Toward Autonomous Robots for Demolitions in Unstructured Environments

Francesco Corucci[1]* and Emanuele Ruffaldi[2]**

[1] The BioRobotics Institute, Scuola Superiore Sant'Anna, Pisa, Italy
[2] PERCRO, TeCIP Institute, Scuola Superiore Sant'Anna, Pisa, Italy

Abstract. The construction industry is a capital-intensive sector that is steadily turning towards mechanized and automated solutions in the last few decades. However, due to some specificities of this field, it is still technologically behind other sectors, such as manufacturing. Robotic technologies provide room for improvements, that could lead to economical, technical, and also social benefits. We present a possible conceptual framework for an autonomous robot for indoor demolitions, featuring enhanced perceptual capabilities, situational awareness, as well as intuitive Human-Robot Interaction (HRI) paradigms. The paper deals with several aspects of the demolition task, ranging from perception, to planning, to HRI. With respect to perception, we focus on the design and development of some of the perceptual capabilities needed for such robots, as essential to support autonomy, safety, and situational awareness in unstructured construction sites. Particularly, we propose a novel segmentation algorithm that allows the robot to work in highly unstructured scenarios, as well as a mechanism for detecting and quantifying spatial changes during the task. As far as HRI is concerned, a novel interaction paradigm based on laser designation is proposed. Proposed concepts were implemented and tested on a real, scaled-down, controlled mock-up that, while simplifying some aspects of the task, is able to mimic some general characteristics of a real demolition scenario. From lessons learned in this controlled environment we point out some requirements and foreseen issues in facing the complexity of a real demolition set-up.

1 Introduction

In the last few decades the adoption of large-scale machinery systems in the construction industry is increased. The trend toward mechanization is expected to continue in the next years, and thanks to the constant technological progress *construction robots* [1] have become a tangible technical and economical possibility gradually adopted by industries. Despite the mentioned trend, the construction industry is still technologically behind other industries, such as manufacturing. This technological gap can be attributed to some specificities of this sector: the unstructured and cluttered nature of construction sites, the massive presence

---

* Contact: f.corucci@sssup.it
** Contact: e.ruffaldi@sssup.it

of human workers to deal with, the unique characteristic of construction products (as opposed to the mass production of identical objects), to name a few. Although posing some issues, construction robotics also offers room for beneficial improvements, and constitutes an intriguing and challenging research field. As for demolition robots, some possible benefits arising from their employment are: *a*) the possibility to increase work efficiency and prevent human errors; *b*) the possibility to replace or assist humans in exhausting or hazardous tasks (e.g. demolition of an unsafe building, operation in polluted areas, etc.); *c*) the possibility to face the expected future shortfall in personnel available for the construction labor pool.

For what concerns existing robotic platforms for construction-related tasks [1], commercial platforms consist almost exclusively of general purpose remote-controlled robots, with neither autonomy nor user assistance [2]. Among research applications, some innovative tele-operated were proposed [3]. These works mainly focus on mechanical design and control aspects, while autonomy and perceptual issues are usually not investigated. As regards these aspects a lot has been studied with respect to earth-moving [4] where, however, both the interaction with the environment and perceptual aspects are coarser if compared to indoor demolition, that presents challenging peculiar aspects on a different scale. In this context research studies mainly focused on dismantling of interior facilities [5] and decontamination of polluted environments [6]. These robots can take advantage of several elements (e.g. lamps, screws, or beams) present in a static, structured environment. Limited tasks are executed in a semi-assisted mode (e.g. attach to and scrap a beam, assist the positioning of a platform). As a consequence perceptual and planning issues are quite simplified, and the situational awareness of these robot is low, lacking a comprehensive modeling of the surrounding environment. Human-robot interaction is also limited, and performed by means of traditional interfaces (e.g. joysticks).

This preliminary work – developed in the context of an industrial partnership – aims at providing a first contribution towards the development of a novel generation of demolition robots, featuring enhanced autonomy, perceptual capabilities, situational awareness and HRI paradigms. To the best of our knowledge there are no demolition robots integrating such advanced features. The task of autonomous indoor demolition itself was never directly investigated in literature, although posing several challenges: the interaction with the environment is considerable and fine-grained, the environment is extremely dynamic, and very few structured elements can be exploited by the robot. Moreover, the situational awareness of the robot is fundamental in order to automatically accomplish the task while satisfying safety and technical requirements. The above mentioned characteristics highlight the importance of conferring perceptual capabilities to demolition robots, that is one of the major directions of investigation of our present and future work.

The main contributions of this work are as follows: *a*) we define a conceptual architecture for the task of autonomous demolition (Sect. 2); *b*) we investigate the perceptual capabilities needed to support autonomy and situational awareness

on a demolition site, presenting a novel approach to scene segmentation for the demolition task as well as a method for quantifying deformations of the wall (Sect. 4); *c*) we propose a novel HRI paradigm for construction machines based on *laser designation* (Sect. 5); *d*) we present a simplified scaled-down mock-up, that is used for the qualitative and quantitative experimental evaluation (Sect. 6). The paper ends with discussions on scaling issues and conclusions in Sect. 7 and 8. The video at `http://youtu.be/Kq11lYJjHKI` demonstrates the autonomous execution of the task, some highlights on the perceptual system, as well as the HRI paradigm based on laser designation.

