hanrahan, P. 2014. On being the right scale: Sizing large collections of 3D models. *Stanford University Technical Report CSTR 2014-03*.

Schnabel, R., Wahl, R., and Klein, R. 2007. Efficient ransac for point-cloud shape detection. 214–226.

Schnabel, R., Wessel, R., Wahl, R., and Klein, R. 2008. Shape recognition in 3d point-clouds. In *The 16-th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision'2008*, UNION Agency-Science Press, V. Skala, Ed.

Shao*, T., Monszpart*, A., Zheng, Y., Koo, B., Xu, W., Zhou, K., and Mitra, N. 2014. Imagining the unseen: Stability-based cuboid arrangements for understanding cluttered indoor scenes. *SIGGRAPH Asia 2014*, to appear.

Shotton, J., Winn, J. M., Rother, C., and Criminisi, A. 2009. Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *International Journal of Computer Vision 81*, 1, 2–23.

Silberman, N., Hoiem, D., Kohli, P., and Fergus, R. 2012. Indoor segmentation and support inference from rgbd images.

Su, H., Huang, Q., Mitra, N. J., Li, Y., and Guibas, L. 2014. Estimating image depth using shape collections. *Transactions on Graphics (Special issue of SIGGRAPH 2014)*.

Sun, J., Ovsjanikov, M., and Guibas, L. 2009. A concise and provably informative multi-scale signature based on heat diffusion. In *Proceedings of the Symposium on Geometry Processing*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, SGP '09, 1383–1392.

van Kaick, O., Xu, K., Zhang, H., Wang, Y., Sun, S., Shamir, A., and Cohen-Or, D. 2013. Co-hierarchical analysis of shape structures. *ACM Trans. Graph. 32*, 4 (July), 69:1–69:10.

Wong, Y.-S., Chu, H.-K., and Mitra, N. J. 2014. Smartannotator: An interactive tool for annotating rgbd indoor images. *CoRR abs/1403.5718*.

Wu, L., Hoi, S. C., and Yu, N. 2009. Semantics-preserving bag-of-words models for efficient image annotation. In *Proceedings of the First ACM Workshop on Large-scale Multimedia Retrieval and Mining*, ACM, New York, NY, USA, LS-MMRM '09, 19–26.

Wu, C., Lenz, I., and Saxena, A. 2014. Hierarchical semantic labeling for task-relevant rgb-d perception. In *Robotics: Science and Systems (RSS)*.

Xu, K., Ma, R., Zhang, H., Zhu, C., Shamir, A., Cohen-Or, D., and Huang, H. 2014. Organizing heterogeneous scene collection through contextual focal points. *ACM Transactions on Graphics, (Proc. of SIGGRAPH 2014) 33*, 4, to appear.

Zhang, L., Gao, Y., Hong, C., Feng, Y., Zhu, J., and Cai, D. 2014. Feature correlation hypergraph: Exploiting high-order potentials for multimodal recognition. *IEEE T. Cybernetics 44*, 8, 1408–1419.

Zheng, B., Zhao, Y., Yu, J. C., Ikeuchi, K., and Zhu, S.-C. 2013. Beyond point clouds: Scene understanding by reasoning geometry and physics.

Zheng, Y., Cohen-Or, D., and Mitra, N. J. 2013. Smart variations: Functional substructures for part compatibility. *Computer Graphics Forum (Eurographics) 32*, 2pt2, 195–204.

Zheng, Y., Cohen-Or, D., Averkiou, M., and Mitra, N. J. 2014. Recurring part arrangements in shape collections. *Computer Graphics Forum (Special issue of Eurographics 2014)*.

Zitnick, C., Kang, S., Uyttendaele, M., Winder, S., and Szeliski, R. 2004. High-quality video view interpolation using a layered representation. In *ACM SIGGRAPH*, Association for Computing Machinery, Inc., vol. 23, 600608.

Zitnick, C. L., Parikh, D., and Vanderwende, L. 2013. Learning the visual interpretation of sentences. In *Proceedings of the 2013 IEEE International Conference on Computer Vision*, IEEE Computer Society, Washington, DC, USA, ICCV '13, 1681–1688.