EURECA: Enhanced Understanding of Real Environments via Crowd Assistance

Sai R. Gouravajhala, Jinyeong Yim, Karthik Desingh, Yanda Huang, Odest Chadwicke Jenkins, Walter S. Lasecki
CROMA Lab | Laboratory for Progress | MISC Group
Computer Science & Engineering, University of Michigan – Ann Arbor
{sairohit,jyyim,kdesingh,yodaa,ocj,wlasecki}@umich.edu

Abstract

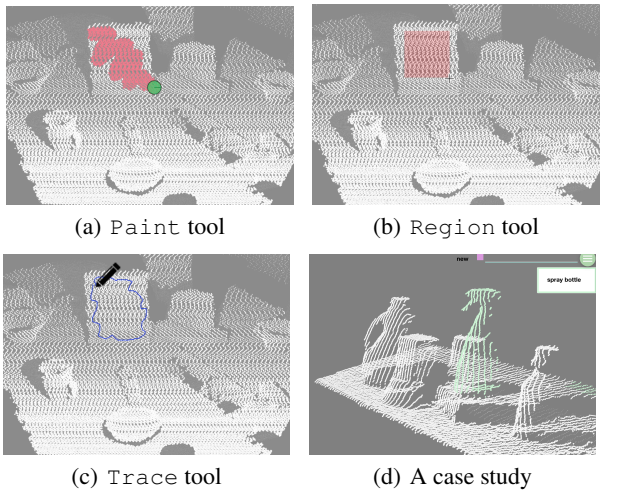
Indoor robots hold the promise of automatically handling mundane daily tasks, helping to improve access for people with disabilities, and providing on-demand access to remote physical environments. Unfortunately, the ability to understand never-before-seen objects in scenes where new items may be added (e.g., purchased) or altered (e.g., damaged) on a regular basis remains an open challenge for robotics. In this paper, we introduce EURECA, a mixed-initiative system that leverages online crowds of human contributors to help robots robustly identify 3D point cloud segments corresponding to user-referenced objects in near real-time. EURECA allows robots to understand multi-object 3D scenes on-the-fly (in ∼40 seconds) by providing groups of non-expert crowd workers with intelligent tools that can segment objects more quickly (∼70% faster) *and* more accurately than individuals. More broadly, EURECA introduces the first real-time crowdsourcing tool that addresses the challenge of learning about new objects in real-world settings, creating a new source of data for training robots online, as well as a platform for studying mixed-initiative crowdsourcing workflows for understanding 3D scenes.

Introduction

Autonomous robots capable of fulfilling high-level end-user requests could revolutionize in-home automation and assistive technology, potentially improving access to the world for people with disabilities, providing a helping hand, and enabling more complete on-demand access to remote physical environments. Yet, robots' ability to identify objects in diverse environments, particularly for objects in settings that have not been previously encountered, remains a barrier to creating and deploying such systems in the wild. Existing 3D computer vision algorithms often fail in new contexts where training data is limited, or in complex real-world settings where scene contents cannot be fully specified in advance. Furthermore, supporting natural language (NL) interaction with end users introduces the significant additional challenge of associating linguistic information with visual scenes (e.g., to identify the target of a request).

We leverage real-time crowdsourcing to create EURECA, a system that helps bridge the gap in understanding between visual scenes and the language used to describe the objects in them in order to make systems that can robustly operate in real-world settings possible.

Copyright © 2018, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.



(a) `Paint` tool (b) `Region` tool (c) `Trace` tool (d) A case study

Figure 1: (a)-(c) Intelligent and collaborative selection tools in EURECA. Crowd workers can choose from three tools to select a group of points for segmenting and labeling 3D point clouds. EURECA takes initiative to automatically augment initial user selections; unintentionally selected points are "filtered" out, and missed points are "filled" in, making final worker selection easier, faster, and more accurate. (d) The scene used in our robot case study, as well as a real crowd worker's annotation of "spray bottle."

As an example, imagine a scenario in which an in-home assistive robot routinely fields requests to pick up or move different household objects. The robot is trained to carry out these tasks and can rely on a wealth of training data available for most objects. However, if the user asks for a newly introduced object (e.g., something they recently purchased) to be retrieved, the robot may fail to complete the requested task using automated methods alone if it does not understand the reference to a new object. EURECA helps robots overcome such failure modes by leveraging on-demand crowds of human workers to collaboratively segment and label unfamiliar objects based on an NL request and a 3D point cloud view of the current scene. Within tens of seconds, the robot under-

stands which object is being referenced and can immediately carry out the request, as well as increase the setting-specific training dataset to improve future automation.

While this presents a powerful way to make robots more robust in real-world settings through on-the-fly training, using human workers as part of the sensing process—especially in non-public spaces, such as home or office settings—introduces privacy concerns. Workers may be able to identify individuals, observe information on documents or whiteboards, and more. To address this, we designed EURECA to be effective even with only depth information (without an RGB image overlayed). This both helps preserve privacy and makes EURECA compatible with a wider range of sensor technology currently used on robotic platforms (e.g., LIDAR sensors).

By combining the machine's ability to precisely select content with people's ability to understand scene semantics, EURECA presents a *hybrid intelligence* approach to 3D annotation—allowing it to benefit from as much automation as possible, while using human intelligence to fill in the gaps. To improve crowd workers' ability to quickly and accurately select objects in a 3D scene, EURECA takes steps towards a mixed-initiative workflow, allowing the crowd to work collaboratively with the system to refine selections for segmentation. Based on initial worker selections, the system automatically infers points to augment those selections (which can even draw on existing 3D vision approaches), with workers able to progressively correct those automatic refinements.

EURECA comprises an interface for selection and scene manipulation (allowing workers to rotate, pan, and zoom) using a series of selection tools, and automated assistance for selection refinement. To further reduce segmentation task latency, EURECA recruits multiple workers on-demand to synchronously complete tasks faster than any lone worker. Coordination mechanisms are provided to prevent redundant or conflicting worker effort.

