2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)
July 19-21, 2021. Munich, Germany

2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids) | 978-1-7281-9372-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/HUMANOIDS47582.2021.9555800

Garbage Collection and Sorting with a Mobile Manipulator using Deep Learning and Whole-Body Control

Jingyi Liu*[1], Pietro Balatti*[2,3], Kirsty Ellis*[1], Denis Hadjivelichkov*[1], Danail Stoyanov[1], Arash Ajoudani[2], and Dimitrios Kanoulas[1]

***Abstract*— Domestic garbage management is an important aspect of a sustainable environment. This paper presents a novel garbage classification and localization system for grasping and placement in the correct recycling bin, integrated on a mobile manipulator. In particular, we first introduce and train a deep neural network (namely, GarbageNet) to detect different recyclable types of garbage. Secondly, we use a grasp localization method to identify a suitable grasp pose to pick the garbage from the ground. Finally, we perform grasping and sorting of the objects by the mobile robot through a whole-body control framework. We experimentally validate the method, both on visual RGB-D data and indoors on a real full-size mobile manipulator for collection and recycling of garbage items placed on the ground.**

I. INTRODUCTION

Rapid urbanization over the past several years resulted in an excessive increase of waste generation per capita, from which a third is not managed in an environmental-friendly manner [1]. In domestic environments, a large amount of garbage is daily thrown or left on the ground, polluting the environment heavily and preventing it from being sustainable and pleasant. Garbage collection and recycling (i.e., sorting garbage into different types) is a common solution that addresses this issue. Garbage separation is essential in this process, however, it is a labor-intensive job that might also affect the labors' health. There are two different types of garbage sorting: 1) centralized classification, where a large amount of garbage is dumped on a conveyor and workers sort out the recyclable waste and 2) piecemeal sorting, which often happens outdoors, such as in parks and streets, where sanitation workers pick up different garbage and place them into corresponding bins. In this paper, we focus on the second type, which significantly reduces the need of extra sorting in the factory and reduces hazardous contact between workers and garbage. Our intention is to allow mobile robots to collect and sort garbage, preventing in this way workers from physical health issues and improving the recycling efficiency.

Garbage collection from the ground (Fig. 1-left) for the purpose of recycling is considered a challenge to be solved using robots. It involves the integration of several subsystems. Firstly, visual or another type of perceptual sensing is required to identify the existence of garbage in the environment and further localize them. Moreover, the type of the garbage must be identified, given that the collection is for recycling, and thus it needs to be placed in the right bin. Secondly, the grasp pose of the garbage object needs to be extracted. Lastly, a planning and control method for the robot to grasp the garbage and place it into the right bin. This whole process needs to be done with all garbage items in the scene in the most efficient way.

Fig. 1: Typical garbage on the ground [2] (left); the IIT-MOCA/UCL-MPPL mobile robot (right).

In this paper, we introduce a novel integration of the aforementioned scheme, in order to allow a mobile manipulator to collect garbage from the ground, after identifying their location, grasping pose, and type. An overview of the system can be visualized in Fig. 2, while the mobile robot that was used is visualized in Fig. 1-right and has been modified to carry three different recycling bins (paper, metal plastic). The process is as follows. First, RGB-D data are acquired from the visual sensor on the robot. These data are fed to a deep learning network (we call it GarbageNet) that is trained to segment, classify (based on their material type), and localize all garbage in the 2D RGB scene. The 3D location of each garbage object can then be extracted from the associated depth data, as well as the grasp pose of the closest one. Last, the robot starts approaching the closest garbage and a whole-body controller enables the robot to grasp the target object and place it in the right recycling bin.

Next, we review related work of garbage collection and sorting robots (Sec. I-A). Then, we present our novel integration of three subsystems, namely the deep garbage recognition and localization, the grasp pose extraction, and

[1]Department of Computer Science, University College London, Gower Street, WC1E 6BT, London, UK. {j.liu.19, kirsty.ellis, dennis.hadjivelichkov, danail.stoyanov, d.kanoulas}@ucl.ac.uk
[2]HRI[2] Lab, Istituto Italiano di Tecnologia, via Morego 30, 16163 Genova, Italy {pietro.balatti,arash.ajoudani}@iit.it
[3]Department of Information Engineering, University of Pisa, Pisa, Italy
*equal contribution

