Towards Categorization and Pose Estimation of Sets of Occluded Objects in Cluttered Scenes from Depth Data and Generic Object Models Using Joint Parsing

Basevi, Hector; Leonardis, Ales

DOI:
10.1007/978-3-319-49409-8

Document Version
Peer reviewed version

Citation for published version (Harvard):
Basevi, H & Leonardis, A 2016, Towards Categorization and Pose Estimation of Sets of Occluded Objects in Cluttered Scenes from Depth Data and Generic Object Models Using Joint Parsing. in G Hua & H Jégou (eds), Computer Vision – ECCV 2016 Workshops Part III. vol. 9915, Lecture Notes in Computer Science, vol. 9915, Springer, pp. 665-681, 14th European Conference on Computer Vision (ECCV 2016), 15/10/16. https://doi.org/10.1007/978-3-319-49409-8

Towards Categorization and Pose Estimation of Sets of Occluded Objects in Cluttered Scenes from Depth Data and Generic Object Models Using Joint Parsing

Hector Basevi and Aleš Leonardis

School of Computer Science, University of Birmingham, United Kingdom
{H.R.A.Basevi, A.Leonardis}@cs.bham.ac.uk

Abstract. This work addresses the task of categorizing and estimating the six-dimensional poses of all visible and partly occluded objects present in a scene from depth image information, in the absence of ground truth training examples and exact geometrical models of objects. A novel multi-stage algorithm is proposed to perform this task by first estimating object category probabilities for each depth pixel using local depth features computed from multiple viewpoints. It then generates a large set of object category and pose pairs, and reduces this set via joint parsing to best match the observed scene depth and per-pixel object category probabilities, while minimizing the physical overlap between objects within the subset. A decision forest is trained on synthetic data and used to estimate pixel category probabilities which are then used to generate a set of pose estimates for all categories. Finally a combinatorial optimization algorithm is used to perform joint parsing to find a best subset of poses. The algorithm is applied to the challenging Heavily Occluded Object Challenge data set which contains depth data of sets of objects placed on a table and generic object models for each category, but does not include registered RGB data or human annotations for training. It is tested on difficult scenes containing 10 or 20 objects and successfully categorizes and localizes 29% of objects. The joint parsing algorithm successfully categorizes and localizes 56% of objects when ground truth poses are added to the set of pose estimates.

Keywords: Joint parsing; categorization; object recognition; pose estimation; scene understanding; robotic vision.

1 Introduction

Computer vision algorithms use a wide range of features to locate, categorize, and estimate the poses of objects in depth, RGB, and RGB-D images [1,2,3]. Many recent works use machine learning algorithms, which learn features from training rather than using hand-designed features [4,5,6], and popular examples include convolutional neural networks (CNNs) [7,8,9,5,10,11] and decision forests [4], which were famously used in Microsoft's Kinect human pose estimation algorithms [12]. Machine learning techniques rely on data sets of annotated examples in order to learn features via supervised learning and generally require large data sets to learn robust features [7,13]. However, these are not always available.

For example, take the case of a mobile robot navigating an unknown office environment. Assume that the robot is using a Lidar system for vision and has a low bandwidth cellular network connection for communication. The robot has been told by its operator to find and retrieve a teacup, but the robot has not previously learned to recognize teacups. It would not be practical to send a data set of annotated images of teacups to the robot over the low bandwidth connection (and in the case of some categories of object, such a data set may not yet exist), but it is possible to send a model of the geometry of a generic teacup, which may not precisely match any of the actual teacups in the scene. It is also not sufficient to be able to recognize the presence of a teacup in the scene as the robot needs to estimate its 6D pose in order to manipulate the teacup.

If this is taking place in a typical office environment then there are also other challenges that cannot be avoided. The scene may contain a number of different categories of object. The robot's ability to move within the environment may be limited, and it may only able to image the scene from a few viewpoints. The teacup may or may not be in the scene and it may be partially occluded by other objects. The robot may need to categorize and estimate pose for these objects as well in case that these must be manipulated in order to gain access to the teacup.

A variety of approaches have been used for categorizing and estimating the pose of objects in RGB images, including the use of image gradient-based part templates and deformable part models [1,14], and complete 3D CAD models which are either aligned to 2D RGB images [15] or are used to train detectors with task-dependent performance [6]. Recent growth in the availability of RGB-D sensors has enabled the generation of a large number of RGB-D data sets, and the creation of many algorithms that perform object detection [5,11] and pose estimation [16,4,17,10] on RGB-D images. Some algorithms that operate on RGB-D data sets deliberately discard RGB cues to remove the influences of illumination and texture, and operate purely on depth data [18], and some augment or replace parts of RGB-D data sets with synthetic data using CAD objects [10]. An alternative goal is scene completion, where the categories of the objects in a scene are not important but a geometric representation of occluded parts of the scene are desired [19].

Some algorithms detect individual objects of specific categories in depth images [18] or RGB-D images [11]. Others detect individual objects and estimate object 6D poses in RGB-D images [4,17,10]. Certain approaches fit for a number of objects jointly on RGB-D images, but do not attempt to categorize the objects [19]. This work presents a method for categorizing and estimating 6D pose for all objects in a scene simultaneously, including those that are partially occluded, from single or sparse multiple-view depth data. It does not require annotated depth data for training and generates its own synthetic labelled training data using generic object models.