While EURECA's approach requires no prior human or machine training (and can actually generate training data), it is possible to integrate the output of computer vision approaches for even better results. In fact, we explicitly avoid relying on preprocessing because we target settings where automated systems have already failed. However, if output from vision approaches exist (e.g., preprocessed clusters, labels, etc.), EURECA can use that to make selection easier for workers. This reduces the effort needed from crowd workers and, over time, enables our approach to smoothly transition towards full automation as 3D computer vision methods improve and as more data is collected.

We validate our approach on scenes from an established, publicly-available dataset (Lai et al. 2011) and demonstrate that our annotation baseline tool, `Paint`, leads to per-object segmentation times of 85 seconds for individual workers. From this base approach, we then show that our machine-augmented selection tools, `Region` and `Trace`, which infer final selections based on worker input, further decrease segmentation times by 32%, while increasing object precision and recall by 5% and 9%, respectively. Next, we demonstrate that our techniques for supporting coordination among workers lead to speedups that increase with the number of contributors, further decreasing the average time it takes to annotate objects to just 26.5 seconds each.

We conclude with a demonstration of the end-to-end EURECA system with a Fetch robot[1] that is able to respond to a user's natural language command and accomplish a grasping task. Our work will allow automated object recognition systems to be trained on the fly, creating a seamless, reliable experience between end users and robots. Specifically, we contribute the following in this paper:

- **EURECA**, a mixed-initiative crowd-powered system that leverages non-expert human workers to annotate objects in 3D scenes on the fly.
- **Mixed-Initiative Annotation Tools** for EURECA that help coordinate multiple simultaneous workers on an annotation task to further reduce latency.
- **Validation** that EURECA can achieve high precision (84%) and recall (92%) while keeping latency on par with fully-automated methods (26.5s/object).

Background

Our work is related to crowdsourcing, human computation, 3D sensing for robotics, and visual scene understanding.

Robotics and Semantic Mapping Point cloud data has enabled geometric mapping of 3D space (Endres et al. 2014; Meilland and Comport 2013; Golovinskiy, Kim, and Funkhouser 2009) and a proliferation of robots capable of autonomous navigation in both indoor and outdoor environments. However, the perception capabilities are often limited to the mapping of space without a semantic parsing of individual objects, as well as their afforded actions and language groundings. Even for simple object affordances ("picking" and "placing"), language annotation of objects is essential to establishing a common ground of object references that is both intuitive for humans and perceptible by robots.

Creating Object Geometries For the robotic manipulation of objects, model-free approaches (Ten Pas and Platt 2016; Garage 2008) reason geometrically over 3D point clouds to grasp objects. Such methods do not attempt to semantically distinguish individual objects, and are unable to provide a common grounding for human-robot interaction or reason in a goal-directed manner. Methods using object geometries (Sui et al. 2017; Desingh et al. 2016; Papazov et al. 2012; Narayanan and Likhachev 2016) address these shortcomings, often through a combination of generative and discriminative inference. However, such methods then rely upon object models to be provided *a priori*. EURECA, as a crowdsourcing-based data annotation system, offers one viable option to building such object geometries suitable for real-world scenarios.

Scene Annotation Interfaces In general, there is a lack of annotation interfaces for visual scenes; this is typically because existing work has focused on creating datasets for these applications offline, often curated by experts. Helpful tools like (Russell et al. 2008) are used to create large

[1] http://fetchrobotics.com/platforms-research-development/

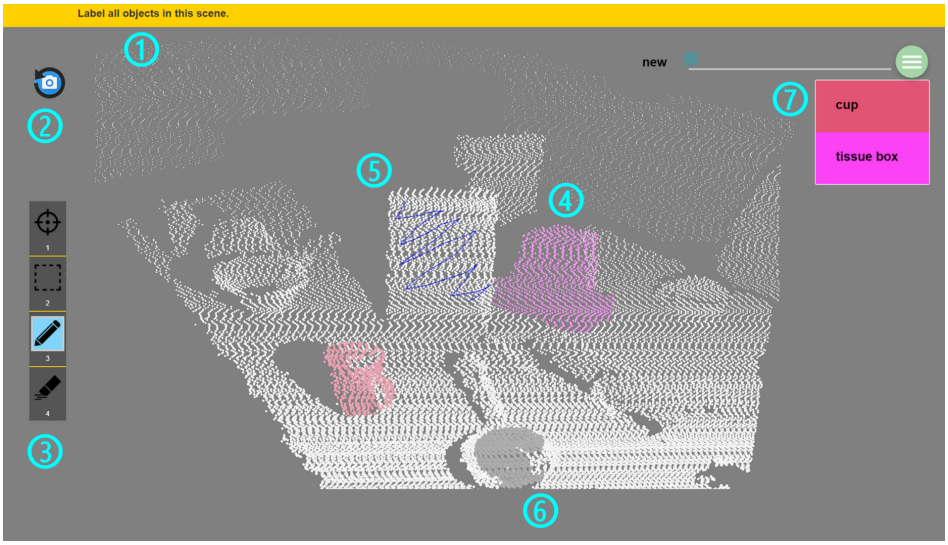


Figure 2: EURECA's worker labeling interface. A typical view includes: (1) Natural language query issued by the end user.; (2) Camera controls allow a worker to easily zoom, pan, and orbit around in the scene; (3) Collaborative selection tools make it easy to select objects (as well as undo any erroneous selections); (4) An object already segmented and labeled by the worker; (5) An object that is currently being selected by the worker; (6) Gray points indicate a remote worker's real-time activity for collaboration tasks; (7) Labeling interface for associating the NL query to object segments.

datasets, but are often time-consuming to create. Furthermore, our envisioned use cases are for the in-home or in-office setting, where preserving privacy becomes an important concern, especially in crowd-powered systems (Kaur et al. 2017). Indeed, removing the RGB data from an image will make it resemble a flat grid; without RGB, performing the kind of outlining discussed in Russell et al. would not be as feasible. If an application wishes to work with another sensor, such as a Velodyne, lack of RGB would be the norm. However, with the 3D point selection that EURECA makes possible, privacy is maintained and segmentation is done online and in real-time.

Crowdsourcing and Human Computation Recent work has explored "hybrid intelligence" workflows that leverage both human and machine intelligence to solve tasks that neither could accomplish alone (Russakovsky, Li, and Fei-Fei 2015; Song et al. 2018).

