3D Object Detection and Localization using Multimodal Point Pair Features

Bertram Drost
MVTec Software GmbH
München

drost@mvtec.com

Slobodan Ilic
Department of Computer Science, CAMP,
Technische Universität München (TUM)

slobodan.ilic@in.tum.de

Abstract

Object detection and localization is a crucial step for inspection and manipulation tasks in robotic and industrial applications. We present an object detection and localization scheme for 3D objects that combines intensity and depth data. A novel multimodal, scale- and rotation-invariant feature is used to simultaneously describe the object's silhouette and surface appearance. The object's position is determined by matching scene and model features via a Hough-like local voting scheme. The proposed method is quantitatively and qualitatively evaluated on a large number of real sequences, proving that it is generic and highly robust to occlusions and clutter. Comparisons with state of the art methods demonstrate comparable results and higher robustness with respect to occlusions.

1. Introduction

Detection and localization of 3D objects is a crucial part in many machine vision systems. It is also a fundamental step for subsequent inspection and manipulation tasks. Recent methods [6, 23, 15, 2, 19] that localize objects in range data provide excellent results and can cope with moderate clutter and occlusions. However, they are often sensitive to the occlusion of key regions of the object [19, 2, 23] and have problems detecting objects which are planar, self-similar or similar to background clutter [6, 15]. Object detectors based on template matching that use both intensity and range data [9] have proven to be very fast and highly robust in the presence of clutter, but are sensitive to occlusion and do not recover the full pose of the object.

The use of edges for object detection in intensity images has a long tradition and have proven to be a powerful feature in object detection [17, 8, 20, 10]. By contrast, remarkably little work [21, 19, 11] has been published on using edges or depth discontinuities in range images for such tasks. This is probably due to a number of challenges which are unique to 3D edges, most notably because the range sensors tend to fail exactly at or around the edges. Triangulating recon-
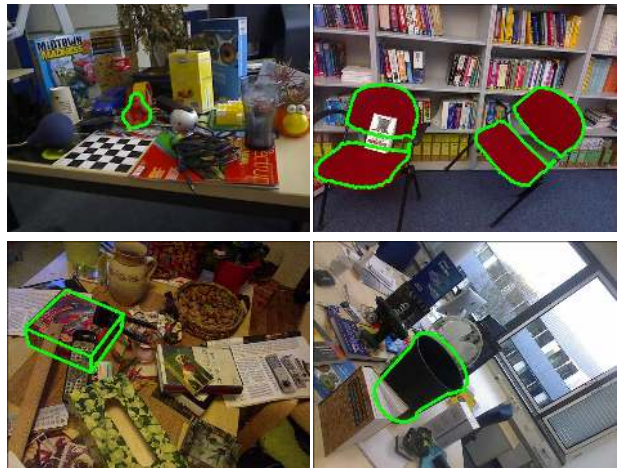

Figure 1. Example localizations for our method for highly cluttered scenes, multiple instances of texture-less objects in arbitrary poses, planar objects and high amounts of occlusion (best viewed in color).

struction methods, such as stereo or structured light, suffer from local occlusion around such edge points. Other methods, such es time-of-flight, tend to smooth over edges and introduce veil points, *i.e.*, points at 3D positions that do not correspond to any real 3D points in the scene. Such effects make it difficult to detect and accurately localize geometric edges and to measure their direction from range images alone. By contrast, edges in intensity images can be detected and measured precisely, but it is generally not possible to distinguish between texture and geometric edges.

We propose a method that combines the accuracy of the intensity edges with the expressiveness of range information. For this, edges are extracted from the intensity image and filtered using the range image to obtain accurate geometric edges. Those geometric edges are combined with the 3D data from the range image to form a novel multimodal feature descriptor that combines intensity and range information in a scale and rotation invariant way. The object's appearance from different viewpoints is described in terms of those features, which are stored in a model database. This

1

model database allows efficient access to similar features and captures the overall appearance of the object. In the online phase, the multimodal features are extracted from the scene and matched against the model database. An efficient local voting scheme is used to group those matches and find the pose that simultaneously maximizes the overlap of the object's silhouette to the detected geometric edges as well as the overlap of the model surface and the 3D surface of the scene. The resulting pose candidates are filtered using clustering and non-maximum suppression, and the final result is refined using iterative closest point (ICP) algorithm to obtain a highly precise pose.

