Robotic System Capable of Identifying Objects Indicated by Pointing Gestures

Juris Klonovs, Dennis Herzog, Mikkel Rath Pedersen, Bjarne Großmann, Lazaros Nalpantidis, Volker Krüger

Robotics, Vision and Machine Intelligence Lab.,
Department of Mechanical and Manufacturing Engineering,
Aalborg University Copenhagen
Email: {juris,deh,mrp,bjarne,lanalpa,vok}@m-tech.aau.dk

***Abstract*—This work presents a robotic system able to visually segment an unknown object that was indicated by a human through a pointing gesture. The robot uses RGB-D sensors to observe the human and find the 3D point indicated by the pointing gesture. The system can use this point to initialize a fixation-based, fast object segmentation algorithm, inferring thus the outline of the whole object. A series of experiments with different objects and pointing gestures show that both the recognition of the gesture, the extraction of the pointing direction in 3D, and the object segmentation perform robustly. The discussed system can provide the first step towards more complex tasks, such as object recognition, grasping or learning by demonstration with obvious value in both industrial and domestic settings.**

I. Introduction

As robots are getting more autonomous their communication with humans is becoming crucial. In industrial and household environments, non robot-expert users would need to interact with robots, giving them instructions or requesting their assistance. The use of gestures provides a natural means of communication and has been widely adopted for intuitive Human-Robot Interaction (HRI). Among the most basic but also essential gestures is pointing. Pointing towards an object, a tool, or a person should make it/him the center of attention, and any related request should implicitly involve it/him.

Recognizing the target of a pointing gesture is innate for us humans, but is absolutely not trivial for robotic systems; especially when unknown objects are considered. It involves 3D perception of the environment and segmenting the corresponding object or person out of the rest of the scene. The robust transition from one single point (as resulting from the closest physically occupied image/volume element across the direction of the pointing gesture) to the whole object or person around that point is still an open research topic.

This work discusses a robotic system able to infer the outline of an unknown object out of a pointing gesture. More specifically, this work considers the mobile manipulator robot "Little Helper", equipped with two RGB-D sensors, a Microsoft Kinect sensor and an Asus Xtion sensor, for HRI purposes. The scenario involves a user (e.g. a patient at home or an industrial worker) pointing towards an object, which is previously unknown to the robot. Our system can recognize that a pointing gesture is performed, staying idle or performing other activities in the meantime. Then, it captures a depth and RGB image that contain the point where the pointing direction intersects with the closest physical object. This point is used to initialize a fixation-based, fast object segmentation algorithm. This algorithm initially transforms the image to the log-polar domain using the given image point as a pole and applies Graph Cut and Grab Cut optimization steps, taking into consideration both color and depth information. The final result is the outline of the corresponding object in the image.

Fig. 1. The considered scenario: human is indicating an object to the robot through a pointing gesture

The discussed system can provide the first step towards more complex tasks, such as object recognition, grasping or learning by demonstration with obvious value in both industrial and domestic settings.

A. Related Works

The use of pointing gestures to interface with computer systems and robots is not a new concept [1]–[3]. In [1] a real-time system able to detect and interpret pointing gestures, performed with one or both arms, is presented. The system then deduces the pointed direction by extrapolating the line between the users eye and fingertip. Furthermore, in [4] the authors propose a method for perceiving pointing gestures using a Time-of-Flight camera and train a model of pointing directions using Gaussian Process Regression. Recently, the work by Quintero et al. [5] proposed an interface based on the Kinect sensor for selecting by pointing in a 3D real-world situation, where the user points to a target object or location and the interface returns the 3D position coordinates

of the target. They performed three experiments based on their interface to study precision and accuracy of human pointing in typical household scenarios: pointing to a "wall", pointing to a "table", and pointing to a "floor".