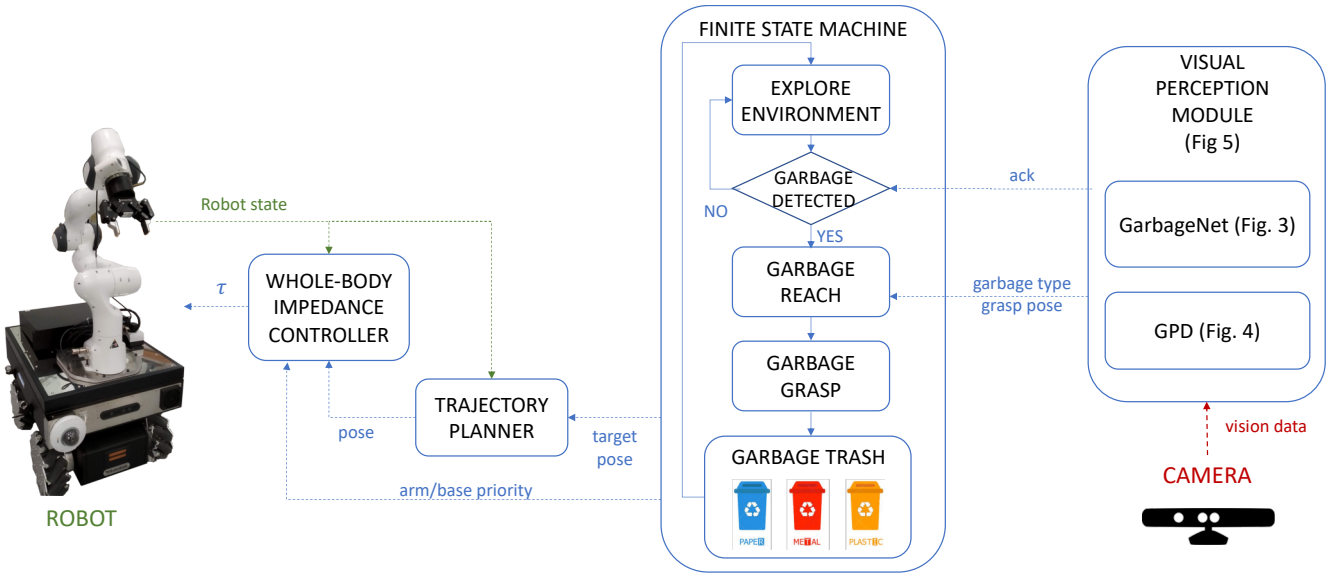


Fig. 2: Software architecture of the whole system.

the whole-body mobile manipulation (Sec. II). Moreover, we demonstrate the performance of our introduced system through experimental results (Sec. III). Finally, we conclude with future directions (Sec. IV).

A. Related Work

The vast majority of autonomous garbage sorting robots mainly focus on the centralized classification, i.e., an automated conveyor along with one or more arms and a visual detection system are combined to sort garbage in the factory. The most representative one is the sorting station developed by ZenRobotics Recycler in Finland [3]. A high-resolution 3D sensor is used to get an isometric 2D height map of the conveyor, then a machine learning method is employed for object recognition and manipulation. The sorting efficiency of this system is as high as 98%, and the average sorting speed is 3000 times per hour. Another successful commercial product is the Waste Robotics developed by FANUC [4], where convolutional neural networks are employed to classify data that are collected by RGB-D cameras. After the model is trained successfully, the robot arm uses suction grippers to pick the recyclable waste. A similar approach has been recently investigated using a fast parallel manipulator with a suction gripper, for sorting items on a conveyor [5]. Several other similar systems have been developed recently [6], [7], [8], and the difference from our approach is that they usually classify items in a known background environment (conveyor) in the factory, while we are looking into sorting items during their collection by grasping from the ground and placing them in the right bin.

The second type of garbage sorting (i.e., piecemeal) that we are interested in this paper, still remains an active research area in robotics, with several open challenges. For instance, the potential unstructured surrounding environment that garbage may lie in, or the fact that a robot operating robustly and efficiently in such a task, involves many aspects of operations, such as object recognition, grasp pose estimation, grasp control algorithm, path planning, etc. Even though there is work to been done on garbage detection [9], [10], the only mobile manipulation robotic system that has developed a pick-up garbage method on the grass is the one presented by Bai et al. [11]. In particular, a deep learning method is deployed to classify the waste on the grass (i.e., as waste or not) and a novel navigation algorithm is presented based on grass segmentation. However, this system does not work in real-time and is not able to classify garbage by type for the purpose of recycling. In this paper, we propose a novel integration of systems that detect the type and pose of the garbage on the floor and use state-of-the-art whole-body control to collect them and sort them in the right bin, based on their type.

II. METHODS

In this section, we discuss the approaches we employ to realize the garbage recycling robot, including finding what and where the garbage is and how the robot can grasp it.

A. GarbageNet: Deep Garbage Recognition & Localization

While object detection methods satisfy the demands of garbage classification and localization, by providing class labels and bounding boxes, instance segmentation methods have the advantage of also providing pixel-level masks. These masks can then be projected onto a depth image and significantly simplify the robot grasp search for a given target object.

Given the need to detect and localize garbage in real-time with the mobile robot, we decided to use the YOLACT framework introduced in [12] and train it for garbage objects. We have named the new trained network GarbageNet. Using this type of network structure it is possible to infer the bounding box and type of an object, as well as to acquire pixel-level object masks that could better help the robot comprehend

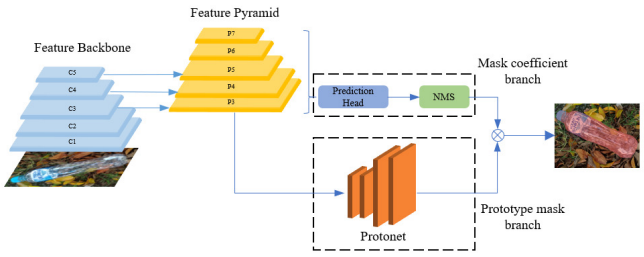
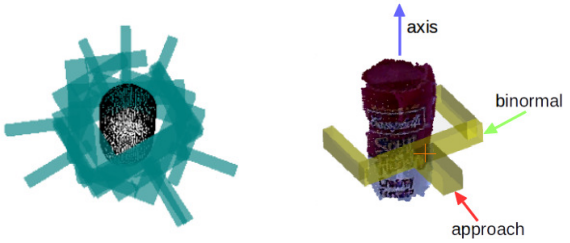


Fig. 3: GarbageNet: Convolutional image features are produced and passed onto two branches - the Protonet branch produces mask prototypes, while the other estimates their coefficients. Both are combined into an instance-level mask [12].



