INTECH
open science | open minds

OPEN ACCESS
ARTICLE

International Journal of Advanced Robotic Systems

Towards a Stable Robotic Object Manipulation Through 2D-3D Features Tracking

Regular Paper

Sorin M. Grigorescu[1,*] and Claudiu Pozna[1,2]

1 Department of Automation, Transylvania University of Braşov, Braşov, Romania
2 Informatics Department, Széchenyi István University, Gyor, Hungary
* Corresponding author E-mail: s.grigorescu@unitbv.ro

Received 16 Apr 2012; Accepted 24 Jan 2013

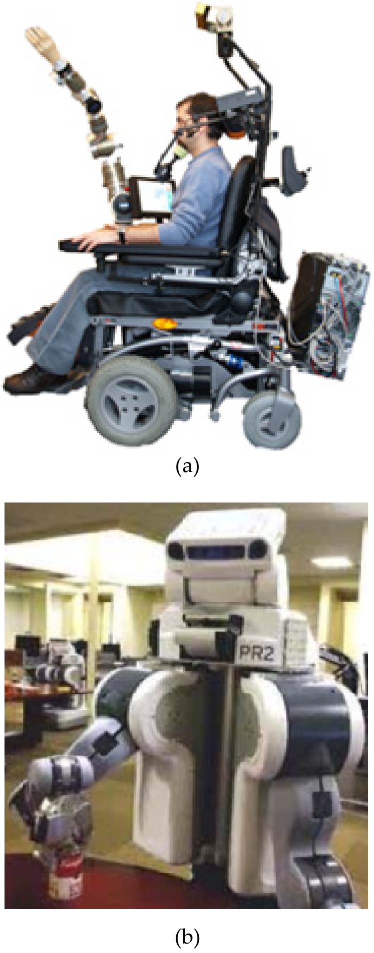
DOI: 10.5772/55952

Abstract In this paper, a new object tracking system is proposed to improve the object manipulation capabilities of service robots. The goal is to continuously track the state of the visualized environment in order to send visual information in real time to the path planning and decision modules of the robot; that is, to adapt the movement of the robotic system according to the state variations appearing in the imaged scene. The tracking approach is based on a probabilistic collaborative tracking framework developed around a 2D patch-based tracking system and a 2D-3D point features tracker. The real-time visual information is composed of RGB-D data streams acquired from state-of-the-art structured light sensors. For performance evaluation, the accuracy of the developed tracker is compared to a traditional marker-based tracking system which delivers 3D information with respect to the position of the marker.

Keywords Object Tracking, Robot Vision, Mobile Manipulation, Service Robotics

1. Introduction

One of the main tasks of service robotics systems operating in human environments, such as the FRIEND (*Functional Robot with dexterous arm and user-frIENdly interface for Disabled people*) [1] and PR2 (*Personal Robot* 2) [2] platforms displayed in Figure 1, is to reliably handle common household objects, such as plates, bottles or boxes, usually placed in heavy cluttered scenes. The common approach in many mobile manipulation scenarios is to recognize and calculate the pose (*position and orientation*) of the objects of interest [3], followed by further manipulative actions [4]. Throughout the manipulation procedure, there is no visual information available with respect to the imaged scene, that is, of the pose of the objects of interest and of the obstacles. Such a high uncertainty increases the risk of robotic handling failures if the state of the environment changes, namely if the poses of the objects, as well as of different obstacles present in the scene, varies.



(a)

(b)

Figure 1. Mobile manipulation platforms: (a) FRIEND, (b) PR2.

