Fusing Color and Geometry Information for Understanding Cluttered Scenes

Ulrike Thomas, Simon Kriegel and Michael Suppa

***Abstract*— In this paper, we introduce a new image processing pipeline for scene recognition and pose estimation in robotic applications. Unknown objects are autonomously modeled resulting in geometric 3D models and color images. Theses models are then used for object recognition in cluttered scenes by merging color and geometry information. Our recognition approach generates new suitable feature vectors and uses RANSAC to obtain promising hypotheses of recognized object poses for the scene. RANSAC is widely used for scene understanding. For making RANSAC applicable, it is very important to implement this algorithm efficiently and to reject hypotheses as early as possible in the scene understanding pipeline. By using color information many hypotheses can be rejected early in the recognition pipeline. With our approach we provide an efficient implementation of a scene analyzing pipeline while fusing color and geometric information. Moreover, we are able to learn new objects by a fast autonomous scanning process and no further runs through time consuming learning algorithms are necessary. The complete pipeline from scanning to scene understanding is described. The evaluated scenes consist of several household objects. Some of them vary only in texture and not in shape.**

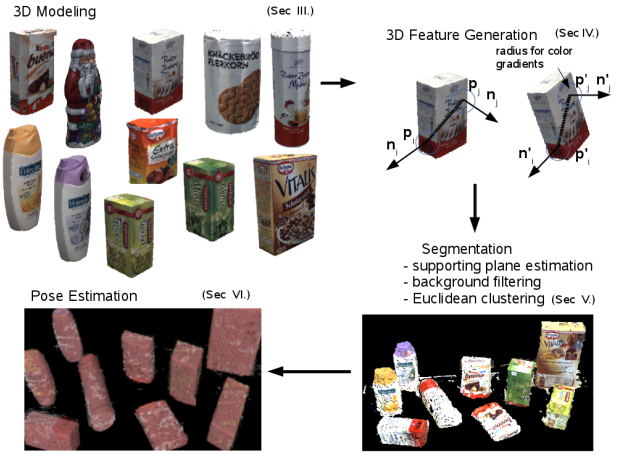

Fig. 1. The image processing pipeline from scanning to pose estimation.

I. INTRODUCTION

Interpretation of complex scenes and pose estimation of known objects is one of the main topics in computer vision for robotics. In many industrial and service robotic scenarios, it is desirable to autonomously acquire properties of new objects. The fused color and geometry information can then be stored in a data base, for which important feature vectors are generated from. One key topic is how to fuse geometric data with color information. A possibility would be to apply SIFT-features or SURF-features. Merging these features with a geometric based model fitting approach has been investigated, e.g. in [1]. In contrast, we extend a model based RANSAC approach with color values, such that models can be matched faster into a scene. The used features can be computed very fast based on colored point clouds. Throughout this paper 3D point clouds are denoted by its point sets $P := \{\mathbf{p_1}, \ldots, \mathbf{p_n}\}$ and its surface normals $N = \{\mathbf{n_1}, \ldots, \mathbf{n_n}\}$, which can be obtained by principle component analysis of its neighbors. There, color information is represented in the RGB color space by $C = \{\mathbf{c_1}, \ldots, \mathbf{n_n}\}$. The model are obtained by autonomous 3D scanning, instead of applying time consuming learning algorithms. In the last years, reasonable and affordable 3D sensors have become available. Instead of using an active approach such as the Kinect, we haven chosen a semiglobal matching (SGM) stereo system [2]. It has the advantage that no additional light sources are necessary during the scene understanding process. Pose estimation in general is equivalent with matching the correct model into the scene, more formally it is described by estimating the correct pose $\Theta := (\mathbf{R},\ \mathbf{t})$ with $\mathbf{R} \in SO(3)$ and $\mathbf{t} \in \mathbb{R}^3$ which fits best into the point cloud P such that $\sum_{\forall p_i \in M} || \mathbf{p}_i - \mathbf{p}_j ||^2$ is minimized under certain constraints such as collisions or physics, where p_i is a model point and $\mathbf{p}_j$ is the closest scene point. Our approach additionally uses color and tries to find the optimal solution for table top scenes, where we do not take any assumptions of the object poses. In Fig. 1 our image processing pipeline for generic scene analysis is shown. At first a triangle mesh of the unknown object is autonomously generated with a laser striper. After obtaining the mesh, the color information is collected from 2D images. In another step a hash map of features is generated for each object. After acquisition of a new scene with a stereo camera, the preprocessing step, i. e. segmenting the scene is triggered. The RANSAC approach is then started for each segment. The best assumption regarding certain cost function and physical collisions is then reported.