2 Conceptual Framework

2.1 Objectives and Proposed Setup

We are interested in empowering construction robots with *perceptual capabilities*, in order to achieve a *full situational awareness* that can be exploited both for implementing fully autonomous systems or user-assistance in semi-autonomous/tele-operated ones. The robot should be able to build (and provide to human supervisors) rich representations of the working site, being fully aware of surrounding obstacles and of the effects that its actions have on the target site. For what concerns autonomy, we investigate the possibility of developing a robot that is capable of executing a task (or part of it) in autonomous mode, with some level of reasoning/planning and with constant feedback from the environment.

The task we take in consideration is the one of demolishing a wall, that can present some obstacles: visible (e.g. tubes attached on its surface) or not visible (e.g. walled-in tubes or cables). In Fig. 1 the proposed scenario is depicted. The robot, consisting of a mobile base mounting a robotic arm, is equipped with a 3D sensing device (e.g. laser scanner) that allows it to build a real-time 3D model of the construction site. This model is used by the robot to plan its actions and check how these actions modified the environment. Moreover, the sensing system is used to build augmented representations to be returned to the remote human supervisor. An intuitive, bi-directional, on-field interaction paradigm is also provided, based on acoustic signals and *laser-designation* (Sect. 5). It is important to note that in such a set-up the on-site operator can stand at a safe distance since the robot exhibits considerable level of autonomy and the demolition task does not rely on the operator's perception and control skills.

2.2 The Demolition Task: Requirements and Assumptions

We assume the robot to be driven roughly in the proximity of the wall to be demolished (in a tele-operated mode), then it must: *a*) autonomously recognize the target, approach it and align at a given operative distance; *b*) autonomously formulate a *demolition plan* that allows it to accomplish the task; *c*) execute the demolition task, avoiding every identified obstacle (e.g. tubes, beams, pillars, etc.); *d*) at each moment, be aware of the state of the surrounding environment,
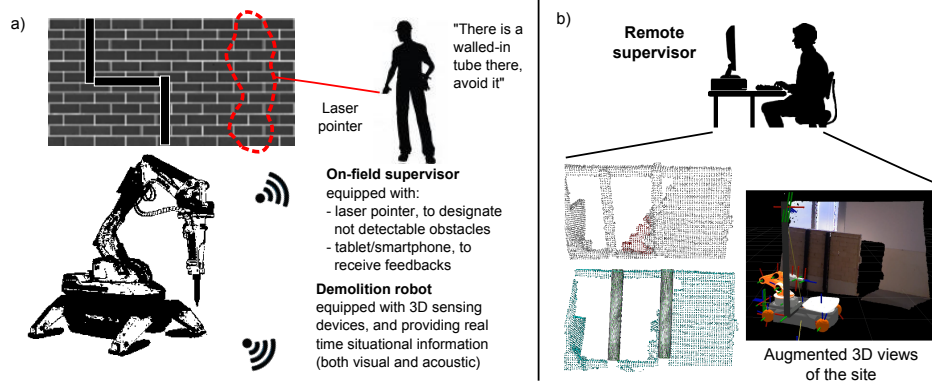
Fig. 1. Proposed scenario. a) The on-site scenario is depicted. The robotic system is shown, together with an on-field supervisor. HRI by means of laser designation is also shown (Sect. 5). b) The remote supervisor, receiving detailed visual feedback.

knowing which parts of the wall have been destroyed, which parts are still intact, and the overall progress; *e)* at each moment, be ready to receive input from human supervisors (e.g. accept new constraints). In what follows, the main assumptions are the following: *a)* the space between the robot and the wall does not present obstacles to be avoided while relocating the base (we do not address the problem of collision avoidance in the approaching phase, for which there exist solutions); *b)* the wall can be of any kind, it can be partially deformed or demolished: the only assumption is that it presents, initially, a dominant planar component. The mock-up implementation presented in Sect. 6 follows these general requirements and assumptions.

2.3 Autonomous Approach to the Demolition task

In Fig. 2 the proposed autonomous approach to the demolition task is outlined. The task has been broken-down into *sub-tasks*, each one being the sequence of demolition actions that the robot is able to perform without relocating its base. The perceptual module offers a number of complex functions that allow the robot to segment the 3D scene, recognize obstacles, understand the changes that its actions determined on the surrounding environment (i.e. transformation of the wall) and estimate the current task progress (Sect. 4). Before detailing the perceptual system (that is the most advanced at the current state of the work), we give some details on how the planning module can be organized with respect to the task of autonomous demolition.

3 Hierarchical Planning

Planning is a crucial aspect for each autonomous robot. Particularly in the context of autonomous demolitions it poses several research challenges, ranging from
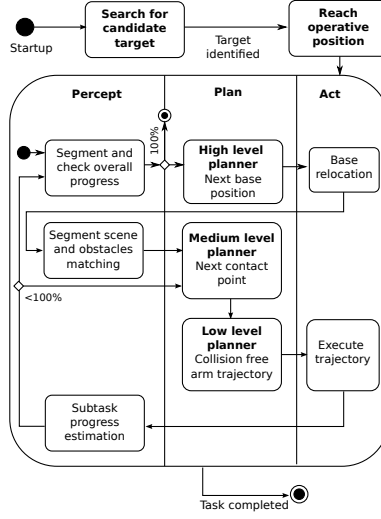
Startup • → **Search for candidate target** → Target identified → **Reach operative position**

Percept | Plan | Act

100% ⊙

Segment and check overall progress → **High level planner** Next base position → Base relocation

Segment scene and obstacles matching → **Medium level planner** Next contact point

