3D Object Recognition and Pose Estimation for Multiple Objects using Multi-Prioritized RANSAC and Model Updating

Michele Fenzi, Ralf Dragon, Laura Leal-Taixé, Bodo Rosenhahn, and Jörn Ostermann

Institute for Information Processing (TNT), Leibniz University Hannover, Germany

Abstract We present a feature-based framework that combines spatial feature clustering, guided sampling for pose generation, and model updating for 3D object recognition and pose estimation. Existing methods fails in case of repeated patterns or multiple instances of the same object, as they rely only on feature discriminability for matching and on the estimator capabilities for outlier rejection. We propose to spatially separate the features before matching to create smaller clusters containing the object. Then, hypothesis generation is guided by exploiting cues collected off- and on-line, such as feature repeatability, 3D geometric constraints, and feature occurrence frequency. Finally, while previous methods overload the model with synthetic features for wide baseline matching, we claim that continuously updating the model representation is a lighter yet reliable strategy. The evaluation of our algorithm on challenging video sequences shows the improvement provided by our contribution.

1 Introduction

3D object recognition is a well established field of research in computer vision, and feature-based approaches have become increasingly popular due to their robustness to clutter, occlusions, changes in scale, rotation and illumination. In the feature-based paradigm, as pioneered in [10,18], a 3D sparse point cloud representing the target object is reconstructed by applying Structure from Motion to features tracked over a set of training images. Once the model is obtained offline, on-line recognition and pose estimation is performed by matching the image features against the model features and solving the Perspective-n-Point problem for the 2D-3D correspondences. Given a set of correct matches, pose estimation is a well-solved problem, and various solutions have been devised [13,6].

Feature-based methods can be grouped on the basis of the feature used, *e.g.*, edges [7], shape [5], patches [17], interest points [10,18]. We share the same approach of the latter, as we use SIFT features as interest points [15]. Our choice is motivated by SIFT high discriminability and invariance towards rotation, scale and illumination changes, as evaluated in [16].

Recent approaches based on this paradigm rely on feature discriminability for correct matches and on the robust estimator capabilities for outlier rejection [2,11,12]. However, this presents numerous critical issues:

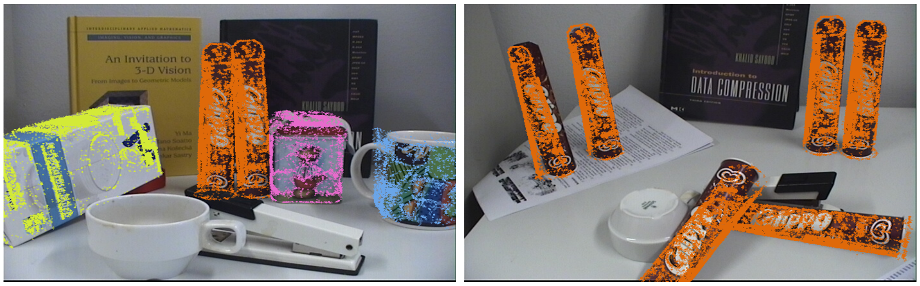

Figure 1. Performance of our method for challenging scenes with 5 objects (left) and 6 instances of the same object (right). (Figure best viewed in color)

1. Local patterns, although strongly discriminative *per se*, often appear in symmetric and repeated fashion. Feature descriptors at those locations are very similar and thus, the typical image-to-model discriminative matching employed by previous approaches rejects many correct matches. In a domino effect fashion, robust estimators work poorly when the inlier ratio drops, often providing wrong hypotheses if the number of trials is small.
2. Robust estimators, like RANSAC [8], generate pose hypotheses without exploiting any information contained in the model, thus making hypothesis generation prone to the potential inconsistency of the matches. In [11], a simple guided sampling based on co-visibility among correspondences is used, without investigating on other cues. In [14], priority is used in the matching stage by considering only the most recurring features. However, pose estimation still fails if these features cover just a small region of the object.
3. Since features are not perspectively invariant, object recognition fails in case of wide baseline matching. Many approaches handle this by adding synthetic features the training images [12,11]. However, this makes feature matching ambiguous as the number of features grows.