The authors are with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Wessling, Germany {firstname.lastname}@dlr.de

II. RELATED WORK

In the following sections, the most recent work on autonomous object modeling and recognition based on geometry and color information is summarized.

A. Autonomous Object Modeling

Kasper et al. [3] present a system for semi-automatic object modeling. It requires a very large, fixed and expensive

setup and does not allow for scanning the bottom part of an unknown object. The acquisition time of 20 minutes is very high and the models in the database are noisy and still contains holes.

Autonomous object modeling systems usually consists of a robot and sensor for which a Next-Best-View (NBV) is iteratively planned until a complete 3D model of the unknown object is given. In [4], an industrial robot is used in combination with a turntable, on which the object is placed. However, the user needs to manually input object size and stand-off distance for each object individually, which does not represent a fully autonomous system. Torabi et al. [5] try to scan a set of points on the occlusion surface which they call target points using spheres as search space. The average waiting time in between NBVs was four minutes, resulting in a total acquisition time beyond one hour. In [6], unknown objects are autonomously modeled within table scenes utilizing an industrial robot. Furthermore, in [7] a mirror cabinet is introduced for partially autonomous modeling of human bodies. In [8], new objects are learned by grasping them, moving them in front of a RGB-D camera and planning NBV regrasps. From all presented approaches, only in [8] the bottom part and color of the object are obtained. However, the 3D models are noisy due to the RGB-D camera.

B. Object Pose Estimation

For pose estimation several RANSAC approaches exist. RANSAC is well understood and can be easily implemented [9], [10]. Two key issues arise: How to draw samples and how to evaluate hypotheses. Moreover the generation of hypotheses is critical. Many feature descriptors for 6D pose estimation such as (fast) point feature histograms [11], [12], surflet pair relation histograms [13] or point triplets exist. Their usage for pose estimation regarding accuracy is evaluated in [14]. All these features have widely been used in combination with RANSAC for pose estimation in robotic applications [15], [16], [17], [18]. Another RANSAC approach for model fitting into depth images can be found in [19]. It suggests a very generic penalty term for model matching in the global scene. In [20] texture information is combined with depth information for model fitting. Local descriptors on surfaces patches are applied to find good pose hypotheses. Among the above mentioned feature descriptors spin images are introduced [21] or three-dimensional Tensors are applied for registration [22]. First approaches combining geometry and texture information for object recognition such as in [1], estimate the object's pose with texture-based features and verify the pose hypothesis in the depth image using a geometry model. A voting based approach exists which uses an image in combination with Hough transforms and obtains the depth information, where texture information is available [23]. In [24], [25] a CAD-model and camera images are used. For their approaches shape edges are applied to match CAD-model into recognized scenes. Furthermore, CSHOT [26] was introduced for feature matching, which adds color difference between points computed in CIELab space to geometry.

Fig. 2. Autonomous object modeling of a shower gel. *Left:* Laser striper and stereo camera are attached to the flange of an industrial robot. The unknown object is placed on a pedestal. *Top right:* Image from bottom mono camera view. *Bottom right:* Textured 3D Model.

III. AUTONOMOUS MODELING

In this work, the method for autonomous 3D modeling of unknown objects by Next-Best-Scan (NBS) planning, presented in [27], is extended. The method does not consider scanning the bottom part of the objects or mapping color onto the 3D surface model. We will address both these issues in this work. The autonomous modeling system consists of a Kuka KR16 industrial robot, which is used to get the sensor pose and to move the sensor to and along the Next-Best-Scan (NBS). Fig. 2 shows the pedestal on which the unknown object is placed. The geometry is obtained with a laser striper, the ScanControl 2700-100 from Micro-Epsilon, and the texture with a mono camera. A stereo camera system of two Guppy Pro F-125 (1292 x 964 pixels) is mounted beside the laser striper, which will be used for object pose estimation as described in the next sections. The autonomous method generates two different 3D models, a triangular mesh and a probabilistic voxel space (PVS), online during a laser scan. Here, the mesh represents the application goal and is used to generate further scan path candidates and to calculate a sampling quality, which is used as termination criteria. The PVS represents the exploration aspect and is used for occlusion avoidance, collision free path planning and for selection of a NBS. Scan path candidates are calculated based on the actual geometry of the current surface model and not by simply sampling candidates over a sphere. First, several boundaries in the mesh are detected and for each the surface trend is estimated and a candidate is calculated with overlap considering the optimal sensor configuration. Second, after the object model is fairly complete all around, holes are detected and scan paths for these are planned. The system terminates if the model has a certain completeness and quality level. Using the mono camera, one color image is obtained from the top of the object and eight on a circular path around the object considering the dimensions of the now

