A Global Hypotheses Verification Method for 3D Object Recognition

Aitor Aldoma[1], Federico Tombari[2], Luigi Di Stefano[2], and Markus Vincze[1]

[1] Vision4Robotics Group, ACIN, Vienna University of Technology
[2] Computer Vision Lab., DEIS, University of Bologna
{aa,vm}@acin.tuwien.ac.at,
{federico.tombari,luigi.distefano}@unibo.it

Abstract. We propose a novel approach for verifying model *hypotheses* in cluttered and heavily occluded 3D scenes. Instead of verifying one hypothesis at a time, as done by most state-of-the-art 3D object recognition methods, we determine object and pose instances according to a *global optimization* stage based on a cost function which encompasses geometrical cues. Peculiar to our approach is the inherent ability to detect significantly occluded objects without increasing the amount of false positives, so that the operating point of the object recognition algorithm can nicely move toward a higher recall without sacrificing precision. Our approach outperforms state-of-the-art on a challenging dataset including 35 household models obtained with the Kinect sensor, as well as on the standard 3D object recognition benchmark dataset.

1 Introduction

Object recognition has been extensively pursued during the last decade within application scenarios such as image retrieval, robot grasping and manipulation, scene understanding and place recognition, human-robot interaction, localization and mapping. A popular approach to tackle object recognition - especially in robotic and retrieval scenarios - is to deploy 3D data, motivated by its inherently higher effectiveness compared to 2D images in dealing with occlusions and clutter, as well as by the possibility of achieving 6-degree-of-freedom (6DOF) pose estimation of arbitrarily shaped objects. Moreover, the recent advent of low-cost, real-time 3D cameras, such as the Microsoft Kinect and the ASUS Xtion, has turned 3D sensing into an easily affordable technology. Nevertheless, object recognition remains an unsolved task, particularly in challenging real world settings involving texture-less and/or smooth objects, significant occlusions and clutter, different sensing modalities and/or resolutions (i.e. see Fig. 1).

Algorithms for 3D object recognition can be divided between *local* and *global*. Local approaches extract repeatable and distinctive keypoints from the 3D surface of the models in the library and the scene, each being then associated with a 3D descriptor of the local neighborhood [1–5]. Scene and model keypoints are successively matched together via their associated descriptors to attain point-to-point correspondences. Once correspondences are established, they are usually

A. Fitzgibbon et al. (Eds.): ECCV 2012, Part III, LNCS 7574, pp. 511–524, 2012.

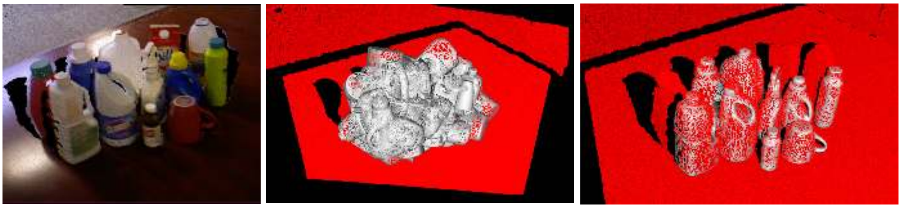


Fig. 1. With the proposed approach, heavily occluded and cluttered scenes (left) are handled by evaluating a high number of hypotheses (center), then retaining only those providing a coherent interpretation of the scene according to a global optimization framework based on geometric cues (right)

clustered together by taking into account geometrical constraints derived from the underlying assumption that model-to-scene transformations are rigid. Clustered correspondences define model *hypotheses*, i.e. subsets of correspondences holding consensus for a specific 6DOF pose instance of a given model in the current scene [1, 3, 6, 5, 7, 8]. Conversely, global methods, e.g., [9, 10], compute a single descriptor which encompasses the whole object shape: this requires, in presence of clutter and/or occlusions, the scene to be pre-processed by a suitable 3D segmentation algorithm capable of extracting the individual object instances.

These 3D pipelines usually comprise an additional final step whereby object hypotheses are further verified so as to reject false detections. However, unlike previous stages, this final *Hypothesis Verification* (HV) step has been relatively unexplored so far, with only a few techniques explicitly addressing it [11, 3, 6, 5]. The most common HV paradigm consists in considering one hypothesis at a time and thresholding a consensus score depending on the amount of scene points explained by the transformed model points. Hence, this paradigm disregards interactions between different hypotheses, this implying the inability to detect strongly occluded objects (scoring low in terms of explained scene points), unless the consensus threshold is kept low resulting in numerous false detections.