(a) (b)

Fig. 4: Grasps produced by GPD [19]: (a) candidate pool and (b) axes defining each grasp.

its surrounding environment. The real-time performance and high accuracy contribute to its advantage over other types of object segmentation methods, such as Mask R-CNN [13], SOLO [14] and TensorMask [15]. We have integrated the original network in a ROS wrapper, where the robot visual sensor is used as input and the garbage object segmentation, bounding box, type, and grasping pose messages are generated. Our framework produces instance masks and scores them with mask coefficients. Masks are combined using Non-Maximum Suppression (NMS) to ensure there is no overlap between instances while retaining useful information. The core structure is shown in Fig. 3.

1) Dataset: The original YOLACT network is trained on the COCO [16] dataset, originally used for image recognition and does not fulfill the requirements of garbage type characterization and segmentation. Thus, a novel dataset to train GarbageNet for garbage identification was needed. For this reason, we used the newly introduced TACO dataset [2], which is specialized for garbage segmentation and classification. The dataset uses an object taxonomy that can be directly used for garbage sorting purposes. In particular, it includes 1500 images with 4784 annotations, 60 categories which belong to 28 super-categories (e.g., paper, glass, metal, carton, plastic, polypropylene, etc). Moreover, the objects' background environment includes both indoors and outdoors environments, such as tiles, pavements, grass, roads, etc. In this way, even deformed garbage objects in the wild can be classified and segmented.

2) Training: To exploit our framework, we randomly split the TACO dataset into training (80%), cross-validation (10%), and testing (10%) sets. We used an ImageNet [17] pre-trained model of YOLACT to fine-tune the weights on the TACO dataset, using a batch size of 8 on two Titan XP GPUs for 1 day and $40,000$ iterations (learning rate: 10^{-3}, weight decay: 5×10^{-4}, momentum: 0.9). Using ResNet-50 as backbone, we achieved a mAP_{75} of 40.43 (mean Average Precision with an IoU threshold of 0.75), in roughly 30 frames per second (i.e. almost the speed of the input RGB-D sensing). This is slightly better than the original mAP_{75} of YOLACT on the COCO dataset, which is 31.2, or Mask R-CNN, which is around 37.8. Notice here that the exact mask segmentation of the object is not particularly important in this stage, since the grasping pose is extracted from a different process, as described in the next section.

3) Implementation: To allow the system be integrated on our ROS-based architecture, a wrapper was used to interact easily with the other components and the real robot through ROS topics. In particular, an interface node subscribes to the input point cloud and the GarbageNet-produced masks, which in turn projects the masks onto the point cloud. The approximate position of the closest garbage piece is produced using these projections. The interface also filters the detected garbage category into three super-categories: paper, metal and plastic, based on keyword search. Finally, the interface publishes the approximate position of the nearest object, its projected mask points and its super-category.

B. GPD: Grasp Pose Detection

Traditional grasp pose generation methods [18] require either the geometric properties or an exact 3D model of the targeted object. However, litter thrown on the ground often has a non-rigid structure with varying textures and shapes. Providing precise models or establishing a large garbage grasping database is impractical. Moreover, a mobile robot dealing with cluttered scenes would only have access to RGB-D information from a single view.

A more general solution that deals with these challenges would be to generate grasps directly from a voxelized point cloud. That is the principle on which Grasping Pose Detection (GPD) [19] is based. GPD has successfully been integrated with object detectors in cluttered environments.

1) Method: The GPD algorithm follows several steps as briefly outlined in Algorithm 1.

Algorithm 1: Grasp Pose Detection

input : Pointcloud $\mathbb{C}$;
Subset of points where the grasps are to occur S;
Grasp filtering parameters Θ;
output: Grasp Configurations G;
1) H = HandSearch($\mathbb{C}, S$);
2) G = SelectGraspConfigurations($H, \mathbb{C}, \Theta$);

In Step 1, the received point cloud data $\mathbb{C}$ is voxelized and filtered. Points uniformly sampled from the subset of points