known object geometry. In order to scan the object from the bottom, the object needs to be moved onto a defined side. After the desired mesh completeness for the object without the underside is reached, the object for now is manually rotated by 90°. As the approximate rotation of the object is known, two scans are performed along the two most significant edges of the rotated object. The data of the two scans is merged and transformed by the approximate 90° rotation. Then the exact transformation between the previous object position and the rotated object is estimated using the Iterative Closest Point algorithm [28]. The transformation is applied to the scan paths and the generated surface model, and the NBS planning is continued until also the bottom part in the triangle mesh is filled and a final color images is taken from the bottom. Finally, the triangle mesh from the laser striper and color images from the mono camera are merged by acquiring a color value for each vertex from the color image, of which the view direction is most similar to the inverse of the corresponding surface normal. The time for acquisition of a textured 3D model is approximately 6-7 minutes per object.

IV. FEATURE GENERATION

Pairs of points have become very popular as feature vectors, as they can be computed very fast in point clouds, where normal vectors of surfaces are available. Based on such features a hashing function is commonly applied to sort each feature vector into a bin. The more non-ambiguous such functions are the more efficient the search for good hypotheses is. Certainly, the optimum is a one to one mapping in between features and hypotheses, but this is not feasible. To this end, various features and hash functions can be applied. Furthermore, the RGB values are used to achieve a more non-ambiguous representation of feature vectors. The feature vectors used in this paper are surflet pairs in combination with RGB difference values. Difference values are applied to become more independent from lighting conditions. The hash map for each object is filled in such a way that a constant number of vertex pairs of each object is randomly drawn with the constraint, that the distance between the objects is not less than a third of the largest distance of two points. This requires that at least approximately a third of the object must be not occluded. In addition to the geometry, the color differences of the two points in the RGB-space are taken. Hence, the feature vectors are generated in following manner: Given two points denoted as $\mathbf{p}_i$ and $\mathbf{p}_j$ with their respective surface normals $\mathbf{n}_i$, $\mathbf{n}_j$ as well as their color values $\mathbf{c}_i := (r_i, g_i, b_i)$, $\mathbf{r}_i$ and $\mathbf{d}_{ij} := \mathbf{p}_i - \mathbf{p}_j$, the feature vector is obtained by:

$$\begin{pmatrix} || \mathbf{d}_{ij} || \\ \angle(\mathbf{n}_j, \mathbf{d}_{ij}) \\ \angle(\mathbf{n}_j, \mathbf{d}_{ij}) \\ atan2(\ \mathbf{n}_i \cdot ((\mathbf{d}_{ij} \times \mathbf{n}_j), (\mathbf{n}_i \times \mathbf{d}_{ij}) \cdot (\mathbf{d}_{ij} \times \mathbf{n}_j) \\ | r_i - r_j | \\ | g_i - g_j | \\ | b_i - b_j | \end{pmatrix} \quad (1)$$

Fig. 3 illustrates this feature vector. The size of the bin for each degree of freedom of the feature vector is adjusted according to the object size, sensor noise and sensitivity. The bin size for the first index is between the maximum distance between object vertexes and one third of it, with a resolution of 2 mm. The rotational degrees of freedom are set to 30° and the difference between colors in the RGB space are each divided into 5 chunks. In addition for each vertex the color gradient is estimated and later on used for comparison of vertexes.