This paper proposes a study focused on the HV stage including three main contributions: (i) a careful analysis of geometrical cues to be deployed within the HV stage, taking into account model-to-scene/scene-to-model fitting, occlusions and model-to-model interactions. (ii) A more principled approach to address the HV stage where, instead of considering each model hypothesis individually, we take into account the whole set of hypotheses as a global scene model by formalizing the HV problem in terms of the minimization of a suitable global *cost* function, trying to maximize the number of recognized models while taking into account the aforementioned cues. Due to the computational burden involved in the minimization of such a global *cost* function for relatively large solution spaces, we explore the use of Simulated Annealing, an approximate method, to retrieve accurate solutions within a limited amount of time and computational resources. Finally, (iii) a complete local 3D recognition pipeline to efficiently generate model hypotheses to be validated by the HV stage. By means of experimental results we demonstrate that the proposed approach neatly outperforms state-of-the-art HV algorithms. In this respect, a major advantage provided by the global HV approach deals with the dramatic increase of correct recognitions, in particular those that are "weak", without increasing the number of false positives. Furthermore, our approach brings in a significant reduction of the number of *hard* thresholds required by the recognition pipeline, thus providing a general framework capable to handle different scenarios governed by a few parameters, some easily derivable from the characteristics of the processed data.

2 Related Work

So far only a few methods have outlined a specific proposal for the HV stage. In [1, 11] using the correspondences supporting a hypothesis as seeds, a set of scene points is grown by iteratively including neighboring points which lie closer than a pre-defined distance to the transformed model points. If the final set of points is larger than a pre-defined fraction of the number of model points (from one fourth to one third of the number of model points), the hypothesis is selected and ICP is selectively run on the attained set of points in order to refine object's pose. Obviously, one disadvantage of such an approach is that it can not handle levels of occlusions higher than 75%.

The HV method proposed in [3] ranks hypotheses based on the quality of supporting correspondences, so that they are then verified sequentially starting from the highest rank. To verify each hypothesis, after ICP, two terms are evaluated: the former, similarly to [1], is the ratio between the number of model points having a correspondent in the scene and the total number of model points, the latter is the product between this ratio and the quality score of supporting correspondences. This step requires to set three different thresholds. Two additional checks are then enforced, so as to prune hypotheses based on the number of *outliers* (model points without a correspondent in the scene) as well as on the amount of occlusion generated by the current hypothesis with respect to the remaining scene points. Again, these two additional checks require three thresholds. If an hypothesis gets through each of these steps, it is accepted and its associated scene points are eliminated from the scene, so that they will not be taken into account when the next hypothesis is verified.

In [6], for each model yielding correspondences, the set of hypotheses associated with the model is first pruned by thresholding the number of supporting correspondences. Then, the best hypothesis is chosen based on the overlap area $A(H_{best})$ between the model associated with that hypothesis and the scene, and the initial pose is refined by means of ICP. Finally, the accuracy of the selected hypothesis is given by the ratio $\frac{A(H_{best})}{M_a(H_{best})}$ where $M_a(H_{best})$ is the total visible surface of the model within the bounding box of the scene. The model is said to be present in the scene if its accuracy is above a certain threshold and, upon acceptance, the scene points associated with $A(H_{best})$ are removed.

Papazov and Burschka [5] evaluate how well a model hypothesis fits into the scene by means of an *acceptance function* which takes into account, as a bonus, the number of transformed model points falling within a certain distance from a scene point (*support*) and, as a penalty, the number of model points that would occlude other scene points (i.e. their distance from the closest scene point is above threshold but they lie on the line of sight of a scene point). A hypothesis is accepted by thresholding its support and occlusion sizes. Given the hypotheses fulfilling the requirements set forth by the acceptance function, a conflict graph is built, wherein forks are created every time two hypotheses share a percentage of scene points above a -third- threshold. Surviving hypotheses are then selected by means of a non-maxima suppression step carried over the graph and based on the acceptance function. This approach is the most similar to ours, as, thanks to the conflict graph, interaction between hypotheses is taken into account. Nevertheless, their method is only partially global, since the first stage of the verification still relies on pruning hypotheses one at a time and a *winner-take-all* strategy is used for conflicting hypotheses.

Relevant to our work but aimed at piecewise surface segmentation on range images, Leonardis et al. proposed in [12] a model selection strategy based on the minimization of a cost function to produce a globally consistent solution. Even though the minimization is formalized in terms of a Quadratic Boolean Problem, the final solution is still attained taking into accounts hypotheses sequentially by means of a *winner-take-all* strategy.

3 Proposed Method

This Section illustrates the proposed HV approach for 3D object recognition. After introducing notation, we analyze the geometrical cues that ought to be taken into account for global hypotheses verification. Then, we illustrate how to formulate the cost function and tackle the optimization problem. Finally, we describe the overall 3D object recognition pipeline.

3.1 Notation, Grounds and Geometrical Cues

We consider a model library consisting of m point clouds, $\mathbf{M} = \{\mathcal{M}_1, \cdots, \mathcal{M}_m\}$, together with a scene point cloud, $\mathcal{S}$. We address the general case of $\mathcal{S}$ containing any number of instances from $\mathcal{M}$ (as well as no instance at all), including the case of multiple instances of the same model. The pose, $\mathcal{T}$, which relates a model to its instance in $\mathcal{S}$ is given by a 6 DOF rigid body transformation (i.e. a 3D rotation and translation). We assume that the previous stages of the 3D pipeline provide a set of n recognition hypotheses $\mathcal{H} = \{h_1, \cdots, h_n\}$, each hypothesis h_i given by the pair $(\mathcal{M}_{h_i}, \mathcal{T}_{h_i})$, with $\mathcal{M}_{h_i} \in \mathbf{M}$ being the model hypothesis and $\mathcal{T}_{h_i}$ the pose hypothesis which relates $\mathcal{M}_{h_i}$ to $\mathcal{S}$.