Our contribution is a fully automatic method that individually solves these drawbacks by combining inverse feature matching and spatial feature clustering with multiple instances detection (1), prioritized hypothesis generation (2) and model updating (3). By doing so, our system is able to reliably detect multiple objects and multiple instances of the same object, as shown in Figure 1.

1.1 Our Contribution

We present our contribution by correspondingly addressing the drawbacks given in the previous section.

1. To cope with repeated patterns and image-to-model discriminative matching, we propose to introduce an inverse matching paradigm, *i.e.*, to match the model against the image. Since this approach is still prone to fail when multiple instances of the same object are in the scene, we introduce a spatial feature clustering with multiple instances detection (Sec. 3).

2. As robust estimators rely on a completely random sampling, we propose a consistent guided sampling named Multi-Prioritized RANSAC (Sec. 4). It exploits individual and grouping cues in a probabilistic framework, in contrast to [4], in which samples are simply sorted on the basis of their matching score. In particular, we exploit off- and on-line information on the number of occurrences of each sample, 3D co-visibility among samples and temporal occurrence frequency.
3. To handle wide baseline matching, our solution is to continuously update the model description to adapt it to its on-line appearance (Sec. 5).

In Section 2, an overview of the off-line and on-line stages is given. In the following sections, the contributions outlined above are individually detailed. After experimental evaluation, we give a conclusion.

2 Overview

Off-line Stage Firstly, SIFT features are detected from a set of training images covering the object. Each view provides a set of features, $S_{v_i} = \{f_1, \dots, f_{N_i}\}$, where v_i is the view index. By tracking each feature over the entire set of views, multi-view correspondences are created and input to a Structure from Motion algorithm. The output is a 3D point cloud $\mathcal{M}$. Each point in $\mathcal{M}$ is augmented to take the form of the following 3D feature descriptor

$$\mathbf{X} = \{(x, y, z), \mathcal{F}, \mathcal{V}, l_f\}, \tag{1}$$

where (x, y, z) are the 3D coordinates of the point; $\mathcal{F} = \{f_1, f_2, \dots, f_n\}$ is the set of 2D feature descriptors located at the 2D positions to which the 3D point projects; $\mathcal{V} = \{v_1, v_2, \dots, v_n\}$ is the set of view indices where the 2D feature is visible. l_f is the index of the last frame where the 3D feature was detected as inlier during on-line operation; it is initially set to 0. Since 3D feature descriptors are highly redundant in case of long tracks, the point cloud appearance is compressed by using mean-shift clustering in the high-dimensional feature space. The 3D descriptor $\mathbf{X}$ now takes the following compressed form

$$\mathbf{X} = \{(x, y, z), \tilde{f}, \mathcal{V}, l_f\}, \tag{2}$$

where $\tilde{f}$ is the cluster representative.

On-line Stage Once the model database is assembled, 3D object recognition can be performed. For each frame t, image features are first detected and clustered on the basis of their location in the image. Each cluster is then verified for the presence of multiple instances of the same object, and possibly split in further clusters (Sec. 3). Then, for each database model $\mathcal{M}_j$, correspondences between its 3D descriptors and each cluster are established by applying SIFT matching. An *inverse matching* approach, *i.e.*, the model is matched against the cluster, is adopted. Once 3D-2D matches are obtained, pose estimation is performed with our novel Multi-Prioritized RANSAC approach (Sec. 4). In case of detection, the model appearance is updated by using the information recovered in the last frames (Sec. 5).

3 Spatial Feature Clustering

Using *inverse matching* against the whole set of image features S is disadvantageous as many false matches arise due to the low inlier ratio. Furthermore, when this strategy is used, multiple instances of the same object interfere with each other's recognition, preventing the system to detect some or even any instance.