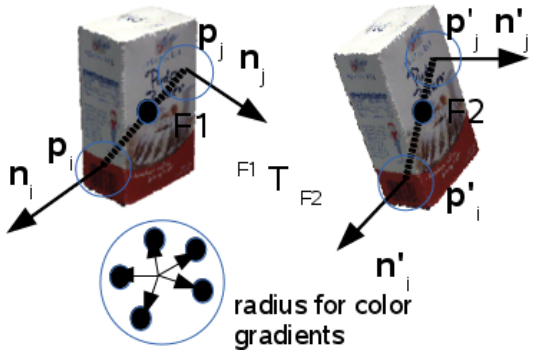

Fig. 3. The 7D feature vector consists of 4D geometry and 3D texture information. In addition, the color gradients are generated for each vertex.

V. PREPROCESSING THE SCENE

First of all a supporting plane is searched in the scene. Fig. 4 illustrates the estimated plane in the colored point cloud. The depth image is obtained by SGM and the plane is matched into the scene by using RANSAC and a least square error function. Everything below the plane and the background is filtered, see Fig. 4.

Fig. 4. *Left:* A scene acquired by the SGM stereo algorithm; *Right:* The estimated supporting plane and clustered scene.

Then, the Euclidean algorithm is applied which is available in the point cloud library. It separates the scene into clusters. In the best case, each cluster represents one single object. In order to accelerate the entire image processing pipeline, the RGB-spectrum is analyzed for each cluster and compared with the RGB-spectrum from known objects. It provides the possibility to exclude improbable hypotheses as early as possible in the image processing pipeline. This is achieved by comparing the differential color histograms. Fig. 5 illustrates the histograms for three objects in the upper lines and the color histograms of the clusters from the illustrated scene in Fig. 4. The maximum values of $\max(r,g,b)$ are subtracted from the second largest values in (r,g,b). The histogram according to theses difference values are determined. Then

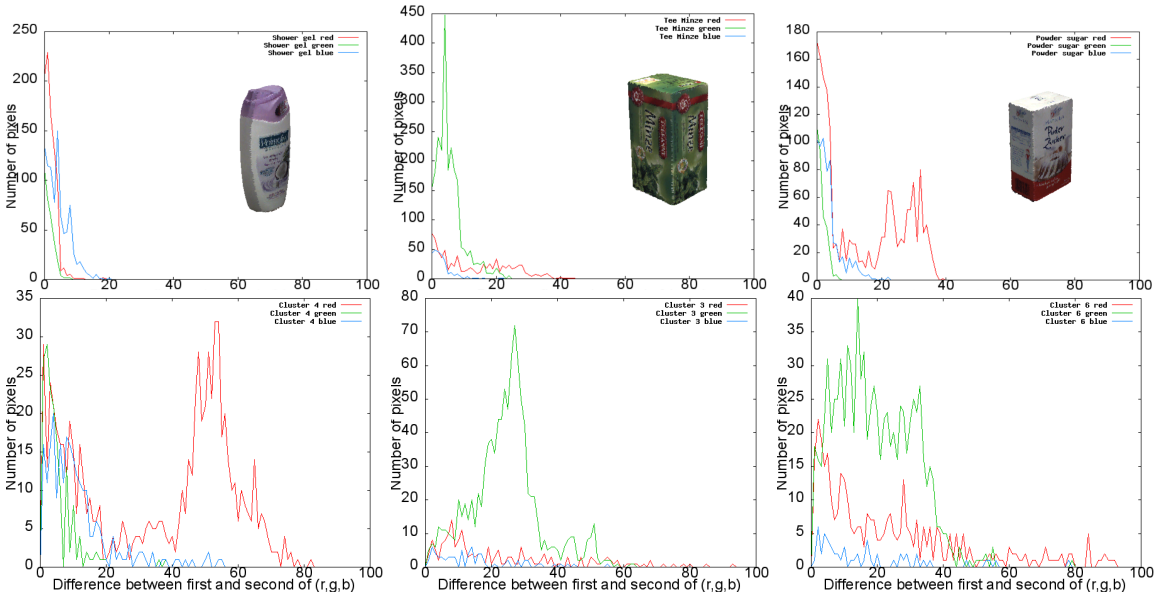

Fig. 5. Histograms for three objects and three clusters from the scene in Fig.4

.

the peaks are compared. If for example there is a green peak in the color histogram of a cluster, but not in the cluster of the object it is unlikely that this object lies within the cluster. The first shown cluster (image 4 in Fig. 5) represents the cluster with the sugar powder, (image 5) represents the cluster with the tee mint and the last cluster represents the tee fennel. The shower gel can be excluded from all clusters, because there are peaks in the red and green bands, which can not be found in the histogram of the shower gel.The cluster illustrated in the middle will probably not contain the sugar powder, which holds also for the cluster represented by the histogram depicted at the right side.