Our work draws heavily on real-time crowdsourcing, which makes it possible to get rapid responses from crowds of workers—often in less than a second. This response speed makes it possible to create crowd-powered interactive systems. By leveraging real-time crowds to quickly annotate 3D scenes, we make it possible to interact with robots using natural language in real-world scenarios, even when the robot has no prior training on them. For example, Lasecki et al. have created systems capable of generating captions with less than 3 seconds of latency per word (Lasecki et al. 2012) and creating functional UI prototypes (Lasecki et al. 2015). Berstein et al. have created systems for finding the best image from a short video, and generating varied images from a single source (Bernstein et al. 2011).

Since robust, general-purpose computer vision is still a distant goal, visual scene understanding via human computation has been explored by several prior projects. Objects and activities have also been recognized in video using the crowd—Glance (Lasecki et al. 2014) coded behavioral events, Salisbury et al. used augment live video with natural language markers using real-time crowds (Salisbury, Stein, and Ramchurn 2015a), and Legion:AR (Lasecki et al. 2013) recognized human activities in real time. By building upon this body of work, we will integrate visual scene understanding into EURECA.

Robotics and Autonomous Control Prior work has also explored how to use crowdsourcing to augment robotics. Crick et al. show that users provide reliable demonstrations and training for robots when the robots were in sensory-constrained environments (Crick et al. 2011). Moreover, Legion (Lasecki et al. 2011) used real-time crowds to provide continuous control for an off-the-shelf robot that enabled it to follow natural language commands. While Legion provided generalized user interface control, Salisbury et al. (Salisbury, Stein, and Ramchurn 2015b) introduced additional control mediators that improved performance by focusing specifically on robotics applications. de la Cruz et al. (de la Cruz et al. 2015) got feedback from a crowd of workers in ∼0.3 seconds for mistakes made by an automated agent. Chung et al. (Chung et al. 2014) explored learning from initial demonstrations using crowd feedback for motion planning problems.

We clarify that while these references contribute useful selection UIs, none solve the NL resolution problem on the fly. Instead, they are systems for offline segmentation

and data creation. Further, a majority rely on high quality segmentations or classifications from automated systems, which we do not assume is available to EURECA due to our focus on novel objects and settings. Our solution provides near real-time segmentation based on an NL query even in domains where references and objects may be completely unknown to the system (i.e., no available training data).

EURECA: Collaborative 3D Tagging

We build on this related work to recognize objects in settings where automated approaches fail or lack sufficient training data. EURECA recruits crowds of workers on demand, then takes initiative to augment user selections, after which users can further correct updated selections. In this section, we describe EURECA's architecture, including the mixed-initiative workflow, worker UI for interacting with the point cloud, automated support to refine users' object selections, and annotation tools for collaborating with remote workers.

Web-based Annotation Tool

EURECA presents workers with an interactive visualization and annotation tool for 3D point clouds (Figure 2) built in JavaScript using the ThreeJS library[2]. Full 3D point clouds can contain more datapoints than can be rendered at interactive speeds (e.g. a Kinect generates over 300,000 points). To address this, EURECA keeps only every eighth point for a final point cloud size of ∼35,000 points. On page load, crowd workers are shown the point cloud and asked to select and label objects mentioned in a natural language query. Workers can adjust their view of the 3D space using camera controls that let them easily pan, zoom, and orbit a scene. Workers see color highlights of the points they select. To select points, workers are provided with the `Paint` tool (Figure 1(a)) which works by dragging an adjustable-size cursor over the 3D points in a continuous motion (akin to "painting" on the 3D canvas).

To help the crowd select 3D objects more efficiently, we create two additional tools, `Region` and `Trace`. The `Region` tool (Figure 1(b)) allows workers to drag-select a rectangle over a region of interest. Once the click-and-drag event is finished, points that are inside the 2D rectangular region are selected by ray casting a shape matching the worker-indicated region and including all intersected points. For objects that are harder to select with just the `Region` tool—e.g., objects with a more organic shape, or objects that are partially occluded—workers can use the `Trace` tool (Figure 1(c)). Unlike `Region` tool, `Trace` allows workers to draw a free-form region of interest. The points enclosed within the region are highlighted using a ray casting method similar to that used for the `Region` tool.

Mixed-Initiative Workflow

Selection using the tools described above will not always result in perfect object boundaries. Automated refinement is one way to overcome this limitation. A user's ultimate goal of fine-grained selection of a novel object can be thought of as a two-part approach: there is the user's *latent intent*, which involves wanting to perform a fine-grained segmentation of an object (the "goal state"), and then there is the user's *expressed intent*, which involves using the tools in the system. A user's expressed intent is often limited by the selection tools' capabilities (there will be imperfection in this process).

One approach would be to provide smarter and more capable tools to the users. However, direct manipulation using selection tools might not always help achieve the goal state because the user's latent intent is unknown to the system. Our key insight into overcoming this limitation is to instead use a mixed-initiative workflow (Hearst et al. 1999; Horvitz 1999). Within this mixed-initiative framework, users can now *collaborate* with the system's initiative to interactively refine the machine's selection (by taking back initiative). Based on the initial user selections, EURECA takes initiative to **filter** out points that were unintentionally selected by workers, and **fill** in points that it believes were missed in the initial worker selection. As users make repeated point selections for the same object, EURECA starts to better understand the user's high-level (latent) intent that is being expressed through low-level selection actions, thereby building a shared context to achieve the goal of fine-grained segmentations (Figure 3). This lets EURECA's automated selection

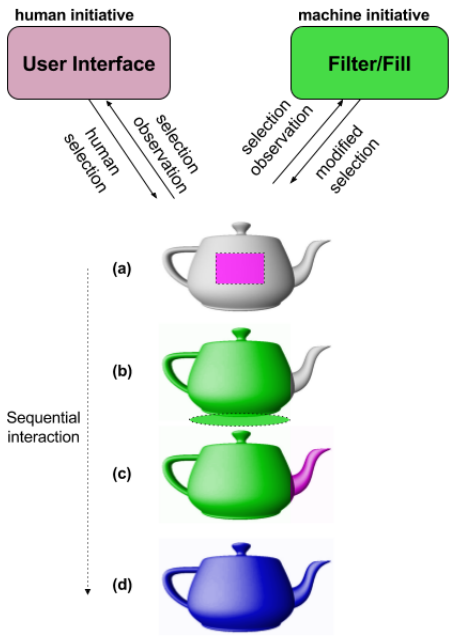


Figure 3: EURECA's iterative, mixed-initiative approach. In (a), the user makes an initial selection (magenta); in (b), the machine observes the selection, takes initiative, and modifies it to fill the rest of the base (green); in (c), the user sees that the system overfilled points (dotted oval), and retakes initiative to clean up that excess selection, and then selects the tea pot's spout; finally, in (d), the machine enters a "no-op" state since there is no more filter/fill to be had.