Naturally, the next step of a pointing gesture is the segmentation of the pointed object, out of a whole image. Object segmentation can be considered as an instance of the more general problem of image segmentation [6]. The recent work of Mishra and Aloimonos [7], [8] proposed the use of polar transformation and then Graph Cut segmentation of objects depicted in single images, achieving very accurate results. This algorithm largely relies on accurate detection of edges in the image. Within the framework of Mishra et al., the possibility of including disparity or optical flow information has been also considered. On the other hand, the work of [9] proposes a simple way to combine information coming from sequences of images, gathered by mobile robots. Along the same path, the work of [10] is using the same underlying algorithm as the previous works, but also proposes a symmetry-based technique to choose suitable fixation points. Furthermore, the works of [11] and [12] perform accurate object segmentation, but again they rely on an initial rich 3D representation of the scene. Finally, GrabCut [13] is an efficient segmentation method, but it requires the definition of a coarse mask containing the object in order the segmentation process to be initialized.

II. Implementation

This work considers the mobile manipulator robot "Little Helper", equipped with two RGB-D sensors, in particular the Microsoft Kinect and the Asus Xtion cameras, for HRI purposes. The HRI scenario involves a user pointing towards an object, which is previously unknown to the robot (see Fig.2). The Microsoft Kinect camera, which is mounted on top of the mobile platform, is intended to capture colour and depth information of objects placed in front of the robot. The Asus Xtion camera, which is mounted on top of a pole high above the robot, is intended to monitor human body gestures in front of the robot and thus has a broader view comparing to the Kinect camera. The graphical user interface (GUI) is visualised on the large LCD screen, which is mounted on the front of the mobile platform. The GUI is specifically designed to provide an intuitive visual feedback to the user, which helps to adjust the pointing direction towards an object more precisely. The core system is set up on a PC using OpenCV [14] and OpenNI [15] libraries with integrated middleware ROS (Robot Operating System) [16], which is embedded inside the "Little Helper" platform, with an *Intel®Xeon®* CPU *E5620* @ 2.40GHz×16 and 8 GB of RAM with *Ubuntu* 64-bit operating system.

Our system is divided into two main steps. In the first step, it can recognize that a pointing gesture is performed, staying idle or performing other activities in the meantime. If the pointing gesture is performed, it captures a depth and RGB image that contain the point where the pointing direction intersects with the closest physical object. In the second step, this point is used to initialize a fixation-based, fast object segmentation algorithm [17]. The final result is the outline of the identified object of interest in the image. The output information can be useful for initialising more complex tasks, such as object recognition, as well as manipulation tasks, such as grasping, which can be further executed by the KUKA lightweight robot arm mounted on top of the mobile manipulator platform (see Fig.2).

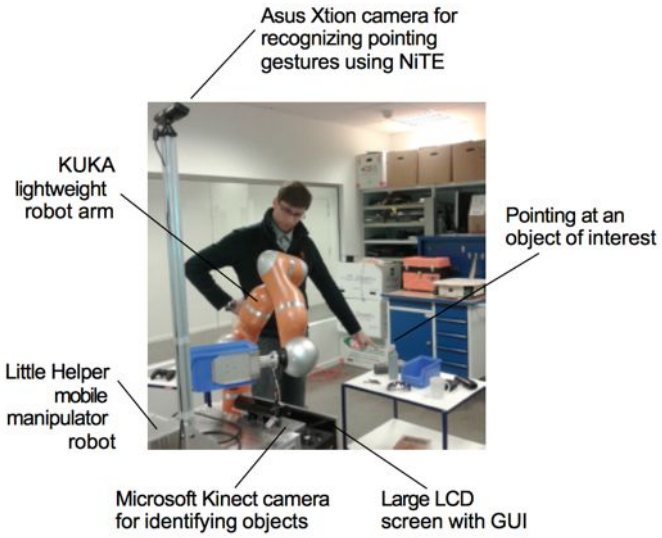
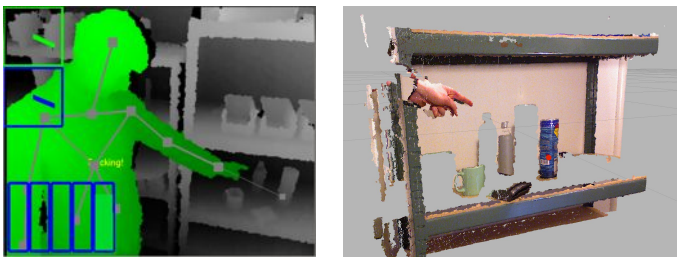