<100%

Low level planner Collision free arm trajectory → Execute trajectory

Subtask progress estimation

Task completed ⊙

Fig. 2. Autonomous approach of the robot to the demolition task, as implemented in Sect. 6. Some aspects are intentionally abstracted at this stage of the work. However, just seeing this this simple setup in action suggested us interesting starting points. For example, requiring the full completion of a sub-task is not, in some cases, the most efficient solution, given that some parts of the current operative region can collapse later on.

low-level force control to high-level reasoning: all aspects that would deserve an in-depth dedicated work. For completeness, though, we consider useful to lay down some general concepts about how the planning module can be organized in the context of autonomous demolition. Doing so, we also give some details on how different functions were implemented in our experimental set-up (Sect. 6). We propose to hierarchically organize the planning module in three levels (Fig. 2). The *high-level planner* identifies a certain number of sub-tasks, and outputs the positions that the robot's base has to visit in order to perform each sub-task. In our experiments this level has been based on *Euclidean cluster extraction* [7], that partitions the wall into sub-regions. The *medium-level planner* is important to define the demolition style of the robot, since it calculates, within each sub-task, the *contact points* that the robotic arm will reach in order to execute the sub-task. In our experimental setup this level simply computes contact points that vertically span each sub-region, having care to check if a contact point is valid before executing the reaching action. Finally the *low-level planner* is in charge of computing collision-free trajectories for arm navigation. State of the art algorithms for collision free arm navigation can be adopted to implement this level: in our experimental setting the SBL planner (Single query Bi-directional probabilistic road-map planner with Lazy collision checking) [8] was adopted. This flexible organization of the planning module allows to decouple the different functions: a wide variety of behaviors can arise simply modifying one of the

levels. For these reasons we suggest it as promising for the future design of an autonomous demolition machine.

4 Model and Perception

4.1 Overview

A substantial part of this work involves the design and implementation of perceptual capabilities that allow the robot to process unstructured 3D scenes with any or very few a priori assumptions. In this section we will discuss in greater detail the perceptual challenges we faced. Particularly, we present in an incremental manner several variants of a novel segmentation algorithm, leveraging low-level perceptual building blocks in order achieve segmentation of unstructured 3D scenes during a demolition task.

4.2 World Representation

To be able to autonomously execute the task in real-time with the required *situational awareness*, the robot must hold a convenient representation of the surrounding world. The chosen representation should be rich enough to allow the robot to reason on and extract useful information from it. On the other hand, the model of the world should be simple enough to be processed in real-time. For these reasons, we propose the use of *3D sparse point clouds* to represent the surrounding environment: in our experimental setup input data from a 3D sensor are thus heavily *down-sampled*, reducing the dimension of the point clouds from approximately 300k points to less than 10k. Semantically identified objects (e.g. obstacles to be avoided while demolishing) may be matched and approximated by geometrical primitives, when possible. This can be convenient in order to simplify the planning module in generating collision free trajectories. The final result is a light 3D augmented reconstruction of the demolition site, that can be efficiently processed.

4.3 Scene Segmentation: Motivations and Rationale

The goal of the segmentation module is to allow the robot to distinguish, in every moment, between the wall and the obstacles to be avoided. Initially the wall is mainly planar with obstacles lying on it (a scene that can be easily segmented), but once the robot starts to demolish the wall can be, in general, excavated and deformed in a complex fashion. State of the art techniques for point-cloud segmentation usually search for planes [9][10], known shapes [11][12], geometric primitives [13], or surfaces with homogeneous curvatures [14], but this does not apply to our reference setup. In the general case we want to address we have to segment a scene in which noisy, partial views of generic obstacles (hardly identifiable as geometric shapes before segmenting the scene) lie on an irregularly shaped, partially deformed wall (Fig. 3a). After unsuccessful tests with state of

the art techniques, we designed and developed a new algorithm for segmenting this kind of scenes in the context of a demolition setting, leveraging low level pre-existing perceptual building blocks. A peculiarity of the demolition task is that the world is initially quite structured, and becomes more and more unstructured as the demolition proceeds. We tried to exploit this peculiarity to our benefit, basing the segmentation algorithm on this simple observation. The idea behind our proposed approach is to propagate an easy-to-acquire initial knowledge to subsequent frames, where this knowledge would be much more difficult to acquire directly. We define the concept of *segmentation plane*, being the plane that spatially separates the foreground obstacles from the background wall, allowing to segment a scene. At the beginning of the task we assume that the intact wall presents a dominant planar component, with obstacles attached to it. The first segmentation plane is thus easily calculated by mean of RANSAC-based planar segmentation [15] (RANdom SAmple Consensus). A RANSAC-based approach is also used to approximate obstacles with geometric primitives (e.g. cylinders). When the wall gets excavated, we can't rely in general on planar matching to segment the scene. Moreover, the robot will, in general, move while executing the task, so we can't rely on the initial segmentation to directly segment subsequent scenes. What our algorithm does is to estimate the transformation that occurred between a segmented scene and a new scene, and use it to transform accordingly the previously known segmentation plane in the one that allows to segment the new scene. We now give some more details on the proposed approach.
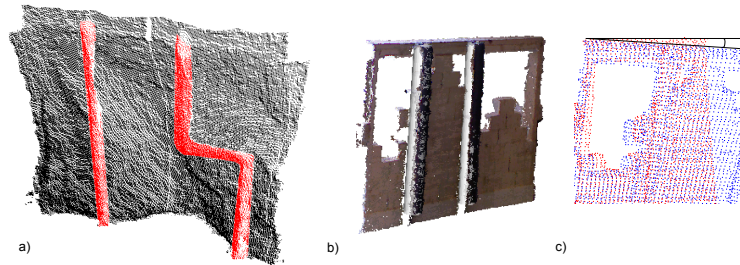

Fig. 3. (Best viewed in colors) a) Non-planar scene being segmented by our algorithm. Points belonging to recognized obstacles are marked in red. b) Planar scene being segmented by our algorithm. Obstacle matching with geometric primitives is also shown. c) Unrealistic rotation introduced by SAC-IA, as discussed in Sect. 4.4.