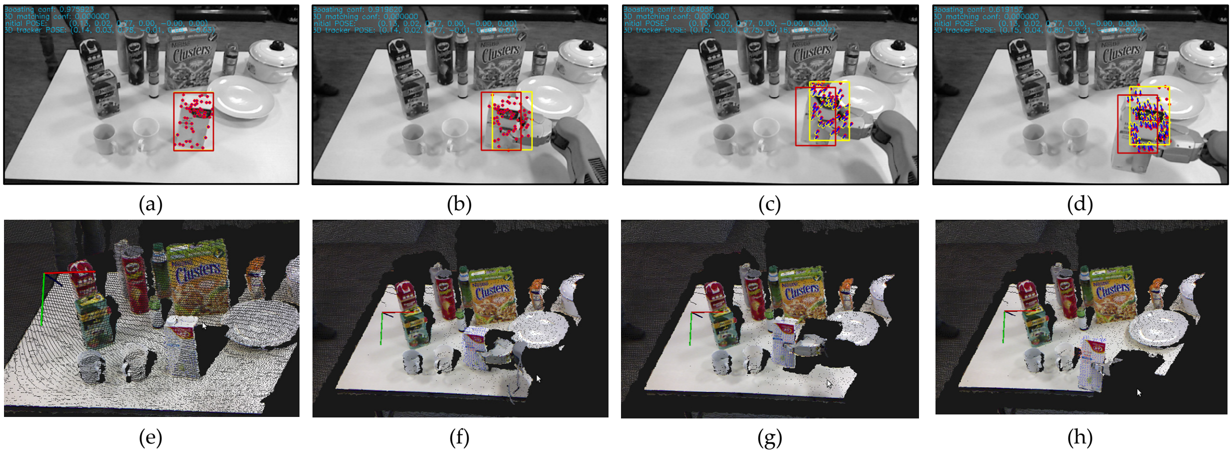
The main goal of tracking rigid bodies in robotics is to determine their pose with respect to a common reference coordinate system W, such that, if the state of the environment changes, the manipulation path of the robotic system is changed accordingly. This challenge is illustrated in Figure 2, where different tracking snapshots of an object of interest are illustrated. The object is tracked both at the 2D image level (see Figure 2[a-d]), as well as in the 3D point cloud data (Figure 2[e-h]) delivered by an RGB-D (*Red*, *Green*, *Blue*, *Depth*) MS Kinect® camera. The main advantage of such sensors is the calculated *depth map*, or *point cloud*, structure, which, along with the RGB image, describes the colour and real-world distances between the sensor and the visualized surfaces. Recently, they have become an intensively used perception sensor in a large number of indoor robotic applications.

In this work, we tackle the challenge of a single rigid object of interest pose O tracking with respect to W. The approach is based on a collaborative tracking framework which determines the pose of the object in the 2D image, while simultaneously tracking 3D point cloud features on the object of interest.

The contributions of the paper may be summarized as follows:

- development of a collaborative robotic tracking framework for fusing information from more trackers;
- implementation of a 2D-patch-based particle filtering algorithm for tracking a reference object of interest;
- occlusion detector based on ray casting using an RGB-D camera system;
- usage of object tracking for improving mobile manipulation scenarios.

As will be shown in the performance evaluation section, it is relatively difficult to compare the results of the proposed tracker, which by its nature processes 3D information, to existing 2D systems. Although there are a large number of image-based trackers which accurately deliver visual information in the 2D image domain, the tracker presented in this paper is intended to be used in service robotics application where 3D information regarding the objects of interest is imperative.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 2. Tracking during object grasping and manipulation. Snapshots of 2D features (a-d) and 3D point cloud (e-f) tracking.

1.1 Related Work

Although object tracking is a well-developed research area in the computer vision community, its application both in mobile manipulation and robotics in general [5] is quite restricted. Recently, Krainin et al. [6] applied the concept of object tracking during manipulation for building online 3D models of objects of interest. As they remark, the online learning and tracking of new objects is an imperative task for successful robotic manipulation scenarios. Also, tracking approaches have been proposed by Wang et al. [7] and Krainin et al. [6] for the purpose of hand tracking and modelling.

Teichman and Thrun [8] proposed a semi-supervised, boosting classification approach to the problem of track classification in dense 3D range data. The method uses a series of 2D-3D features, such as Spin images, *Histogram of Oriented Gradients* (HOG) and the object's oriented bounding box size.

A novel paradigm for training a binary classifier in the context of tracking has been proposed in [9, 10, 11] for the problem of pure 2D image *Region of Interest* (ROI) tracking. The learning process is guided by *positive* (P) and *negative* (N) constraints which restrict the labelling of the unlabelled set. P-N learning evaluates the classifier on the unlabelled data, identifies examples that have been classified in contradiction with structural constraints and augments the training set with the corrected samples in an iterative process. Another online boosting tracking technique has been proposed by Grabner and Bischof [12] and Stalder et al. [13]. These authors propose an on-line AdaBoost feature selection method. Contrasting them with traditional features such as Haar wavelets [14], Wang et al. [17] proposed the usage of so-called *superpixels* for the purpose of object tracking.