The solution we propose is to cluster the image features before matching. Since features tend to naturally group over objects, individual objects can be isolated before matching. We use mean-shift clustering as there is no information on how many objects are in the scene. Therefore, S is spatially split into several clusters $S_1, \dots, S_q$. Thereby, the inlier ratio increases for the clusters containing target objects, and decreases otherwise. If several instances of the same object are spatially distant in the scene, they are effectively assigned to different clusters.

Nevertheless two drawbacks exist. Firstly, different objects can belong to the same cluster. In this case, the cluster is reconsidered for matching if the number of inliers is too small. Secondly, multiple instances that are spatially close in the image can belong to the same cluster. Our contribution treats the latter as follows.

3.1 Intra-cluster Detection of Multiple Instances

If multiple instances of the same object are grouped together, matching the model against that cluster fails. We propose to detect the instances by treating them as different views of the same object under epipolar geometry constraints.

For each feature, multiple correspondences within the same cluster are created by thresholding on their normalized scalar product. Each match shall identify two instances of the object. Let the matches set be $\mathcal{X} = \{\mathbf{m}_i\}_{i=1}^N$ where $\mathbf{m}_i = (\mathbf{m}_{i_1}, \mathbf{m}_{i_2})$. Given a set of putative hypothesis $\mathbf{F}_1, \dots, \mathbf{F}_M$, where $\mathbf{F}_j$ is a fundamental matrix, we define a residual vector for each match and hypothesis as

$$\mathbf{r}^i = \left[r_1^i \; r_2^i \; \dots \; r_M^i\right], \quad \text{where } r_j^i = \mathbf{m}_{i_1}^T \mathbf{F}_j \mathbf{m}_{i_2} \text{ and } i = 1, \dots, N. \tag{3}$$

Let $\tilde{\mathbf{r}}^i$ be $\mathbf{r}^i$ sorted in ascending order, it is possible to rank the M hypothesis according to the *preference* of each match, as described by [3]. Inliers for the same pair of instances are likely to share many common hypotheses at the top of their sorted residual vectors. To quantify the similarity between correspondences, the following measure is used

$$w(\mathbf{m}_i, \mathbf{m}_j) = \frac{1}{h}[\tilde{\mathbf{r}}_{1:h}^i \cap \tilde{\mathbf{r}}_{1:h}^j], \tag{4}$$

where w is the normalized number of hypotheses shared in the first h positions. To choose a minimal subset of size n, the first sample $\mathbf{s}_1$ is randomly selected. Then, to select the k-th sample, the remaining samples are first weighted as follows,

$$w_k(\mathbf{m}_p) = \prod_{i=1}^{k-1} w(\mathbf{m}_p, \mathbf{s}_i), \qquad \text{where } k = 2, \dots, n. \tag{5}$$

Then, the k-th sample is chosen according to $P_k(\mathbf{m}_p) > P_k(\mathbf{m}_q)$ if $w_k(\mathbf{m}_p) > w_k(\mathbf{m}_q)$, where $P_k(\mathbf{m}_p)$ is the probability of $\mathbf{m}_p$ being selected as the k-th sample. Each minimal subset feeds a RANSAC loop and the inlier set is retained if the consensus is large enough.

As a result, the multiple instances can now be isolated by splitting the feature cluster. K-means clustering is used here because the number of instances, provided by the number of inlier sets without repetition, is now known.

4 Object Recognition and Pose Estimation

Once feature clusters are established, object recognition and pose estimation can be performed. Firstly, the 3D descriptors of the model are matched against each feature cluster to produce a set of 3D-2D matches $(\mathbf{X}_i, \mathbf{x}_i)$. A projection matrix $\tilde{\mathbf{P}}$ is computed in order to minimize the sum of the reprojection errors between the 3D points $\{\mathbf{X}_i\}$ and the 2D points $\{\mathbf{x}_i\}$ in the image. Due to the presence of outliers among the putative matches, a robust approach as RANSAC is needed. However, in the basic RANSAC, hypotheses are generated from a minimal subset of randomly selected samples. No additional information regarding the importance of each sample and the relations among the samples is taken into account. We show that exploiting additional information can be highly beneficial.