4.4 Variants of the Segmentation Algorithm: Changing the Method for the Initial Alignment

Once described the rationale behind our segmentation algorithm, we now discuss how it was designed in terms of lower-level perceptual building blocks. Our

segmentation algorithm is, in fact, an high-level algorithm that leverages primitive functions (detailed in Fig. 4) in order to achieve scene segmentation in a dynamic and general demolition setting, in which state of the art algorithms for point clouds segmentation are not applicable. To better understand the following explanation, the reader may refer to Fig. 4.

As already mentioned, in order to transform a previously computed segmentation plane into a new scene, we need to estimate the transformation that aligns two subsequent scenes, i.e. we need *3D registration*. Registration is performed in two steps: *a*) a *rough initial alignment* between the two scenes is calculated (two possibilities were considered: the features-based SAC-IA [16], and a method based on odometry, that we call *odometry-IA* – later described); *b*) ICP (Iterative Closest Point) [17] refinement. Algorithm's parameters are tuned so that the whole algorithm is executed in real-time (meaning that no noticeable wait is introduced with respect to the timing of a demolition task). By changing the underlying building blocks used for registration and improving the logic of our segmentation algorithm, several versions of the algorithm are developed. Starting from the basic intuition given in the previous paragraph, we now detail those variants (compared in Sect. 6) in an incremental manner.

The most general version of the algorithm (*naive SAC-IA based* algorithm) exploits the SAC-IA algorithm for implementing a *features based* initial alignment. After testing some of the most used features for 3D point clouds, we chose NARF (Normal Aligned Radial Feature) features [18], since they resulted faster than other descriptors preserving, at the same time, good registration performances. This variant of the algorithm can estimate an initial alignment between two scenes even if complex motion occurred between the two, thus being very general. However, some problems can arise when the number of iterations of the SAC-IA sub-module is kept low in order to achieve near real-time performances. The algorithm can sometimes fall in a local minima, providing a bad initial registration (e.g. specular, flipped, rotated) that the ICP module is not able to compensate for. Recovering from such errors can be difficult. Moreover, SAC-IA uncertainties can occur in every possible dimension, since the estimated transformation is not bounded in any way to fit realistic motions of the robot (Fig. 3). Thus, the algorithm is not taking advantage of some a priori known constraints, and this can result in unnecessary uncertainties or computation. We then designed an alternative algorithm (*naive odometry-IA based* algorithm) in which the initial alignment between two subsequent scenes is achieved by means of *odometry* information, i.e. we directly estimate the transformation matrix aligning two scenes from odometry information being measured between the two. The estimated transformation is more constrained than in the features-based case. Moreover odometry uncertainties are limited, since movements are short and simple in the demolition phase (consisting of short translations parallel to the wall), and can thus be compensated by the subsequent ICP refinement stage. This solution resulted in benefits to the overall segmentation algorithm, in terms of complexity (less parameters to be tuned), execution time (less computation needed), robustness (less uncertainties).

4.5 Variants of the Segmentation Algorithm: Feedback Mechanism and Refined Version

In the above described *naive* variants of our segmentation algorithm we registered pairs of subsequent frames, transforming accordingly the previously known segmentation plane to segment a new scene (Fig. 4). Without appropriate countermeasures, registration errors tend to accumulate, and the algorithm will eventually fail if the number of registrations (i.e. the number of robot's relocations) needed to perform the task is considerable. For this reason, we introduced a *feedback mechanism*, that allows to reset the registration error every few frames. The feedback mechanism is based on the only invariant elements presents in the scene, i.e. obstacles. When the robot is able to identify a sufficient number of obstacles, it estimates the plane on which they lie on (*obstacles plane*): it will then use that plane for the next transformation process instead of a plane that already accumulated registration errors.

A subsequent version of the algorithm (that we call *refined* version) works without the need of a feedback, thus resulting in a more general and stable algorithm (Fig. 4). The rationale is simple: instead of registering each snapshot with the previous one (possibly relying on multi-step transformations to estimate a segmentation plane when feedback lacks for some time), we register each frame with the first one in which, being the wall still mainly planar, we are able to robustly estimate a segmentation plane using RANSAC based planar matching (the basic assumption is that there are some invariant features that allow to register the two scenes even in presence of modifications of the wall: invariant obstacles can be sufficient for that to hold). This way we always work with a segmentation plane that is a one-step transformation of the first estimated plane, a situation that is only the best case in the naive variants (i.e. the case in which they find a feedback in each snapshot). With this approach the need of a robust obstacle matching algorithm (that was crucial in the previous version) is also relaxed, since obstacles matching does not affect the capability to segment the scene. The refined approach has been evaluated with respect to the odometry-IA registration method. All the three variants of the segmentation algorithm will be evaluated in Sect. 6.