Boosting is a machine learning technique used in a variety of computer vision applications such as image segmentation, text and object recognition, natural language processing, medical diagnostics, etc. In this paper, a boosting approach similar to the one described by Grabner and Bischof [12] has been used to track the object of interest in the 2D image domain.

One of the first applications of boosting in the area of computer vision was conducted by Viola and Jones [14] for the purpose of fast object detection. In this research, boosting, in its *AdaBoost* variant [19], was used for the purpose of feature selection. Traditionally, the classifier training approach was performed off-line using a large amount of training samples [14]. In this work, the focus is on object tracking without using off-line training. Hence, the training examples are composed of the first streams of images from the image acquisition system. The tracked features are composed of *Haar-like* patches.

In a number of recent papers, such as that by Choi et al. [15], the output results from more trackers are fused together using a weighting scheme to improve the performance of the overall tracking procedure. The same concept has been successfully applied by Yang et al. [16] in relation to the problem of tracking the motion of the human heart. In Yang et al. [16] a collaborative tracking framework is suggested.

Tracking has also been heavy investigated in relation to camera pose and the dense 3D reconstruction of human environments. The DTAM (*Dense Tracking and Mapping*) system proposed by Newcombe et al. [17] relies not on feature extraction, but on dense every-pixel processing. As a single hand-held RGB-D camera passes over a static scene, detailed textured depth maps at selected keyframes are estimated in order to produce a surface patchwork with millions of vertices.

2. A collaborative tracking framework

The collaborative nature of the tracking framework is defined within the classical Bayesian tracking approach. Let x_t be the state, that is, the true 6-DoF (*Degrees of Freedom*) pose of the tracked rigid body, or object, at time instance *t*:

$$\mathbf{x}_t = [x_t \quad y_t \quad z_t \quad \varphi_t \quad \psi_t \quad \theta_t], \tag{1}$$

where x_t, y_t and z_t give the 3D position along the three Cartesian axes, while φ_t, ψ_t and θ_t represent the orientation.

The tracking challenge can be defined as the estimation of the posterior probability $p(\mathbf{x}_t \mid \mathbf{z}_{1:t})$, where $\mathbf{z}_{1:t}$ are the past t poses, or state measurements:

$$\mathbf{z}_{1:t} = \{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_t\}. \tag{2}$$

Sequential Bayesian tracking follows a Markov modelling approach, where the prediction step is defined as:

$$p(\mathbf{x}_t \mid \mathbf{z}_{1:t-1}) = \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) \cdot p(\mathbf{x}_{t-1} \mid \mathbf{z}_{1:t-1}) d\mathbf{x}_{t-1} \tag{3}$$

The update state is defined as:

$$p(\mathbf{x}_t \mid \mathbf{z}_{1:t}) = p(\mathbf{z}_t \mid \mathbf{x}_t) \cdot p(\mathbf{x}_t \mid \mathbf{z}_{1:t-1}). \tag{4}$$

In order to improve the tracking capability, we propose a set of collaborative trackers that can take advantage of each other's data. The tracking framework is built around a set of two trackers:

- a 2D *patch-base* tracker $p(r_1)$ which tracks a reference patch in the RGB-D input stream;
- a 2D *point-base* tracker $p(r_2)$ that aims to track 2D point features in the 2D image domain, as well as determining direct *2D-to-3D* correspondences for estimating the 3D pose of the tracked object.