The advantages of the proposed method are numerous: It is able to localize textured and untextured objects of any shape and finds the full 3D pose of the object. Multiple instances of the object can be found with little loss of performance. The method can be trained either using a CAD model of the object or using registered template images taken from different viewpoints. When using templates, one template per viewpoint is sufficient since the method is invariant against scale changes and in-plane rotations. The method also shows high robustness to background clutter and occlusions.

The proposed approach is evaluated both quantitatively and qualitatively and compared against other state-of-the-art methods. It shows comparable results and higher robustness with respect to occlusions. In the remainder of this paper, we will first discuss related works, describe our method and finally present our results before concluding.

2. Related Work

A number of different surface descriptors were proposed to detect free-form 3D objects. The most famous one are arguably spin images [12], where the surface around a reference point is described as a histogram created by rotating a half-plane around the point's normal and intersecting it with the surface. 3D shape contexts and harmonic shape contexts [7] are an extension of 2D shape contexts [3] to three dimensions and represent the surface as a histogram and its harmonic transformation. Tombari *et al.* [23] proposed a framework that combines histogram-based and signature-based descriptor approaches. Rusu *et al.* [18] use a histogram of point pair features as local surface descriptor, which is then used for point cloud registration. Extensive reviews over other descriptors are covered in [4, 14, 16].

Stiene *et al.* [21] proposed a detection method in range images based on silhouettes. They rely on a fast Eigen-CSS method and a supervised learning method. However, their object description is based on a global descriptor of the silhouette and is thus unstable in the case of occlusions. They also require a strong model of the environment which does not generalize well. Steder *et al.* [19] use an edge-based keypoint detector and descriptor to detect objects in

range images. They train the descriptors by capturing the object from all directions and obtain good detection results on complex scenes. Wu *et al.* [24] used a perspective correction based on 3D data similar to the correction presented in this paper.

Hinterstoisser *et al.* [9] proposed a multimodal template matching approach that is able to detect textureless objects in highly cluttered scenes but is sensitive to occlusion and does not recover the 3D pose of the object. Sun *et al.* [22] use multimodal information to simultaneously detect, categorize, and locate an object. However, while working well in many scenarios, both approaches require large training datasets. Lai *et al.* [13] proposed a distance-based approach for object classification and detection in multimodal data and provided a large evaluation dataset. However, they also do not recover the pose of the object and show no results for close-range clutter.

A couple of edge detectors for range images were proposed, such as [19, 11]. However, designing a generic edge detector for range images is a very difficult task. As discussed above, different range sensors, such as time-of-flight cameras, stereo systems, structured light, laser triangulation, depth-from-focus, or photometric stereo exhibit very different characteristics in terms of noise, missing data, occlusion, smoothing and other variables.

Drost *et al.* [6] use a local Hough-like voting scheme that uses pairs of points as features to detect rigid 3D objects in 3D point clouds. The voting scheme optimizes the overlap of object and scene surface. While the method is efficient and general, it has shortcomings when objects appear similar to background clutter since edge information is not taken into account. Our proposed method uses their voting scheme for the localization step, however, using a different edge-based point pair feature.

In this paper, we introduce novel edge based point pair feature and couple it with the local Hough-like voting scheme. We demonstrate that even with very basic edge detection we can obtain very good results and overcome the problems of related works. Compared to the method of Hinterstoisser *et al.* [9], our approach is more robust to occlusions and, thanks to the scale- and rotation-invariant feature descriptor, requires less template images. Additionally, the proposed approach is faster and more robust than method of Drost *et al.* [6] when detecting objects which appear similar to background clutter. This is often the case if the object's surface contains large planar patches: Since the method of [6] does not take the object boundaries or the viewpoint-dependent appearance into account, such objects are often detected in walls or tables. Compared to the method of Steder *et al.* [19], the proposed approach is more robust as it does not require a feature-point detector which relies on object parts with corner-like characteristics. In that sense our proposed method is more generic than the related
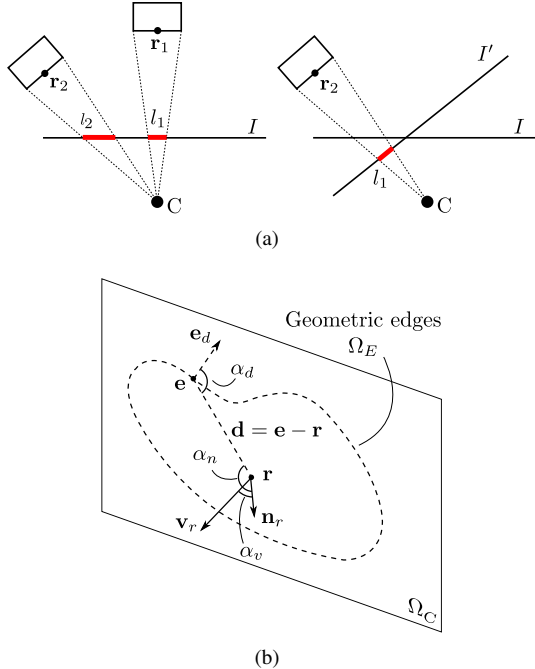
(a)