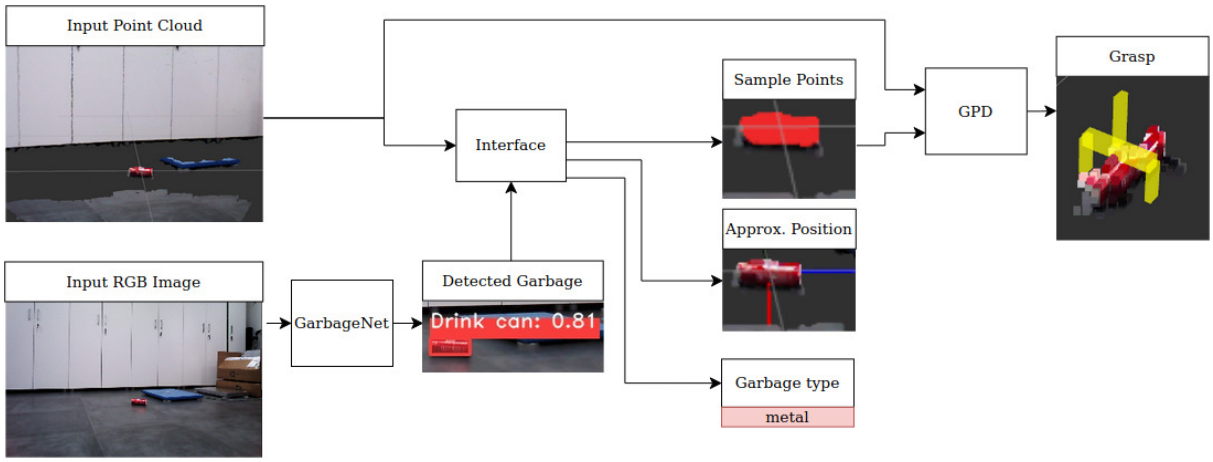


Fig. 5: Perception pipeline: Input image is passed through GarbageNet to detect garbage. In the interface, masks of detected objects are projected onto the point cloud. The approximate position of the nearest garbage is outputted, while its mask projection is used as sampling points for GPD. A garbage type label is also produced. Finally, GPD produces a grasp.

S are used to produce hand candidates (see Fig. 4a) at the axes aligned with the points' normals. Each hand candidate is defined by axes for approach, hand binormal and object axis as shown in Fig. 4b. Filtering is applied to reject any candidates that would collide with the point cloud or do not contain at least one point in the closing region of the hand.

In Step 2, grasp candidates are produced from the hand candidates, given some allowable angle deviation and approach restrictions Θ. The candidates are encoded into several image embeddings, which are passed through a trained convolutional neural network based on LeNet [20]. The output of the network classifies the candidates as successful grasps by assigning them a score. Finally, the grasp configurations G with the highest scores are selected as the best ones.

2) Implementation: The pre-trained original implementation of the GPD package [21] is used within a GPD ROS wrapper. The input to GPD is set as the RGB-D view received from a camera, along with sampling points based on the detected garbage instance masks to provide a region of interest. The outputted grasp with the highest score is selected and transformed into a ROS pose message type.

Following the aforementioned framework, a unified garbage detection, classification, localization and grasp generation pipeline is created by connecting GarbageNet and GPD through an interface node as shown in Fig. 5.

C. Whole-Body Mobile Manipulation Grasping

With the aim of localizing and collecting garbage items from the ground with a robotic system, we introduce in this section the control module that has been implemented on the research platform IIT MOCA/UCL MPPL [22]. This versatile cobot is composed by a Robotnik SUMMIT-XL STEEL mobile platform (3-Degrees of Freedom (DoFs)), and a Franka Emika Panda robotic arm (7-DoFs). Since the control of the former is achieved through admittance control while the robotic arm is torque-controlled, a Whole-Body Impedance Controller has been developed to deal with their different causalities, extending our methods introduced in [23], [24]. The implementation of such control system allows both to achieve the desired end-effector behavior, and to exploit the redundant DoFs of the robot. This is a fundamental requirement to successfully execute autonomous and complex manipulation tasks.

Considering the mobile-manipulator with 3-DoFs (rigid body motion) at the mobile base and n-DoFs at the manipulator, we can define the generalised coordinates $\boldsymbol{q} = [\boldsymbol{q}_v^T \quad \boldsymbol{q}_r^T]^T \in \mathbb{R}^{3+n}$, with $\boldsymbol{q}_v$ and $\boldsymbol{q}_r$ the coordinates of the mobile base and the manipulator. We describe the dynamics equations of the combined system as follows, taking into account the admittance causality of the mobile base that is velocity controlled:

$$\overbrace{\begin{bmatrix} \boldsymbol{M}_{adm} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{M}_r \end{bmatrix}}^{\boldsymbol{M}} \begin{bmatrix} \ddot{\boldsymbol{q}}_v \\ \ddot{\boldsymbol{q}}_r \end{bmatrix} + \overbrace{\begin{bmatrix} \boldsymbol{D}_{adm} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{C}_r \end{bmatrix}}^{\boldsymbol{C}} \begin{bmatrix} \dot{\boldsymbol{q}}_v \\ \dot{\boldsymbol{q}}_r \end{bmatrix} + \overbrace{\begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{g}_r \end{bmatrix}}^{\boldsymbol{g}} = \begin{bmatrix} \boldsymbol{\Gamma}_v^{vir} \\ \boldsymbol{\Gamma}_r \end{bmatrix} + \begin{bmatrix} \boldsymbol{\Gamma}_v^{ext} \\ \boldsymbol{\Gamma}_r^{ext} \end{bmatrix}, \quad (1)$$