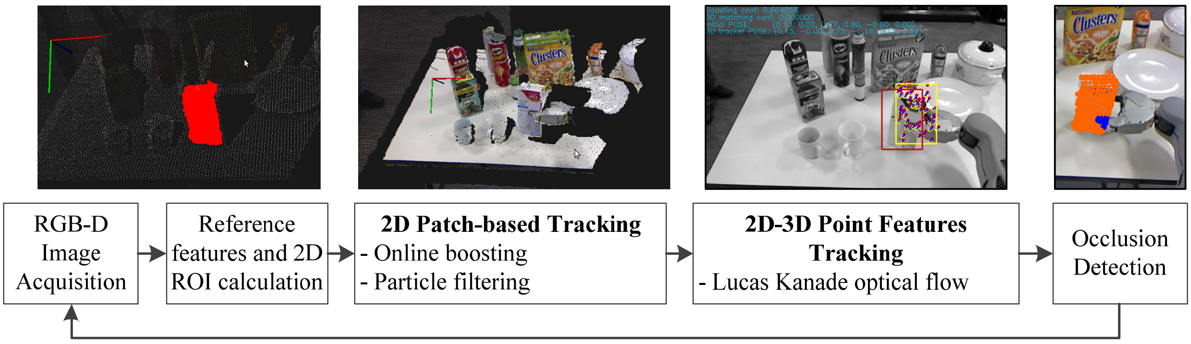


Figure 3. Block diagram of the proposed 2D-3D object tracking approach for robotic grasping and manipulation (best viewed in colour).

Both trackers contribute to the final state estimate as:

$$p(\mathbf{z}_t \mid \mathbf{x}_t) = p(\mathbf{x}_t \mid \mathbf{z}_t,\ p(r_1)) + p(\mathbf{x}_t \mid \mathbf{z}_t,\ p(r_2)) \qquad (5)$$

As suggested by Yang et al. [16], the information from both trackers may be fused as:

$$\mathbf{x}_t = \arg\max_{\mathbf{x}_t} p(\mathbf{x}_t \mid \mathbf{z}_{1:t}) \qquad (6)$$

3. 2D-3D Object Tracking

The block diagram of the proposed tracking loop for improving the mobile manipulation capabilities of service robots is shown in Figure 3. The first step in the diagram is to acquire a stream of RGB-D data, that is, data of images with corresponding depth information provided by structured light sensors (e.g., MS Kinect®, Asus®) or stereo cameras (e.g., Point Grey's Bumblebee®).

The reference features are calculated from an object recognition module. In this paper, a clustering object detection system is used, which segments objects of interest on flat surfaces such as tables [18]. The determined 3D object cluster is further projected on the 2D image, and an initial tracking patch is calculated in the form of an ROI. As will be explained, the initial patch is tracked using an on-line boosting method. Parallel with the patch-based tracker, an optical flow system determines the 2D-3D correspondences between consecutive frames. Finally, an occlusion detector evaluates possible object occlusions.

3.1 2D Patch-Based Tracking

A patch is considered an ROI in the 2D image that is being tracked by a specific tracker. This patch provides a search area for the second tracker, which establishes the 2D-3D point correspondences and estimates the transformation between consecutive poses of the object of interest. For tracking the initial patch, an on-line boosting learning technique has been implemented, as will be further explained below.

The boosting approach is a general framework used to improve the accuracy of a certain machine learning algorithm. This is performed by combining a weighted voting scheme using N hypotheses which have been generated by a repeated training built around different subsets of training data. A boosting classifier is composed of so-called *weak* and *strong* learners, or *classifiers*:

- **Weak learner**: has to perform only slightly better than random guessing, that is, its overall classification error has to be slightly smaller than 50%. The hypothesis h_{weak} is obtained from a learning algorithm.
- **Strong learner**: from a set of N weak learners, a strong learner, or classifier, is obtained as a linear combination of weak learners:

$$h^{strong}(\mathbf{x}) = sign(conf(\mathbf{x})), \qquad (7)$$

where $\mathbf{x}$ is a data sample.

The $conf(\cdot)$ operator is a confidence measure that provides the classification precision of the boosting learner:

$$conf(\mathbf{x}) = \sum_{n=1}^{N} \alpha_n \cdot h_n^{week}(\mathbf{x}) \qquad (8)$$

where α_n is the assigned weight:

$$\alpha_n = \frac{1}{2} ln\left(\frac{1-e_n}{e_n}\right). \qquad (9)$$

The basic boosting classification algorithm works as follows. Given a set of training samples:

$$\chi = \{\langle \mathbf{x}_1, y_1\rangle, \langle \mathbf{x}_2, y_2\rangle, \dots, \langle \mathbf{x}_L, y_L\rangle \mid \mathbf{x}_i \in \mathbb{R}^m, y_i \in \{-1, +1\}\}, \quad (10)$$

where L is the number of training samples, with positive $+1$ and negative -1 labels, and an initially uniform distribution over the examples:

$$p(\mathbf{x}_i) = \frac{1}{L}. \qquad (11)$$

A weak learner h_{week} is trained based on $p(\mathbf{x})$. Based on the error e_n the weak classifier h_{week} is assigned a weight α_n. $p(\mathbf{x})$ is updated as the probability of the misclassified samples increases. If a sample is correctly classified, the corresponding weight is decreased, such that the algorithm will focus on difficult samples. This procedure is repeated until the algorithm converges on a certain stopping criterion.