4.1 Multi-Prioritized RANSAC

As a second contribution, we propose to exploit the information contained in our 3D feature descriptor to drive the minimal subset selection. Firstly, each sample $\mathbf{s} = (\mathbf{X}, \mathbf{x})$, *i.e.*, a 3D-2D match, receives a weight w_1 based on the number of training views n in which the 3D descriptor was visible,

$$w_1(\mathbf{s}) = n. \tag{6}$$

The motivation is that 3D descriptors appearing in many views represent more reliable information on the object appearance.

Secondly, geometrical inconsistency can affect the sample subset if the selected samples belong to 3D points that are not simultaneously visible. This can occur if objects have similar patterns on opposite sides. To avoid this, after the first sample $\mathbf{s}_1$ is chosen, each remaining sample $\mathbf{s}_i$ is further weighted as follows

$$w_2(\mathbf{s}_i, \mathbf{s}_1) = \frac{|\mathcal{V}_1 \bigcap \mathcal{V}_i|}{|\mathcal{V}_i|}, \tag{7}$$

where the numerator is the number of views shared by the current sample and the first sample, and the denominator is its total number of views. In other words, the co-visibility consistency of the samples is examined, assigning a null weight to samples that do not share any view in common with the first one.

A third weight is given by considering the temporal distance between the current frame t and the last frame l_f where the 3D descriptor was an inlier,

$$w_3(t, \mathbf{s}) = \frac{1}{t - l_f}. \tag{8}$$

Selecting samples which were inliers in frames close in time to the current one shall increase the inlier ratio of the minimal subset. Thus, minimal sampling is guided by the probability $P(\mathbf{s}) \propto w(\mathbf{s})$, where $w = w_1 w_2 w_3$, *i.e.*,

$$w(\mathbf{s}_i) \geq w(\mathbf{s}_j) \Rightarrow P(\mathbf{s}_i) \geq P(\mathbf{s}_j). \tag{9}$$

Given the minimal subset, a pose estimation via EPnP [13] and an eventual non-linear minimization is performed in a sample-and-test framework to estimate the pose $\tilde{\mathbf{P}}$ that best fits the matches. In Table 1, each weight is evaluated in terms of the average number of iterations needed to find, for the first time, at least 75% of the inliers and it is averaged over 1000 runs per frame on a sample sequence. Whereas in [11] only w_2 is used, we prove that w_1 is a stronger cue and their combination exceeds both. The best performance is obtained by far with the complete guided sampling, reducing the number of iterations by up to ten times as the inlier ratio decreases. Thus, our method is highly beneficial in applications where the permitted number of iterations is small.

Table 1. Mean and std. deviation of the number of iterations for several inlier ratios.

Inlier ratio	No weight	w_1	w_2	$w_1 w_2$	$w_1 w_2 w_3$
60%	39.9 ± 40.6	9.8 ± 9.8	11.9 ± 13.9	6.2 ± 6.77	$\mathbf{5.8} \pm \mathbf{6.2}$
50%	110.5 ± 113.3	19.2 ± 21.0	28.3 ± 30.4	12.9 ± 13.5	$\mathbf{9.4} \pm \mathbf{12.6}$
40%	309.0 ± 286.7	46.7 ± 53.9	89.4 ± 101.4	28.3 ± 30.2	$\mathbf{17.4} \pm \mathbf{19.9}$
30%	627.4 ± 515.0	113.2 ± 128.9	272.6 ± 276.7	71.5 ± 71.5	$\mathbf{19.0} \pm \mathbf{27.6}$
20%	1428.5 ± 1294.6	411.4 ± 395.2	1047.7 ± 899.8	302.0 ± 317.9	$\mathbf{29.1} \pm \mathbf{56.1}$

5 Model Updating

To improve recognition performance, our solution is a model updating step where its description is adapted to the current appearance. Given a successful detection, all the inliers $\mathbf{m_i} = (\mathbf{X}_i, \mathbf{x}_i)$ are considered. Each 2D feature $\mathbf{x}_i$ is added to the descriptor set $\mathcal{F}_i$ of $\mathbf{X}_i$. Then, each descriptor set is clustered as in the off-line stage, considering both the training view descriptors and the 2D features collected within the last k frames. By retaining the training views, drift is avoided.