The proposed algorithm generates random synthetic images of scenes using generic object models and learns decision forests on local depth features incorporating all available camera views, adapting the approach of Brachmann *et al.* [4]. These decision forests estimate the probability that each pixel belongs to one of the object categories or to the background. For each category, sampling a random pixel from one of the views proportional to its category probability enables a bounding box to be placed on each of the available views to extract a 3D object fragment from the depth images. The generic

category model is then matched to the 3D object fragment to estimate a 6D pose. Many such object fragments and resulting poses are generated for each category, and the best fitting poses for each category are then processed by a joint parsing algorithm. This algorithm finds a subset of categories and poses that fits well with the depth data and pixel category probabilities while minimising the physical overlap between objects. Unlike Brachmann *et al.*, we generate features from multiple camera views in the absence of exact object instance information, and search for all objects of classes of interest rather than searching for a single instance of a specific object.

The contributions of this work lie in the use of local depth features incorporating multiple views to categorize objects and estimate their poses combined with the use of global properties of the scene to refine the set of all categories and poses simultaneously. Unlike Brachmann *et al.* [4], the use of multiple views to form a feature allows full 3D features to be learnt. The use of local depth features enables the algorithm to detect occluded objects but inevitably produces spurious categories and poses due to ambiguous object fragments and the use of generic models that could not be expected to match object fragments exactly. Global depth, pixel category probabilities, and object overlaps enable the algorithm to remove spurious categories and poses.

In section 2 we discuss a selection of related work. In section 3 we discuss the proposed algorithm, and in section 4 we discuss how the algorithm is implemented. In section 5 we discuss the HOOC data set, and in section 6 we discuss the application of the presented algorithm to the HOOC data set. Finally, in section 7 we give conclusions.

2 Related Work

There are many publications on object detection, pose estimation, and scene completion, which are topics relevant to this work. This section contains the most related work.

A classic approach to object categorization and pose estimation is to use object features such as SIFT features [20,21], or learned features [22,23]. These features can be used to find correspondences, identify parts, or to generate global descriptors for object classification [24]. However, these require RGB or intensity images to identify features, and a training set of RGB images of the objects of interest in order to construct a feature set. Training RGB images are not accessible in the HOOC data set due to an absence of RGB camera calibration information. There has been some research into identifying analogous features in depth images [25], which could potentially form part of an alternative approach to object categorization and pose estimation. Another approach is to use a set of RGB (or RGB-D) images and associated poses as templates, and compare test images to template images directly [26].

It is possible to measure the 3D geometry and RGB texture of objects using an apparatus such as a depth sensor and a turntable. These dense RGB-D measurements can then be used to construct a model and extract features specific to the object to maximise performance. Hinterstoisser *et al.* [16] presented an approach using templates consisting of RGB image gradients and depth image normal vectors of each object at a number of viewpoints to identify objects in RGB-D images. 6D pose was then refined using the depth data. Their method was dependent on possessing exact object representations in order to generate templates.

Brachmann *et al.* [4] presented an approach that trained decision forests to estimate pixel-wise category probabilities and poses based on local RGB-D features. They also experimented with pure depth features which did not perform as well. They generated pose estimates by associating object coordinates with the leaves of the decision forest, relying on the fact that their test objects did not possess RGB-D symmetries under rotation. The depth data and pixel probabilities were then used to decide whether to discard a pose via a designed energy function. Krull *et al.* [17] extended their previous work by replacing their designed energy function with a CNN which again evaluated individual poses. We adapt their decision trees for estimating pixel category probabilities but do not use the decision forests to directly estimate object poses, as the generic models, symmetric objects, and the lack of RGB data does not allow this. Unlike their approach we also do not attempt to detect a single object of a specific category and estimate its pose, but instead search for all objects belonging to categories of interest and use an energy function for the global scene to refine the set of potential object poses. By performing this for all objects simultaneously we compensate for the lack of specific object information.

Sun *et al.* [27] presented an approach using Hough voting to detect objects and estimate poses from RGB and RGB-D images. Depth was used to address challenges involving object scales in images and so was used to correct image patches for object distance, but the primary features used by the algorithm were RGB image patches. Tejani *et al.* [28] and Doumanoglou *et al.* [29] combined Hough voting with decision forests, in conjunction with other techniques.

Song *et al.* [18] presented an approach for object detection from depth data only, but did not focus on object pose. Similarly to this work, they used generic object models in the detection process. However, they used designed features as inputs to support vector machine (SVM) classifiers, and trained a classifier for each synthetic training object pose. They also adopted a 3D sliding window and processed the depth data in 3D form, rather than the approach in this work which operates on depth images directly. Song *et al.* later presented a CNN-based approach [11] which instead operated on RGB-D data.

Gupta *et al.* [5,10] proposed an algorithm for object categorization and 6D pose selection from RGB-D data. It used a CNN to detect objects using depth data converted to a geocentric coordinate system and RGB edge data, then a CNN to estimate object pose using normal vectors derived from the depth data, and finally the iterative closest point (ICP) algorithm to refine the pose. Interestingly, they demonstrated that their pose estimation CNN performed better when trained on synthetic data generated using generic object models than when trained on real data. Their algorithm successfully identified multiple categories of object in a single RGB-D image and produced 6D pose estimates. It was tested on a number of object categories and performed well on most of them. However, detection performance on geometrically simple objects such as desks (separate from tables) and boxes was poor, and this carried through to the pose estimation. Similarly to this work, they partially trained their algorithm on synthetic data generated using generic models and produced full 6D pose estimates for multiple instances of multiple categories for a single image. However, they relied on RGB data for object categorization.