Fig. 2. Outline of the considered robot's components in the pointing scenario



(a) (b)

Fig. 3. Visual feedback provided through the graphical user interface on the *Little Helper's* frontal display

A. Graphical User Interface

The GUI is designed with a purpose of providing an intuitive visual feedback to the user, while he or she is pointing towards an object that is previously unknown to the robot. Specifically, this is done by providing an indication of whether and when the robot recognises the pointing gesture, and then the pointing direction is represented as a ray plotted on top of the mirrored reflection of the user's pointing arm (see (a) in Fig.3). In such a way, user can clearly see where he or she is pointing from the robot's point of view and whether the object of interest is in the sensible range of the robot. The main two windows of the GUI are shown in Fig.3, where (a) is a User Viewer window with real-time depth information from the Asus Xtion camera and (b) is the Object Viewer window with the 3D reconstructed scene with mapped color and depth information from the Microsoft Kinect sensor. The User Viewer window (a) shows the skeletal mapping of recognized main parts of the user's upper body. The bars in the bottom-left corner of this window are representing the success

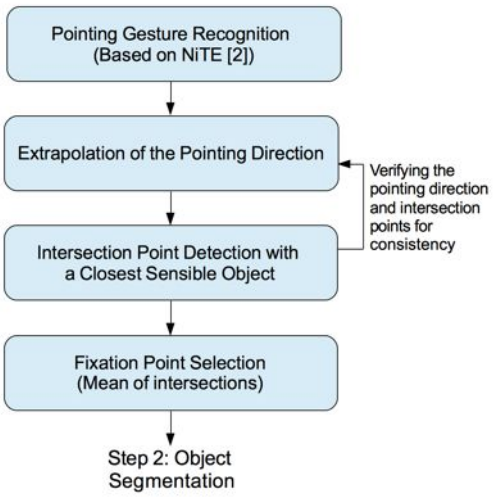


Fig. 4. Fixation Point Generation (Step 1)

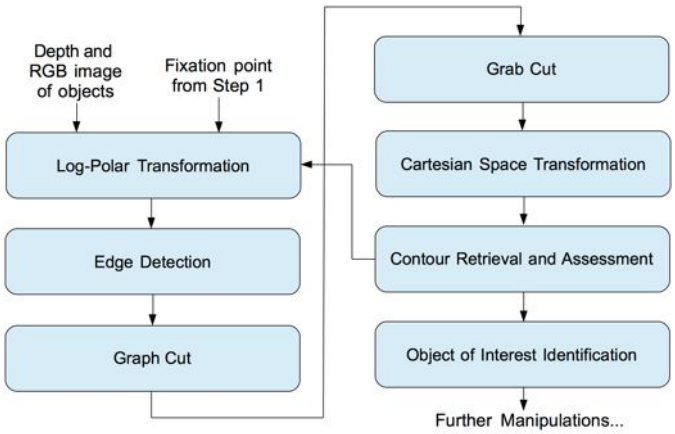


Fig. 5. Object Segmentation (Step 2)

of recognised gestures. When the pointing gesture is performed and remains still, the fifth bar gradually reaches the upper limit and becomes green indicating that the pointing gesture is stable. If the intersection point of the pointing direction with the closest sensible object is found, then in the Object Viewer window (b) the user can see a red dot, indicating where he or she is currently pointing. This information helps user to adjust the pointing direction towards an object of interest more precisely.