[2]http://threejs.org

methods iteratively refine the current selection state in tandem with the worker, thereby informing future selections.

As an example of where this mixed-initiative approach is beneficial is in cases where users might not always know the exact object boundaries in the 3D scene. If they mistakenly lump two objects into one selection (if, say, their viewpoint hid the boundaries), a confident system can take initiative, jump in, and adjust the filter / fill process. This mixed-initiative approach, then, can let both the user and the system collaborate effectively.

Point-Filtering ("Filter") To infer points to be removed from a worker's initial selection, EURECA uses a combination of two methods: filtering first by performing outlier detection, and then finding the selection of interest using the Kernel Density Estimation from an off-the-shelf JavaScript library (Davies 2011).

Standard outlier detection, in which points that are significantly distant from the bulk of the selected points are removed, is first performed. This method is not resilient to filtering out points that are within the distance threshold, but still clearly belong to another object (e.g., if there are two objects that occlude each other, a worker's wayward selection can catch points from both objects). Outlier detection is augmented with the KDE method. (We note that EURECA's architecture supports any method that takes in an initial user selection and outputs a refined segmentation, and so use KDE as one such method.)

EURECA builds a density curve of points from the camera's line-of-sight to the initial selection set, based on camera distance. Since the goal is to filter out erroneous selections within the user's line of sight, the algorithm splits on the first local minimum and discards points outside the first cluster. This method filters out points that are behind the object that was "intended" to be selected. A threshold learned from training data is used to avoid splitting off and selecting a cluster that contains only a few points.

Selection-Completion ("Fill") For fill, there is a higher likelihood that points close together belong to the same object. To infer points to add to the initial selection set, EURECA uses a label propagation-based method that is similar to "flood fill" tools in modern graphic editing software (the simplest example of which is "bucket fill" in Microsoft Paint and similar applications).

For each unselected point, EURECA first calculates a constant influence value from a selection point to all points within its neighborhood. Using the kd-tree structure allows for rapid calculation of each point's distance relations to all its neighbors. This is augmented with a term that takes into account how far away this unselected point is from the selection center. Since we assume a worker's initial selection lies mostly within their target object, the second term helps prevent runaway propagation, as points that are too far away will be less likely to be filled in. An inclusion threshold is used to determine which points to add to the final filled-in selection set. A version of the Brushfire algorithm (Choset 2005) is used to estimate the influence on subsequent points. In practice, every point influences its neighbors within a radius that is proportional to the average distances between neighborhoods of points. To slow down the effect of the "brushfire," EURECA adds a penalty on the length of the propagation chain. An inclusion threshold is again used to determine points that are added to the final selection set.

Collaboration and Scaling with Crowd Size

Moreover, EURECA facilitates coordination between multiple workers via real-time feedback on the selection and labeling of synchronous workers. Because we have little to no information about a given scene in our problem formulation, it is difficult to direct workers to non-overlapping parts of the scene to avoid redundant work. Lasecki et al. previously explored using "soft locking" in Apparition (Lasecki et al. 2015), where workers manually placed markers to signal to others that they were contributing in the 2D scene's physical location. We adapt this idea by automatically providing real-time feedback on what other workers are marking via highlighting. This approach, while intuitive, is novel in crowdsourcing systems and generalizes to broader classes of real-time coordination problems. EURECA uses Meteor[3] to create a shared tagging state that allows remote events to be synchronized between workers' local views.

Evaluation

Making interactive robotics applications possible via crowd-augmented sensing requires a combination of speed and accuracy. In the previous section, we described EURECA's architecture. In this section, we introduce the experiments we use to validate the efficacy of this architecture.

Recruiting Crowd Workers

We recruited 78 unique workers with a minimum approval rating of 95% from Amazon Mechanical Turk (AmazonMechanicalTurk 2005). For each task, we paid at an effective hourly rate of $10 per hour, along with a built-in bonus amount for successfully completing a multi-stage tutorial. Once workers pass the tutorial, they are routed to the main task in which they use EURECA to respond to the posted query (e.g. "Select and label the dinner plate").

Point Cloud Dataset

Our evaluation uses scenes from the RGBD Object Dataset (Lai et al. 2011), which consists of color and depth images of naturalistic household and office scenes. Because we wish to explore sensing modes that preserve user privacy, we use only the depth images to generate a 3D point cloud. We selected five scenes with enough diversity in object type, clutter, and orientation to validate object reference resolutions and crowd segmentations (Figure 4). To create the ground truth for evaluation, two researchers carefully annotated the various object segments for each scene.

Measures

We evaluate worker performance using four measures: latency, precision, recall, and the F1 score. We measure latency in terms of the entire session's duration, from understanding to segmentation to NL annotation, and not simply