The previously discussed algorithms adopted a local approach processing single objects. Guo *et al.* [19] examined the global scene instead, but with a different goal. Rather than performing object categorization and finding poses, Guo *et al.* estimated a complete 3D geometric representation of a scene including occluded regions, but did not attempt to decompose the scene into individual objects with specific categories and poses. Instead, generic objects were selected from a library irrespective of object category, and positioned such that the collection of objects matched the depth and appearance data and minimized object overlap in order to provide an estimate of the *occluded* scene geometry. Their use of global scene optimization and object overlap is similar to this work, but they also use RGB appearance data in their optimization function.

Finally, there are works that explore physical relations. Zheng *et al.* [30] proposed a method of forming 3D volumes from RGB-D images by projecting onto a voxel representation and executing a physical simulation to calculate parts that must be connected for the scene to be stable. Contacts between objects, and physical stability in general, is a property that could be used to extend the joint parsing stage of the algorithm in future work. Jia *et al.* [31] also examined supports and stability in RGB-D images, but used 3D blocks to represent objects rather than using a voxel representation. 3D blocks may not be an appropriate representation for some categories of object.

3 Method

The algorithm presented here consists of two stages: estimating a number of pose candidates for each object category, and selecting a globally consistent subset of those pose candidates. Fig. 1 illustrates this process.

Estimating Pose Candidates

The task of estimating pose candidates for each category was adapted from the approach of Brachmann *et al.* [4] in that a decision forest is used to estimate category probabilities for each pixel in each available depth image. However, the situation in this work differs significantly from that tackled by Brachmann *et al.* in several ways:

- They assumed a single view of a scene, whilst we consider one or sparse multiple views of a scene acquired from very different viewpoints.
- They used RGB-D data, whilst we use only depth data.
- They used detailed RGB-D information of the exact objects present in the scenes to create object models, whilst we use generic object models which do not correspond exactly to the objects imaged in terms of shape or general size, and not at all in terms of color.
- They searched for instances of a specific object category in each scene, whilst we do not assume knowledge of which categories are present in any scene.
- They searched for individual objects in isolation, whilst we search for all objects present in the scene.
- They used annotated data for training, whilst we have none.

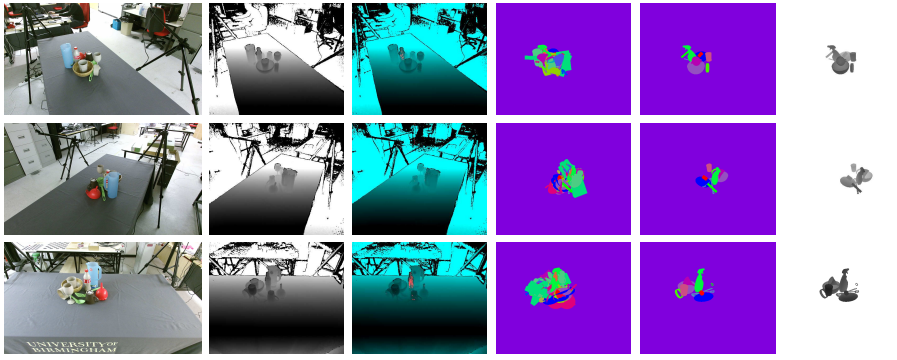

Fig. 1. The processing stages of the algorithm shown for the first scene containing 10 objects. The algorithm calculates pixel probabilities for each object category **(b)** using depth images acquired from multiple views **(a)**, generates a number of pose candidates for each category using the pixel probabilities and depth images **(c)**, and then selects a subset of pose candidates to match the depth images and pixel probabilities and to minimise the physical overlap between objects **(d)**. The rows correspond to different camera views.

In order to use all viewpoints we modify each decision forest to make decisions based upon all of the available views of the scene, and train a separate decision forest for each view but which uses features constructed from all views. Using multiple views enables truly 3D features to be learnt. Training these features requires the relative positions and orientations of the cameras to be known. This limits the general applicability of the multi-view features, but there are a number of cases where fixed cameras are standard, such as industrial environments and security cameras. A potential alternative approach would be to use features derived from point clouds combined from all views, but the use of the structure in two dimensional depth images provide computational advantages over unstructured three dimensional point clouds. The computational burden of training a decision forest for a view increases linearly with the number of views available, and a decision forest must be trained for each available view.

We train the decision forests by generating synthetic depth data for all of the views for random subsets of the generic object models placed in random positions in the scene. Because of the generic object models, lack of RGB data, and object symmetry, we cannot use the decision forests to estimate an object pose directly with any reliability. Instead we use the decision forests to segment pixels belonging to an object fragment in all of the available views, and convert the pixel depths into points in 3D space. A fitting process then estimates the 6D pose for the generic object model associated with the object fragment points.

Selecting a Globally Consistent Subset of Pose Candidates

The first stage of the approach produces a set of candidate object category and pose pairs from detected object fragments. This set should contain the set of correct poses,

but the majority of the set candidates will be false. The second stage of the approach is to take the set of pose candidates and reduce the set to the true set of pose candidates or at a minimum to a globally consistent set of pose candidates. A globally consistent set of pose candidates is one that is consistent with the depth data and pixel category probabilities, and includes no or minimal volume overlap between objects.