B. Fixation Point Generation

A general sequence of the fixation point generation algorithm (step 1) is illustrated in Fig. 4. Initially, the human tracking is based on the continuous data stream from the Asus Xtion RGB-D sensor using the free cross-platform driver OpenNI and the NiTE skeletal tracking library [18]. The Asus Xtion camera is mounted on top of a pole on the robot as illustrated in Fig. 2 for a broader view of a scene. Based on the depth estimation data, the pointing gesture can be recognized only when a significant angle in Cartesian space between the direction of one hand and the ground normal vector is detected and remains without any certain variation for one second. Further, when a pointing gesture is identified, the pointing direction is represented as a 3D vector with direction from the elbow to the wrist. This pointing direction is consistently retrieved and assessed for two more seconds, and if no major deviations occur, several intersection points of the pointing direction with the point cloud from Xtion sensor are iteratively extracted and stored as potential fixation points, and the current pointing direction is calculated as a mean of the pointing vectors extracted from the last seconds frames.

When a certain number of potential fixation points is extracted and the current pointing direction is known, the Asus Xtion camera is switched off and the Microsoft Kinect camera is then turned on, which is mounted on the frontal part of the mobile robot platform (Fig.2) for a closer view of objects placed in front of the robot. Both RGB-D cameras, which have overlapping views, are not operating simultaneously in order to avoid any interference in the reflected infra-red patterns of these cameras that can potentially disturb the accuracy and precision of the measured depth by both sensors. Both cameras are previously calibrated in order to accurately register their output with respect to the robot's coordinate system. Previously stored potential fixation points from the Asus Xtion camera are then projected on the 3D reconstructed scene (see e.g. Fig.6) generated by the Kinect sensor. If they fall within a specified distance range with no major deviations between the depth values of these points, the centroid is then identified as a final fixation point, which can be used as an input for the object segmentation algorithm.

C. Object Segmentation

The general sequence of the object segmentation algorithm (step 2) is illustrated in Fig. 2 and is mainly based on our previous work [17]. The initialisation of the object segmentation algorithm requires two inputs, provided by step 1:

- a 3D reconstructed scene containing one or more objects appearing in front of the robot,
- 3D coordinates of the fixation point assuming that it hits the object of interest.

The input image is then transformed to the log-polar space, which stretches textures close to the fixation point and allows generating the color models for the subsequently applied Grab Cut algorithm more precisely, comparing to the polar space representation, by increasing the object region [17]. Moreover, a growing kernel size following the increasing cell-size of the log-polar grid for the edge detection is implemented, thereby imitating an aspect of the human visual system: the blurred vision outside the focus. The edge detection algorithm is a simple Difference of Gaussians (DoG) kernel, instead of "Globalized Probability of Boundary" (gPb) as proposed in Mishra's work [7], however it decreases the quality of edge map. Thus, higher error ratio of the Graph Cut is expected, especially for regions with blurry contours. In order to compensate the errors, the output of the Graph Cut algorithm is initializing the Grab Cut algorithm. Thus, high-quality results can be achieved, even for problematic initial segmentations made by the Graph Cut. If the edge map converges into the closed contour and it does not hit the border of an image, then the silhouette, which has

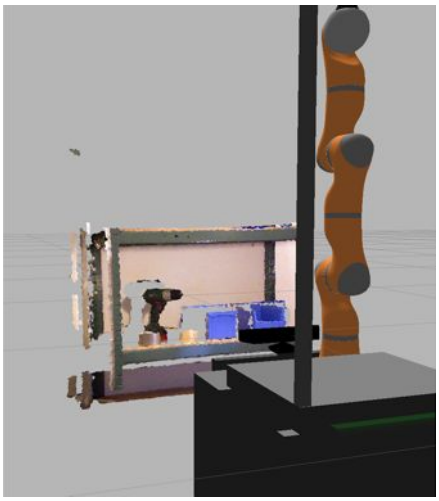


Fig. 6. 3D reconstructed scene with respect to the robot's model

been segmented, is identified as an object. Given the estimated distance towards the segmented object and the geometry of its outline, further reasoning and manipulations can be applied, such as object recognition and/or grasping.