The goal of the proposed method is to choose an arbitrary (up to n) number of items belonging to $\mathcal{H}$ in order to maximize the number of correct recognitions (TPs) while minimizing the number of wrong recognitions (FPs). Purposely, we determine and minimize a suitable *cost* function defined over the solution space of the HV problem. In particular, we denote a solution as a set of boolean variables $\mathcal{X} = \{x_0, \cdots, x_n\}$ having the same cardinality as $\mathcal{H}$, with each $x_i \in \mathbb{B} = \{0, 1\}$ indicating whether the corresponding hypothesis $h_i \in \mathcal{H}$ is dismissed/validated (i.e. $x_i = 0/1$). Hence, the *cost* function can be expressed as $\mathfrak{F}(\mathcal{X}) : \mathbb{B}^n \to \mathbb{R}$, $\mathbb{B}^n$ being the solution space, of cardinality 2^n.

Occlusions. Given an hypothesis h_i, model parts not visible in the scene due to self-occlusions or occlusions generated by other scene parts should be removed since they cannot provide consensus for h_i. Thus, given an instance of $\mathcal{X}$, for each $x_i = 1$ we compute a modified version of $\mathcal{M}_{h_i}$ by transforming the model according to $\mathcal{T}_{h_i}$ and removing all occluded points. Hereinafter this new point cloud will be denoted as $\mathcal{M}^v_{h_i}$.

Establishing whether a model point is visible or occluded can be done efficiently based on the range image associated to the scene point cloud, possibly generating the range image from the point cloud whenever the former is not available directly. Thus, similarly to [5, 3], a point $p \in \mathcal{M}_{h_i}$ is considered occluded if its back-projection into the rendered range map of the scene falls onto a valid depth point and its depth is bigger than the corresponding one of the model. The same reasoning applies to self-occlusions.

Cues i, ii) Scene Fitting and Model Outliers. Once the set of visible points of a model, $\mathcal{M}^v_{h_i}$, has been calculated, we want to determine whether these points have a correspondent on the scene, i.e. how well they *explain* scene points. This cue was exploited by thresholding scene points based on a fixed distance value in the HV stage proposed in [1, 3, 6, 5]. Here, we want to refine such approach, by measuring how well each visible model point locally fits the scene. Hence, for each $x_i = 1$ we associate to each scene point, p, a weight which estimates how well the point is explained by h_i by measuring the local fitting with respect to its nearest neighbor in $\mathcal{M}^v_{h_i}$, denoted as $\mathcal{N}\left(p, \mathcal{M}^v_{h_i}\right)$:

$$\omega_{h_i}(p) = \delta\left(p, \mathcal{N}\left(p, \mathcal{M}^v_{h_i}\right)\right) \tag{1}$$

Local fitting between two points p and q is measured by function $\delta(p, q)$, which accounts for their distance as well as the local alignment of surfaces, as typically done, e.g., to assess the quality of registration between two surfaces. Referring to the normals at p and q respectively as n_p and n_q, $\delta(p, q)$ is defined as

$$\delta(p,q) = \begin{cases} (-\frac{\|p-q\|_2}{\rho_e} + 1)(n_p \circ n_q), & \|p-q\|_2 \leq \rho_e \\ 0, & elsewhere \end{cases} \tag{2}$$

where $\circ$ is the dot product, which is rounded to 0 whenever negative to avoid negative weights (note that all normals have a consistent orientation based on the position of the sensor). As for the distance weight, so far we have employed a simple linear function truncated to 0 when the distance between p and q gets bigger than a threshold ρ_e, though in principle other choices may be considered, e.g. according to one of the several M-estimators proposed in the literature. Additionally, we point out that the use of weights performs a *soft* thresholding for visible points, which helps in case ρ_e is not chosen properly.

For each solution $\mathcal{X}$, we can associate to each scene point p the sum of all weights related with active hypotheses:

$$\Omega_{\mathcal{X}}(p) = \sum_{i=1}^{n} \omega_{h_i}(p) \cdot x_i \tag{3}$$

Then, a scene point is said to be *explained* by $\mathcal{X}$ if $\Omega_{\mathcal{X}}(p) > 0$, *unexplained* otherwise ($\Omega_{\mathcal{X}}(p) = 0$). Moreover, a point $p \in \mathcal{M}_{h_i}^{v}$ is termed an *outlier* for hypothesis h_i if it is not fitted by any scene point according to (2), it is termed *inlier* otherwise. Hereinafter, we will denote as Φ_{h_i} the set of outliers for hypothesis h_i and as $|\Phi_{h_i}|$ the cardinality of each such a set. In the bottom left of Fig. 2-a) and -b), we provide an example of the classification of model points associated to a solution into outliers and inliers.

The amount of explained scene points and outliers are powerful geometrical cues for evaluating the goodness of a solution $\mathcal{X}$ within a global HV framework. In particular, i) the number of explained scene points should be maximized; and ii) the number of outliers associated with all active hypotheses should be minimized.