Use of this information can enable discrimination between pose candidates for which the object is expected to be significantly occluded. The consistency requirements with respect to depth data and pixel category probabilities account for positive visibility (parts of the pose candidates that *should* be visible), and also negative visibility (parts of the pose candidates that *should not* be visible). The requirements that pose candidate volumes do not overlap and that the set of pose candidates and environment match well with the visual data in general can act as a natural constraint on solution complexity (subset size), particularly in the case where multiple camera views can minimize the occluded region. However, an artificial constraint on complexity could potentially be added for single view scenes.

The joint parsing algorithm combines object and background depth discrepancy, pixel category probabilities, and object volume overlap into an energy function which converts the task into an optimization problem. This is a combinatorial optimization problem as solutions to this problem are sets of objects. It is not practical to evaluate all of the possible subsets because there is a computational cost to evaluating the energy function, and the number of possible subsets also grows with the size of the full pose candidate set. Consequently it is necessary to find an approximately or locally optimal solution. A genetic algorithm (GA) [32] is used to find these solutions.

GAs can be attractive optimization algorithms because they are not necessarily greedy and so can avoid local optima, and because the nature of the crossover operation used to generate new solutions can preserve subsets of the existing solution. This can be beneficial because the relations between objects in a scene are likely to be local to a large extent. For example, three objects in close proximity are likely to largely determine each other's visibility. Removing one of those objects would then greatly modify the effect of the remaining two objects on the energy function, but not necessary objects further away. Therefore, once this subset of three objects have been identified, the subset should ideally persist as a single unit, so that the subset members are either all present in a solution or none are present. Doing so would simplify the optimization problem. An appropriately designed GA crossover operation can achieve this to an extent.

At the completion of the GA, a pose candidate subset with a locally minimum observed energy function value is retained, regardless of the iteration of the algorithm at which it was encountered. This is treated as the final output of the approach. See Fig.1 for an example of the stages of the process.

4 Algorithm Implementation

The decision forest (DF) features were modified so that a pixel in the camera view associated with the DF defines the center of bounding regions in all camera views, and individual features can arise from sub-features within these regions in any of the camera views. The DF training process was modified to maximize information gain for

sub-features in any of the views. The features for each decision tree were trained from a set of 500 synthetic scenes containing random subsets of the generic object models. The models were placed upright in the scene in a manner that ensured that there would be no physical overlap between objects. To prevent overfitting of the probability distributions in the leaves of the trees, the probability distributions of the leaves were estimated using a different set of 500 random synthetic scenes once the decision tree features had been learnt through the initial training.

The lack of RGB information and generic object models prevents the use of the DF for initial pose estimation as performed in the Brachmann *et al.* algorithm [4]. Instead the pixel category probabilities and bounding regions in each view (for a given chosen central pixel) were used to extract probable 3D points in space to which a uniformly sampled version of the generic object mesh was registered using the ICP algorithm. Central pixels were sampled from the pixel category probability images for all views. In each scene a total of 200 pose candidates were sampled for each object category. Each pose candidate was generated with a random fixed size and a random pose which was optimized using the ICP algorithm. The 5 best pose candidates for each category were added to a pose candidate set and presented to the joint parsing portion of the algorithm.

The joint parsing energy function used to quantify the global consistency of a pose candidate subset incorporates depth images discrepancies, pixel-based category probabilities, and object percentage volume overlap. The energy function itself is of the following form:

$$E(D,P,V;\tilde{D},d_{max}) = \frac{1}{N_{pixels}} \sum_{i=1}^{N_{pixels}} \frac{1}{d_{max}} \min(\left|d_i - \tilde{d_i}\right|, d_{max}) + \frac{1}{N_{pixels}} \sum_{i=1}^{N_{pixels}} (1-p_i) + \frac{1}{N_{objects}} \sum_{i=1}^{N_{objects}} v_i \quad (1)$$

Where $\tilde{D}$ is the set of measured pixel depths of the true scene, D is the set of calculated pixel depths of the current pose candidate subset, d_{max} is a maximum limit on the pixel depth discrepancy, P is the set of pixel category probabilities corresponding to the current pose candidate subset and background, V is the set of of fractional volume overlaps calculated for each pose candidate based of the volume fraction overlapping with any of the other pose candidates within the current subset, N_{pixels} is the number of pixels, and $N_{objects}$ is the number of pose candidates within the current subset. The first term in Eq.1 calculates the average pixel depth discrepancy. The discrepancy at each pixel is limited to a maximum value of d_{max} (chosen to be 100mm in this case) as distances larger than this are likely to result from incorrect categories or missing objects. In these cases the discrepancies may be large, but small changes in their magnitudes should not influence the optimization process because the degree of discrepancy between the measured depth and the depth generated from an incorrect pose candidate does not carry any information beyond that the pose candidate is incorrect.

The energy function here is similar to that of Guo *et al.* [19], except that where Guo *et al.* include a RGB-based appearance term we include a categorization probability term. However, the categorization probability term can be thought of as a depth-based local appearance term as it contains the probability that a local depth patch belongs to a certain category.