4.6 Changes Detection and Progress Estimation

In order to support autonomy and situational awareness, we wanted the robot to be aware of the modifications that its actions imply on the surrounding environment in terms of changes in the spatial configuration of the wall. This is important also in order to make the robot aware of the task progress, a key requirement in order to support autonomy. We thus implemented a perceptual module called *superficial diff* (where "diff" stands for *difference*), since it is able to compare two point clouds (before and after a demolition move) and return the spatial difference among them. This way we are able to identify precisely what regions of the wall were demolished after each move (Fig. 5). The superficial diff is based on the comparison between binary coded *octree* data structure
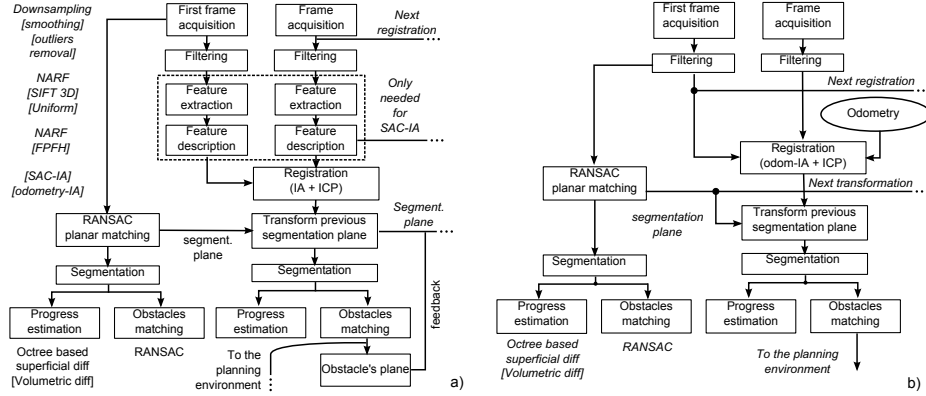
Fig. 4. a) Naive, feedback-based approaches to scene segmentation. Squared brackets denote possible alternatives or additional stages. b) Refined segmentation pipeline, here based on odometry-IA.

constructed from the two point clouds (pre- and post-move) to be compared (see [19] for a similar set-up). By recursively comparing the octrees, the spatial changes – represented by differences in voxel configuration – are identified. Given the pre-move and the post-move clouds, we identify voxels that were occupied in the pre-move cloud but are no longer present in the post-move cloud, thus belonging to a region of the wall that has been demolished. A filtering stage is then applied in order to remove spurious points due to sensor noise. Once the difference surface is identified, it is possible to calculate its area exploiting the surface discretization offered by the octrees (thus supporting autonomous task progress estimation). Pushing beyond this concept, we also implemented a *volumetric diff* in order to cope with the general case in which the wall gets excavated. The volumetric diff was efficiently achieved with a XOR comparison of the binary octrees: this way we obtain all the points belonging to the two surfaces that enclose the excavated volume. We can then calculate the *convex (or concave) hull* on the resulting point cloud, so as to visualize the excavated volume and be able to numerically estimate its volume (Fig. 6). A concave hull is, in general, more accurate in representing the excavated volume, but the numerical calculation of its volume is not trivial. The convex hull, on the contrary, is in general less precise, but its volume can be computed more easily.

5 Human-Robot Interaction

Given the safety critical nature of the application, in a real demolition scenario some forms of human supervision would be certainly required, no matter the level of autonomy and reliability of a demolition machine. For this reason we also designed and implemented an intuitive, precise, on-site *Human-Robot Interaction* (HRI) paradigm. While traditional interfaces for construction machines
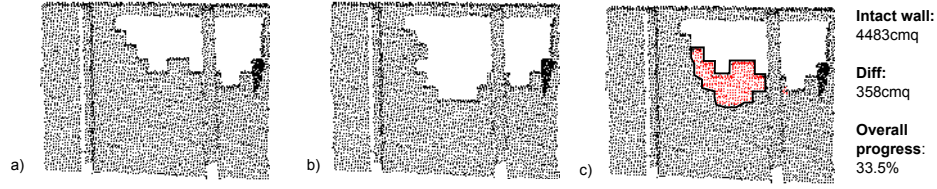
Fig. 5. Superficial difference. a) pre-move cloud, b) post-move cloud, c) the superficial diff is highlighted, together with quantitative information.
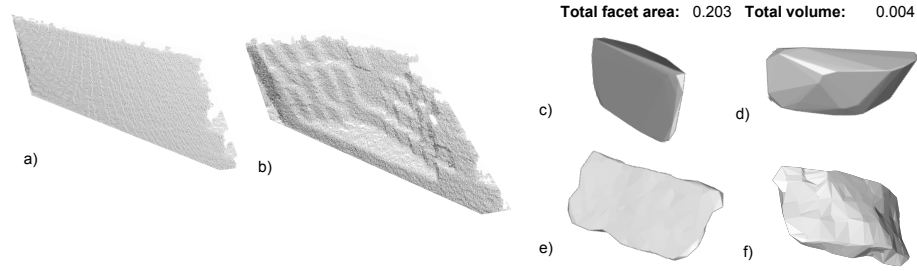