VI. POSE ESTIMATION

Like in each typical RANSAC implementation, the key steps are how to draw samples and how to evaluate hypotheses. Usually it is important to reject unlikely hypotheses as early as possible in the processing pipeline. Thus, the non-ambiguity in the hash map is an important issue as discussed in the last section. Furthermore, drawing samples from the scene which lead to good hypotheses speeds up the entire scene analyzing approach. Therefore, the following steps for hypotheses generation, filtering and evaluation are implemented:

1. For all objects estimate $\varepsilon_{\min}$ and ε_{max} which serve well as maximum distance for drawing point pairs.
2. For each segment do: Draw one point $\mathbf{p}_i$ of the point cloud $P = \{\mathbf{p}_1 \dots \mathbf{p}_n\}$ at random. Draw a second point out of a ball with the radius given by either ε_{max} or by r (for point features) according to the matching model.
3. For the drawn point pairs compute the feature vectors either $f(\mathbf{p}_i, \mathbf{p}_j)$ and try to find corresponding entries in the hash maps. Let $\mathscr{H}_k$ be a set of hypothesis for object k in the corresponding bin, then for each hypothesis determine the alignment for rigid motion with $\mathbf{R}$ and $\mathbf{t}$, see Fig. 3. This pose hypothesis $\Theta := (\mathbf{R}, \mathbf{t})$ can be alignment with:

$$\mathbf{t}_\Delta = -\mathbf{p}_i^B - \frac{\mathbf{p}_j^B - \mathbf{p}_i^B}{2} + \mathbf{p}_i^A + \frac{\mathbf{p}_j^A - \mathbf{p}_i^A}{2} \tag{2}$$

and the rotational part is yielded by

$$\mathbf{R}_F^W := \left(\frac{\mathbf{d}_{ij}}{||\mathbf{d}_{ij}||} \quad \frac{\mathbf{d}_{ij} \times (\mathbf{n}_i \times n_j)}{||\mathbf{d}_{ij} \times (\mathbf{n}_i \times \mathbf{n}_j)||} \quad \frac{(\mathbf{d}_{ij} \times \mathbf{n}_i \times \mathbf{n}_j) \times \mathbf{d}_{ij}}{||(\mathbf{d}_{ij} \times \mathbf{n}_i \times \mathbf{n}_j) \times \mathbf{d}_{ij}||} \right) \tag{3}$$

By this, the estimated pose can be calculated with respect to the world's reference system by $\mathbf{R}_A^W \cdot \mathbf{R}_B^{W-1}$ leading to Θ, which is inserted into $\mathscr{H}_k$.
4. Now, hypotheses for all objects are collected in $\mathscr{H} := \cup_{\forall k} \mathscr{H}_k$ and will be evaluated with two different functions. The first one indicates how well the object matches into the scene and with the second function one obtains a quality measure how probable the hypothesis is regarding the viewpoint. The first cost function is denoted as

$$\frac{1}{\sum_{\mathbf{p}_i \in P_{seg}}} \sum_{\mathbf{p}_i \in P_{seg}} g(\mathbf{p}_i) \wedge h(\mathbf{p}_i) \wedge q(\mathbf{p}_i) \tag{4}$$

with:

$$g(\mathbf{p}_i) = \begin{cases} 1 & \text{if} \quad \min_{\forall \mathbf{p}_j \in M} \{|| \mathbf{p}_j - \mathbf{p}_i ||\} < \varepsilon_g \\ 0 & \text{else} \end{cases} \tag{5}$$

$$h(\nabla(\mathbf{p}_i)) = \begin{cases} 1 & \text{if} \quad \min_{\forall \mathbf{p}_j \in M} \{|| \nabla(\mathbf{p}_j) - \nabla(\mathbf{p}_i) ||\} < \varepsilon_h \\ 0 & \text{else} \end{cases} \tag{6}$$

$$q(\mathbf{p}_i) = \begin{cases} 1 & \text{if} \quad \min_{\forall \mathbf{p}_j \in M} \{|| \mathbf{c}_j - \mathbf{c}_i ||\} < \varepsilon_q \\ 0 & \text{else} \end{cases} . \tag{7}$$