The optimization process is performed by a GA which modifies a population of 100 pose candidate subsets in an iterative manner. The population is initialized randomly so that each individual within the population contains an average of 10 pose candidates. At each iteration, the energy function value is calculated for each individual and the top 20% of individuals are retained in the new population. A further 20% of individuals are randomly retained in the new population via sampling, with a probability proportional to the inverse of their energy function values. 10% of the new population is generated entirely randomly. Finally, 50% of the new population is generated from the old population. Roulette selection using the energy function values is used to select two individuals from the old population to generate each member of the new population. The two individuals are combined using random three point crossover to form a new individual and then this individual is randomly mutated so that each pose candidate is added or removed with a given probability. This probability is initially high to encourage global search, and is reduced at each iteration.

5 The HOOC Data Set

The presented algorithm was tested on the highly occluded object challenge (HOOC) data set [33]. Each scene within the HOOC data set consists of three views of a table on which a number of objects are placed. The objects are placed on the table in close proximity to one another, including in some cases being stacked, which results in significant object occlusion. See Table 1 for quantitative measures of object visibility. These objects originate from a number of categories, and generic object models are available for each category. See Fig. 2 for some examples. Depth images and depth camera calibration data are provided for each of the views. RGB images are also provided, but these are not registered with the depth images so are for reference only and cannot be used within the approach. Human annotated ground truth categories and poses are provided for the test scenes, but no annotated training data is provided. As such this data set is uniquely complex and presents a great challenge. We contacted the creators of the data set and were provided with the data set and with the ground truth category and pose annotations.

Fig. 2. Generic object models. From left to right: teacup, frying pan, banana, ball, spatula, tube, Sellotape core, stapler, glass.

Object	Fractional pixel visibility			Fractional surface area visibility			
	1	2	3	1	2	3	Total
Teacup	0.86	1.00	0.69	0.18	0.36	0.20	0.49
Frying pan	0.55	0.52	0.64	0.24	0.21	0.29	0.47
Banana	0.50	0.57	0.76	0.24	0.26	0.33	0.47
Ball	0.89	0.32	1.00	0.40	0.17	0.47	0.68
Spatula	0.73	0.55	0.82	0.36	0.29	0.40	0.54
Tube	0.92	0.72	0.75	0.24	0.19	0.20	0.49
Sellotape core	1.00	0.96	1.00	0.35	0.45	0.47	0.71
Stapler	0.45	0.43	1.00	0.12	0.12	0.26	0.39
Glass	0.69	0.72	0.76	0.22	0.27	0.26	0.61

Table 1. Examples of object visibility for each camera in the first scene containing 20 objects in the HOOC data set. The fractional pixel visibility is the fraction of the total image pixels associated with an object in isolation that are still visible in the presence of other objects. The fractional surface area visibility is the fraction of the total surface area of an object visible to the camera in the presence of other objects. The total fractional surface area visibility is the fraction of the total surface area of an object visible to at least one camera.

In detail, the HOOC data set contains 5 scenes each containing 10 objects on a table, and 4 scenes each containing 20 objects on a table. Every object within a given scene is of a different category, and there are a total of 25 object categories. A generic object model is provided for each of the 25 categories. Three Kinect II cameras are placed approximately at the vertices of an equilateral triangle with the scene in the center. Intrinsic and extrinsic calibration data is provided for each depth camera, but not for each RGB camera. The depth images contain significant spherical distortion which primarily affects the background, and the calibration data was used to correct each of the depth images using Bouguet's camera calibration toolbox [34].

A number of properties of the HOOC data set complicate the process of scene understanding. Firstly, a number of the generic object models match poorly with the associated real object. Small objects can be lost within the noise of the depth sensor, which is significant. Some objects, such as the jug, are translucent, and the recorded depth contains large systematic error that is different for each camera view. Many objects are partially occluded in each scene. The presence of three camera views reduces occlusion but does not remove it entirely.

Fig. 3 shows an example from the jug category. This category exhibits many of the challenges associated with the HOOC data set. The jug is translucent, and this drastically affects the quality of the resulting depth data. The point cloud image shows the 3D points associated with the jug from the three camera views as seen from above. The points should form an approximate cylinder, but in practice do not resemble a cylinder due to the interactions of the depth camera light source with the translucent object. The generic mesh of the jug has a different width to height ratio and the shape, position and size of the handle is different. The generic mesh also contains a top surface that is not present in the actual object. Once processed to be closed, the mesh diverges further from the true object.

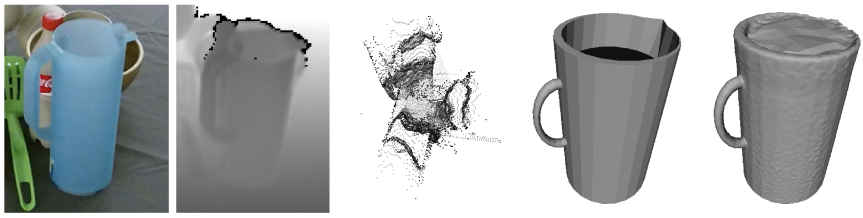

Fig. 3. The jug category. From left to right, the images show a RGB image of the jug, a depth image of the jug, the 3D points associated with the jug in three registered camera views as viewed from above, the generic mesh of the jug, and a closed version of the generic mesh of the jug.

The generic object models for the dustpan and tray object categories were incompatible with object overlap calculation and so the dustpan and tray categories were excluded from the experiments, leaving a total of 23 object categories.

