Interactive singulation of objects from a pile

Lillian Chang*†, Joshua R. Smith†‡, Dieter Fox†

***Abstract*—Interaction with unstructured groups of objects allows a robot to discover and manipulate novel items in cluttered environments. We present a framework for interactive singulation of individual items from a pile. The proposed framework provides an overall approach for tasks involving operation on multiple objects, such as counting, arranging, or sorting items in a pile. A perception module combined with pushing actions accumulates evidence of singulated items over multiple pile interactions. A decision module scores the likelihood of a single-item pile to a multiple-item pile based on the magnitude of motion and matching determined from the perception module. Three variations of the singulation framework were evaluated on a physical robot for an arrangement task. The proposed interactive singulation method with adaptive pushing reduces the grasp errors on non-singulated piles compared to alternative methods without the perception and decision modules. This work contributes the general pile interaction framework, a specific method for integrating perception and action plans with grasp decisions, and an experimental evaluation of the cost trade-offs for different singulation methods.**

I. Introduction

An ongoing challenge in robotics is interaction in unstructured environments. In particular, objects may be placed close together or overlapped such as in a pile of toys or groceries. The composition of a pile depends on the properties of the base unit items. A common example is a pile of primarily rigid objects, such as a boardgame pieces or a stack of books. In other cases, items may be articulated or deformable objects, such as pile of dishrags or a jumble of rope. The pile of fine granularity may itself even be considered as a single deformable material, such as a heap of chopped vegetables or sugar. Robust exploration and interaction with piles will be necessary for a service robot that encounters clutter in household environments.

The interaction strategies may range from the singulation of individual units, to operation of groups of units, or even direct classification and manipulation of a material pile as one entity. In this work, we focus on the first case of determining and singulating individual items from among a pile of multiple and previously unseen objects (Fig. 1). Tasks that involve perception and manipulation of multiple items, such as counting, arranging, or sorting, have a common requirement for knowledge of what constitutes an individual object within a pile.

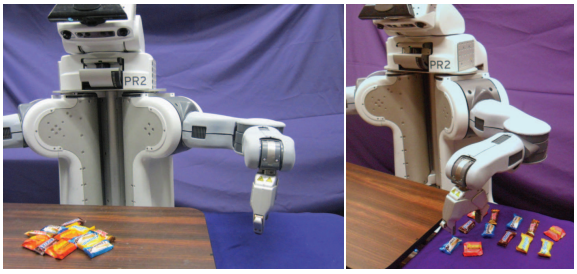


Fig. 1. One useful interaction with piles is the singulation of individual items. Interactive singulation enables a robot to perform tasks such as counting, arranging, or sorting of previously un-encountered objects.

Cluttered spatial placement complicates both perception and manipulation. Object recognition may fail due to occlusion by or proximity to surrounding objects. Grasping may also fail when unintentionally applied to a target that is actually multiple objects (Fig. 2), either by grasping more than one item (Fig. 2(a)) or losing the grasp due to shifting between items (Fig. 2(b)). Tasks requiring an individually-grasped object depend on a chance "lucky grasp" of a single item out of a pile (Fig. 2(c)). Thus, physical singulation can improve the performance of retrieving and modeling novel objects, as well as recognition and search for known objects among piles.

Our goal is that the robot be able to identify and physically separate individual objects from a pile of novel items. We focus on the advantage of singulation in reducing grasping errors rather than improving object recognition since the items are unknown. Our framework includes the integration of perception of the scene state with a manipulation plan for interacting with the pile. This singulation process is intended as an initial learning stage that would precede and facilitate future behaviors such as object modeling, object recognition, or grasping, where spatial separation improves performance.

The following sections present related work (Section II) and a general interaction framework (Section III) that is applicable to multiple tasks and strategies. A specific singulation strategy is developed for the example task of finding and arranging individual items from a pile (Section IV). The singulation method integrates actions that perturb the pile with an evaluation of the motion to accumulate singulation evidence before grasp attempts. The perception module used in the proposed strategy is described in Section V. The

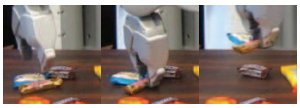
*L. Chang is with Intel Corporation, at the University of Washington Intel Science and Technology Center, Seattle, WA, USA

†L. Chang, J. Smith, and D. Fox are with the Department of Computer Science & Engineering, University of Washington, Seattle, WA, USA

‡J. Smith is also with the Department of Electrical Engineering, University of Washington, Seattle, USA

E-mail: {lchang,jrs,fox}@cs.washington.edu

This material is based upon work supported by the National Science Foundation under Grant # 1019343 to the Computing Research Association for the CIFellows Project. This work was funded in part by an Intel grant and by ONR MURI grant N00014-09-1-1052.