The functions h and g ensure that only such points will be counted which fit into the scene regarding color information as well. For the color gradient ∇, a constant radius is chosen, within each of theses balls the color gradient is estimated. In addition, the certain color values are compared if they do not differ too much, e.g. if their value is smaller than ε the vertex is counted as proper match. The second cost function is given by assuming the view direction with $\mathbf{v}$ then for the quality of a hypothesis $\Theta = (\mathbf{R}, \mathbf{t})$ follows

$$\frac{1}{\sum_{\mathbf{p}_j \in M | \mathbf{p}_j \cdot v > 0}} \sum_{\mathbf{p}_j \in M | \mathbf{p}_j \cdot \mathbf{v} > 0} || \mathbf{R}\mathbf{p}_j + \mathbf{t}_\Delta - \mathbf{p}_i^\star ||^2 , \qquad (8)$$

where $\mathbf{p}_i^\star$ is the closest scene point.

These functions are used for evaluation, where each hypothesis $h_i \in \mathscr{H}$ is rejected if it does not lead to a value above a certain threshold. All remaining hypotheses are collected in the set $\mathscr{H}_{best}$ which is sorted in descendant order $h_i \succ h_j \succ \ldots \succ h_{min}$ according to a weighted sum of both cost functions. More hypotheses are inserted into the set by exploiting the object's symmetries. Often objects with symmetrical similar poses need to be distinguished for grasping and manipulation. Then, with the evaluation function accurate hypotheses are found.

Fig. 6. Symmetry-axis for the cup and a power socket.

Fig. 7. From left to right depth images of table top scene A, B, C, and one with real clutter D.

TABLE I

RECOGNITION RATES IN PERCENT BY POSE ESTIMATION BASED ON GEOMETRY AND COLOR (LEFT) AND SIMPLY GEOMETRY (RIGHT) INFORMATION FOR THE EVALUATED SCENES.

Objects	(A)	(B)	(C)	(D)
Shower gel (purple)	95 | 55	95 | 45	90 | 90	– | 55
Shower gel (yellow)	95 | 45	95 | 50	90 | 90	95 | 45
Tee pack (fennel)	80 | 45	90 | 40	- | 50	80 | 50
Tee pack (mint)	85 | 45	90 | 50	95 | 45	65 | 40
Sugar powder	75 | 60	80 | 80	80 | 75	–
Sugar mill	90 | 60	90 | 60	80 | 60	90 | 70
Kinder Buono 1	75 | 55	65 | 60	65 | 65	60 | 50
Kinder Buono 2	70 | 50	70 | 50	80 | 70	–
Santa Claus	100 | 100	–	100 | 100	85 | 65
Jam sugar	–	–	90 | 80	90 | 85
Cereal box	100 | 95	100 | 100	95 | 95	95 | 85
Crisp bread	–	–	85 | 80	80 | 75

VII. RESULTS

The RGB-D images including depth and color information are obtained from various scenes with a stereo camera system (see Fig. 2 left) using the SGM approach. For feature generation, $1 \cdot 10^6$ point pairs are chosen and their feature vectors are generated. The radius for color gradient estimation is set to 8 mm. The applied epsilons are $\varepsilon_\nabla = 0.1$ and $\varepsilon_{color} = 0.3$, where the RGB values are normalized between 0 and 1. The scenes illustrated in Fig 7 are applied for evaluation. Here, the number of RANSAC iterations for pose estimation of the scenes was set to $20 \cdot 10^4$. Tab I illustrates the recognition rate for each object per scene. Each recognition rate represents an average value estimated over 20 iterations of different depth images obtained of the same scene. The object poses are considered to be well estimated if they lie in the tolerance space of 4 mm translational and 4° rotational error. The left side of the cells represents the recognition rate, when depth and color values are included as described in this paper. The right side contains the recognition rates obtained without considering color values. As it can be expected objects with identical geometry (tee packs and shower gels) can not be distinguished properly without considering color information. With color information, the recognition rate significantly increases and the objects with identical geometry can be separated quite well. The algorithm has problems in separating the sugar powder from the 'Kinder Buono' as the color distribution is quite similar (large white and small red area), and they differ only slightly in geometry. A ground truth for the pose of each object in the scenes A, B, and C was obtained with the laser striper as it is more accurate than the stereo camera. Therefore, several scans of the scene from different views were performed and the accumulated depth images are clustered by the dominant plane of the tabletop. For each point cloud cluster and each detected object model, a principal component analysis is applied and the center and orientation of these are compared. For the scenes A, B, and C, the pose error regarding the ground truth is illustrated in Fig. 8 for the translational error (left) and for the rotational error (right). In scene A (red line), the number of objects with a translational error of 2 mm or less is 6, only one object has a translational error of 3.5 mm. Note, that the pose estimation is done in full 6D neglecting the assumption from the table top scene within less than 20 sec. The error is estimated in x,y plane translational and about the z-axis rotational and contains combined errors from pose estimation, calibration,