Cue iii) Multiple Assignment. An important cue highlighting the existence of incoherent hypotheses within a solution deals with a surface patch in the scene being fitted by multiple models. According to our definitions, this can be exploited by penalizing scene points explained by two or more hypotheses (see Fig. 2-a) and -b) for a graphical explanation). Thus, given a solution $\mathcal{X}$ and a scene point p, we define a function $\Lambda_{\mathcal{X}}(p)$

$$\Lambda_{\mathcal{X}}(p) = \begin{cases} \sum_{i=1}^{n} sgn\left(\omega_{h_i}(p)\right), & \sum_{i=1}^{n} sgn\left(\omega_{h_i}(p)\right) > 1 \\ 0, & elsewhere \end{cases} \tag{4}$$

which counts the number of conflicting hypotheses with respect p according to the definition given in (1). Again, in bottom-right of Fig. 2 -a) and -b), we show an example of scene points explained by either a single or multiple hypotheses.

Hence, another cue for global HV to be enforced through $\Lambda_{\mathcal{X}(p)}$ is that iii) the number of multiple hypothesis assignments to scene points should be minimized.

Cue iv) Clutter. In many application scenarios not all sensed shapes can be fitted with some known objects model. Exceptions might occur, for instance, in some controlled industrial environments where all the objects making up the scene are known *a priori*. More generally, though, several visible scene parts which do not correspond to any model in the library might locally - and erroneously - fit some model shapes, potentially leading to false detections. Maximizing the number of explained scene points (i.e. cue i)), although useful to increase the number of correct recognitions, nevertheless favors this circumstance. On the other hand, computing the outliers associated with these false positives (cue ii)) might not help, since the parts of the model which do not fit the scene might

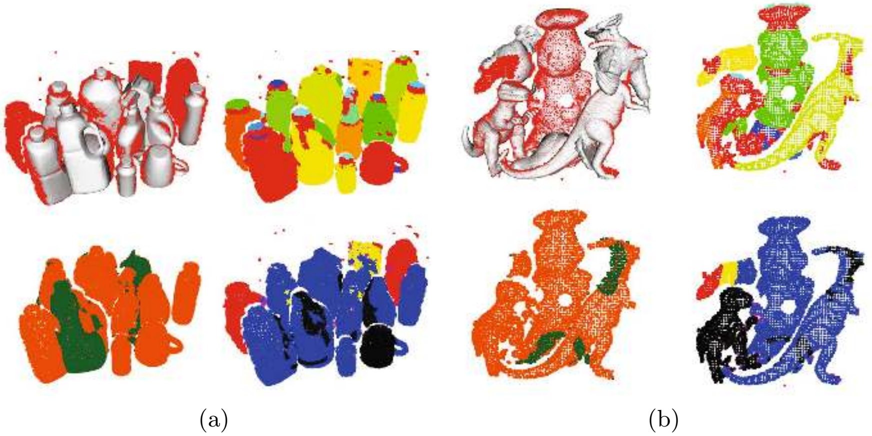


Fig. 2. Proposed cues for global optimization. In both a) and b), top left: a solution consisting of a set of active model hypotheses super-imposed onto the scene. Top right: scene labeling via smooth surface segmentation. Bottom left: classification of visible model points between inliers (orange) and outliers (green). Bottom right: classification of scene points between explained by a single hypothesis (blue), by multiple hypotheses (black), unexplained (red), cluttered due to region growing (yellow) and to proximity (purple).

turn out occluded or outside the field of view of the 3D sensor. This is the case, e.g., of the *chicken* in Fig. 2-b) being wrongly fitted to the *rhino* (in particular, the bottom left image shows that the potential outliers turn out indeed occluded and hence the wrong hypothesis not significantly penalized).

To counterattack the effect of clutter, we devised an approach, inspired by surface-based segmentation [13], aimed at penalizing a hypothesis that locally explains some part of the scene but not nearby points belonging to the same smooth surface patch. This is also useful to penalize hypotheses featuring correct recognition but wrong alignment of the model in the scene. Surface-based segmentation methods are based on the assumption that object surfaces are continuous and smooth. Continuity is usually assessed by density of points in space and smoothness through surface normals. Following this idea, scene segmentation is performed by identifying smooth clusters of points. Each new cluster is initialized with a random point, then it is grown by iteratively including all points p_j lying in its neighborhood which show a similar normal:

$$||p_i - p_j||_2 < t_d \wedge n_i \circ n_j > t_n \tag{5}$$

At the end of the process, each scene point is associated with a unique label $l(p)$. In top right of Fig. 2-a) and -b), we report two examples of scene segmentation.