where $\boldsymbol{M}_{adm} \in \mathbb{R}^{3\times3}$ and $\boldsymbol{D}_{adm} \in \mathbb{R}^{3\times3}$ represent the virtual inertial and virtual damping terms for the admittance control of the mobile base, $\dot{\boldsymbol{q}}_v \in \mathbb{R}^3$ is the velocity of the generalised motion of mobile platform, $\boldsymbol{\Gamma}_v^{vir} \in \mathbb{R}^3$ and $\boldsymbol{\Gamma}_v^{ext} \in \mathbb{R}^3$ are the virtual and external torques. $\boldsymbol{M}_r \in \mathbb{R}^{n\times n}$ is the symmetric and positive definite inertial matrix, $\boldsymbol{C}_r \in \mathbb{R}^{n\times n}$ is the Coriolis and centrifugal matrix, $\boldsymbol{g}_r \in \mathbb{R}^n$ is the gravity vector, $\boldsymbol{\Gamma}_r \in \mathbb{R}^n$ and $\boldsymbol{\Gamma}_r^{ext} \in \mathbb{R}^n$ are the joint torque vector and external torque vector of the robotic manipulator, respectively.

Let us consider $\boldsymbol{x} \in \mathbb{R}^6$ as the task coordinates in Cartesian space. It follows that the desired task-space dynamics behaviour in response to the external wrench $\boldsymbol{F}_{ext} \in \mathbb{R}^6$, (leading to the external torques $\boldsymbol{\Gamma}^{ext} = [\boldsymbol{\Gamma}_v^{ext^T} \quad \boldsymbol{\Gamma}_r^{ext^T}]^T$ in (1)), can be obtained as:

$$\boldsymbol{F}_{ext} = \boldsymbol{\Lambda}(\boldsymbol{q})\ddot{\tilde{\boldsymbol{x}}} + (\boldsymbol{\mu}(\boldsymbol{q}) + \boldsymbol{D})\dot{\tilde{\boldsymbol{x}}} + \boldsymbol{K}\tilde{\boldsymbol{x}}, \quad (2)$$

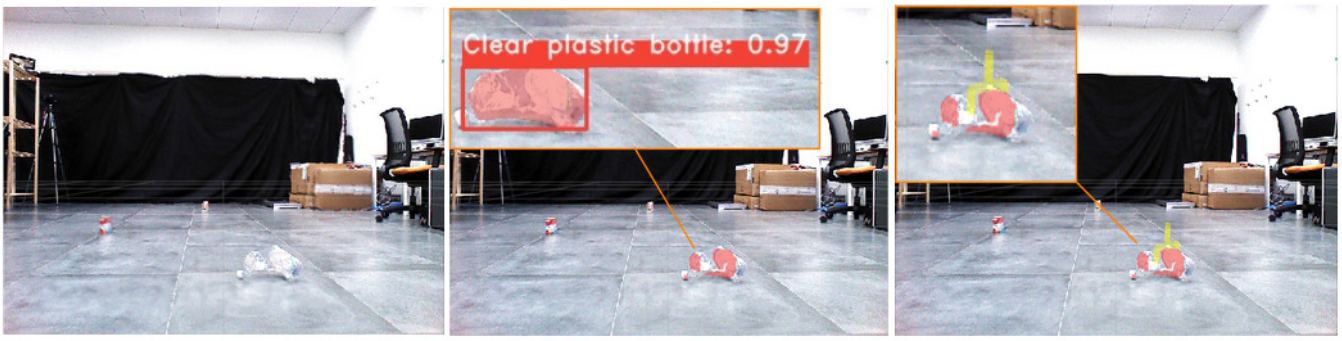


Fig. 6: Example output of input point clouds (left), GarbageNet mask and classification of the closest garbage (middle, zoomed), mask projected onto the pointcloud (middle) and grasps generated via GPD (right).

where $\tilde{\boldsymbol{x}} = \boldsymbol{x} - \boldsymbol{x}_d$ is the Cartesian error from the desired task $\boldsymbol{x}_d$, and $\boldsymbol{K} \in \mathbb{R}^{6\times6}$ and $\boldsymbol{D} \in \mathbb{R}^{6\times6}$ are the desired Cartesian stiffness and damping matrices, respectively. $\boldsymbol{\Lambda}(\boldsymbol{q}) \in \mathbb{R}^{6\times6}$ represents the Cartesian inertial and $\boldsymbol{\mu}(\boldsymbol{q}) \in \mathbb{R}^{6\times6}$ the Cartesian Coriolis and centrifugal matrix, respectively. For more details, please see our previous work on this [22].

In order to navigate through unstructured environment and to grasp garbage items from the ground, it is crucial to selectively assign different mobility priorities to the mobile base or to the robotic arm, when a desired trajectory is executed at the end-effector level. Specifically, during the exploration of the environment, the robot movements must be performed mostly by the mobile base, while when collecting objects from the ground the priority needs to be set to the arm movements.