[3] https://www.meteor.com/

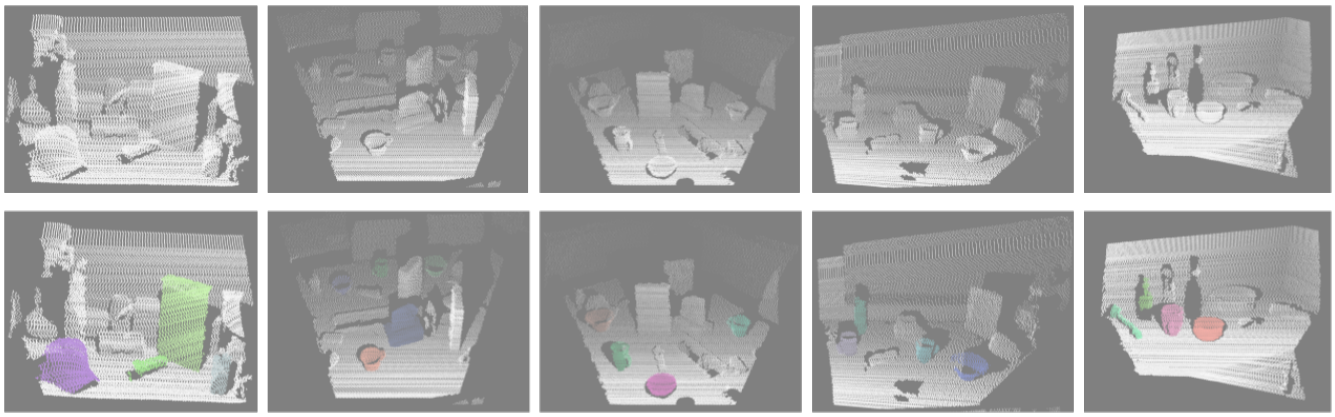


Figure 4: Point clouds used in worker studies. Workers are instructed to segment common household objects. The exact natural language query differs for each scene. For each scene, non-trivial camera movements are required to overcome object occlusion, shadow effects, and orientation in order to identify objects.

the time spent selecting the object. This includes time to understand the scene ("perception") and time to select objects ("selection"). Factors that impact perception include dealing with occlusion, understanding an object's orientation within the scene, and recognizing how distinct an object is by its shape. Factors that impact selection include how easily separable the objects are, as well as how difficult it is to select the object's shape. Therefore, latency will always be longer than time spent segmenting objects.

We divide latency by the number of objects we detect that the worker has labeled. This normalization on a per-object basis lets us compare across scenes with different numbers of objects. We then automatically align worker selections to the best-fit ground truth objects to calculate precision and recall. We report precision and recall for both objects (important for object recognition) and points (important for grasping / motion planning). The F1 score (harmonic mean of precision and recall) gives a combined accuracy measure. We perform paired two-tailed t-test to measure significance.

Study Conditions

We focus on evaluating 1) EURECA's overall efficacy, 2) the effect of our selection tools (with automated refinement) on how quickly and accurately workers can segment objects, and 3) the impact of workers collaborating in teams.

Study 1: EURECA's Effectiveness. To measure the overall effectiveness of EURECA in enabling workers to segment and label objects in 3D point clouds, we identify at the object level how many object instances were correctly identified by the workers. Across the five scenes, there are 21 unique objects (four scenes with four objects each, and one scene with five objects).

We evaluate this recognition in terms of precision (how many objects did the worker correctly identify?) and recall (did the worker correctly identify all the requisite objects in the scene?). We treat each iteration of the scene as a new data point, as there is a chance that a worker might not recognize an object when they see the scene the first time around, but do end up recognizing it the second time they see the scene.

Study 2: The Effect of Automatic Refinement on Selection. To study the efficacy of EURECA's initiative when automatically refining user selections, we task workers with segmenting and labeling objects in the same scene twice: once with only the `Paint` tool enabled (PAINTMODE) and then once with all tools at their disposal (TOOLSMODE), presented in a randomized order.

Study 3: Collaboration in Teams. Next, we want to demonstrate that EURECA's collaborative features enable it to efficiently scale with the number of workers available. We select two scenes with a total of nine distinct objects that needed to be identified. Workers are recruited to a "retainer pool" whenever EURECA is running, and can be directed to a task within one second of a query that the system does not understand arriving. By varying the team size from one to three workers, we can investigate the efficacy of EURECA in enabling worker coordination and collaboration when performing multiple selections.

Results

In this section, we describe the experimental results of related to the core EURECA system, the mixed-initiative tools that support workers, and the benefits of collaboration.

Study 1: EURECA (It Works!)

We recruited 34 workers to use EURECA using only the `Paint` tool. We dropped one outlier whose task duration was more than 3σ from the mean. The remaining 33 workers were distributed across the five scenes: three scenes had seven workers each, and two scenes had six workers each.

We find that the average total time to task completion (both perception and selection for the never-before-seen scene) for all 33 workers, when normalized on a per-object basis, was 85 seconds ($\sigma = 56$s; $p < 0.005$), with 99.6% object-level precision (only one false positive), and 93.9% recall. Of the total of 17 object instances were missed (not recalled) by workers, 10 were completely missed and 7 were

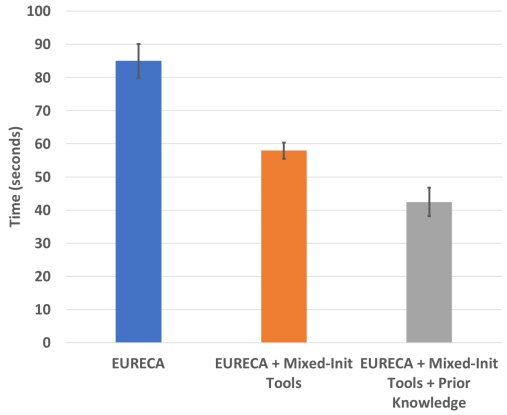


Figure 5: Overall latency per object among the crowd workers in EURECA's various iterations.

combined with other objects into one label. As mentioned earlier, because scene understanding involves both perception and selection factors, we see scene latency times range from 65s to 92s on average.

Study 2: Mixed-Initiative Selection Tools

Although 85 seconds latency per object segmentation and labeling already allows for on-the-fly understanding of novel 3D scenes, we seek to improve this further with EURECA's mixed-initiative tools. Does the progressive refinement of selections help speed workers up?

For the 33 workers, as we see in Figure 5, when TOOLSMODE is enabled, we see a 35% relative improvement in annotation speed to 58 seconds ($\sigma = 28$s). Additionally, we observe an improvement in the average precision and average recall: precision improves from 0.82 to 0.86, whereas recall improves from 0.82 to 0.90).