Hence, given a solution $\mathcal{X}$, likewise in (3), we compute a clutter term, $\Upsilon_{\mathcal{X}}(p)$, at each unexplained scene point p, so as to penalize those that are likely to belong to the same surface as nearby explained points:

$$\Upsilon_{\mathcal{X}}(p) = \sum_{i=1}^{n} x_i \cdot \gamma\left(p, \mathcal{N}\left(p, \mathcal{E}_{h_i}\right)\right) \tag{6}$$

$\Upsilon_{\mathcal{X}}(p)$ consists of contributions, $\gamma\left(p, \mathcal{N}\left(p, \mathcal{E}_{h_i}\right)\right)$, associated with each active hypothesis h_i, where $\mathcal{E}_{h_i}$ is the set scene points explained by h_i. Analogously to $\delta\left(p, q\right)$, we want $\gamma\left(p, q\right)$ to weight clutter based on the proximity of p to its nearest neighbor, $q \in \mathcal{E}_{h_i}$, as well as according to the alignment of their surface patches:

$$\gamma(p,q) = \begin{cases} \kappa, & \|p-q\|_2 \le \rho_c \ \wedge \ l(p) = l(q) \\ (-\frac{\|p-q\|_2^2}{\rho_c^2} + 1)(n_p \circ n_q), & \|p-q\|_2 \le \rho_c \ \wedge \ l(p) \neq l(q) \\ 0, & elsewhere \end{cases} \tag{7}$$

The radius given by ρ_c defines the spatial support related to $\gamma(p,q)$, while κ is a constant parameter used to penalize unexplained points that have been assigned to the same cluster by the smooth segmentation step. Thanks to the above formulation, wrong active hypotheses, such as the milk bottle in Fig. 2-a) and the *chicken* model in Fig. 2-b), cause a significantly valued clutter term $\Upsilon_{\mathcal{X}}$ (e.g. the purple and yellow regions in the bottom right part of the Figure), which will penalize their validation within the global cost function. Therefore, we have derived the last cue: iv) the amount of unexplained scene points close to an active hypothesis according to (7) should be minimized.

3.2 Cost Function

We have so far outlined four cues i)-iv). While i) is aimed at increasing as much as possible the number of recognized model instances (thus TPs and FPs), ii), iii) and iv) try to penalize unlikely hypotheses through geometrical constraints, so as to minimize false detections (FPs). The cost function $\mathfrak{F}$ we are looking for is obtained as the sum of the terms related to the cues that need to be enforced within our optimization framework:

$$\mathfrak{F}\left(\mathcal{X}\right) = f_{\mathcal{S}}\left(\mathcal{X}\right) + \lambda \cdot f_{\mathcal{M}}\left(\mathcal{X}\right) \tag{8}$$

where λ is a regularizer, and $f_{\mathcal{S}}$, $f_{\mathcal{M}}$ account, respectively, for cues defined on scene points and model points:

$$f_{\mathcal{S}}\left(\mathcal{X}\right) = \sum_{p \in \mathcal{S}} \left(\Lambda_{\mathcal{X}}(p) + \Upsilon_{\mathcal{X}}(p) - \Omega_{\mathcal{X}}\left(p\right)\right) \tag{9}$$

$$f_{\mathcal{M}}\left(\mathcal{X}\right) = \sum_{i=1}^{n} \left|\Phi_{h_i}\right| \cdot x_i \tag{10}$$

The global cost formulation in (8) easily allows plugging in additional cost terms derived from geometric constraints or from specific application characteristics. For instance, the relatively common assumption, in indoor robotic scenarios, that objects are placed over a planar surface [9, 10] would allow to penalize hypotheses having object parts lying below - or intersecting with - this surface.

3.3 Optimization

Having defined the cost function, we need to devise a solver for the proposed optimization problem. As previously pointed out, we are looking for a solution $\tilde{\mathcal{X}}$ which minimizes function $\mathfrak{F}(\mathcal{X})$ over the solution space $\mathbb{B}^n$:

$$\tilde{\mathcal{X}} = \underset{\mathcal{X} \in \mathbb{B}^n}{\operatorname{argmin}} \left\{ \mathfrak{F}(\mathcal{X}) = f_{\mathcal{S}}(\mathcal{X}) + \lambda \cdot f_{\mathcal{M}}(\mathcal{X}) \right\} \tag{11}$$

As the cardinality of the solution space is 2^n, even with a relatively small number of recognition hypotheses (e.g. in the order of tens) exhaustive enumeration becomes prohibitive. As the defined problem belongs to the class of non-linear pseudo-boolean optimization, we adopt a classical approach from operation research based on simulated annealing [14] (SA). SA is a meta-heuristic algorithm that optimizes a certain function without the guarantee to find the global optimum. It randomly explores the solution space applying *moves* from a solution $\mathcal{X}^i$ to another valid solution $\mathcal{X}^j$. In order to deal with local optima, the algorithm allows *moves* which increase the cost of the target solution. Such "bad" moves are usually performed during the initial iterations (when the *temperature* of the system is high), whilst they become less and less probable as the optimization goes on (system *cooling down*). The algorithm stops when the temperature has reached a minimum or no improvement has been achieved in the last N *moves*, which occurs either when the algorithm reaches the global optimum or trapped into a local minimum. We initialize SA assuming all hypotheses to be active, i.e. $\mathcal{X}^0 = \{1, \cdots, 1\}$, each move consisting then in switching on/off a specific hypothesis at a time.