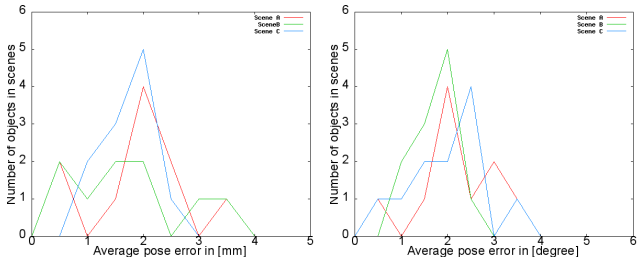

Fig. 8. Translation (left) and rotational (right) error for object poses.

and stereo camera system.

Scene D shows a real cluttered scene. In this case the segmentation of the scene does not support the recognition process and hence all objects are matched within the clutter. Due to some occlusions the recognition rate of one of the tee packs is only at 65 % for the combined color and geometry based pose estimation. All the other objects reach higher recognition rate. Note that for the simply geometry based pose estimation both shower gels (purple and yellow) are detected although only a yellow one exists in the scene. Due to the clutter, we could not obtained the ground truth in the same manner. Furthermore, for scene D the collision checker used for consistent tests is modified such that objects are modeled only for the collision test smaller than they are. This could be improved by using real virtual tests allowing penetration detection.

Overall, the recognition rates for the pose estimation algorithm based on both color and geometry information are significantly better than for just the geometry and also objects with same geometry and different color can be distinguished, which was to be assumed.

VIII. CONCLUSIONS AND FUTURE WORK

In this work, we have described a complete recognition pipeline for pose estimation of unknown objects. After the unknown objects are autonomously modeled, the described feature vectors for object pose estimation are generated based on the acquired models and mapped into a hash map. Due to the applied feature vector, the hash map can be computed and accessed very quickly. A new approach for fusing color and geometry information for recognizing objects in scenes is presented. The challenge is to adjust the weighting between geometry and color in an appropriate way. Moreover, which feature to select as the most important feature is also an interesting topic for recognition.

When humans perceive object scenes they acquire a very inaccurate representation of objects as long as they do not want to grasp the objects. This implies that a fast uncertain understanding could be implemented in the processing pipeline before accurate pose estimation is done. Moreover, another possibility would be to detect features in the 2D image before they are mapped to the mesh. Further, for making the modeling process fully autonomous, the rotation of the objects could be performed by the robot itself.

REFERENCES

[1] M. Brucker, S. Léonard, T. Bodenmüller, and G. D. Hager, "Sequential scene parsing using range and intensity information," in *IEEE ICRA*, St. Paul, Minnesota, USA, May 2012, pp. 5417–5424.

[2] H. Hirschmüller, "Stereo processing by semiglobal matching and mutual information," *IEEE PAMI*, vol. 30, no. 2, pp. 328–341, 2008.

[3] A. Kasper, Z. Xue, and R. Dillmann, "The KIT object models database: An object model database for object recognition, localization and manipulation in service robotics," *IJRR*, vol. 31, no. 8, pp. 927–934, 2012.

[4] S. Larsson and J. A. P. Kjellander, "Path planning for laser scanning with an industrial robot," *RAS*, vol. 56, no. 7, pp. 615–624, 2008.

[5] L. Torabi and K. Gupta, "An autonomous six-DOF eye-in-hand system for in situ 3D object modeling," *I. J. Robotic Res.*, vol. 31, no. 1, pp. 82–100, 2012.