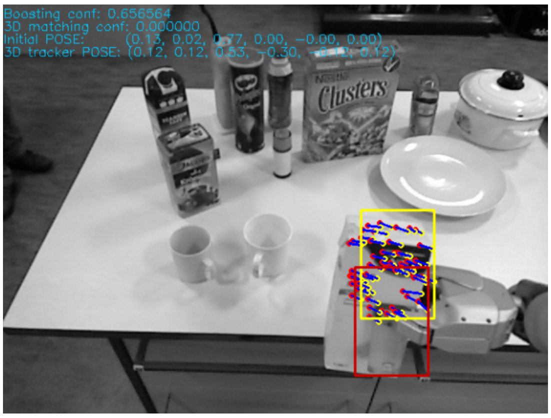
In the case dealt with in this paper, which concerns on-line boosting, the initial training set is composed only of an initial set of RGB-D frames. From these first frames, an initial training set composed of a series of Haar features is built. Further, the classifier is trained on-line by redetecting the tracked features.

At the 2D image level, the on-line boosting performances have been improved using a classical *particle filtering* framework [20]. In our experiments we have used a non-linear motion model governed by a set of 200 particles and a constant acceleration of the object.

The boosting tracking architecture provides a 2D ROI in the RGB image, which can be used to track the 3D pose of an object of interest. As will be further detailed below, this pose is obtained from a *Lucas-Kanade* (LK) optical flow estimation system [21].

3.2 Optical Flow for 3D Object Pose Estimation

The 3D pose of the reference point cloud is calculated based on the 2D ROI obtained from the boosting method and the available 3D point cloud data. 2D point features from the ROI are calculated based on the LK point tracking method [22], as shown in Figure 4. The 2D points used in LK tracking are extracted through the Harris corner detector, followed by correspondence matching using a traditional cross-correlation similarity measure. In order to account for larger motions of the object, a pyramid-based segmentation is used.



```
Boosting conf: 0.656564
3D matching conf: 0.000000
Initial POSE:      (0.13, 0.02, 0.77, 0.00, -0.00, 0.00)
3D tracker POSE: (0.12, 0.12, 0.53, -0.30, -0.12, 0.12)
```

Figure 4. LK tracking displayed as blue lines between two consecutive feature points. The two trackers are shown as red (patch-based tracking) and yellow (LK tracking) rectangles (best viewed in colour).

For each tracked 2D point in the image, a corresponding 3D point in the point cloud data is available, that is, there exists a direct 2D to 3D correspondence between the points obtained from the optical flow and the 3D point available in the point cloud delivered by the RGB-D system. Keeping this in mind, the consecutive 3D pose of the reference point features P_{ref} is determined using a *Singular Value Decomposition* (SVD) rigid body transformation method.

3.3 Occlusion Detection

One of the main components of the proposed 2D-3D tracker is the occlusion detection system. Its goal is to restart the tracker in case certain obstacles occlude the object of interest. In order to detect occlusions in real time, the ray-casting approach has been used. This means the reference point features P_{ref} are reprojected at every frame, based on the depth information available from the RGB-D sensor. The obtained 2D reprojections are used to search the current point cloud P_i for occlusions, that is, if a point in P_i has a depth smaller than its correspondence in P_{ref}, that specific point in the tracked cloud is occluded. An example of occlusion detection can be seen in Figure 5. An occlusion is considered to have taken place if the number of detected point occlusions exceeds a specific threshold value.

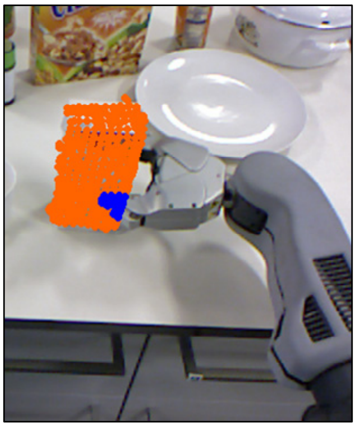


Figure 5. Occlusion detection. Orange points: reprojection of the tracked point features. Blue points: occlusion points coming from the intersection with the robotic arm (best viewed in colour).