(b)

Figure 2. (a) *Left*: Even if the object is seen from the same direction and from the same distance, lengths and angles appear distorted in the image plane. *Right*: We apply a perspective correction by reprojecting on a new image plane I' with the same focal distance, but orthogonal to the line of sight to the reference point $\mathbf{r}$. (b) Description of the used feature descriptor. Dashed vectors and lines live in the image plane, while solid vectors are in 3D. Our feature uses the angles α_r, α_n, and α_v as well as the scaled distance $Z(\mathbf{r})|\mathbf{r} - \mathbf{e}|/f$. All four components are invariant against in-plane rotations and scaling.

approaches and works well in many practical applications.

We designed a method that is robust to background clutter, can efficiently handle occlusions and can find multiple instances of the given object of interest. The key element of our approach is a novel multi-modal point pair feature that uses both geometric object edges in the intensity image and depth from the range image. The proposed feature combines the stable information from both modalities, is invariant against scale and rotation changes and has a low dimension that allows fast clustering using hashing. Since it is based on the edge information the overall scene complexity is reduced, making this approach faster than the one of Drost *et al*. that uses all 3D points in the scene for the search.

3. Method

The objective is to detect a given 3D object and determine its pose in an RGBD image. We assume that the 3D model of the object we want to detect and localize is available to us. It can be provided as a man-created 3D CAD model or reconstructed from multiple RGBD images by moving the sensor around the object. The scene in which we search for the object of interest is captured with an RGBD sensor and may contain clutter and occlusions.

In the reminder of this section we will first introduce our novel multi-modal feature and then discuss its use for object model description and then detection and localization.

3.1. Multimodal Feature

Suppose that a 3D CAD model M of the object we want to detect and localize is given to us as a surface with points and normals. The scene in which we search the object of interest is an RGBD image composed of intensity or color image I_C defined for domain Ω_C and range or depth image I_R defined for domain Ω_R. In practice, the range image has to be calibrated so that we know the metric measurement of the scene. Our principal intuition for the creation of our multi-modal feature is that the intensity image provides accurate information at geometric edges where depth sensors tends to fail, while the depth sensor provides information on the inner surface of the object where, in the absence of texture, the intensity image provides little or unreliable information. Therefore, intensity image and range image complement each other and our feature combines the stable information from both modalities. The feature pairs a reference point $\mathbf{r} \in \Omega_R$ from the range image, selected from the visible part of the object, and a point $\mathbf{e}$ from the geometric edge in the intensity image.

Since our feature is based on geometric object edges, we will fist discuss how the geometric edges are extracted from the RGBD image of the scene. Since the geometric boundaries, and thus the geometric edges, of an object depend on the viewpoint, our feature is inherently viewpoint-dependent. Since we use a perspective camera model, measurements of distances and angles are disturbed by the perspective distortion. In order to compensate for this distortion we perform a perspective correction described below in respect to a chosen reference point $\mathbf{r}$. Finally, we describe our multi-modal point pair feature.