(a) Multiple-item grasp

(b) Empty grasp

(c) Lucky grasp

Fig. 2. Grasp outcomes on non-singulated piles. (a) **Multiple-item grasps** and (b) **empty grasps** of no items fail to retrieve individual objects due to surrounding clutter. Without singulation, grasping individual items depends on the chance occurrence of (c) **lucky grasps** from an object cluster.

proposed singulation strategy is evaluated against two alternative strategies of (1) singulation by only grasping and (2) singulation by fixed pushing (Section VI). The accumulation of perceptual evidence for singulation decreases the number of grasping failures on non-singulated piles. We conclude in Section VII with observations about directions for further increased singulation efficiency and applications to behavior learning.

II. Related Work

Singulation of objects from piles has been examined previously for the bin-picking application. Several methods for workpiece acquisition reframe the problem as the identification of potential grasping or hold sites based on the local shape of objects, without explicitly segmenting an individual workpiece from the pile. For example, Kelley et al. [1] describes vision algorithms for identifying smooth surfaces suitable for vacuum grippers or hold sites for parallel jaw grippers (see also [2]). More recently, this approach of matching or filtering surface features for a particular gripper geometry was also achieved using 3D range data and the PR2 grippers for a cashier checkout operation [3]. Other investigations of bin-picking have focused on machine vision techniques for pattern recognition of a known workpiece shape [4, 5] or segmentation of the topmost piece in a pile [6]. Our approach does not assume prior modeling of the objects and can also be used to interact with objects that are not graspable by the robot end effector.

Related work has also investigated "interactive techniques" that integrate robot actions with object recognition, perception, or modeling. Work by Sinapov et al. [7] investigates object recognition from auditory cues as a result of object manipulations such as dropping or crushing, and assumes individual items have already been singulated. Object segmentation from video sequences of robot pushing actions has been demonstrated for rigid objects by Fitzpatrick [8], Kenney et al. [9], for articulated objects by Katz and Brock [10, 11], and for symmetric rigid objects by Li and Kleeman [12]. These works focused on the segmentation of objects from a video of a single pre-planned robot motion for the target object, and the techniques could be adapted as the perception modules in our framework. Our work builds upon these investigations of perception component to achieve a full framework that includes the decision model to aggregate evidence of singulation over a sequence of multiple interactions, which individually may not provide sufficient evidence for singulation.

Another area of related work is interactive object modeling, such as next-best view planning investigated by, e.g. Krainin et al. [13] and Kriegel et al. [14]. These methods focus on a single object either already grasped in-hand by a robot or fixed in the world relative to a moving camera. The aim of these methods is to create a complete model of the object geometry, and these techniques would complement our results to create an object model once it is separated from clutter.

Modeling and planning push actions has been investigated recently by Kopicki et al. [15], Dogar and Srinivasa [16] and Kappler et al. [17]. These works assume a context where the individual object model is known. In particular, Dogar and Srinivasa [16] presents push-grasping as a framework for dealing with clutter in the environment, and Kappler et al. [17] derives humanoid pushing actions for novel objects based on template patterns. The robot motion plans depend on previous knowledge of object models for the target and the cluttering objects. Our method could be used as an initial exploration step for identifying individual movable items among the clutter.

III. Pile interaction framework

This section describes a common framework for representing possible strategies of interactive singulation. Here we intentionally present general functions that would be applicable to multiple strategies and tasks, before Section IV describes a particular singulation strategy for item arrangement. Our framework assumes that the application task is suited to an iterative or recursive approach where individual items are singulated until task completion and/or no uncertain segments remain from the original input pile.

A. Terminology

A singulation strategy consists of a set of discrete action types and evaluation policies that are applied to *spatial units* within a scene. We use the following terminology to describe the proposed singulation framework.

- **Item**: an actual physical entity, the real object.
- **Spatial unit**: the representation of a component of the scene. Spatial units are candidates for actual items but may include multiple items.
- **Target**: the identity of particular single or multiple spatial units involved during an action.
- **Scene State**: a set of spatial units and the attributes associated with the spatial units.
- **Action**: a discrete type of primitive interaction that is applied to a target.
- **Action History**: a sequence of actions, the targets of those actions, and the action outcomes. Action outcomes may be implicitly included in updates of the Scene State, or may be explicitly included in the action history.

Specific definitions of these concepts depend on the perceptual module and action planner of a particular singulation strategy. For example, in our work, a spatial unit essentially represents a pile of interest items and is represented as a spatially separated point cloud, or cluster of 3D colored points. Alternative definitions, not implemented in this work, may define a spatial unit as a 3D volume unit in the scene (which may contain a pile and/or other structures), a 2D region of an image, sets of multiple piles, or sets of surface patches that represent object faces.