The experiments considered two types of occlusion: objects occluded by other objects and the occlusion of an object of interest by the manipulator arm. In the second case, the major occlusion is produced by the gripper of the robotic arm. Since the occlusion detection system is based on ray-casting, it can be stated that the detection is invariant to the shape of the tracked object or of the objects present in the scene.

4. Visual Pipeline and Experimental Results

The processing pipeline for the proposed tracking system is implemented within the *Robotic Operating System* (ROS) framework. The input to the tracking module represents a real-world metric-registered point cloud together with its

RGB image correspondence. The processing pipeline within ROS is illustrated in Figure 6.

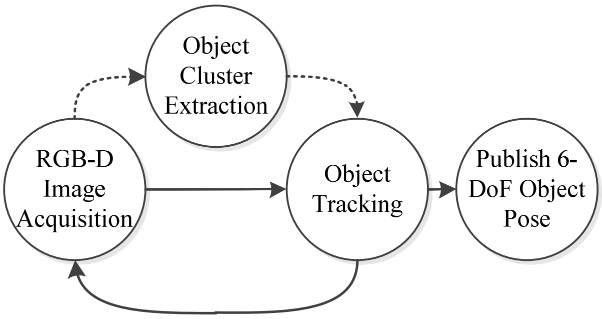


Figure 6. Object tracking loop implementation within the ROS operating system.

The evaluation of the overall visual tracking system is performed with respect to the real 3D poses of the objects of interest. Although a lot of 2D image-based object tracking methods exist, the literature is relatively scarce on 3D object trackers, which are a mandatory requirement for service robots operating in uncontrolled real-world conditions. One of the most reliable 3D object tracking systems, which relies only on 2D image information, is the ARToolKit library [23], against which the collaborative tracker presented in this paper has been evaluated.

The real 3D positions and orientations of the objects of interest were manually determined using the following setup. On the imaged scene, a visual marker, considered to be the ground truth information, was installed in such a way that the poses of the objects could be easily measured with respect to the marker. The 3D pose of the marker was detected using the ARToolKit library, which provides subpixel accuracy estimation of the marker's location with an average error of approx. 5 mm [23]. By calculating the marker's 3D pose, a ground truth reference value for camera position and orientation estimation could be obtained using the inverse of the marker's pose matrix. Further, the positions of the object's features were calculated using the proposed system. Both results, i.e., the 2D and 3D poses, were compared to the ground truth data provided by the ARToolKit marker. The 2D image marker position was calculated using its reprojection in the 2D image plane. As can be seen from Figure 7, the 2D and 3D position errors are within a tolerable range. The statistical analysis of the results is summarized in Table 1. It is important to emphasize here that the ARToolKit tracker was considered to be a reference value, or ground truth, against which the proposed approach was measured. In many situations, the classical marker-based approach fails to deliver proper pose estimation because of different lighting and surface reflection phenomena.

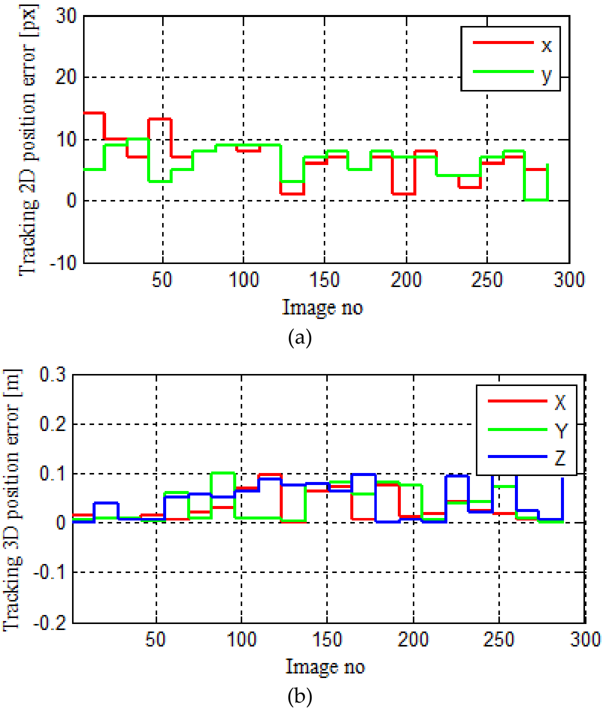


Figure 7. Variations in the calculated 2D and 3D poses with respect to the reference ARToolKit marker (best viewed in colour).