The motivation for this contribution is twofold. Firstly, detection success is dependent on the current object pose as SIFT features are not perspectively invariant. Invariance is indeed rather limited, as its repeatability drops under 80% for an angular difference greater than 20°[15]. Therefore, object detection fails in case of wide baseline matching. Secondly, by updating the model description the model size remains constant, and it is more efficient than the brute force approach of adding features recovered from synthetic views [12,11]. In the latter, the increase in size and the many wrong matches generated by synthetic views having similar appearance, respectively, need additional countermeasures.

6 Experimental Results

3D object databases are usually composed by small objects on monotone background [9]. When challenging situations are envisaged [11], only recognition methods for still images can be tested. Since our system is designed for videos, we created several 200-frame-long sequences to evaluate the performance of our method. To raise the bar, we assembled a database of 10 household objects, comprising complex items like shoes or toy planes, reflection-prone objects like cups and objects with repetitive structures like milk boxes. The database and the sequences are available at [1]. The experiments focus first on the recognition of a single object in terms of pose accuracy and stability, and then on the recognition of multiple objects and multiple instances of the same object.

6.1 Pose Accuracy and Stability

For each object, we created three frame sequences by freely moving a calibrated camera around the object in challenging scenarios, envisaging occlusion, clutter and a combination of both. Four systems are compared in their performance: the paradigm system proposed in [10] (G&L), where only image-to-model matching and RANSAC is employed, and our system by sequentially adding spatial feature clustering (S.C.), Multi-Prioritized RANSAC (MP-R), and model updating.

The quantitative evaluation is given in terms of the Jaccard index:

$$J = \frac{A \cap A_{gt}}{A \cup A_{gt}}, \tag{10}$$

where A_{gt} is the ground-truth area in the current frame and A is the area of the hull determined by the 3D descriptors re-projected with the recovered pose. We avoided using the mean reprojection error as it can be non-meaningful. Firstly, because arbitrarily small errors can be obtained in RANSAC-like frameworks by tuning the reprojection error and the inlier thresholds. Secondly, because it is not robust to inconsistent poses due to ambiguous configurations. Tab. 2 shows the mean value and the standard deviation of the Jaccard index, as $\mu \pm \sigma$, evaluated over the 200 frames of each sequence. The mean value μ is considered as a measure for pose accuracy and the standard deviation σ for pose stability.

With respect to the state of the art, the combination of feature clustering and MP-RANSAC improves pose accuracy by 20% in terms of correct overlapping. Furthermore, it allows for successful detections over all frames, where the standard method fails in the more complex "Clutter+Occlusion" scenario. Updating the model provides improvement to pose accuracy mostly in the "Clutter+Occlusion" scenario. But its real benefit comes into play for stability, as it is increased by a factor of 2 in all scenarios. To show how pose stability benefits from updating the model, J is shown in Fig. 2 on a frame-by-frame basis for the object "Hexa Tea" in the "Occlusion" scenario. While pose accuracy is slightly improved, pose stability is significantly increased. A comprehensive set of sample pictures regarding the experiments is given in the supplemental material.

Table 2. J as $\mu \pm \sigma$ in the "Occlusion", "Clutter", "Occlusion+Clutter" scenarios