Example action primitives used in this work are grasp attempts and perturbation pushes of a spatial unit. These categories could be further granulated into specific types of grasps or pushes depending on the sophistication of the manipulation action. For a robots with different actuation capabilities, action primitives could include vibrating or perturbing a container or tray holding the items or combined bimanual actions such as sweeping and fencing attempts.

B. Pile interaction algorithm

A general algorithm for pile interaction is an iterative or recursive process that acts on spatial units until the task is complete. The base cycle is a sequence of steps for target selection, action selection, action evaluation, and state updates, illustrated in Fig. 3.

A particular interaction strategy includes a specific set of action primitives and the perceptual module for action evaluation. The criterion for process termination and target selection also depend on the application task. For example, a counting task ends when all spatial units have been accounted, and a search task ends when the matching piece has been located.

Our approach considers that target selection depends not only on the current state of scene but also the history of previous actions. Thus a useful form for the SelectTarget function includes an option for continuing interaction with a previous target, as shown in Fig. 3.

For an interaction strategy that processes or recurses over all spatial units in the scene, two categories of action primitives are (1) Completion or removal of a unit from further consideration and (2) Interaction with the unit to acquire further observations or change the scene state.

The perceptual module is the key component of action evaluation and also affects the method for updating attributes or the scene state. Evaluation results can include changes in both

- the "topology" of the scene, e.g. from splitting or merging of spatial units, or vanishing of a unit due to removal, and
- the state of a single spatial unit, such as a motion that changes the attributes of the action's target spatial unit.

The following sections describe specific action and perceptual modules for an example application of counting and arranging items found in a pile.

IV. Singulation Strategy for Arrangement Task

We investigate how the general pile interaction framework described in Section III can be applied for a task of singulating items from a pile for the purpose of counting or

Pile interaction strategy

```
ActionHistory = ∅
SceneState = GetCurrentState()
while TaskIncomplete(SceneState, ActionHistory) do
    // Interact with the scene
    Target=SelectTarget(SceneState, ActionHistory)
    Action=SelectAction (Target, SceneState, ActionHistory)
    Result=EvaluateAction (Target, Action)
    // Evaluation includes execution of action
    SceneState, ActionHistory = UpdateState (Target, Action, Result, SceneState, ActionHistory )
end while
```

SelectTarget(*SceneState*, *ActionHistory*)

```
if ContinueTestingLastTarget(ActionHistory) then
    return LastTarget
else
    return HighestUtilityTarget(SceneState)
end if
```

SelectAction(*LastTarget*, *SceneState*, *ActionHistory*)

```
if ConfidentOfSingulation(Target) then
    return CompleteAction
else
    return TestAction
end if
```

Fig. 3. Framework for representing general pile interaction strategies. The main loop consists of four stages to select and test a target for singulation. The structure of the **EvaluateAction** and **UpdateState** stages will depend on the action primitives and the perception module. Definitions of task completion and termination criteria for continuing testing on a target will depend on the application task.

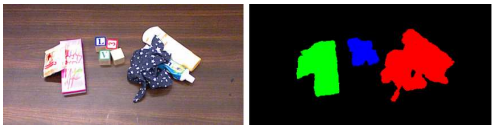


Fig. 4. Multiple objects may appear as a single segment after clustering 3D data due to lack of spatial separation.

spatial arrangement (Fig. 1). The objects composing the pile are not known a priori. Here we describe the main aspects of the singulation approach, and Section V presents the specific perception module and pile classification in further detail.

For this specific singulation task, we define a spatial unit as a cluster of 3D points, or point cloud, in the scene obtained from a depth camera. A target is a single spatial unit, and actions are not planned for multiple units. For each cycle of interaction, spatial units are identified from the cloud data of the entire scene by first removing the plane of the table support surface and then spatially clustering the remaining cloud points. The scene state is the set of these tabletop point clouds. As shown in Fig. 4, spatial clustering is not necessarily sufficient to segment individual objects in clutter. The goal is to separate individual items such that a target spatial unit represents only a single item before the removal action.

The action primitives available for this task are either (1) pushes as the only type of *TestAction* or (2) grasps as the *CompleteAction* to remove or finish processing a target.

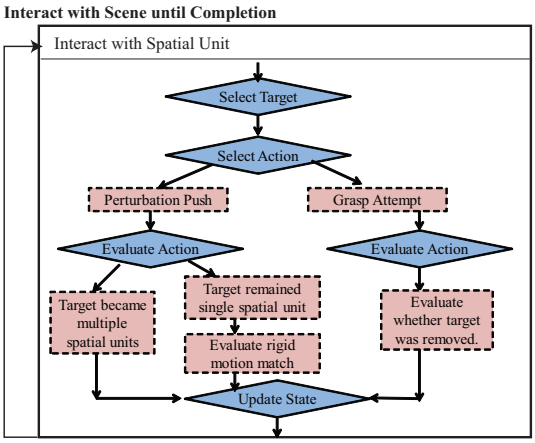


Fig. 5. Pile interaction flow chart. The diamond blocks represent the steps of the general pile interaction framework. The dashed boxes are the components of the specific singulation strategy investigated here for the arrangement task.