To this end, we implemented a weighted dynamically-consistent pseudo-inverse to achieve such behaviours. This is done by applying the desired motion constraints through real-time variable weighting factors. The weighted dynamically consistent pseudo-inverse is defined as

$$\bar{\boldsymbol{J}}_W = \boldsymbol{W}^{-1}\boldsymbol{M}^{-1}\boldsymbol{J}^T\boldsymbol{\Lambda}_W\boldsymbol{\Lambda}^{-1}, \tag{3}$$

where $\boldsymbol{\Lambda}_W = \boldsymbol{J}^{-T}\boldsymbol{M}\boldsymbol{W}\boldsymbol{M}\boldsymbol{J}^{-1}$ represents the weighted Cartesian inertia, $\boldsymbol{J} \in \mathbb{R}^{6\times(3+n)}$ denotes the whole-body Jacobian matrix, $\boldsymbol{M} \in \mathbb{R}^{(3+n)\times(3+n)}$ is the whole-body inertial matrix, and $\boldsymbol{W} \in \mathbb{R}^{(3+n)\times(3+n)}$ is the diagonal and positive-definite weight matrix defined by $\boldsymbol{W} = \text{diag}\left([w_1 \quad w_2 \quad \cdots \quad w_n]\right)$, with $w_i \geq 0$. Therefore, a higher value of w_i at the i-th joint will impede the motion of that joint, and $\boldsymbol{W} = \boldsymbol{I}_{3+n}$ will make no effect on the motion mapping.

Finally, the whole-body Cartesian impedance controller's commanded torque for the main task are calculated as:

$$\boldsymbol{\Gamma}_{\text{imp}} = \boldsymbol{g} + \bar{\boldsymbol{J}}_W(\boldsymbol{\Lambda}_w\ddot{\boldsymbol{x}}_d + \boldsymbol{\mu}_w\dot{\boldsymbol{x}}_d - \boldsymbol{K}_d\tilde{\boldsymbol{x}} - \boldsymbol{D}_d\dot{\tilde{\boldsymbol{x}}}). \tag{4}$$

The robot desired poses are retrieved through the *Trajectory planner* unit, that, once received as input a target pose, computes the intermediate waypoints by means of a classical fifth-order polynomial law.

III. EXPERIMENTAL RESULTS

In this section, we present a brief experimental analysis of the garbage segmentation and classification (GarbageNet), grasp pose proposal (GPD), and overall system performance that identifies and collects for recycling three different types of garbage (paper, metal, plastic) using the whole-body controlled mobile manipulation robot.

Fig. 7: GarbageNet classification and segmentation: images with single items are classified correctly with high confidence scores (top). Images containing multiple items are classified with smaller confidence score due to occlusions (bottom).

A. GarbageNet: Garbage Segmentation and Classification

To test the quality of GarbageNet segmentation and classification introduced in Sec. II-A, we have first validated on the testing TACO dataset (see Sec. II-A.2), with a resultant mAP_{75} of 40.43 at 30 frames per second. We further segmented several unseen test images (roughly 1h of recorded data, including objects from the categories into which we will be sorting), both from a handheld RGB-D RealSense camera and the visual sensor of the mobile robot.

It is found that instance segmentation of spread out pieces of garbage is successful (Fig. 7-top), while in some scenes containing a cluster of many pieces of garbage it is less successful and needs a further research investigation (Fig. 7-bottom). This localization failure of cluttered scenes has been identified as one of two typical errors encountered in mask generation by GarbageNet, the second being leakage - noise that is included in the instance mask when a bounding box is not accurate [12]. The success of the classification of garbage provided by GarbageNet is influenced by the quality of the images that are provided to the system. It is found that in overexposed images, the algorithm struggles to detect features that differentiate the garbage item from the surrounding environment.

B. GPD: Garbage Grasp Proposal

An advantage of our introduced system is that the precision of the garbage mask segmentation and bounding-box estimation does not highly influence the grasping pose extraction, since this is estimated from the GPD method, introduced in Sec. II-B. Items of garbage to be picked are provided to the GPD node sequentially by order of proximity to the robot. Some example grasp generation sequences are shown in Fig. 6. The quality of the grasps generated by GPD depends on the number of sample points on the item, e.g., a sparse point cloud can result in no grasp candidates. This was observed in some scenes, but it was quickly rectified by capturing new RGB-D data. With a well populated point cloud, GPD produces very good grasp proposals with a grasping success rate of almost 90%, tested with 50 grasps on the robotic manipulator.

Notice that we had to restrict all grasps to be from the top of the object, to respect the reachability constraints of the robot manipulator. GPD parameters allow for easy selection of approach direction as well as allowable angle deviation from it. It was found that when generating grasps on objects that were seen only from the side, GPD, as expected, struggles to produce grasps from above and data recapturing is required from a different pose. Generated grasps have been successfully transferred from simulation to the real robots with a two fingered mobile manipulator.