Worker Improvement Over Time In addition to worker speedups from the mixed-initiative tools, we want to know if workers learn with repeated exposure to the conditions in Studies 1 and 2. We further break down the performance on tasks accounting for the order of task conditions (i.e., whether workers used PAINTMODE or TOOLSMODE first). There were 19 workers who used PAINTMODE first and 14 who used TOOLSMODE first. Workers using PAINTMODE first completed the first task in 87.4s (followed by a second task using TOOLSMODE completed in 61.1s). Workers using TOOLSMODE first completed the first task in 55.6s (followed by a second task using PAINTMODE completed in 81.8s). Comparing across both orders, we find that workers improve over time when they use EURECA.

Moreover, we note that 10 workers (30.3%) did not use TOOLSMODE when they had the option available, which means that they used the `Paint` tool for all object tagging. With this in mind, we can focus specifically on those workers who used at least one of our TOOLSMODE selection tools. For these 23 workers, when in the TOOLSMODE condition, we see a statistically significant 36% improvement ($p < 0.02$) in time taken to tag objects when compared with the time taken in the PAINTMODE condition.

Leveraging A Priori Clustering Information With EURECA, we obtain performance improvements when we add our new selection tools with the system initiative to refine user selections. However, active research is being conducted on devising systems that can segment out objects or surfaces in visual scenes. Such information can provide our tools with improved knowledge and understanding of the scene. In fact, for all new elements that have never been seen before, this kind of automated segmentation is the best that can be done. But, even though we can delineate it in the scene, we still require the proper NL annotation for it. Since both of EURECA's selection tools have the ability to integrate results from any off-the-shelf segmentation algorithm, can performance be improved if our envisioned robot has such *a priori* understanding of its environment?

To test this hypothesis, we recruited 10 workers from Amazon Mechanical Turk and ran the same experimental setup as seen in Study 1 with one of our scenes. We use perceptual grouping of RGBD segments to form object cluster information (Richtsfeld et al. 2012). When we take the average worker performance across all of TOOLSMODE *with* clustering information available, we find a further 37% improvement in speed when compared with the average across all of TOOLSMODE *without* clustering information. Therefore, if prior clustering knowledge exists, EURECA's selection tools can leverage that information to further reduce per-object tagging time (Figure 5).

Study 3: Collaboration Leads to Lower Latency

To understand the ability of teams of workers to complete the annotation tasks, we recruited 24 workers to create four different teams for four scenes. Each team had to segment between four and five objects. We find a large decrease in segmentation time required as we add more workers (Figure 6). Individual workers (teams of size one) took on average 89 seconds ($\sigma = 24$s) to segment the objects, with an overall precision of 0.96 and an overall recall of 0.99 (F1 score of 0.97). When we add one worker (teams of size two), we see a 62% relative decrease in time taken to 34 seconds ($\sigma = 15$s); however, we also see a 14% decrease in preci-

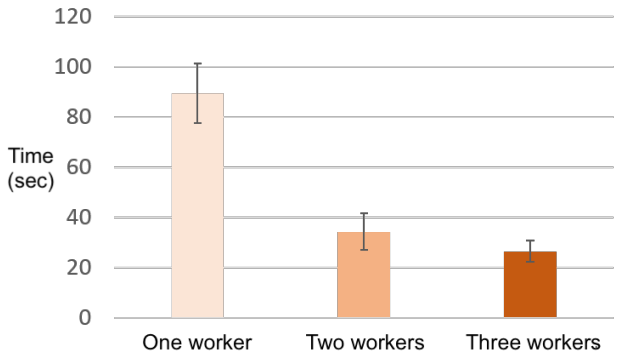


Figure 6: Latency per object in a collaborative setting.

sion along with a 0.2% increase in recall (F1 score of 0.91). Finally, for teams of size three, we see a further 22.5% decrease in time taken for segmentation and labeling to 26.5 seconds ($\sigma = 8.4$s), with a relative decrease in precision of 0.12%, and a relative decrease of 7.3% in recall (F1=0.88).

These results suggest that having workers collaborate with each other offers immediate speed benefits; increasing team size to two leads to a drastic reduction in latency, but at the expense of a decrease in precision. However, precision losses seem to stabilize when an additional worker is added.

During one of the trials, we observed that one team of three did not complete the task because they entered a conflict state in which an error confused all workers into tagging erroneous objects. This suggests EURECA still needs to find more effective ways of enabling explicit worker coordination, especially to rectify mistakes. Future work may explore addressing such conflicts by automatically changing a worker's camera view such that no one worker is looking at the same part of the scene during collaborative tasks.

Case Studies

We have experimentally validated that EURECA enables crowd workers to quickly identify and accurately select, segment, and annotate objects in 3D point clouds, all with near real-time latency. With the novel `Paint` tool, we see speeds of 85 seconds, which we are able to reduce to a best-case scenario of 26.5 seconds with teams of 3 workers. In this section, we explore some case study scenarios to better evaluate EURECA's performance for real life applicability.

Case Study: End-to-End Test with a Robot

We are using the Fetch robot, a mobile manipulation platform mounted with an ASUS depth camera to sense the environment. For this case study, we assume that the robot has bounding boxes and training data for numerous objects. Provided the object locations in the point cloud, the robot uses handle grasp localization (Ten Pas and Platt 2016) and MoveIt! (Sucan and Chitta 2013) (a motion planning library) to manipulate an object. However, when a new object—a *spray bottle*—is introduced, the robot has no way of detecting it, so it places an on-demand request to EURECA. In our case study, the robot successfully picked up the spray bottle—of which it had zero training data on—based on the crowd-generated annotation. Our case study validates that the precision obtained from the crowd's segmentation and annotation using EURECA is enough to enable object manipulation, which is typically seen as a harder task than object annotations for room navigation (as manipulation requires higher segmentation accuracy).

Case Study: Using RGB Color Information

If the lack of RGB color information constraint were relaxed, would that improve worker performance? To investigate this, we repeat Study 1, but workers now see the RGB point clouds. After accounting for two outliers, we find that with N=25 workers, segmentation is 5.9% faster with PAINTMODE (80s), with a 4.9% gain in precision and 2.4% gain in recall. Worker feedback seems to shed some light on this result, as workers found it difficult to delineate the selection colors from the point cloud colors. Future work could look into ways of toggling point cloud colors to make the selection object stand out more clearly.