In our experiments we relied on the SA implementation available on MetsLib[1] based on linear cooling and on the default parameter values, except for the number of iterations, as we used a high enough value (6000) so that different runs yielded the same results. We wish to point out that the proposed formulation allows the terms included in the cost function to be pre-computed for each single hypothesis (those related to $f_{\mathcal{M}}$) and scene point (those related to $f_{\mathcal{S}}$), so that at each new move the cost function can be computed efficiently, and independently of the total amount of scene points and number of hypotheses, by updating only the hypothesis (and related scene points) being switched on/off. In Sec. 4 we will show how, despite being an approximate optimization algorithm, in the addressed scenarios SA can yield accurate results while requiring reasonably low computation times (typically below 2 seconds for hypothesis sets up to 200 elements – see Fig. 3-a) and Fig. 4-b).

[1] `www.coin-or.org/metslib`

3.4 3D Recognition Pipeline

We briefly present the complete pipeline which will be used for our object recognition experiments. This pipeline is based on local features, although we wish to point out that the proposed algorithm can be straightforwardly plugged into pipelines based on global features or recognition pipelines combining several descriptors with different strengths as long as a 6DOF pose is provided.

Input Data. We assume input data to be either in the form of 3D meshes (2.5D/3D) or point clouds. In most practical scenarios, scenes will be represented by range images obtained from a single viewpoint. To build up the model library, we transform each full 3D model into 2.5D clouds by placing a virtual camera on a tessellated sphere around the mesh and rendering it from the centroids of the triangles building up the tessellated sphere. In our pipeline, a icosahedron is tessellated once, giving a total of 80 camera positions (i.e., similarly to [9]).

Keypoint Detection and Description. Keypoints are extracted at uniformly sampled positions on the surface of models and scene, parameter σ_s being the sampling distance. Then, the SHOT local descriptor [2] is computed at each keypoint over a support size specified by radius σ_d. As for SHOT parameter values, we have used those originally proposed in [2].

Correspondence Matching and Grouping. Descriptors are then matched to attain point-to-point correspondences. To handle the case of multiple instances of the same model, each scene descriptor is matched, via fast indexing [15], against all models descriptors. We explicitly avoid using a matching threshold to reject weak correspondences, given the ad-hoc choice of such thresholds and their strong dependency to the metric being used.

Successively, a Correspondence Grouping (CG) algorithm is run to obtain the model hypotheses to be feed to the proposed global HV process. Our CG approach, inspired by [16], iteratively groups subsets of correspondences based on checking the geometric consistency of pairs of correspondences. In particular, starting from a seed correspondence $c_i = \{p_i^{\mathcal{M}}, p_i^{\mathcal{S}}\}$, $p_i^{\mathcal{M}}$ and $p_i^{\mathcal{S}}$ being, respectively, the model and scene 3D keypoints in c_i, and looping over all correspondences not yet grouped, the correspondence $c_j = \{p_j^{\mathcal{M}}, p_j^{\mathcal{S}}\}$ is added to the group started by c_i if the following relation holds:

$$\left| ||p_i^{\mathcal{M}} - p_j^{\mathcal{M}}||_2 - ||p_i^{\mathcal{S}} - p_j^{\mathcal{S}}||_2 \right| < \varepsilon \tag{12}$$

ε being a parameter of the method, intuitively representing the inlier tolerance for the consensus set. A threshold τ is usually deployed to discard subsets supported by too few correspondences. Given that each subset of correspondences yielded by CG defines a model hypothesis, threshold τ influences the final cardinality of the hypothesis set $\mathcal{H}$ (i.e. n) and, consequently, the computational efficiency of the HV stage. As a further refinement, we run RANSAC on each subset obtained out of the previous stage, the model being the 6DOF transformation provided via Absolute Orientation (AO) [17]. The consensus set tolerance

Table 1. Parameters used for the proposed object recognition pipeline

	Parameters							
	Det./descr.		CG		HV			
Dataset	σ_s	σ_d	τ	ε	ρ_e	ρ_c	κ	λ
Laser Scanner	0.5 cm	4 cm	5	0.5 cm	0.5 cm	3 cm	5	3
Kinect	1 cm	5 cm	5	1.5 cm	1 cm	5 cm	5	3

and minimum cardinality for RANSAC are set, respectively, to ε and τ. Finally, ICP is deployed on each subset to refine the 6DOF pose given by AO. At this point, the hypothesis set $\mathcal{H}$ is ready to be fed to the proposed HV stage.