6 Experiments

The approach was applied to the scenes in the HOOC data set. The algorithm performance was tested using the full set of pose candidates generated from the DF. This consisted of 5 pose candidates for each of the remaining 23 categories (excluding the dustpan and tray categories), resulting in a total of 115 pose candidates. The joint parsing performance was separately tested specifically by modifying the full set of pose candidates by replacing the worst pose candidate of each ground truth category with the ground truth pose candidates themselves, resulting in a set of 115 pose candidates containing 10 or 20 ground truth poses depending on the scene.

The HOOC data set is uniquely complex and challenging. The combination of depth data without RGB data, noisy imaging using Kinect II, an absence of human category and pose annotations for training, three viewpoints, generic object models, and occluded objects makes the challenge extremely difficult. Algorithm performance must be considered in that context. These unique properties of the HOOC data set also mean that most of the existing state-of-the-art algorithms for simultaneous object categorization and pose estimation are incompatible with the HOOC data set and so cannot be directly compared with the presented approach.

Table 2 shows the categorization performance of the algorithm. The algorithm identifies instances of 44% of the ground truth labels in the scene on average. When the pose for the label is required to be in spatial proximity of the ground truth pose (within a translation distance of half of the ground truth object size), the accuracy reduces to 29%. Table 3 shows the categorization performance of the algorithm when using pose candidate sets modified to contain the ground truth labels. This tests the ability of the joint parsing algorithm and energy function to choose a good subset of pose candidates. Here the categorization performance rises to 62% without requiring locality, and rises to 56% when requiring locality. These results suggest that the initial pose candidate

generation process is a major limitation on performance. Fig. 4 shows the output of the algorithm for the first scene involving 20 objects. Pose rotation error is not shown because many of the objects in the HOOC data set contain symmetries and the concept of rotation error in this context is unclear.

	Scene								
	10 (1)	10 (2)	10 (3)	10 (4)	10 (5)	20 (1)	20 (2)	20 (3)	20 (4)
Ground truth	0.40	0.50	0.33	0.50	0.33	0.47	0.56	0.44	0.44
Localized ground truth	0.20	0.50	0.22	0.50	0.22	0.32	0.22	0.28	0.17
False categories	5	8	2	5	7	1	1	2	2

Table 2. Categorization performance on the HOOC data set. The fraction of the ground truth categories found is listed, as is the fraction of ground truth categories where the ground truth pose center differs from the estimated pose center by less than half of the generic model size. Finally, the number of categories found that do not belong to set of ground truth categories is listed. The localized ground truth fraction is the most important quantity, and the ideal value is 1.

	Scene								
	10 (1)	10 (2)	10 (3)	10 (4)	10 (5)	20 (1)	20 (2)	20 (3)	20 (4)
Ground truth	0.50	0.88	0.44	0.60	0.56	0.63	0.67	0.67	0.67
Localized ground truth	0.30	0.88	0.44	0.60	0.56	0.58	0.56	0.61	0.56
False categories	4	5	2	6	5	0	3	2	1

Table 3. Categorization performance on the HOOC data set when the ground truth poses are added to the input set of the joint parsing algorithm. The fraction of the ground truth categories found is listed, as is the fraction of ground truth categories where the ground truth pose center differs from the estimated pose center by less than half of the generic model size. Finally, the number of categories found that do not belong to set of ground truth categories is listed. The localized ground truth fraction is the most important quantity, and the ideal value is 1.

Table 4 shows the values of the energy function and of its components for the selected pose candidate subset in the case where ground truth poses are not available and in the case where the ground truth poses are available, and for the ground truth pose set itself, in that order. The overall energy function value favors the ground truth poses and the best subset with the option to choose ground truth poses. The depth component of the energy function favors the ground truth poses. The probability and overlap components both favor the best subset without the option to choose ground truth poses. The overall energy function value suggests that subsets in the case where ground truth poses are available perform similarly to the set of ground truth poses themselves, but both sets consistently outperform the set of poses where ground truth poses are not available.

Clear patterns are observed in each of the components: depth discrepancy, pixel category probability, and object overlap. As might be expected, the depth component con-

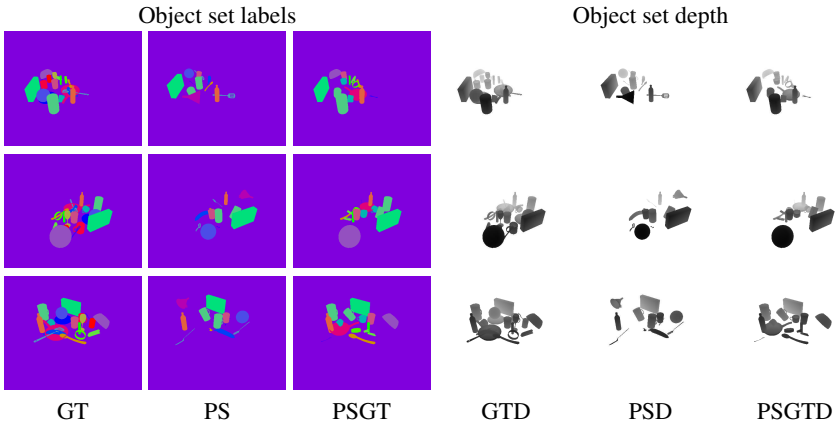

Fig. 4. The output of the algorithm for scene 20(1). From left to right: ground truth object category labels **(GT)**, pose candidate subset labels **(PS)**, pose candidate subset with ground truth labels **(PSGT)**, ground truth object depth **(GTD)**, pose candidate subset depth **(PSD)**, and pose candidate subset with ground truth depth **(PSGTD)**. The rows correspond to different camera views.