III. Experimental Evaluation

We conducted experiments to clarify the feasibility of identifying objects of interest indicated by a pointing gesture. We gathered two datasets with segmentation results of some common (1) domestic and (2) industrial objects, which were placed on a shelf in front of the mobile robot platform. The final outcomes of the previously described algorithms are presented in Fig.7 and Fig.8. The basic domestic objects (Fig.7) included (a) a bottle of 0.5*l* volume, (b) a potato chip can, (c) a standard mug, (d) sun glasses, and (e) a wallet. The basic industrial objects (Fig.8) included (a) a front-facing small load carrier (SLC) box, (b) a rear-facing SLC box, (c) a drill, (d) a gray duct tape, and (e) a transparent duct tape. The left-side images present the results of fixation point generation from pointing gestures indicated as red dots; while the right-side images show the object segmentation results, where the fixated objects' contours are marked in green colour. The qualitative results show that the integrated system was able to provide fairly accurate outlines of most of the objects, which is expected to be sufficient for initialising further more complex tasks.

IV. Conclusion

We introduced the development of a robust robotic system capable of identifying unknown objects indicated by humans using pointing gestures. The discussed system can provide the first step towards more complex tasks, such as object recognition, grasping or learning by demonstration with obvious value in both industrial and domestic settings. The first results proved the feasibility of the proposed system, because the extracted outlines of the tested objects appeared to be accurate enough for estimating basic dimensions, such as width or height, of fixated objects. Furthermore, several advantages of the system have been observed. One of the main usability advantages of this robotic system is that only an upper body of a user is necessary to be in the view of a robot vision system (e.g. a user can be sitting), which is sufficient for detecting user's arm and for extracting the pointing direction when the user starts pointing at something. Another major advantage is that the robotic system is using a cross-platform fast and accurate unknown object segmentation method, which can detect and outline a big variety of industrial or household objects, depending on a preferred scenario, and can be implemented on other robotic systems equipped with RGB-D sensor(s).

Currently the system is undergoing more exhaustive testing with high variety of objects in different settings in order to produce quantitative and thus more objective evaluation of the system's performance. For the future work we expect to evaluate the proposed robotic system by quantitatively assessing both (1) the correctness of choosing the right fixation points and (2) the precision of the extracted contours. One way of evaluating the correctness of fixation points is to provide users' subjective feedback of whether the robotic system identified the right object or not. The precision of detected object contours can be evaluated objectively by comparing each contour to the known object model and its known location in the space. Different pointing gestures at varying distances are considered to be evaluated as well.

Acknowledgment

This work has been supported by the European Commission through the research project "Sustainable and Reliable Robotics for Part Handling in Manufacturing Automation (*STAMINA*)", FP7-ICT-2013-10-610917, "Robotics-enabled logistics and assistive services for the transformable factory of the future (*TAPAS*)", FP7-ICT-260026, and by the Danish Strategic Research Council through the project *Patient@home*.

References

[1] R. Kehl and L. Van Gool, "Real-time pointing gesture recognition for an immersive environment," in *IEEE International Conference on Automatic Face and Gesture Recognition,*, 2004, pp. 577–582.

[2] K. Nickel and R. Stiefelhagen, "Visual recognition of pointing gestures for human-robot interaction," *Image and Vision Computing*, vol. 25, no. 12, pp. 1875–1884, 2007.

[3] R. B. Rusu, A. Holzbach, G. Bradski, and M. Beetz, "Detecting and segmenting objects for mobile manipulation," in *Proceedings of IEEE Workshop on Search in 3D and Video (S3DV), held in conjunction with the 12th IEEE International Conference on Computer Vision (ICCV)*, Kyoto, Japan, September 27 2009, pp. 47–54.