Fig. 6. Volumetric difference between point clouds. a) Intact wall b) Excavated wall, c,d) Excavated volume (convex), with calculated area and volume e,f) Excavated volume (concave)

are based joysticks, levers and buttons, the proposed paradigm is based on *laser designation* (Fig. 1, 7). As a case study, we allowed the supervisor to designate additional obstacles to be avoided during the demolition task, such as walled-in pipes that the robot is not able to identify autonomously. This is done by directly projecting a laser pointer (tracked by the robot) onto the demolition target. Currently the interaction is based on sequences of *way-points* that the supervisor specifies by pointing a certain spot of the wall for prolonged time (2-3 seconds): each recognized way-point is acknowledged by the robot with an acoustic feedback. By means of way-points the supervisor is able to delineate special regions of the wall or walled-in obstacles. Represented by geometric primitives, these additional constraints are integrated in the robot's representation of the world, and considered by the planning architecture in the same way as obstacles that are autonomously recognized. This paradigm solves very effectively the challenging task of avoiding walled-in obstacles and defining additional constraints. This flexible HRI paradigm based on laser designation can be used also for different purposes, e.g. to show the path that the robot should follow to reach the proximity of the target, or to provide demonstrations that will affect the demolition style of the robot in a *programming by demonstration/apprenticeship learning* scenario (a direction in which we also conducted some studies, here not detailed).
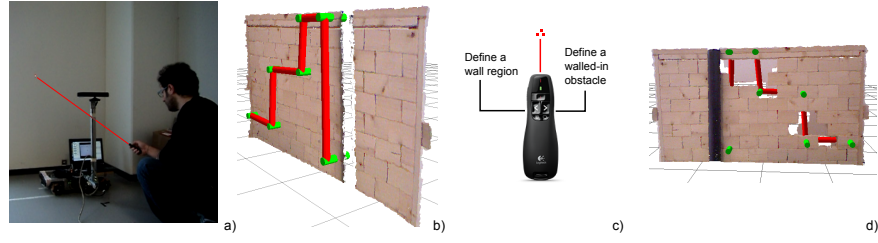
Fig. 7. HRI paradigm based on laser designation: a) Example setup. b) 3D view of the back of the wall, with a walled-in designated tube represented as geometric primitives (green dots are the recognized way-points). c) The laser pointer used for the experiment. d) Example of augmented representation: a front view showing real objects together with walled-in designated obstacles.

6 Experimental Work

6.1 Experimental Setup

In order to test the validity of the proposed concepts and algorithms we built a scaled-down representative scenario for a demolition task. It consists of a wooden brick wall that the robot is able to demolish by simply pushing the bricks with its manipulator (Fig. 8). Some cylindrical tubes were attached onto the surface of the wall in order to mimic obstacles that could be found in a real scenario. While the mock-up wall may seem quite oversimplified with respect to a real demolition site (e.g. it is planar and not able to mimic the dynamics of an excavation), it is a good starting point to demonstrate the overall principles of the proposed scenario. What is important is that key algorithms (e.g. the segmentation algorithms) do not leverage simplified aspects of the mock-up (e.g. the planar configuration of the wall - Fig. 3). In order to study subsequent aspects of the demolition task, we plan to build a more realistic mock-up. The adopted robotic platform is the *KUKA youBot*, a mobile manipulator for research and education. The robot features an omnidirectional base and a 5-DOF manipulator. The platform has been equipped with two *Microsoft Kinect* (one is used for navigation, the other for scene reconstruction). Processing occurs on a dedicated external laptop (ASUS N53S, featuring an Intel Core-i7 2630QM 2.0 GHz, 6GB DDR3 RAM, Nvidia GeForce Optimus 540M 2GB for graphics), connected via EtherCAT to the robot. The robot runs the *Robot Operative System* (ROS) [20], and the whole application was organized in such a framework as several new ROS packages. Perceptual processing is based on the *Point Clouds Library* [21].

6.2 Evaluation of the Segmentation Algorithm: Methodology

In addition to the qualitative evaluation of the overall system, in this section we report the quantitative evaluation of the main variants of the proposed seg-
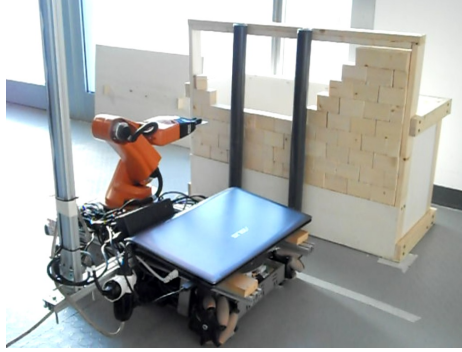
Fig. 8. The mobile robotic platform facing the wooden brick wall in a partially-demolished state. Each brick is 7 x 4.5 x 3.5 cm (width, height and depth). The wall is 70cm x 42cm being thus composed of 90 bricks. Some cylindrical obstacles are also visible.

mentation algorithm. Other aspects of the system will be evaluated in future work.

In order to quantitatively evaluate the variants of the segmentation algorithm presented in Sect. 4 we needed a ground truth. In order to easily obtain it, we allowed the ground truth algorithm (RANSAC based planar matching) to leverage a simplified aspect of our mock-up, e.g. the planar configuration of the wall (see Fig. 8). It is important to note that the ground truth algorithm only works in presence of a dominant planar component throughout the task, a very simplistic assumption that is not always satisfied in a real scenario (e.g. when the wall gets excavated, Fig. 3). Our segmentation algorithm, instead, does *not* exploit in any way the planar configuration of the mock-up wall, thus working in the more general conditions of a real scenario (detailed in Sect. 4, see also Fig. 3). The evaluation has been performed using a data set composed of 30 point cloud snapshots acquired from a Microsoft Kinect camera during a demolition task. To compare the algorithms, in addition to the mean execution time and the number of errors, we define two more metrics: *a*) *displacement of the estimated segmentation plane*, with respect to the plane computed by the ground truth algorithm; *b*) *segmentation index*, a measure of the capability of the estimated segmentation plane to classify points as belonging to the wall or to the obstacles. The segmentation index is thus the misclassification rate, with respect to the classification operated by the ground truth algorithm (the lower, the better). In addition to the aforementioned metrics, we performed visual inspection to evaluate the quality of each segmentation, in order to check if every obstacle was correctly matched with geometric primitives or some error occurred (e.g. visible obstacle not detected, wrong parts of the scene detected as obstacle).