sistently favors the set of ground truth poses. However, this is not the case for the both the pixel category probability and object overlap components, where the set of poses where ground truth poses are not available outperforms the other two. This suggests that discrepancy between the generic models and actual objects may be significant. The probability component results from similarity between local depth features in the scene and local depth features on the pose candidates, which suggests that the algorithm is better able to match these features by using poses other than the ground truth poses, and given the categorization results (see Table 2) other categories entirely. The object overlap component supports this interpretation as the ground truth pose set all have significant overlap. Object overlap may cause difficulty for a human annotator, as the annotator is concerned with visible discrepancies, while object overlaps are not obvious to the eye.

In this context, improvements to the DF stage are likely to provide the greatest improvement to general algorithm performance. However, the discrepancy between generic object models and actual objects would continue to affect performance. A potential solution would be to randomly locally perturb the geometry of the generic object models when generating synthetic training data to increase variety. The use of a technique such as deformable models [15] and a priori statistical information of category shape variation would allow this to be performed in a more principled manner. Alternatively, additional generic models from an external source could be included in the training process. The integration of a physics engine into the synthetic training data generation process would allow the production of a wider variety of physically consistent pose sets, which may also increase performance.

Scene	Energy function	Depth component	Probability component	Overlap component
10 (1)	1.4007, 1.3973, **1.3944**	0.6199, 0.6094, **0.5994**	**0.7798**, 0.7877, 0.7901	0.0011, **0.0002**, 0.0050
10 (2)	1.3611, **1.3558**, 1.3559	0.5835, 0.5799, **0.5744**	**0.7751**, 0.7751, 0.7799	0.0025, **0.0008**, 0.0015
10 (3)	1.3772, 1.3586, **1.3550**	0.6000, 0.5702, **0.5670**	**0.7768**, 0.7852, 0.7838	**0.0003**, 0.0032, 0.0042
10 (4)	1.3417, 1.3321, **1.2913**	0.5588, 0.5360, **0.5124**	**0.7749**, 0.7822, 0.7789	0.0080, 0.0139, **0.0000**
10 (5)	1.3919, **1.3795**, 1.3842	0.6025, **0.5787**, 0.5827	**0.7876**, 0.7988, 0.7976	**0.0017**, 0.0020, 0.0039
20 (1)	1.3598, 1.3392, **1.3348**	0.5961, 0.5503, **0.5236**	**0.7637**, 0.7820, 0.7906	**0.0000**, 0.0069, 0.0207
20 (2)	1.3562, **1.3399**, 1.3875	0.5663, 0.5411, **0.5207**	**0.7899**, 0.7971, 0.8053	**0.0000**, 0.0018, 0.0615
20 (3)	1.3777, **1.3354**, 1.3459	0.5976, 0.5347, **0.5341**	**0.7801**, 0.7968, 0.8023	**0.0000**, 0.0039, 0.0094
20 (4)	1.3664, **1.3289**, 1.3320	0.5902, 0.5453, **0.5298**	0.7759, **0.7741**, 0.7835	**0.0004**, 0.0095, 0.0186

Table 4. Energy function values for chosen subsets. Each triple of numbers is in the order: the best subset without the option to choose ground truth poses, the best subset with the option to choose ground truth poses, and the set consisting of all ground truth poses. In all cases, lower values indicate a better solution, and the number in bold is the best of the triple.

An alternative direction which could be pursued in addition to adding shape variety would be to include additional sources of information. For example, the jug is difficult to image because its translucency results in different systematic error in each view. However, the 2D silhouette of the object is unaffected, and could be used. Additionally physical contacts could be included, but this would also be affected by generic model inaccuracy.

7 Conclusion

In this work we have demonstrated that it is possible to an extent to categorize sets of objects and estimate poses in scenes with occlusion where there are no annotated images and only generic object models are available. This was performed by estimating pixel category probabilities using local depth features and generating a large number of pose candidates according to those probabilities. The set of pose candidates was then reduced to a final subset that was consistent with the measured depth and pixel category probabilities, and contained a minimum of overlap between objects. However, there is much room for improvement. Future work would include incorporation of physical contacts into the joint parsing algorithm, and improvements to the DF technique to generate superior initial pose candidates. Replacement of the DF with a Hough Forest [28,29] or CNN may be appropriate. Further, ideally the two stages of the algorithm would be combined to simultaneously optimize categorizations, poses, and pose sets.

Acknowledgements

We acknowledge MoD/Dstl and EPSRC for providing the grant to support the UK academics' involvement in a Department of Defense funded MURI project through EPSRC grant EP/N019415/1. This work was also supported in part by EU H2020 RoMaNS, 645582.