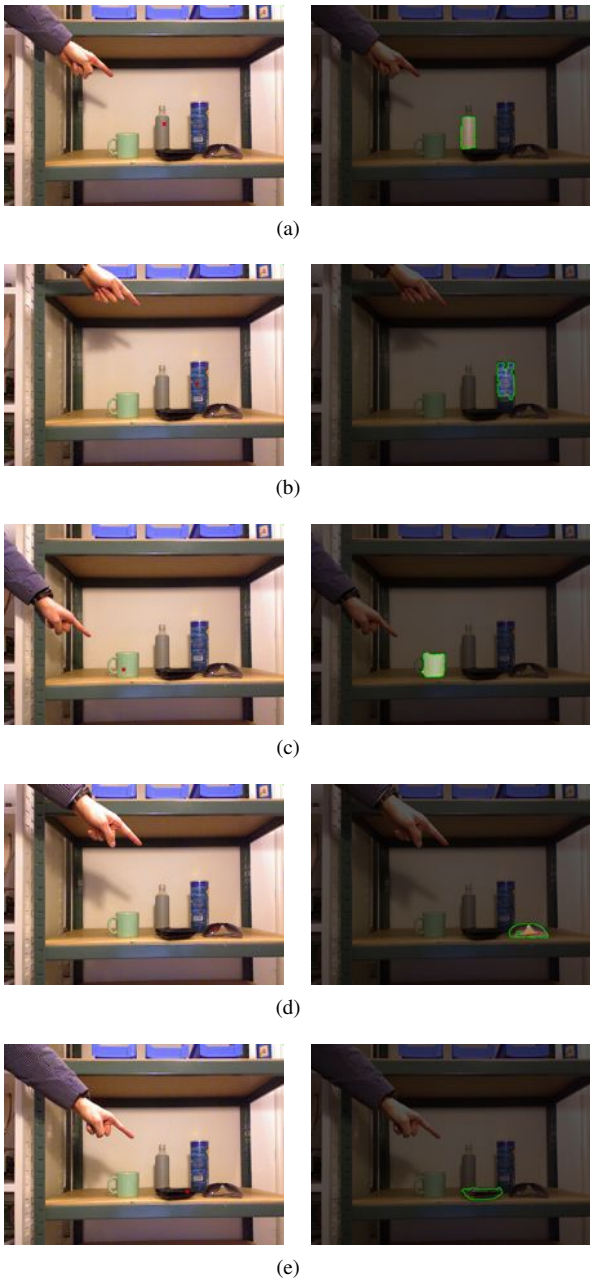
[4] D. Droeschel, J. Stuckler, and S. Behnke, "Learning to interpret pointing gestures with a time-of-flight camera," in *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2011, pp. 481–488.

[5] C. P. Quintero, R. T. Fomena, A. Shademan, N. Wolleb, T. Dick, and M. Jagersand, "SEPO: selecting by pointing as an intuitive human-robot command interface," in *2013 IEEE International Conference on Robotics and Automation (ICRA)*, 2013, pp. 1166–1171.

[6] D. E. Ilea and P. F. Whelan, "Image segmentation based on the integration of colour-texture descriptors - A review," *Pattern Recognition*, vol. 44, no. 10-11, pp. 2479–2501, 2011.