C. Whole-Body Grasping Results

Exploiting the Whole-Body impedance controller introduced in Sec. II-C, we performed a set of experiments with the IIT MOCA/UCL MPPL robotic platform (Fig. 1-left). To describe the phases of such experiments, we follow the control flow of the Finite State Machine (FSM) (see Fig. 2). As in a real world scenario, the mobile robot explores the environment, until an acknowledgment (ack) message is provided by the visual perception module. Fig. 8 shows all the phases taking place after this ack is triggered for three different materials: metal (a), paper (b), and plastic (c). In the *garbage detected* state (light red), the robot halts its motion, so that GarbageNet identifies the garbage type, and GPD extracts the grasp pose. These data are sent to the FSM, that can move on to the next phases. The grasp pose (visualized inside

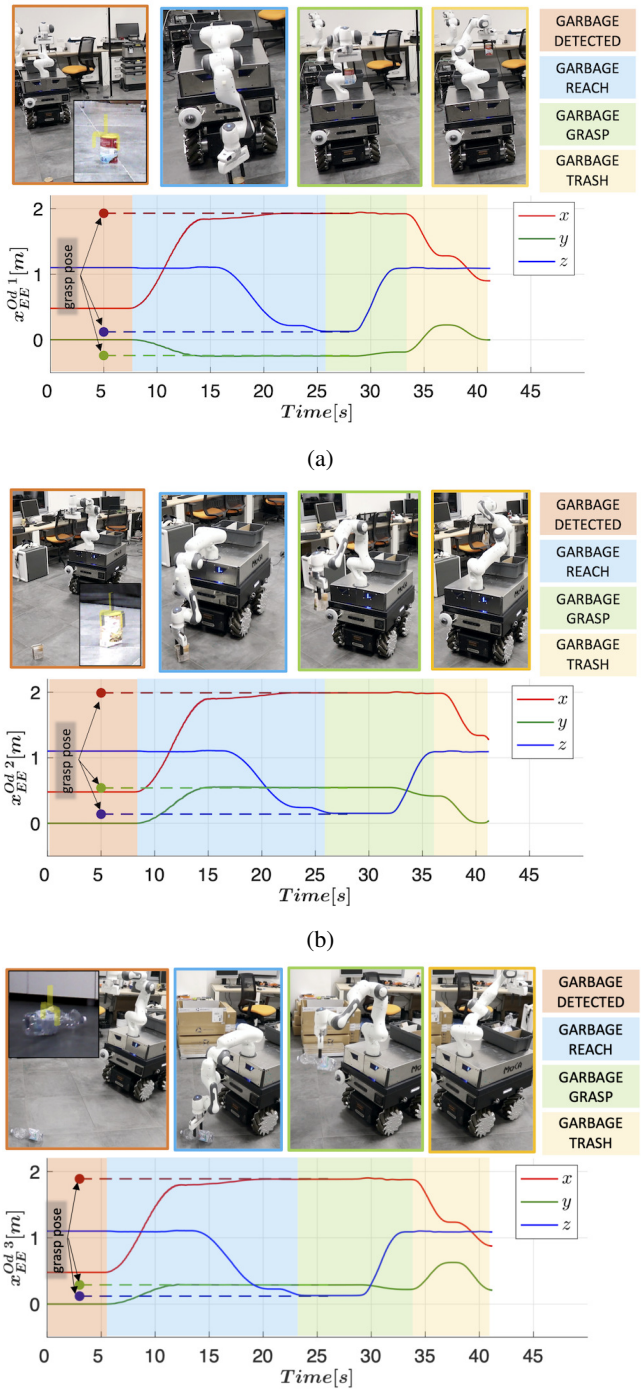


Fig. 8: The grasping results performed by the IIT MOCA/UCL MPPL robotic platform exploiting the Whole-Body impedance controller. Images of garbage detection (with the grasp pose in the embedded image), reach, grasp, and disposal in the correct type of trash bin are visualized. Three different items were identified and collected: a tomato juice can - classified as metal (a), a lentils carton box - classified as paper (b), and a water plastic bottle - classified as plastic (c).

the garbage detection image in Fig. 8) is reported in the plots with point markers at the moment of detection, and reported until the grasp takes place with (dashed lines). Next, in the *garbage reach* state (light blue), the robot moves towards this grasp pose. During this process we can distinguish two sub-phases. In the first one, the robot reaches the vicinity of the goal pose, assigning a higher priority to the mobile base through (3), i.e. setting $w_i = 1$ to the mobile base joints and $w_i = 3$ to the arm joints, with the impedance parameters set to a compliant value $\boldsymbol{K} = diag(500N/m)$. Like this, the mobile robot can approximately reach the item pose in a compliant way, and avoiding unnecessary movements of the arm out of the mobile base support polygon. This guarantees a safety interaction in case of an unexpected collision with the environment. Subsequently, the priority is switched to the arm through (3), i.e. setting $w_i = 5$ to the mobile base joints and $w_i = 1$ to the arm joints, and the impedance parameters are set to be stiffer with $\boldsymbol{K} = diag(1000N/m)$. In this way, the robotic arm can reach the ground towards the grasp pose in a precise manner. From Fig. 8, it is possible to notice that the robot end-effector reaches the grasp pose with a high accuracy, so that the *garbage grasp* state (light green) can be performed successfully. In this state, the robot gripper closes its finger until a force of $3N$ is sensed, to ensure the object is firmly grasped. Lastly, in the *garbage trash* state (light yellow) the robot takes the garbage item to the corresponding trash bin placed on its back, selecting it through the garbage type message received previously.

IV. CONCLUSIONS AND FUTURE WORK

In this work, we present a novel garbage identification and sorting system, integrated on a mobile robot, using whole-body control. This approach works in real-time, identifying, localizing, and sorting garbage.