References

1. Hejrati, M., Ramanan, D.: Analyzing 3d objects in cluttered images. In: Advances in Neural Information Processing Systems. (2012) 593–601 1, 2
2. Lim, J.J., Pirsiavash, H., Torralba, A.: Parsing ikea objects: Fine pose estimation. In: Proceedings of the IEEE International Conference on Computer Vision, IEEE (2013) 2992–2999 1
3. Yoruk, E., Vidal, R.: Efficient object localization and pose estimation with 3d wireframe models. In: Proceedings of the IEEE International Conference on Computer Vision Workshops. (2013) 538–545 1
4. Brachmann, E., Krull, A., Michel, F., Gumhold, S., Shotton, J., Rother, C.: Learning 6d object pose estimation using 3d object coordinates. In: European Conference on Computer Vision. Springer (2014) 536–551 1, 2, 3, 4, 5, 8
5. Gupta, S., Girshick, R., Arbeláez, P., Malik, J.: Learning rich features from rgb-d images for object detection and segmentation. In: European Conference on Computer Vision. Springer (2014) 345–360 1, 2, 4
6. Peng, X., Sun, B., Ali, K., Saenko, K.: Learning deep object detectors from 3d models. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 1278–1286 1, 2
7. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. (2012) 1097–1105 1
8. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 3431–3440 1
9. Zhang, N., Donahue, J., Girshick, R., Darrell, T.: Part-based r-cnns for fine-grained category detection. In: European Conference on Computer Vision. Springer (2014) 834–849 1
10. Gupta, S., Arbeláez, P., Girshick, R., Malik, J.: Aligning 3d models to rgb-d images of cluttered scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 4731–4740 1, 2, 4
11. Song, S., Xiao, J.: Deep sliding shapes for amodal 3d object detection in RGB-D images. CoRR **abs/1511.02300** (2015) 1, 2, 4
12. Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., Cook, M., Moore, R.: Real-time human pose recognition in parts from single depth images. Communications of the ACM **56**(1) (2013) 116–124 1
13. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. (2014) 1725–1732 1
14. Pepik, B., Stark, M., Gehler, P., Schiele, B.: Multi-view and 3d deformable part models. IEEE Transactions on Pattern Analysis and Machine Intelligence **37**(11) (2015) 2232–2245 2
15. Zia, M.Z., Stark, M., Schiele, B., Schindler, K.: Detailed 3d representations for object recognition and modeling. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(11) (2013) 2608–2623 2, 13
16. Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., Navab, N.: Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In: Asian Conference on Computer Vision. Springer (2013) 548–562 2, 3
17. Krull, A., Brachmann, E., Michel, F., Ying Yang, M., Gumhold, S., Rother, C.: Learning analysis-by-synthesis for 6d pose estimation in rgb-d images. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 954–962 2, 4

18. Song, S., Xiao, J.: Sliding shapes for 3d object detection in depth images. In: European Conference on Computer Vision. Springer (2014) 634–651 2, 4
19. Guo, R., Zou, C., Hoiem, D.: Predicting complete 3d models of indoor scenes. CoRR **abs/1504.02437** (2015) 2, 5, 9
20. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceedings of the IEEE International Conference on Computer Vision. Volume 2., Ieee (1999) 1150–1157 3
21. Martinez, M., Collet, A., Srinivasa, S.S.: Moped: A scalable and low latency object recognition and pose estimation system. In: IEEE International Conference on Robotics and Automation, IEEE (2010) 2043–2049 3
22. Wohlhart, P., Lepetit, V.: Learning descriptors for object recognition and 3d pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 3109–3118 3
23. Crivellaro, A., Rad, M., Verdie, Y., Moo Yi, K., Fua, P., Lepetit, V.: A novel representation of parts for accurate 3d object detection and tracking in monocular images. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 4391–4399 3
24. Drost, B., Ulrich, M., Navab, N., Ilic, S.: Model globally, match locally: Efficient and robust 3d object recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Volume 1. (2010) 5 3
25. Holzer, S., Shotton, J., Kohli, P.: Learning to efficiently detect repeatable interest points in depth data. In: European Conference on Computer Vision. Springer (2012) 200–213 3
26. Hodaň, T., Zabulis, X., Lourakis, M., Obdržálek, Š., Matas, J.: Detection and fine 3d pose estimation of texture-less objects in rgb-d images. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE (2015) 4421–4428 3
27. Sun, M., Bradski, G., Xu, B.X., Savarese, S.: Depth-encoded hough voting for joint object detection and shape recovery. In: European Conference on Computer Vision, Springer (2010) 658–671 4
28. Tejani, A., Tang, D., Kouskouridas, R., Kim, T.K.: Latent-class hough forests for 3d object detection and pose estimation. In: European Conference on Computer Vision, Springer (2014) 462–477 4, 14
29. Doumanoglou, A., Kouskouridas, R., Malassiotis, S., Kim, T.K.: Recovering 6D Object Pose and Predicting Next-Best-View in the Crowd. ArXiv e-prints (December 2015) 4, 14
30. Zheng, B., Zhao, Y., Yu, J., Ikeuchi, K., Zhu, S.C.: Scene understanding by reasoning stability and safety. International Journal of Computer Vision **112**(2) (2015) 221–238 5
31. Jia, Z., Gallagher, A., Saxena, A., Chen, T.: 3d-based reasoning with blocks, support, and stability. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2013) 1–8 5
32. Fogel, D.B.: An introduction to simulated evolutionary optimization. IEEE transactions on neural networks **5**(1) (1994) 3–14 7
33. Walas, K., Leonardis, A.: Uob highly occluded object challenge (uob-hooc). http://www.cs.bham.ac.uk/research/projects/uob-hooc/ Accessed: 2016-03-11. 9
34. Bouguet, J.Y.: Camera calibration toolbox for matlab. (2004) 10