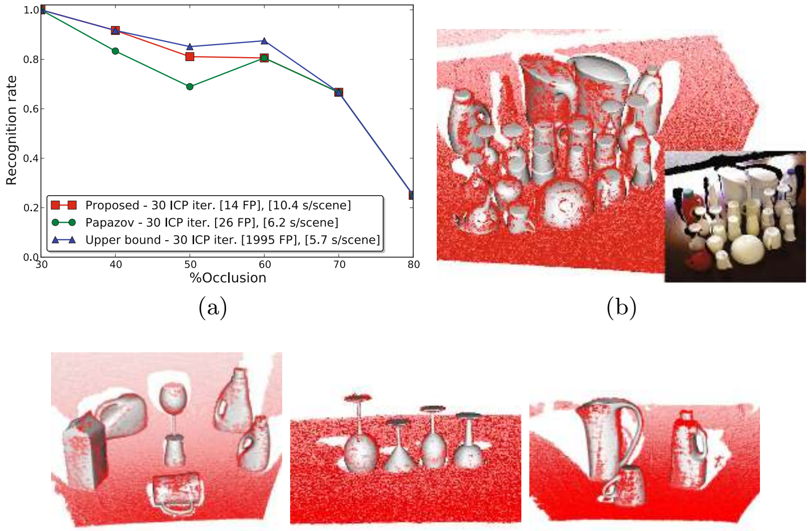
4 Experimental Results

Two experiments were conducted to validate the proposed HV algorithm and the proposed 3D recognition pipeline, as well as to evaluate the suitability of our approach with respect to different kinds of 3D data. The first experiment is performed on a novel dataset, referred to hereinafter as *Kinect*, whereby we match a set of CAD models against scenes acquired with a Kinect sensor. In this experiment we compare our HV algorithm to that proposed in [5], as the latter appears to be currently the best performing HV algorithm (see Fig. 4-a) and is able to handle multiple model occurrences on the scene. For a fair comparison, both algorithms are plugged into exactly the same 3D object recognition pipeline, i.e. that described in Sec. 3.4. Furthermore, to validate the entire 3D object recognition approach proposed in this paper, in the second experiment we evaluate our proposal on the standard benchmark dataset for 3D object recognition presented in [3], which comprises objects acquired by a laser scanner and will be referred to as *Laser Scanner*.

Due to the different nature of the two datasets in terms of scene size, model library size and noise – *Kinect* data is noisier than *Laser Scanner* data, especially as the distance to the camera increases – parameters were slightly tuned to accommodate the algorithms to the underlying data (see Table 1). For the implementation of [5], the same values of ρ_e reported in Table 1 were deployed, while remaining parameters were tuned according to their performance on both datasets: we used a support threshold of 0.08, a penalty threshold of 0.05 and a value of 0.02 to decide when two hypotheses are in conflict. A TP is scored if the *id* of the recognized model matches that in the ground truth and the RMSE computed on the estimated model pose with respect to ground truth is under a certain threshold, otherwise, a FP is scored.

***Kinect* Dataset.** This dataset consists of 50 scenes and a library of 35 models including typical household objects[2]. This dataset is particularly challenging

[2] `http://users.acin.tuwien.ac.at/aaldoma/datasets/ECCV.zip`



(a) (b)

(c)

Fig. 3. *Kinect* dataset: a) results reported by the proposed HV algorithm and that in [5]; the chart also reports the upper bound on the recognition rate related to the deployed 3D pipeline. b-c): qualitative results yielded by the proposed algorithm on a challenging dataset scene (b) and other simpler scenes (c).

given the different nature of the 3D data between scenes (2.5D acquired with a Kinect) and models (fully 3D, mainly CAD, the remaining acquired with a laser scanner). The relatively high number of models present in the library allows testing how well the proposed approach scales with the library size. Another relevant challenge of this dataset is represented by the traits of the models' shape, some of which are highly symmetrical and hardly descriptive (e.g. bowls, cups, glasses), many of them being characterized by a high similarity (e.g. different kinds of glasses, different kinds of mugs, ..), as also vouched by Fig. 3. Fig. 3-a) shows the *Recognition vs Occlusion* curve and number of FPs of the proposed HV algorithm and that in [5], both plugged in on the 3D pipeline presented in Sec. 3.4. The Figure also shows the upper bound curve, i.e. the maximum recognition rate achievable with the proposed recognition pipeline. It can be seen that our approach gets really close to the upper bound while dramatically reducing the number of false positives. Our approach also outperforms that in [5], in terms of Recognition rates as well as in terms of FPs. The overall recognition rate on this dataset is 84.1% for the upper bound, 79.5% for the proposed approach and 73.8% for [5]. Qualitative results obtained by the proposed algorithm are also provided in Fig. 3-b) and 3-c).

***Laser Scanner* Dataset.** This dataset consists of 50 scenes and 5 models, and it can be currently regarded as the most popular benchmark for 3D object recognition. Fig. 4-a) shows the *Recognition vs Occlusion* curve reported by the

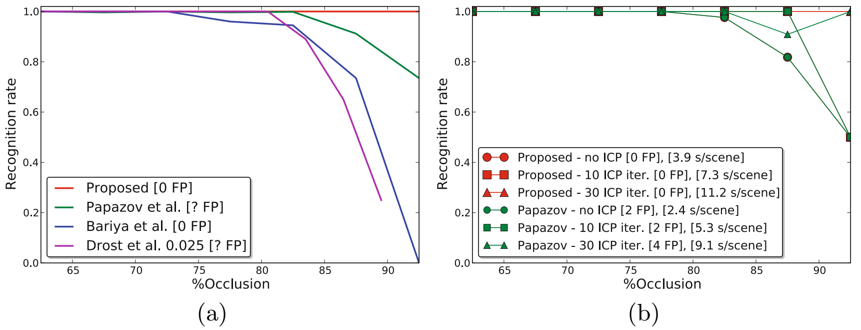


Fig. 4. Evaluation on *Laser Scanner* dataset. a) Comparison of the proposed recognition pipeline and the HV approach against results published in [5, 7, 6]. b) Comparison of the proposed HV stage and our implementation of that in [5] plugged on the proposed 3D pipeline.