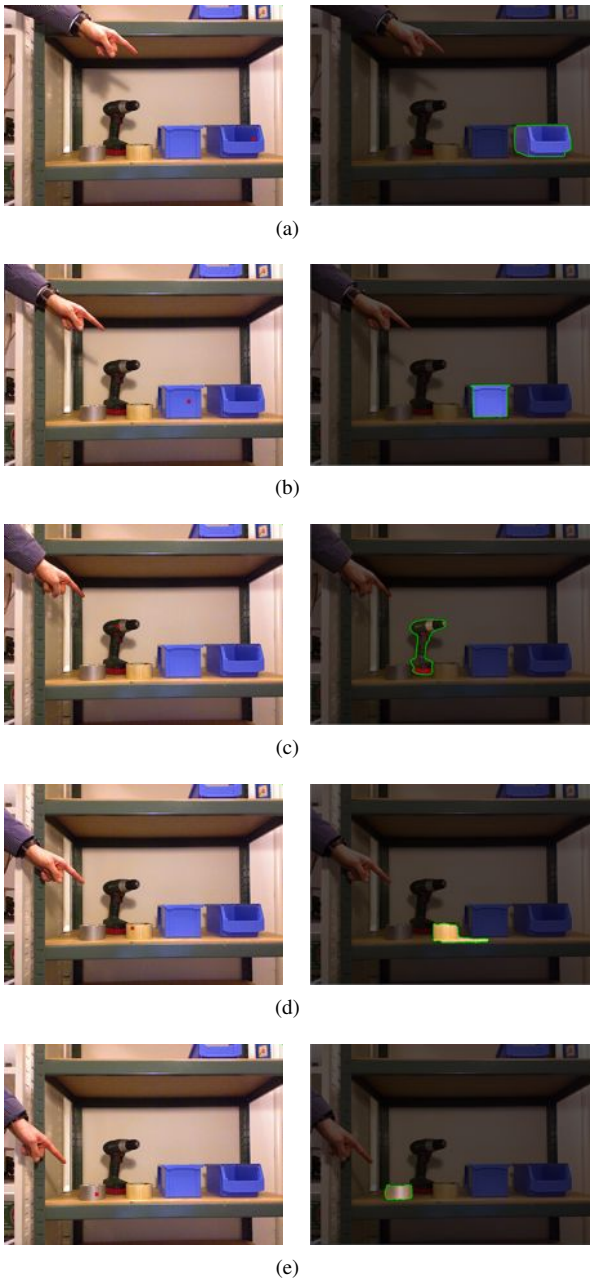
[7] A. Mishra, Y. Aloimonos, and C. L. Fah, "Active segmentation with fixation," in *IEEE International Conference on Computer Vision*, 2009, pp. 468–475.

[8] A. Mishra and Y. Aloimonos, "Visual segmentation of simple objects for robots," in *Robotics: Science and Systems*, Los Angeles, CA, USA, June 2011.



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

Fig. 7. Pointing and segmentation results of domestic objects



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

Fig. 8. Pointing and segmentation results of industrial objects

[9] L. Nalpantidis, M. Björkman, and D. Kragic, "Yes - yet another object segmentation: exploiting camera movement," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vilamoura, Algarve, Portugal, 2012.

[10] G. Kootstra, N. Bergstrom, and D. Kragic, "Fast and automatic detection and segmentation of unknown objects," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2010, pp. 442–447.

[11] M. Johnson-Roberson, J. Bohg, M. Björkman, and D. Kragic, "Attention-based active 3D point cloud segmentation," in *International Conference on Intelligent Robots and Systems*, 2010, pp. 1165–1170.

[12] M. Björkman and D. Kragic, "Active 3D segmentation through fixation of previously unseen objects," in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2010, pp. 119.1–119.11.

[13] C. Rother, V. Kolmogorov, and A. Blake, ""GrabCut": Interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 309–314, 2004.

[14] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[15] O. organization, *OpenNI User Guide*, OpenNI organization, November 2010. [Online]. Available: http://www.openni.org/documentation

[16] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, 2009.

[17] L. Nalpantidis, B. Großmann, and V. Krüger, "Fast and accurate unknown object segmentation for robotic systems," in *International Symposium on Visual Computing (ISVC)*, ser. Lecture Notes in Computer Science, vol. 8034. Rethymnon, Greece: Springer-Verlag, July 2013, pp. 318–327.

[18] "NITE - natural interface technology for end-user," http://www.primesense.com/solutions/nite-middleware/.