Object	G & L	Spat. Clust.	S.C. + MP-RANSAC	S.C. + MP-R + Updating
Hexa Tea	0.47 ± 0.15	0.63 ± 0.15	0.87 ± 0.13	$\mathbf{0.93 \pm 0.02}$
Cube Tea	0.68 ± 0.19	0.81 ± 0.12	0.91 ± 0.10	$\mathbf{0.93 \pm 0.03}$
Coffee	0.73 ± 0.19	0.86 ± 0.09	$\mathbf{0.95 \pm 0.02}$	$\mathbf{0.95 \pm 0.02}$
Flower Cup	0.67 ± 0.18	0.75 ± 0.16	0.78 ± 0.11	$\mathbf{0.87 \pm 0.06}$
Bear Cup	0.74 ± 0.21	0.76 ± 0.14	0.88 ± 0.07	$\mathbf{0.94 \pm 0.03}$
City Cup	0.66 ± 0.21	0.79 ± 0.13	0.94 ± 0.05	$\mathbf{0.97 \pm 0.02}$
Toy Plane	0.47 ± 0.16	0.67 ± 0.19	0.70 ± 0.16	$\mathbf{0.83 \pm 0.08}$
Milk	0.63 ± 0.17	0.69 ± 0.16	0.79 ± 0.18	$\mathbf{0.87 \pm 0.09}$
Calippo	0.52 ± 0.19	0.63 ± 0.21	0.75 ± 0.18	$\mathbf{0.85 \pm 0.14}$
Slipper	0.75 ± 0.16	0.78 ± 0.13	0.86 ± 0.11	$\mathbf{0.90 \pm 0.04}$
Average	0.63 ± 0.18	0.74 ± 0.15	0.84 ± 0.11	$\mathbf{0.90 \pm 0.05}$

Object	G & L	Spat. Clust.	S.C. + MP-RANSAC	S.C. + MP-R + Updating
Hexa Tea	0.61 ± 0.23	0.72 ± 0.15	0.92 ± 0.04	$\mathbf{0.93 \pm 0.02}$
Cube Tea	0.79 ± 0.16	0.83 ± 0.13	0.87 ± 0.12	$\mathbf{0.92 \pm 0.05}$
Coffee	0.53 ± 0.25	0.64 ± 0.19	0.83 ± 0.19	$\mathbf{0.91 \pm 0.03}$
Flower Cup	0.63 ± 0.18	0.69 ± 0.14	0.80 ± 0.14	$\mathbf{0.87 \pm 0.06}$
Bear Cup	0.62 ± 0.35	0.71 ± 0.25	0.88 ± 0.13	$\mathbf{0.93 \pm 0.03}$
City Cup	0.64 ± 0.27	0.79 ± 0.16	0.90 ± 0.11	$\mathbf{0.95 \pm 0.04}$
Toy Plane	0.58 ± 0.14	0.55 ± 0.18	0.62 ± 0.21	$\mathbf{0.79 \pm 0.12}$
Milk	0.67 ± 0.24	0.70 ± 0.19	0.88 ± 0.16	$\mathbf{0.90 \pm 0.08}$
Calippo	0.59 ± 0.29	0.77 ± 0.11	0.85 ± 0.06	$\mathbf{0.90 \pm 0.04}$
Slipper	0.72 ± 0.20	0.76 ± 0.17	0.90 ± 0.06	$\mathbf{0.92 \pm 0.03}$
Average	0.58 ± 0.23	0.72 ± 0.17	0.84 ± 0.12	$\mathbf{0.90 \pm 0.05}$

Object	G & L	Spat. Clust.	S.C. + MP-RANSAC	S.C. + MP-R + Updating
Hexa Tea	//	//	0.57 ± 0.29	$\mathbf{0.91 \pm 0.08}$
Cube Tea	//	0.47 ± 0.35	0.70 ± 0.26	$\mathbf{0.91 \pm 0.15}$
Coffee	//	0.65 ± 0.21	0.83 ± 0.16	$\mathbf{0.87 \pm 0.12}$
Flower Cup	//	0.58 ± 0.23	0.78 ± 0.13	$\mathbf{0.81 \pm 0.06}$
Bear Cup	//	0.61 ± 0.35	0.78 ± 0.26	$\mathbf{0.89 \pm 0.09}$
City Cup	//	0.53 ± 0.27	0.77 ± 0.22	$\mathbf{0.89 \pm 0.07}$
Toy Plane	//	//	0.58 ± 0.18	$\mathbf{0.76 \pm 0.10}$
Milk	//	0.51 ± 0.31	0.70 ± 0.22	$\mathbf{0.84 \pm 0.13}$
Calippo	//	0.55 ± 0.27	0.77 ± 0.17	$\mathbf{0.88 \pm 0.06}$
Slipper	//	0.74 ± 0.21	0.87 ± 0.16	$\mathbf{0.91 \pm 0.10}$
Average	//	0.58 ± 0.28	0.73 ± 0.18	$\mathbf{0.87 \pm 0.09}$