[6] S. Kriegel, M. Brucker, Z.-C. Marton, T. Bodenmüller, and M. Suppa, "Combining Object Modeling and Recognition for Active Scene Exploration," in *IEEE/RSJ IROS*, Tokyo, Japan, Nov. 2013, pp. 2384–2391.

[7] S. Molkenstruck, S. Winkelbach, and F. Wahl, "3d body scanning in a mirror cabinet." in *DAGM*, 2008, pp. 284 –293.

[8] M. Krainin, B. Curless, and D. Fox, "Autonomous Generation of Complete 3D Object Models Using Next Best View Manipulation Planning," in *IEEE ICRA*, May 2011, pp. 5031–5037.

[9] M. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analy- sis and automated cartography," *CACM*, vol. 24, no. 6, pp. 381–395, 1981.

[10] R. Bolles and M. A. Fischler, "A ransac-based approach to model fitting and its application to finding cylinders in range data," in *Proc. IJCAI*, 1981, pp. 637–643.

[11] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (fpfh) for 3d registration," in *ICRA*, 2009, pp. 3212 –3217.

[12] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu, "Fast 3d recognition and pose using the viewpoint feature histogram," in *IROS*, 2010.

[13] E. Wahl, U. Hillenbrand, and G. Hirzinger, "Surflet-pair-relation histograms: a statistical 3d-shape representation for rapid classification," in *IEEE 3DIM*, 2003, pp. 474–481.

[14] U. Hillenbrand, "Pose clustering from stereo data," in *VISAPP International Workshop on Robotic Perception*, 2008, pp. 23–32.

[15] S. Winkelbach, M. Rilk, C. Schoenfelder, and F. Wahl, "Fast random sample matching of 3d fragments," in *PatternRecognition (DAGM 2004), Lecture Notes in Computer Science 3175*, 2004, pp. 129–136.

[16] C. Papazov and D. Burschka, "An efficient ransac for 3d object recognition in noisy and occluded scenes," in *ICCV*, 2010.

[17] D. Buchholz, S. Winkelbach, and F. M. Wahl, "Ransam for industrial bin-picking," in *ISR/Robotics 2010*, 2010, pp. 1317–1322.

[18] B. Drost, M. Ulrich, N. Navab, and S. Ilic, "Model globally, match locally: Efficient and robust 3D object recognition," in *IEEE CVPR*, San Francisco, CA, USA, June 2010.

[19] A. Aldoma, F. Tombari, L. Di Stefano, and M. Vincze, "A Global Hypotheses Verification Method for 3D Object Recognition," in *ECCV*, Firenze, Italy, 2012.

[20] A. G. Buch, D. Kraft, J.-K. Kamarainen, H. G. Petersen, and N. Krüger, "Pose Estimation using Local Structure-Specific Shape and Appearance Context," in *IEEE ICRA*, Karlsruhe, Germany, May 2013.

[21] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3d scenes," *IEEE PAMI*, vol. 21, no. 5, pp. 433–449, 1999.

[22] A. S. Mian, M. Bennamoun, and R. Owens, "Three-dimensional model-based object recognition and segmentation in cluttered scenes," *IEEE PAMI*, vol. 28, no. 10, 2006.

[23] M. Sun, G. Bradski, B.-X. Xu, and S. Savarese, "Depth-encoded hough voting for joint object detection and shape recovery," in *ECCV*, Heraklion, Crete, Greece, 2010, pp. 658–671.

[24] M. Ulrich, C. Wiedemann, and C. Steger, "Cad-based recognition of 3d objects in monocular images," in *IEEE ICRA*, 2009, pp. 1191–1198.

[25] Y. Chen, T.-K. Kim, and R. Cipolla, "Inferring 3D shapes and deformations from single views," in *ECCV*, 2010, pp. 300–313.

[26] F. Tombari, S. Salti, and L. Di Stefano, "A combined texture-shape descriptor for enhanced 3D feature matching," in *IEEE ICIP*, Sept. 2011.

[27] S. Kriegel, C. Rink, T. Bodenmüller, and M. Suppa, "Efficient next-best-scan planning for autonomous 3d surface reconstruction of unknown objects," *JRTIP*, pp. 1–21, 2013.

[28] P. J. Besl and N. D. McKay, "A Method for Registration of 3-D Shapes," *IEEE PAMI*, vol. 14, no. 2, pp. 239–256, 1992.