6.3 Naive SAC-IA Based vs Naive Odometry-IA Based

This test compares the two naive variants of the segmentation algorithm, differing in the technique adopted for the initial alignment of two scenes to be registered (other stages of the two algorithms are identical). Table 1 summarizes the comparison. The first aspect to note is that both algorithms were able to correctly segment all the presented scenes, thus proving the effectiveness of the presented approaches. No errors occurred, every visible obstacle was correctly recognized and fitted with geometric primitives. However, the odometry-based version performed better than the SAC-IA-based in our setup. With the same parameters for the ICP refinement, the first algorithm is, on average, more than a second faster, the estimated segmentation plane is slightly more accurate, as the resulting segmentation. We also report that in four cases out of thirty SAC-IA failed in estimating an initial alignment: although not happening in our experiments (extra work carried out by ICP was able to put right), this could in general produce segmentation errors when movements are broader. We can conclude that when movements between two scenes are simple - as in a typical indoor demolition setup - and odometry is available, it is convenient to exploit this information, achieving good results with a simpler pipeline.

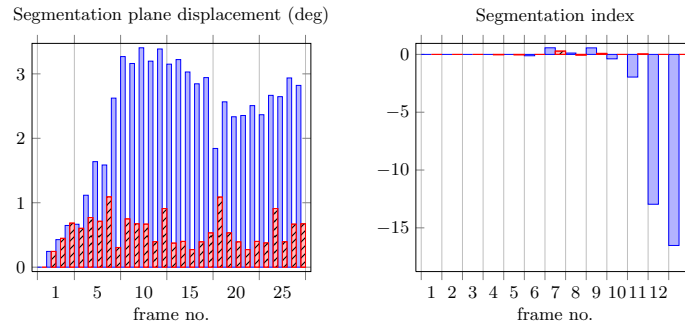### 6.4 Naive Odometry-IA Based vs Refined Odometry-IA Based

We now compare the naive (both with and without enabled feedback) and the refined odometry-IA based algorithms (Table 1). The refined algorithm is slightly slower than the naive: this is due to the fact that while the naive algorithms will often register partial views of the wall (that are composed by a reduced number of points), in the refined version one of the two clouds (the first) will always be complete, thus composed of a greater number of points. Anyway, the difference is not noticeable on the time-scale of a demolition task. The mean plane displacement and segmentation index suggest that the naive algorithm eventually encounters problems when the feedback lacks for prolonged time: in fact we also report four errors out of thirty segmentations. The refined algorithm, instead, performs very well without any feedback, thus proving its effectiveness. Moreover, it also performs better than the naive algorithm with enabled feedback. To better illustrate what happens, Fig. 9 compares the plane displacement and the segmentation index of the two algorithms over the data set. The refined algorithm maintains good performances on all the data set, while the naive version (with disabled feedback) gets worse and worse because of registration errors adding up.

7 Discussion and Scaling Issues

The current mock-up allowed us to perform experiments in a scaled-down, controlled setup, without neither the practical problems of a more realistic set-up (availability of a full-sized robotized platform, required facilities, safety concerns,

Table 1. Performance comparison of the main variants of the segmentation algorithm

	Mean execution time (sec)	Mean plane displacement (deg)	Mean segmentation index (abs)	Errors
SAC-IA	2.66	1.12 ± 0.87	1.87 ± 4.72	0/30
Odom-IA (naive)	1.27	0.98 ± 0.73	0.41 ± 0.97	0/30
Odom-IA (naive, no feedback)	1.32	2.29 ± 1.09	13.67 ± 14.36	4/30
Odom-IA (refined)	1.51	0.55 ± 0.25	0.008 ± 0.13	0/30

Fig. 9. Performance comparison between *refined* (dashed) and *naive* (solid) variants of the odometry-IA based segmentation algorithm (both without feedback). The graph of the segmentation index was truncated at frame number 12 in order to improve readability.

etc.), nor the oversimplifications entailed by simulated environments (that are often adopted in related literature). By designing algorithms that do not leverage simplistic aspects of the mock-up, we were able to test the validity of proposed concepts and algorithms (especially the perceptual ones) in a controlled environment. Although the mock-up may seem oversimplified (and it certainly is in some aspects), we found it particularly useful for a preliminary study of the demolition task. We were able to isolate the aspects of the task that we wanted to investigate first, while simplifying those to be later investigated. As for the simplified aspects of the mock-up (pointing out the necessity of a more sophisticated mock-up, for future work), one of the main current limitations lies in the dynamics of the demolition: the wooden brick wall can only be demolished impulsively and tends to fall apart in an unrealistic manner. This prevents from investigating more sophisticated planning and reasoning strategies, embedding a physical model of the world. Another important limitation is the impossibility to model volumetric aspects of the demolition task (the *volumetric diff* was

tested on a separate static mock-up). Those aspects are certainly important and should be modeled by the mock-up, posing several challenges related to low-level planning and control strategies. As for scaling issues, proposed perceptual algorithms were designed to work in much more general scenarios than the simple mock-up used for experiments, and resulted in fact successful also when tested in more general conditions (see e.g. Fig. 3). The main problem was, in that case, how to achieve a ground truth to test them against. Other problems that should be addressed when scaling up the complexity of the environment are the ability to deal with possibly uneven terrains (although this is not usually the case in indoor scenarios), and to cope with vibrations, dust and difficult light conditions. As for vibrations, we had to solve problems related to oscillations of the 3D camera during relocations (simple filtering techniques were sufficient to cope with the problem in our set-up). Another problem we noticed is arm's occlusion: when designing a real demolition robot, several alternatives should be considered (e.g. a wisely mounted additional camera – on the arm itself, on a little auxiliary arm, on a lateral fixed support etc.). These examples show that a real mock-up, although simplified in some aspects, forces to deal with problems that would have been overlooked in a simulated environment.