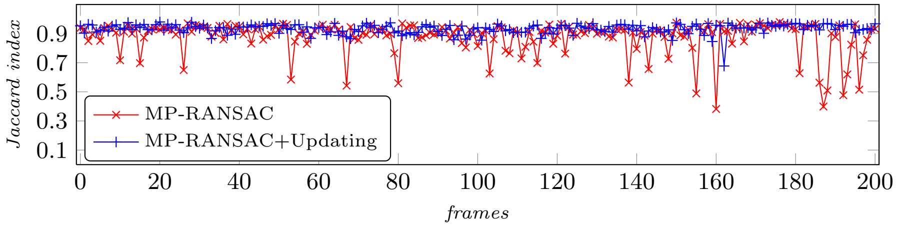

Figure 2. Frame-by-frame plot of J showing the stability improvement given by the model updating. (Figure best viewed in color)

6.2 Multiple Object Recognition

To test the performance of our system in recognizing multiple objects and multiple instances of the same object, we have created two different sequences. A calibrated camera moves freely in two scenarios: multiple objects with and without repetition. To evaluate the performance of our system, a recognition is deemed valid if $J > 0.5$, as proposed in [19]. In the first scenario, five different objects are present in the scene, while the second scenario envisages four instances of the same object and two other objects. The performance of our system is shown in Fig. 3 with respect to the ground truth. Sample frames are given in Fig. 4, while a more comprehensive set of sample pictures is given in the supplemental material. The performance regarding false positive and false negatives is remarkable, as no false negatives and very few false positives were found in both sequences.

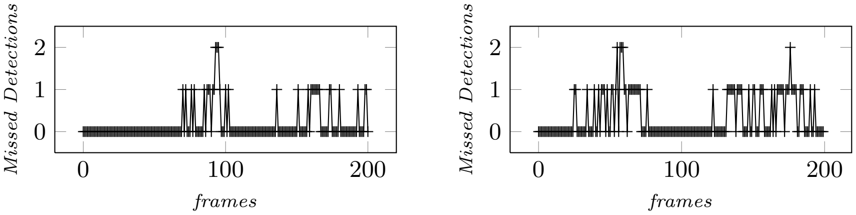

Figure 3. *Multiple Objects (left)*: Our system is able to detect all the objects in the scene in 169/200 frames. *Multiple Instances (right)*: All objects are detected in 135/200 frames. Missed detections are due to heavy blur and occlusion.

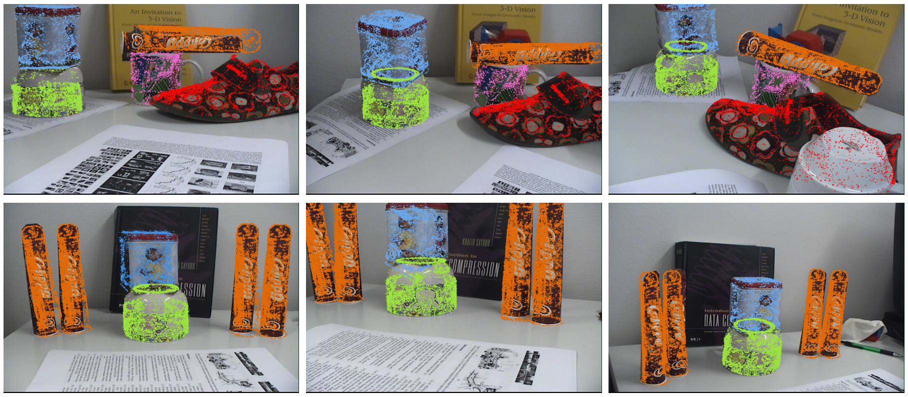

Figure 4. Three frames of the multiple objects (top) and of the multiple instances sequence (bottom). Objects are correctly recognized even in case of heavy clutter and occlusions. (Figure best viewed in color)