	x [px]	y [px]	X [m]	Y [m]	Z [m]
Max. err.	14	10	0.0967	0.0988	0.0958
Mean	6.6818	6.4091	0.0281	0.0350	0.0459
Std. dev.	3.3436	2.5196	0.0253	0.0339	0.0358

Table 1. Statistical results of errors between the proposed and the ARToolKit marker based 3D tracking system.

We have considered a processing cycle to begin with RGB-D data acquisition and end with occlusion detection. In the experimental setup, a calibrated MS Kinect® structured light sensor is used to acquire a sequence of 300 indoor images. The average computational time needed by a processing cycle is just over 400 ms, which is low enough to consider the proposed system as a real-time one for the considered case of object manipulation in domestic environments. The implemented tracking architecture has been tested on a typical portable computer running a 64-bit UNIX operating system on an Intel® i3 dual core CPU, each processor having a 2.40GHz clock speed.

Throughout the object handling routine, the required tracking accuracy is strictly dependent on the manipulative task and the configuration of the robot. In particular, the wider the opening angle of the gripper and the higher the number of degrees of freedom of the manipulator arm, the lower the tracking accuracy. In the presented experiments, a tracking error lower then $0.01m$ was required.

5. Conclusions

In this paper, a 2D-3D object features tracking system has been proposed. It has been proven to stabilize mobile manipulation and provide accuracy in tracking rigid household objects during robotic handling. The proposed approach has been built around a collaborative tracking framework which fuses information from more available trackers. As future work, the author will consider the extension of the collaborative framework with new state-of-the-art trackers in order to improve the accuracy of the proposed system. Also, the speed enhancement of the proposed system using state-of-the-art parallel processing equipment, such as FPGAs and GPUs, would significantly decrease the processing time.

6. Acknowledgment

Sorin Grigorescu was supported by the Sectoral Operational Programme, Human Resources Development (SOP HRD), financed by the European Social Fund and the Romanian Government under project no. POSDRU/89/1.5/S/59323. Claudiu Pozna was financed by the Széchenyi István University. The authors would like to thank Prof. Michael Beetz and Dejan Pangercic for their support and constructive ideas.

7. References

[1] S.M. Grigorescu, T. Lüth, C. Fragkopoulos, M. Cyriacks and A. Gräser. “A BCI Controlled Robotic Assistant for Quadriplegic People in Domestic and Professional Life,” Robotica, Cambridge, vol. 30, no. 3, pp. 419–431, May 2012.

[2] M. Beetz, F. Stulp, P. Esden-Tempski, A. Fedrizzi, U. Klank, I. Kresse, A. Maldonado and F. Ruiz. “Generality and legibility in mobile manipulation: Learning skills for routine tasks,” Autonomous Robots, vol. 2010, no. 28, pp. 21–44, 2009.

[3] S.M. Grigorescu, G. Macesanu, T. Cocias, D. Puiu and F. Moldoveanu, “Robust camera pose and scene structure analysis for service robotics,” Robotics and Autonomous Systems, vol. 59, pp. 899–909, 2011.

[4] S. Ekvall and D. Kragic. “Robot Learning from Demonstration: A Task-level Planning Approach”, International Journal of Advanced Robotic Systems, ISSN: 1729-8806, InTech, Available from: http://www.intechopen.com/journals/international_journal_of_advanced_robotic_systems/ robot_learning_from_demonstration__a_task-evel_planning_approach, 2008.