8 Conclusions and Future Work

In this work we moved the first steps toward the design and development of novel machines for demolition task, featuring enhanced autonomy, perceptual capabilities, situational awareness, HRI paradigms. A reference set-up has been proposed, as well as a conceptual framework to cope with this task. We then focused on some of the perceptual features that such robots should offer in order to provide high levels of autonomy, safety, and situational awareness. Particularly, consistent efforts were put into the design, development, and evaluation of a novel segmentation algorithm, able to operate in unstructured environments under very general conditions and with minimum a priori assumptions. As for perceptual issues, we also proposed a method for identifying and quantifying spatial deformations occurring during the task. As regards HRI, we proposed a novel paradigm for construction machines based on *laser-designation*. As a first step toward tackling the complexity of a real demolition scenario, we implemented a first, scaled-down, representative mock-up, that allowed us to perform experiments in a controlled environment. With respect to this point, it is to be noted that our algorithmic contributions do not leverage simplifications of the set-up. The lesson learned from this first study will be certainly valuable in scaling up toward the complexity of a real demolition scenario. Future work will focus on the design of a more sophisticated test-bed (e.g. allowing excavation), and on further investigating planning, actuation, and HRI issues for the demolition setting. The planning module, in particular, offers several interesting directions of research, at every level of the proposed hierarchy (from low-level force control to high-level reasoning, such as the capability to understand special situations, ask for human intervention, etc.). We also plan to port developed concepts and algo-

rithms on a full-sized commercial demolition machine, currently being robotized at the PERCRO laboratory.

References

[1] Saidi, K.S., OBrien, J.B., Lytle, A.M.: 47. In: Springer Handbook of Robotics: Robotics in construction. Springer (2008) 1079
[2] Brokk AB: website - http://www.brokk.com
[3] Omata, T., Ishibashi, H., Tomita, K.: Development of double arm working machine for demolition and scrap processing. In: ISARC. (2011) 76–81
[4] Lee, S., Kim, J., Park, J., Seo, J., Kim, Y.: Development of a heuristics-based task planning system for intelligent excavating system. ISARC (2009) 307–316
[5] Cruz-Ramírez, S.R., Mae, Y., Arai, T., Takubo, T., Ohara, K.: Vision-based hierarchical recognition for dismantling robot applied to interior renewal of buildings. Computer-Aided Civil and Infrastructure Engineering **26**(5) (2011) 336–355
[6] Arai, M., Hoshino, H.: Robot system for removing asbestos sprayed on beams. In: 28th ISARC, Seoul, Korea. (2011)
[7] Rusu, R.B.: Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments. PhD thesis, TUM, Germany (2009)
[8] Sánchez, G., Latombe, J.C.: A single-query bi-directional probabilistic roadmap planner with lazy collision checking. Robotics Research (2003) 403–417
[9] Bauer, J., Karner, K., Schindler, K., Klaus, A., Zach, C.: Segmentation of building models from dense 3d point-clouds. In: 27th OAGM. Volume 10. (2003)
[10] Vosselman, G., Dijkman, S., et al.: 3d building model reconstruction from point clouds and ground plans. ISPRS **34**(3/W4) (2001) 37–44
[11] Shamir, A.: A survey on mesh segmentation techniques. In: Computer graphics forum. Volume 27., Wiley Online Library (2008) 1539–1556
[12] Benk, P., Vrady, T.: Segmentation methods for smooth point regions of conventional engineering objects. Computer-Aided Design **36**(6) (2004) 511 − 523
[13] Unnikrishnan, R., Hebert, M.: Robust extraction of multiple structures from non-uniformly sampled data. In: IROS, IEEE (2003) 1322–1329
[14] Rabbani, T., van den Heuvel, F., Vosselmann, G.: Segmentation of point clouds using smoothness constraint. ISPRS **36**(5) (2006) 248–253
[15] Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24**(6) (1981) 381–395
[16] Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (fpfh) for 3d registration. In: ICRA. (2009) 3212–3217
[17] Besl, P.J., McKay, N.D.: A method for registration of 3-d shapes. IEEE Transactions on pattern analysis and machine intelligence **14**(2) (1992) 239–256
[18] Steder, B., Rusu, R.B., Konolige, K., Burgard, W.: Narf: 3d range image features for object recognition. In: Workshop on Defining and Solving Realistic Perception Problems in Personal Robotics at the IEEE/RSJ IROS. Volume 44. (2010)
[19] Kammerl, J., Blodow, N., Rusu, R.B., Gedikli, S., Beetz, M., Steinbach, E.: Real-time compression of point cloud streams. In: ICRA, IEEE (2012) 778–785
[20] Quigley, M., Gerkey, B., Conley, K., Faust, J., Foote, T., Leibs, J., Berger, E., Wheeler, R., Ng, A.: Ros: an open-source robot operating system. In: ICRA workshop on open source software. Volume 3. (2009)
[21] Rusu, R.B., Cousins, S.: 3d is here: Point cloud library (pcl). In: ICRA, IEEE (2011) 1–4