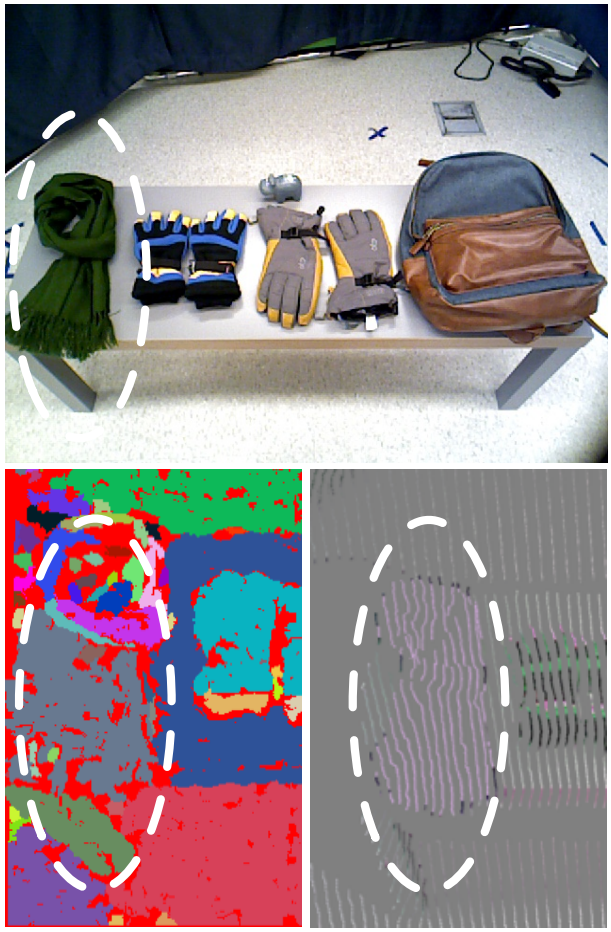


Figure 7: For a deformable object (Top: green scarf within the dotted white oval), PCL's region growing erroneously segments the scarf into multiple distinct regions (Bottom Left), whereas an Amazon Mechanical Turk worker is able to correctly segment and annotate the scarf (Bottom Right).

Case Study: Deformable Objects

We study EURECA's performance on a custom scene with deformable objects where the task is to identify a scarf. Compared with the segmentation using an off-the-shelf region growing algorithm in PCL (Rusu and Cousins 2011) in Figure 7, workers are able to properly segment the scarf.

We can see that an off-the-shelf region growing algorithm erroneously identifies multiple regions for the scarf, whereas crowd workers are able to correctly segment the scarf using EURECA. However, because deformable objects can be rather complex, workers need contextual information to disambiguate between objects if confusion arises (e.g., if the table were to consist of only scarves, then picking out the

correct one would not work). Perhaps crowd workers can separate out the individual scarves and the robot can rely on more clues in the natural language query to annotate the proper scarf (e.g., "the middle one").

Limitations and Future Work

We show that EURECA effectively leverages non-expert crowd workers to annotate 3D scenes in as little as 36s per object for individual workers, and as little as 26s per object when workers collaborate in teams. However, collaboration currently only works in parallel and builds on the "soft-locking" idea seen in (Lasecki et al. 2015). Future work can explore ways of having multiple workers select the same object without conflicts, making the workflow fast enough for natural and continuous interactions with robots.

During our collaboration tests, we did not observe any social loafing behaviors in our tests. We were focused on the functionality of the end-to-end system, and any loafing effect was minimal enough that it did not prevent a significant improvement in the performance of groups over individuals. With further instrumentation, future work can study worker behavior in detail. While this is an interesting problem, studying it sufficiently is beyond this paper's scope.

Furthermore, even when preserving privacy by removing RGB information and downsampling the point cloud by keeping only 10% of initial points, workers are still able to correctly identify common household items with high precision and recall. However, in addition to the scenario we saw in the case study where it is hard to delineate objects into constituent ones (e.g. the saucepan), unique objects could prove problematic for EURECA. Unique objects, such as say a clay dinosaur, would be hard to identify from just the point cloud alone. Indeed, clutter and other scene properties (e.g., camera capture angle) can significantly affect the ability for anyone, computer or human, to both perceive and select objects in 3D scenes.

Our goal was to demonstrate that crowds could be used to segment and annotate objects in real time in "tractable" scenes, which we explored using a common 3D vision dataset in the literature that contains images feasible for highly-trained vision systems to recognize with reasonable accuracy. As a result, EURECA's strength lies in dealing with objects that are familiar to the average worker. Future work could explore how to overcome these bounds by devising workflows that selectively relax privacy constraints.

Finally, EURECA's ability to deal with novelty makes our approach especially relevant to mobile robots. As these robots enter new environments, the likelihood of them encountering unknown and novel objects increases. For these settings, the robot can place on-demand requests to EURECA. We could then take advantage of the class of point-tracking algorithms to map the already-annotated region as the robot moves around its environment. Furthermore, EURECA's fill algorithm can be used to incrementally update this annotated region as more of the object is uncovered (e.g., occlusions disappear as the robot moves around). Future work may address how to introduce approaches that reduce latency and further reduce the amount of human time required for the real-time annotations.

Figure 8: An example case study where the Fetch Robot successfully picked up a spray bottle based on an Amazon Mechanical Turk worker's annotation using EURECA.

Conclusion

In this paper, we present EURECA, a mixed-initiative, hybrid intelligence system that leverages non-expert crowds of human contributors to help robots identify, segment, and label objects in 3D point clouds in near real-time. EURECA allows robots to recognize, on-the-fly, new natural language references to never-before-seen objects. This makes it possible to deploy robots that operate reliably in real-world settings from day one, while collecting training data that can help gradually automate these systems over time.

Acknowledgment

We would like to thank Danai Koutra, Yiwei Yang, Yan Chen, Yilei An, Alan Lundgard, and Stephanie O'Keefe for their valuable input and discussion throughout this work. We also thank the reviewers for their feedback; and finally, we thank all participants in our study. This work was supported in part by IBM and the University of Michigan.

References

AmazonMechanicalTurk. 2005. Accessed: 2017-04-04.

Bernstein, M. S.; Brandt, J.; Miller, R. C.; and Karger, D. R. 2011. Crowds in two seconds: Enabling realtime crowd-powered interfaces. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 33–42. ACM.