7 Conclusions

We showed that for feature-based methods clustering the features before matching makes the system robust in case of multiple patterns. Additionally, multiple object instances belonging to the same cluster can be detected and separated by ordering the features on the basis of their consistency to motion hypotheses.

We also proved that our usage of off- and on-line cues for guided sampling, like feature repeatability and temporal occurrence, is highly beneficial for applications with temporal constraints, as it drastically reduces the number of iterations needed to find a consistent pose. In addition, we showed that combining these two techniques and model updating improves the performance in terms of pose accuracy, from 60% to 90% overlap, and stability, by a factor of two. As a conclusion, by testing our method in challenging sequences [1], we proved that object recognition and pose estimation, irrespectively of the number of objects or object instances present in the scene, is significantly improved by our contribution.

References

1. http://www.tnt.uni-hannover.de/staff/fenzi/
2. Bhat, S., Berger, M.O., Sur, F.: Visual Words for 3D Reconstruction and Pose Computation. In: The First Joint 3DIM/3DPVT Conference (2011)
3. Chin, T.J., Yu, J., Suter, D.: Accelerated Hypothesis Generation for Multistructure Data via Preference Analysis. TPAMI (2012)
4. Chum, O., Matas, J.: Matching with PROSAC Progressive Sample Consensus. In: CVPR (2005)
5. Dambreville, S., Sandhu, R., Yezzi, A.J., Tannenbaum, A.: Robust 3D Pose Estimation and Efficient 2D Region-Based Segmentation from a 3D Shape Prior. In: ECCV (2008)
6. DeMenthon, D., Davis, L.: Model-Based Object Pose in 25 Lines of Code. IJCV (1995)
7. Drummond, T., Cipolla, R.: Real-Time Visual Tracking of Complex Structures. TPAMI (2002)
8. Fischler, M., Bolles, R.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. CACM (1981)
9. Geusbroek, J., Burghouts, G., Smeulders, A.: The Amsterdam Library of Object Images. IJCV (2005)
10. Gordon, I., Lowe, D.: What and Where: 3D Object Recognition with Accurate Pose. In: Toward Category-Level Object Recognition (2006)
11. Hsiao, E., Collet Romea, A., Hebert, M.: Making Specific Features Less Discriminative to Improve Point-based 3D Object Recognition. In: CVPR (2010)
12. Irschara, A., Zach, C., Frahm, J.M., Bischof, H.: From Structure-from-Motion Point Clouds to Fast Location Recognition. In: CVPR (2009)
13. Lepetit, V., Moreno-Noguer, F., Fua, P.: EPnP: An Accurate O(n) Solution to the PnP Problem. IJCV (2009)
14. Li, Y., Snavely, N., Huttenlocher, D.: Location recognition using prioritized feature matching. In: ECCV (2010)
15. Lowe, D.: Distinctive Image Features from Scale-Invariant Keypoints. IJCV (2004)
16. Mikolajczyk, K., Schmid, C.: A Performance Evaluation of Local Descriptors. TPAMI (2005)
17. Özuysal, M., Fua, P., Lepetit, V.: Fast Keypoint Recognition in Ten Lines of Code. In: CVPR (2007)
18. Rothganger, F., Lazebnik, S., Schmid, C., Ponce, J.: 3D Object Modeling and Recognition Using Local Affine-invariant Image Descriptors and Multi-view Spatial constraints. IJCV (2006)
19. Willems, G., Tuytelaars, T., Van Gool, L.: An Efficient Dense and Scale-Invariant Spatio-Temporal Interest Point Detector. In: ECCV (2008)