Perspective Correction. Given a fixed camera, the appearance of an object in a projective image depends on the direction from which it is seen, on the distance to the projection center, and on the position of the projection in the image plane. The perspective distortion due to the position in the image plane disturbs measurements of distances and angles. To be robust against such distortion, we employ a perspective correction step that re-projects the edge $\mathbf{e} \in \Omega_E$, the reference point $\mathbf{r} \in \Omega_R$, and its normal $\mathbf{n}_r$ onto a new image plane. Ω_E is set of all edge pixels detected in the color image. Without loss of generality, we assume the camera center to be in the origin, *i.e.*, the viewing direction towards $\mathbf{r}$ is $\mathbf{v}_r = \mathbf{r}/|\mathbf{r}|$. The new image plane at which we project is defined as the plane perpendicular
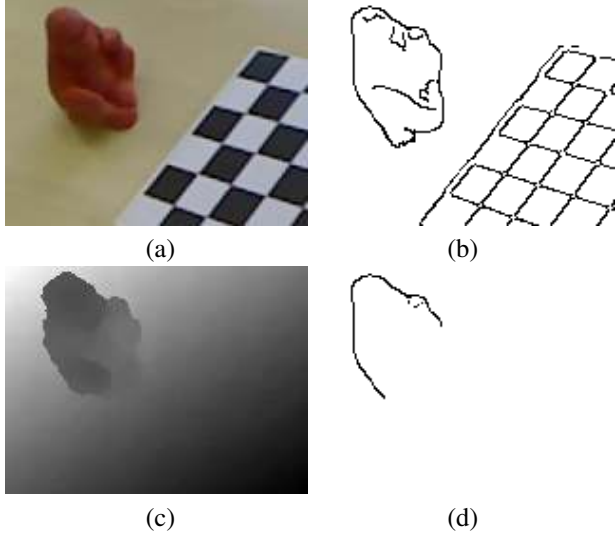
(a) (b)

(c) (d)

Figure 3. Example of the geometric edge detector. (a) Original color image. (b) Edges extracted from the color image. (c) Range image. (d) Filtered geometric edges: Color edges without depth discontinuity perpendicular to the edge direction are removed.

to $\mathbf{v}_r$ and with the same focal distance from the projection center as the original image plane. The reference point $\mathbf{r}$ is thus projected into the center of the new image plane, and the visible features appear as if seen in the center of the image. Fig. 2(a) depicts this correction step. For clarity, we continue to write $\mathbf{e}$ and $\mathbf{r}$ even if the corrected values are meant.

This undistortion is performed both in the online and the offline phase, and boils down to a homography which is efficiently applied on-demand to each edge point.

Geometric Edge Detection. Visible edges in an intensity image can be categorized into texture edges, which are found mainly on the inner parts of a surface, and geometric edges that appear due to geometric boundaries in the scene. The latter occur mainly on the occluding boundaries of objects. To localize an object, the proposed method uses only the object's geometric edges for several reasons. First, every object, textured or untextured, has a silhouette and thus a geometric boundary. The geometric edges are thus a very generic feature. Second, there are typically less geometric edges than texture edges in cluttered scenes, such that less features need to be processed. Third, geometric edges complement the surface information from the depth sensor as described above. And finally, geometric edges are easy to detect in RGBD images, as described in the following.

To obtain accurate geometric edges, we use an edge detector that combines the accuracy of edges in intensity images with the expressiveness of depth discontinuities. For this, we first detect edge pixels $\mathbf{e}$ with gradient direction $\mathbf{e}_d$

in the intensity image I_C using the Canny color edge detector [5]. The detected edges are then filtered using the depth image to obtain the geometric edges. The filter computes the minimum and maximum depth value on a line segment perpendicular to the edge direction. The edge point is classified as geometric edge if the difference of maximum and minimum depth value exceeds a certain threshold. The threshold should be larger than the expected depth noise of the sensor and is in practice set to 1-3 cm for images captured with a Kinect-like device.

This proposed filtering of geometric edges is computationally very efficient, since it needs to evaluate only a couple of pixels per detected color edge point. It is also robust w.r.t. veil points and noise. The orientation of an intensity edge depends on the local color gradient. We re-orient the direction of the edge such that the gradient $\mathbf{e}_d$ points out of the object, *i.e.*, from the surface closer to the camera towards the surface further away. Fig. 3 shows an example of the geometric edge detection.

Multimodal Feature. The multi-modal point pair feature in the perspectively corrected image centered at the reference point $\mathbf{r}$ is described by $F(\mathbf{e}, \mathbf{r})$ using a four-dimensional feature vector. This feature vector is designed to depend only on the viewpoint of the camera w.r.t. the object. It is most notably invariant against scale changes, *i.e.*, against the distance of the object from the camera, against rotations of the object around the viewing direction, and against perspective distortions. This design allows to train the proposed method using only one template per viewpoint. Multiple scales and rotations, such as in [9], are not required. The feature vector is defined as $F(\mathbf{e}, \mathbf{r}) = (d(\mathbf{e}, \mathbf{r}), \alpha_d, \alpha_n, \alpha_v)$ and contains:

- the metric distance $d(\mathbf{e}, \mathbf{r}) = Z(\mathbf{r})|\mathbf{e} - \mathbf{r}|/f$ of the two points. f is the focal length of the projection system and $Z(\mathbf{r})$ is the depth of the reference point $\mathbf{r}$. The scaling factor $Z(\mathbf{r})/f$ transforms the measurement from pixels to metric units, making it invariant against the distance of the point pair from the camera, i.e scale invariant;

- the angle $\alpha_d = \angle(\mathbf{e}_d, \mathbf{e} - \mathbf{r})$ between the difference vector of the two points and the edge gradient $\mathbf{e}_d$;

- the angle $\alpha_n = \angle(\mathbf{n}_r, \mathbf{e} - \mathbf{r})$ between the difference vector and the normal vector; and

- the angle $\alpha_v = \angle(\mathbf{n}_r, \mathbf{v}_r)$ between the normal vector and the direction towards the camera.

Fig. 2(b) depicts the four components of the feature vector. Note that the first two angles are measured in the perspectively corrected image plane, while the third is measured in 3D. α_d is in the range $[0; 2\pi]$, α_n and α_v in the range

$[0; \pi]$. All four components are invariant against rotation around the viewing direction due to the image plane correction. Additionally, they are invariant against the distance of the two points from the camera.

In the following, we describe how this multimodal point pair feature can be used to build a description of the object of interest.

3.2. Model Description

In the offline phase, a model description that contains the appearance of the object from various viewpoints is built. This is done by rendering the model from viewing directions sampled on the sphere around the object. The model description is represented by a hash table that maps quantized multimodal feature vectors to the lists of similar feature vectors on the model. This is similar to the descriptor used by Drost *et al.* [6], but using the proposed, viewpoint dependent multimodal feature. The hash table allows constant-time access to similar features on the object. The following steps are performed to create the model description:

1. Take a set of model reference points by uniformly sampling the model.

2. Select a set of viewing directions that contain all directions from which the object can be seen in the online phase. In practice, we uniformly sampled at least 300 viewpoints from the unit sphere.

3. Obtain the object's appearances I_C, I_R from the different viewing directions. This is done either by rendering the object from the selected directions, or by using template images taken by moving an RGBD sensor around the object and registering the views to each other.

4. For each template, detect the geometric edges on the object as described above.

5. For each edge point $\mathbf{e}$ and each model reference point $\mathbf{r}$ visible in the corresponding template, compute the multimodal feature $F(\mathbf{e}, \mathbf{r})$, quantize it, and store it in the hash table that describes the model.

The sampling parameters of the angle and distance coeficients of the feature depend on the expected noise level. In practice, they are set to $12°$ for the angles and 2% of the object's diameter for the distance value.

3.3. Voting Scheme

The proposed method uses a local voting scheme similar to the Generalized Hough transform [1] to recover the object's pose. The voting scheme is *local* in the sense that it localizes the object using a parametrization relative to a scene
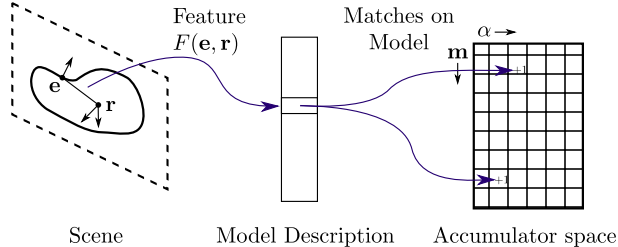

Figure 4. Sketch of the voting scheme. A multimodal point pair feature $F(\mathbf{e}, \mathbf{r})$ is extracted from the scene (left) and is quantized and matched against the model description (center) using a hash table. Each matching feature from the model leads to a vote in the accumulator array (right).

reference point, similar to [6], instead of a global parameter space as in the original GHT. Note that this approach is only outlined below. For a more detailed description, we refer the reader to [6].

Given a scene reference point $\mathbf{r} \in \Omega_R$ from the scene and its normal $\mathbf{n}_r$, and assuming that $\mathbf{r}$ lies on the surface of the object, one needs to recover the corresponding model point $\mathbf{m} \in M$ and the rotation α around the normal $\mathbf{n}_r$. $(\mathbf{m}, \alpha) \in M \times [0; 2\pi]$ are the local parameters of the object w.r.t. $\mathbf{r}$. If $(\mathbf{m}, \alpha)$ are known, the object's pose can be recovered. Since the pose recovery requires that the scene reference point lies on the object of interest, multiple reference points are sampled uniformly from the scene depth image Ω_R and the voting scheme is applied to each of them.