[5] K. Kuehnlenz and M. Buss. “Multi-focal Vision and Gaze Control Improve Navigation Performance, International Journal of Advanced Robotic Systems”, ISSN: 1729-8806, InTech, Available from: http://www.intechopen.com/journals/international_journal_of_advanced_robotic_systems/multi-focal_vision_and_gaze_control_improve_navigation_performance, 2008

[6] M. Krainin, P. Henry, X. Ren and D. Fox. “Manipulator and object tracking for in-hand 3D object modeling”. Int. J. Rob. Res., 30:1311–1327, September 2011. ISSN 0278-3649. doi: http://dx.doi. org/10.1177/0278364911403178. URL http://dx.doi.org/10.1177/0278364911403178.

[7] S. Wang, H. Lu, F. Yang and M.H. Yang. “Superpixel tracking”. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), Barcelona, Spain, November 2011.

[8] A. Teichman and S. Thrun. “Tracking-based semi-supervised learning”. In: Proceedings of Robotics: Science and Systems, Los Angeles, CA, USA, June 2011.

[9] Z. Kalal, J. Matas and K. Mikolajczyk. “Online learning of robust object detectors during unstable tracking”. In: Proceedings of the 3rd On-line Learning for Computer Vision Workshop, Kyoto, Japan, October 2009.

[10] Z. Kalal, J. Matas and K. Mikolajczyk. “PN learning: Bootstrapping binary classifiers by structural constraints”. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, June 2010.

[11] Z. Kalal, K. Mikolajczyk and J. Matas. “Forward-backward error: Automatic detection of tracking failures”. In: Proceedings of the International Conference on Pattern Recognition, Istanbul, Turkey, August 2010.

[12] H. Grabner and H. Bischof. “On-line boosting and vision”. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, NY, USA, June 2006.

[13] S. Stalder, H. Grabner and L. Van Gool. “Beyond semisupervised tracking: Tracking should be as simple as detection, but not simpler than recognition”. In: OLCV 09: 3rd On-line learning for Computer Vision Workshop, Kyoto, Japan, September 2009.

[14] P. Viola and M. Jones. “Rapid object detection using a boosted cascade of simple features”. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Hawaii, 2001.

[15] W. Choi, C. Pantofaru and S. Savarese. “Detecting and tracking people using an RGB-D camera via multiple detector fusion”. In: Workshop on Challenges and Opportunities in Robot Perception, at the International Conference on Computer Vision (ICCV), Barcelona, Spain, November 2011.

[16] L. Yang, B. Georgescu, Y. Zheng, Y. Wang, P. Meer and D. Comaniciu. “Prediction based collaborative trackers (PCT): A robust and accurate approach toward 3D medical object tracking”. IEEE Trans. Med. Imaging, 30(11):1921–1932, 2011.

[17] R.A. Newcombe, S.J. Lovegrove and A.J. Davison. “DTAM: Dense tracking and mapping in real-time”. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), Barcelona, Spain, November 2011.

[18] R.B. Rusu and S. Cousins. “3D is here: Point cloud library (PCL)”. In: International Conference on Robotics and Automation, Shanghai, China, 2011.

[19] Y. Freund and R.E. Schapire. “A decision theoretic generalization of on-line learning and an application to boosting”. J. Comput. Syst. Sci., 55:119–139, August 1997. ISSN 0022-0000. doi: 10.1006/jcss.1997.1504. URL http://dl.acm.org/citation.cfm?id=261540.261549.

[20] P. Azad, D. Munch, T. Asfour and R. Dillmann. “6-DOF Model-based Tracking of Arbitrarily Shaped 3D Objects”. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, May 2011.

[21] J. Shi and C. Tomasi. “Good features to track”. Computer Vision and Pattern Recognition, 1994. Proceedings CVPR ’94, 1994 IEEE Computer Society Conference on, pp. 593–600, June 1994. ISSN 1063-6919. doi: 10.1109/CVPR.1994.323794. URL http://dx.doi.org/10.1109/CVPR.1994.323794.

[22] K. Souhila and A. Karim. “Optical Flow based Robot Obstacle Avoidance, International Journal of Advanced Robotic Systems, ISSN: 1729-8806, InTech, Available from: http://www.intechopen.com/journals/international_journal_of_advanced_robotic_systems/optical_flow_based_robot_obstacle_avoidance, 2008.

[23] P. Malbezin, W. Piekarski and B.H. Thomas. “Measuring ARToolKit Accuracy in Long Distance Tracking Experiments”. In: 1st Int. Augmented Reality Toolkit Workshop, Darmstadt, Germany, September 2002.