Choset, H. M. 2005. *Principles of robot motion: theory, algorithms, and implementation*. MIT press.

Chung, M. J.-Y.; Forbes, M.; Cakmak, M.; and Rao, R. P. 2014. Accelerating imitation learning through crowdsourcing. In *2014 IEEE Robotics and Automation (ICRA)*, 4777–4784.

Crick, C.; Osentoski, S.; Jay, G.; and Jenkins, O. C. 2011. Human and robot perception in large-scale learning from demonstration. In *Proceedings of the 6th international conference on Human-robot interaction*, 339–346. ACM.

Davies, J. 2011. Science.js. `https://github.com/jasondavies/science.js`.

de la Cruz, G. V.; Peng, B.; Lasecki, W. S.; and Taylor, M. E. 2015. Generating real-time crowd advice to improve reinforcement learning agents. In *Workshops at the 29th AAAI Conference on Artificial Intelligence*.

Desingh, K.; Jenkins, O. C.; Reveret, L.; and Sui, Z. 2016. Physically plausible scene estimation for manipulation in clutter. In *16th International Conference on Humanoid Robots (Humanoids)*, 1073–1080. IEEE.

Endres, F.; Hess, J.; Sturm, J.; Cremers, D.; and Burgard, W. 2014. 3-d mapping with an rgb-d camera. *IEEE Transactions on Robotics* 30(1):177–187.

Garage, W. 2008. Pr2 interactive manipulation. Accessed: 2017-04-04.

Golovinskiy, A.; Kim, V. G.; and Funkhouser, T. 2009. Shape-based recognition of 3d point clouds in urban environments. In *Computer Vision, 2009 IEEE 12th International Conference on*, 2154–2161. IEEE.

Hearst, M. A.; Allen, J.; Horvitz, E.; and Guinn, C. 1999. Mixed-initiative interaction. *IEEE Intelligent Systems* 14(5):14–23.

Horvitz, E. 1999. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, 159–166. ACM.

Kaur, H.; Gordon, M.; Yang, Y.; Bigham, J. P.; Teevan, J.; Kamar, E.; and Lasecki, W. S. 2017. Crowdmask: Using crowds to preserve privacy in crowd-powered systems via progressive filtering. In *Proceedings of the AAAI Conference on Human Computation (HCOMP 2017)., HCOMP*.

Lai, K.; Bo, L.; Ren, X.; and Fox, D. 2011. A large-scale hierarchical multi-view rgb-d object dataset. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, 1817–1824. IEEE.

Lasecki, W. S.; Murray, K. I.; White, S.; Miller, R. C.; and Bigham, J. P. 2011. Real-time crowd control of existing interfaces. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 23–32. ACM.

Lasecki, W. S.; Miller, C. D.; Sadilek, A.; Abumoussa, A.; Borrello, D.; Kushalnagar, R.; and Bigham, J. P. 2012. Real-time captioning by groups of non-experts. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*.

Lasecki, W. S.; Song, Y. C.; Kautz, H.; and Bigham, J. P. 2013. Real-time crowd labeling for deployable activity recognition. In *Proceedings of the 2013 conference on Computer supported cooperative work*, 1203–1212. ACM.

Lasecki, W. S.; Gordon, M.; Koutra, D.; Jung, M. F.; Dow, S. P.; and Bigham, J. P. 2014. Glance: Rapidly coding behavioral video with the crowd. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, 551–562. ACM.

Lasecki, W. S.; Kim, J.; Rafter, N.; Sen, O.; Bigham, J. P.; and Bernstein, M. S. 2015. Apparition: Crowdsourced user interfaces that come to life as you sketch them. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 1925–1934. ACM.

Meilland, M., and Comport, A. I. 2013. On unifying key-frame and voxel-based dense visual slam at large scales. In *2013 IEEE/RSJ Intelligent Robots and Systems (IROS)*, 3677–3683. IEEE.

Narayanan, V., and Likhachev, M. 2016. Discriminatively-guided Deliberative Perception for Pose Estimation of Multiple 3D Object Instances. In *Proceedings of Robotics: Science and Systems*.

Papazov, C.; Haddadin, S.; Parusel, S.; Krieger, K.; and Burschka, D. 2012. Rigid 3d geometry matching for grasping of known objects in cluttered scenes. *The International Journal of Robotics Research* 0278364911436019.

Richtsfeld, A.; Mörwald, T.; Prankl, J.; Zillich, M.; and Vincze, M. 2012. Segmentation of unknown objects in indoor environments. In *2012 Intelligent Robots and Systems (IROS)*, 4791–4796. IEEE.

Russakovsky, O.; Li, L.-J.; and Fei-Fei, L. 2015. Best of both worlds: human-machine collaboration for object annotation. In *CVPR*.

Russell, B. C.; Torralba, A.; Murphy, K. P.; and Freeman, W. T. 2008. Labelme: a database and web-based tool for image annotation. *International journal of computer vision* 77(1):157–173.

Rusu, R. B., and Cousins, S. 2011. 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)*.

Salisbury, E.; Stein, S.; and Ramchurn, S. 2015a. Crowdar: augmenting live video with a real-time crowd. In *Third AAAI Conference on Human Computation and Crowdsourcing*.

Salisbury, E.; Stein, S.; and Ramchurn, S. 2015b. Real-time opinion aggregation methods for crowd robotics. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, 841–849.

Song, J. Y.; Fok, R.; Lundgard, A.; Yang, F.; Kim, J.; and Lasecki, W. S. 2018. Two tools are better than one: Tool diversity as a means of improving aggregate crowd performance. In *23rd International Conference on Intelligent User Interfaces*, IUI '18, 559–570. New York, NY, USA: ACM.

Sucan, I. A., and Chitta, S. 2013. moveit!. Accessed: 2017-04-04.

Sui, Z.; Xiang, L.; Jenkins, O. C.; and Desingh, K. 2017. Goal-directed robot manipulation through axiomatic scene estimation. *The International Journal of Robotics Research* 36(1):86–104.

Ten Pas, A., and Platt, R. 2016. Localizing handle-like grasp affordances in 3d point clouds. In *Experimental Robotics*, 623–638. Springer.