For the voting scheme, the space $M \times [0; 2\pi]$ of local parameters is quantized by using the model reference points from the model generation step and by uniformly subdividing the range $[0; 2\pi]$ of rotation angles. In practice, we use around 60–150 model reference points and quantize the rotation angle to 30 bins.

For each scene reference point, an accumulator is assigned to each sample of the local parameter space. The scene reference point is then paired with each edge point detected in the scene, and the corresponding feature vector is computed. Each such feature vector is matched against the model description to obtain the list of matching features from the model, *i.e.*, all possible locations of that feature on the model. Each such location votes for one entry in the accumulator space. After all edge points are processed for the current reference point, peaks in the accumulator space are detected, and the corresponding poses are computed. Fig. 4 outlines the voting scheme for a scene reference point.

The closer the object is to the camera, the larger it appears and the more edges are visible, leading to a higher score in the voting scheme. This bias is removed by multiplying the number of votes with the distance of the scene reference point from the camera.

Pose Clustering. Due to its local nature, several votings with reference points seeded throughout the scene need to be done in order to cover the whole scene. After each voting, the peaks in the accumulator space are detected and the corresponding poses and voting scores are stored. After processing all reference points, a clustering step is applied to group the resulting poses. For this, a new score is assigned to each pose which consists of the sum of weighted scores of its neighboring poses. Additionally, a non-maximum suppression is performed after the clustering to obtain the most prominent peak and remove duplicate, almost similar detections.

Pose Refinement. Due to the discrete nature of the Hough transform, each detected pose will be slightly off the correct pose. This error is in the magnitude of the employed spacial and angular sampling. In practice, an error of around $10°$ in rotation and 0.05 times the model diameter in translation is usual. To obtain a more precise pose and score, we use the iterative closest points algorithm (ICP) to refine the best matches from the clustering step. Finally, a new score is computed from the refined pose that expresses the amount of visible object surface in the scene.

4. Results

We evaluated the proposed method quantitatively and qualitatively on multiple datasets and compared it against state of the art methods. All datasets were captured using a Microsoft Kinect or a Primesense sensor to obtain an RGB and a depth image with resolution 640×480. Both modalities were calibrated and registered.

All models were available as CAD model. The scene reference points were selected by uniformly sampling the scene with a distance of 4% of the object's diameter. All tests were run on an up-to-date computer using an unoptimized C implementation. The evaluation took 2-10 seconds per scene, mostly depending on the scene size and the number of detected geometric edges. We believe that an improved implementation would speed up the method by an order of magnitude.

4.1. Quantitative Evaluation

Dataset Hinterstoisser et al. We first evaluated the method on several sequences from Hinterstoisser *et al*. [9], namely the APE, DUCK, CUP and CAR sequence. The CAMERA and the HOLEPUNCHER sequence were not evaluated as no CAD model was available. Each sequence contains 255 template images that were used for learning the features, and over 2000 evaluation images. Note that the template matching used in [9] does not recover the pose, as opposed to our method. Nevertheless, to allow a comparison between both methods, we employ the criteria of [9] to classify the correctness of matches. This criteria compares
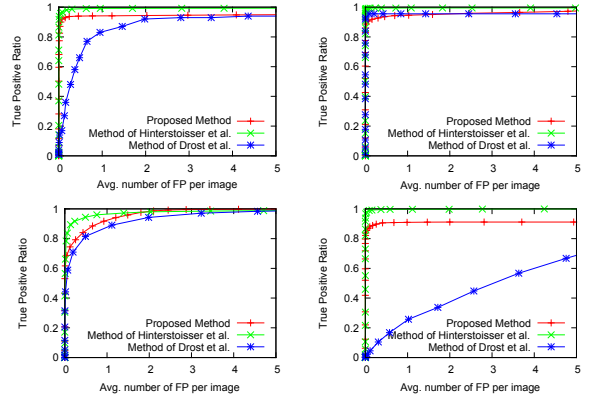