proposed 3D pipeline together with the results reported in [5–7]: our method clearly outperforms the state of the art. Moreover, to the best of our knowledge, we are the first to achieve a recognition rate of 100% without any false positive. Fig. 4-b) shows the *Recognition vs Occlusion* curve and number of FPs of the proposed HV stage compared to that proposed in Papazov et al. [5], with the recognition pipeline proposed in Sec. 3.4 for different ICP iterations. The average time required by each combination is also reported. It is worth noting that although the results obtained by both HV algorithms are equivalent at 0 and 10 ICP iterations in terms of recognition rate, the proposed algorithm is able to yield less FPs (0 against 2). Then, at 30 ICP iterations, our method accurately recognizes all models without yielding any FP, while the number of FPs reported by [5] significantly increases due to the fact that a high number of ICP iterations tend to locally align incorrect hypotheses to the scene points, so that their respective support is large enough to be accepted by that method.

5 Conclusions and Future Work

We have proved the effectiveness of the proposed geometrical cues as well as of simultaneously considering the interaction between model hypotheses as to dramatically reduce the number of false positives while preserving those correct recognitions that, due to occlusions, exhibit a small support in the scene. Overall, the potentialities of the HV stage to boost the performance of 3D object recognition pipelines have also been highlighted, which to the authors' opinion have been underrated so far in literature.

Despite the relative efficiency of the proposed method in comparison to other state-of-the-art strategies (see Fig. 4-b)), future work concerns exploiting GPU parallelism for optimizing the main computational bottlenecks of the proposed algorithm, namely the ICP stage and the initial computation of the cost terms during the SA stage. Another direction regards the use of additional cues, particularly with the aim of penalizing physical intersections between active visible models. We plan to publicly release the implementation of the proposed HV algorithm and recognition pipeline in the open source Point Cloud Library.

Acknowledgements. The research leading to these results has received funding from the Doctoral College on Computational Perception at TU Vienna and the Austrian Science Foundation under grant agreement No. I513-N23.

References

1. Johnson, A.E., Hebert, M.: Using spin images for efficient object recognition in cluttered 3d scenes. IEEE Trans. PAMI (5) (1999)
2. Tombari, F., Salti, S., Di Stefano, L.: Unique Signatures of Histograms for Local Surface Description. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part III. LNCS, vol. 6313, pp. 356–369. Springer, Heidelberg (2010)
3. Mian, A., Bennamoun, M., Owens, R.: 3d model-based object recognition and segmentation in cluttered scenes. IEEE Trans. PAMI (10) (2006)
4. Mian, A., Bennamoun, M., Owens, R.: On the repeatability and quality of keypoints for local feature-based 3d object retrieval from cluttered scenes. IJCV (2010)
5. Papazov, C., Burschka, D.: An Efficient RANSAC for 3D Object Recognition in Noisy and Occluded Scenes. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part I. LNCS, vol. 6492, pp. 135–148. Springer, Heidelberg (2011)
6. Bariya, P., Nishino, K.: Scale-hierarchical 3d object recognition in cluttered scenes. In: Proc. CVPR (2010)
7. Drost, B., Ulrich, M., Navab, N., Ilic, S.: Model globally, match locally: Efficient and robust 3d object recognition. In: Proc. CVPR (2010)
8. Tombari, F., Di Stefano, L.: Object recognition in 3d scenes with occlusions and clutter by hough voting. In: Proc. PSIVT (2010)
9. Aldoma, A., Blodow, N., Gossow, D., Gedikli, S., Rusu, R.B., Vincze, M., Bradski, G.: CAD-Model Recognition and 6DOF Pose Estimation Using 3D Cues. In: 3DRR Workshop, ICCV (2011)
10. Rusu, R.B., Bradski, G., Thibaux, R., Hsu, J.: Fast 3d recognition and pose using the viewpoint feature histogram. In: Proc. 23rd IROS (2010)
11. Johnson, A.E., Hebert, M.: Surface matching for object recognition in complex three-dimensional scenes. IVC (9) (1998)
12. Leonardis, A., Gupta, A., Bajcsy, R.: Segmentation of range images as the search for geometric parametric models. IJCV (1995)
13. Rabbani, T., van den Heuvel, F., Vosselmann, G.: Segmentation of point clouds using smoothness constraint. In: IEVM 2006 (2006)
14. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. Science (4598) (1983)
15. Muja, M., Lowe, D.G.: Fast approximate nearest neighbors with automatic algorithm configuration. In: VISAPP. INSTICC Press (2009)
16. Chen, H., Bhanu, B.: 3d free-form object recognition in range images using local surface patches. Pattern Recognition Letters (10) (2007)
17. Arun, K., Huang, T., Blostein, S.: Least-squares fitting of two 3-d point sets. Trans. PAMI (1987)