In the future, we aim at validating the integrated system outdoors in the wild, under various forecast conditions, and work further on the path planning and exploitation part of the method. In particular, the problem of where to look for garbage in a big outdoors space and how to collect them in an energy and time efficient way are our next steps to address the problem.

ACKNOWLEDGMENT

This work was supported by the UCL Global Engagement Funds 2020/21 and the EU H2020 SOPHIA project (no 871237). The Titan Xp GPUs were donated by the NVIDIA Corporation.

REFERENCES

[1] S. Kaza, L. C. Yao, P. Bhada-Tata, and F. Van Woerden, "What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050," The World Bank, Washington DC, Tech. Rep., 2018.

[2] P. F. Proença and P. Simões, "TACO: Trash Annotations in Context for Litter Detection," *arXiv preprint arXiv:2003.06975*, 2020.

[3] D. T. J. Lukka, D. T. Tossavainen, D. J. V. Kujala, and D. T. Raiko, "ZenRobotics Recycler – Robotic Sorting using Machine Learning," ZenRobotics Recycler, Helsinki, Finland, Tech. Rep., 2014.

[4] W. Liu, H. Qian, and Z. Pan, "Dispersion multi-object robot sorting method in material frame based on deep learning," China Patent 2 017 111 944 941, June 08, 2018.

[5] F. Raptopoulos, M. Koskinopoulou, and M. Maniadakis, "Robotic Pick-and-Toss Facilitates Urban Waste Sorting," in *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*, 2020, pp. 1149–1154.

[6] I. Vegas, K. Broos, P. Nielsen, O. Lambertz, and A. Lisbona, "Upgrading the quality of mixed recycled aggregates from construction and demolition waste by using near-infrared sorting technology," *Construction and Building Materials*, vol. 75, pp. 121–128, 2015.

[7] A. Shaukat, Y. Gao, J. A. Kuo, B. A. Bowen, and P. E. Mort, "Visual classification of waste material for nuclear decommissioning," *Robotics and Autonomous Systems*, vol. 75, pp. 365–378, 2016.

[8] G. SP, H. S, and T. A, "Multi-material classification of dry recyclables from municipal solid waste based on thermal imaging," *Waste Management*, vol. 70, pp. 13–21, 2017.

[9] R. Sultana, R. D. Adams, Y. Yan, P. M. Yanik, and M. L. Tanaka, "Trash and Recycled Material Identification using Convolutional Neural Networks (CNN)," in *2020 SoutheastCon*, 2020, pp. 1–8.

[10] X. Li, M. Tian, S. Kong, L. Wu, and J. Yu, "A modified YOLOv3 detection method for vision-based water surface garbage capture robot," *International Journal of Advanced Robotic Systems*, vol. 17, no. 3, p. 1729881420932715, 2020.

[11] J. Bai, S. Lian, Z. Liu, K. Wang, and D. Liu, "Deep Learning Based Robot for Automatically Picking Up Garbage on the Grass," *IEEE Transactions on Consumer Electronics*, vol. 64, no. 3, pp. 382–389, 2018.

[12] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "YOLACT: Real-Time Instance Segmentation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 9156–9165.

[13] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988.

[14] X. Wang, T. Kong, C. Shen, Y. Jiang, and L. Li, "SOLO: Segmenting Objects by Locations," 2019.

[15] X. Chen, R. Girshick, K. He, and P. Dollar, "TensorMask: A Foundation for Dense Object Segmentation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 2061–2069.

[16] T.-Y. Lin, Y. Cui, G. Paterr, and etc, "Coco: Common objects in context," [EB/OL], https://cocodataset.org/#download/ Accessed September 14, 2020.

[17] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[18] D. Kanoulas, J. Lee, D. G. Caldwell, and N. G. Tsagarakis, "Visual Grasp Affordance Localization in Point Clouds Using Curved Contact Patches," *International Journal of Humanoid Robotics*, vol. 14, no. 01, p. 1650028, 2017.

[19] A. ten Pas, M. Gualtieri, K. Saenko, and R. Platt, "Grasp Pose Detection in Point Clouds," *The International Journal of Robotics Research*, vol. 36, no. 13-14, pp. 1455–1473, 2017. [Online]. Available: https://doi.org/10.1177/0278364917735594

[20] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.

[21] A. ten Pas, "Grasp Pose Estimation," [EB/OL], https://github.com/atenpas/gpd Accessed September 1, 2020.

[22] Y. Wu, P. Balatti, M. Lorenzini, F. Zhao, W. Kim, and A. Ajoudani, "A teleoperation interface for loco-manipulation control of mobile collaborative robotic assistant," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3593–3600, 2019.

[23] P. Balatti, D. Kanoulas, G. F. Rigano, L. Muratore, N. G. Tsagarakis, and A. Ajoudani, "A Self-Tuning Impedance Controller for Autonomous Robotic Manipulation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 5885–5891.

[24] P. Balatti, D. Kanoulas, N. G. Tsagarakis, and A. Ajoudani, "Towards Robot Interaction Autonomy: Explore, Identify, and Interact," in *International Conference on Robotics and Automation (ICRA)*, 2019, pp. 9523–9529.