Figure 5. From top left to bottom right: Detection results for the APE, DUCK, CUP and CAR dataset from [9] for the proposed and the compared methods. Note that the method of Hinterstoisser *et al*. does not recover the object's pose, as opposed to the other two methods. The duck has a rather unique surface and is detected by all three methods with a high detection rate. The car contains planar patches similar to background clutter, leading to misdetections for the method of Drost *et al*. that uses the 3D information only, while the two multimodal approaches keep a high recognition rate.
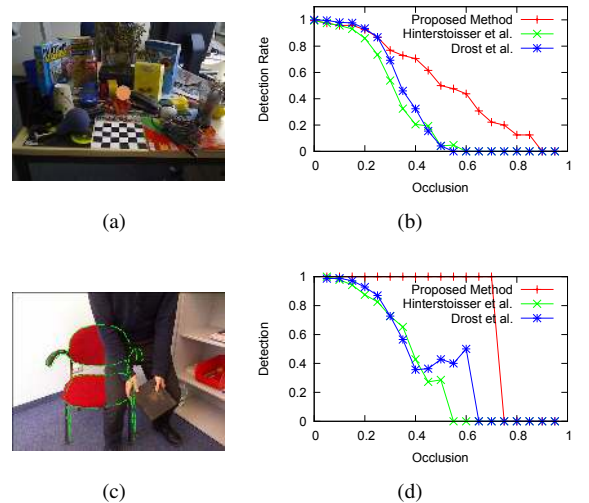

Figure 6. (a) Example image of the artificial occlusion that covers the ape. (b) Detection rate vs. occlusion for the occluded ape sequence. The occlusion measures how much of the original ape surface visible in the image is occluded. (c) Example image showing the occlusion on the real-world occlusion dataset and the detection result of the proposed method in green. (d) Detection rate vs. occlusion for the occluded chair sequence.

the bounding box of the ground truth and the match. Note that this classifies matches at the correct position but with an incorrect rotation as correct. However, we found that such incorrect classifications were rare for our approach. Fig. 8 shows example scenes and detections. Fig. 5 shows the detection rates for our proposed method, the method of Hinterstoisser *et al*. and the method of Drost *et al*. Again, note that the proposed method and the method of Drost *et al*. recover

the pose of the object, while the method of Hinterstoisser *et al*. does not.

Overall, our method performs slightly worse than the method of Hinterstoisser *et al*., but still reaches quite high recognition rates. A manual inspection of the scenes where our method failed shows that most missing detections were due to a failure of the 2D Canny edge extractor. This extractor failed in scenes with high motion blur and in areas with little contrast, *i.e*., when object and background color were similar. Compared to the localization method of Drost *et al*., out method performs approximately equal for the duck and cup sequence but significantly better for the ape and car sequence. This is mostly due to the rather flat back of the ape and the large planar parts of the car: The method of Drost *et al*. optimizes the surface overlap of model and scene, leading to false positives if large parts of the object are similar to clutter. Our method, which optimizes both surface and silhouette overlap, is able to correctly remove such false positives.

Occlusion. We additionally evaluated the three methods against partial occlusion of the target object using two datasets. For the first dataset, one of the images from the ape sequence was disturbed by artificially occluding different parts of the ape in both the RGB and the depth image. A total of 256 images with varying amounts of occlusion was created this way. The detection rates of all three methods and an example image are show in Fig. 6 (a), (b). Note that we set the parameters of the method of Drost *et al*. such that similar timings as our methods were obtained. As explained in [6], higher detection rates can be achieved by allowing the method to run significantly longer.

For the second dataset, we used varying amounts of real occlusion of a chair that was put in a fixed position with respect to the sensor. For the model creation, a reference image without occlusion was used. Fig. 6 (c), (d) shows an example image and the resulting detection rate for this dataset.

Our method clearly outperforms both compared methods in case of non-trivial occlusion. For the method of Hinterstoisser *et al*., occlusion of certain key regions of the object that contribute the most to the detection lead to drastically reduced detection rates. On the contrary, our method treats all edge and inner regions equally. The method of Drost *et al*. suffers from a similar effect, where for larger occlusions the remaining surface parts of the object look more similar to background. Since the proposed method optimizes both surface and geometric edge overlap, significantly larger occlusions are necessary to reduce the detection rate.

4.2. Qualitative Evaluation

The proposed method was also evaluated qualitatively with object's of different size and shape. Fig. 7 and Fig. 1 show several test images and detections. We found that the method performs very well even in case of occlusion and large amounts of clutter. The method works equally well and with the same speed for all tested objects. This is a major advantage over the method of Drost *et al*., which degenerates both in terms of speed and detection performance for planar objects.

5. Conclusion

Intensity images and range data complement each other. The rise of structured light sensors that produce such data at a low cost and with little calibration overhead increases the importance of new methods that effectively use both modalities. We proposed a method that is able to use the most prominent and stable information from both modalities to accurately detect and localize rigid 3D objects. A novel multimodal point pair descriptor and a simple yet effective and stable geometric edge extractor are used to simultaneously optimize surface and silhouette overlap of scene and model. Experiments show a high detection rate even for heavily cluttered scenes, an accurate pose recovery, a faster detection than other localization methods as well as a very high robustness against occlusions.

References

[1] D. H. Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981. 5

[2] P. Bariya and K. Nishino. Scale-hierarchical 3D object recognition in cluttered scenes. In *CVPR*, 2010. 1

[3] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 24(4):509–522, Apr. 2002. 2

[4] R. J. Campbell and P. J. Flynn. A survey of Free-Form object representation and recognition techniques. *Computer Vision and Image Understanding*, 81(2):166–210, 2001. 2

[5] J. Canny. A computational approach to edge detection. *PAMI*, (6):679–698, 1986. 4

[6] B. Drost, M. Ulrich, N. Navab, and S. Ilic. Model globally, match locally: Efficient and robust 3D object recognition. In *CVPR*, 2010. 1, 2, 5, 7, 8

[7] A. Frome, D. Huber, R. Kolluri, T. Bülow, and J. Malik. Recognizing objects in range data using regional point descriptors. *ECCV*, 2004. 2

[8] D. Gavrila and V. Philomin. Real-time object detection for "smart" vehicles. In *ICCV*, 1999. 1

[9] S. Hinterstoisser, S. Holzer, C. Cagniart, S. Ilic, K. Konolige, N. Navab, and V. Lepetit. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In *ICCV*, 2011. 1, 2, 4, 6, 8

[10] S. Holzer, S. Hinterstoisser, S. Ilic, and N. Navab. Distance transform templates for object detection and pose estimation. In *CVPR*, 2009. 1

[11] X. Jiang and H. Bunke. Edge detection in range images based on scan line approximation. *Computer vision and image understanding*, 73(2):183–199, 1999. 1, 2

Figure 7. Example scenes and detection results for several example scenes with high amounts of clutter and occlusion. The detections are outlined in green. Note that for objects that were reconstructed from reference images, such as the chair, have a rather rough outline compared to objects modeled with CAD, such as the box.

Figure 8. Detection results for several scenes from the dataset of [9]. From top to bottom: Original scene, detection result of [9], detection result of [6] (in red), our result (in green). Note that the method of Hinterstoisser *et al.* does not recover the object's pose, but only the bounding box of the best matching template.

[12] A. E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *PAMI*, 21(5):433–449, 1999. 2

[13] K. Lai, L. Bo, X. Ren, and D. Fox. Sparse distance learning for object recognition combining rgb and depth information. In *ICRA*, 2011. 2

[14] G. Mamic and M. Bennamoun. Representation and recognition of 3D free-form objects. *Digital Signal Processing*, 12(1):47–76, 2002. 2

[15] A. S. Mian, M. Bennamoun, and R. Owens. Three-dimensional model-based object recognition and segmentation in cluttered scenes. *PAMI*, 28(10):1584–1601, 2006. 1

[16] A. S. Mian, M. Bennamoun, and R. A. Owens. Automatic correspondence for 3D modeling: An extensive review. *International Journal of Shape Modeling*, 11(2):253, 2005. 2

[17] C. F. Olson and D. P. Huttenlocher. Automatic target recognition by matching oriented edge pixels. *IP*, 6, 1997. 1

[18] R. B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (FPFH) for 3D registration. In *ICRA*, 2009. 2

[19] B. Steder, R. B. Rusu, K. Konolige, and W. Burgard. Point feature extraction on 3D range scans taking into account object boundaries. In *ICRA*, 2011. 1, 2

[20] C. Steger. Occlusion Clutter, and Illumination Invariant Object Recognition. In *IAPRS*, 2002. 1

[21] S. Stiene, K. Lingemann, A. Nuchter, and J. Hertzberg. Contour-based object detection in range images. In *3DPVT*, 2007. 1, 2

[22] M. Sun, G. R. Bradski, B.-X. Xu, and S. Savarese. Depth-Encoded Hough Voting for Joint Object Detection and Shape Recovery. In *ECCV*, 2010. 2

[23] F. Tombari, S. Salti, and L. D. Stefano. Unique signatures of histograms for local surface description. In *ECCV*, 2010. 1, 2

[24] C. Wu, B. Clipp, X. Li, J. M. Frahm, and M. Pollefeys. 3d model matching with viewpoint-invariant patches (vip). In *CVPR*, 2008. 2