Grasp attempts are intended as completion actions once a spatial unit is deemed to be completely singulated. Pushing serves dual purposes to either accumulate additional evidence that a spatial unit is singulated or perturb a spatial unit to elicit a topological change in the scene state to create new spatial units as item candidates. Figure 5 illustrates an overview of how this singulation strategy fits into the previously described framework.

Successful grasps remove units from the scene to an arrangement area for completed items. The task is considered complete when no spatial units remain on the table surface, otherwise **TaskIncomplete** returns *True*.

Target selection by the **SelectTarget** function returns the spatial unit best matching the last action's target in order to accumulate evidence for the same unit over consecutive cycles. For the first pile action or a "reset" state, pile size determines target selection. Thus, the *Highest Utility Target* is modeled as the smallest spatial unit, measured by footprint area on the table plane. When a previous target is completed due to removal or termination due to scene topology changes, the smallest pile is considered the most likely unit to be a candidate for a singulated item.

Pushes to perturb a target are planned based on the target's location relative to surrounding units. Pushes are straight line motion paths of the robot end-effector in the plane of the support surface. The purpose of a push is to provide a motion cue to accumulate evidence of a singulation. In addition, the interaction should avoid topology changes due to re-merging of a spatial unit that is already separated. The push direction is selected to move a target away from other spatial units. Since the spatial units are unknown and potentially un-singulated, the pushes are only planned heuristically assuming the unit is a single item. Thus, the push direction is determined as the direction toward the unit centroid where translation of the sampled footprint boundary points has large forward clearance and small backward clearance. The length of push is determined by the distance from the footprint boundary to the centroid and is capped at a fixed maximum distance. Here the push plan does not model rotations due to moments nor require a model of object surface properties.

The experiments presented in Section VI compare three singulation strategies for the arrangement task. The three strategies are a grasping only method, a fixed pushing and grasping method, and an adaptive strategy with pushing and grasping with the accumulation of singulation evidence over multiple pushes. All three variations share the definition of spatial units as clusters of 3D points and a common grasping action planner. The next section describes the adaptive strategy's perception module that provides singulation evidence for the **EvaluateAction** and **ConfidentOfSingulation** functions.

V. Perception Module and Classification Decision

To reduce the risks associated with grasping non-singulated piles or spatial units (Fig. 2), a completion or removal grasp should only be performed when there is high confidence in item singulation. A perceptual module that evaluates outcomes of perturbation actions provides evidence in support of or against singulation that can be accumulated over multiple actions.

A. Assumptions and approach

In the example arrangement task, we focus on the case of singulating a pile consisting of primarily rigid objects. We assume that the items have visual or geometric features that can be used to identify candidate correspondences between key points in different states of the spatial unit.

Note the several challenges to successful perception of the scene state and action outcome. First, the 3D points representing a spatial unit do not encompass a full surface model of unknown items. This partial observation of spatial units is due to limited views of the scene from the direction facing the camera or depth sensor, as well as limitations common to any real world setting where an object rests on a support surface which essentially occludes the bottom view even in the presence of multiple sensors. Second, interaction often results in occlusion by the manipulator itself during the pushing action, sometimes completely blocking the target from camera view. Third, matching of features between time frames provides candidate correspondences but are not guaranteed to be correct due to noise and the existence of multiple similar points in the scene or the item itself.

Given these conditions, our perception module evaluates an action based on only the input and output scene states before and after a pushing action. This approach achieves a full view of the target spatial unit, free of occlusion by the robot arm which may cover the target during the push manipulation. Unlike the goals of object tracking or continuous segmentation, our application does not require identifying the spatial unit in each time frame of action in order to gather evidence of singulation.

B. Spatial unit matching

The first portion of the perception module matches spatial units between the before and after states of a perturbation.

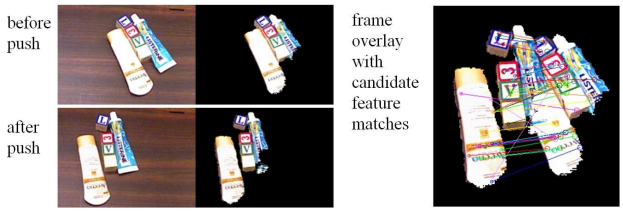


Fig. 6. A pile shifts shape after interaction. The segmented pile before the push (top) is compared to the state after the push (bottom). Local features (right) are sampled to generate candidate transforms to determine if a single rigid motion explains the new state. Rarely do the points in the two frames match completely due to noisy edges of the pile segmentation or changes in the partial object view due to the perturbation.

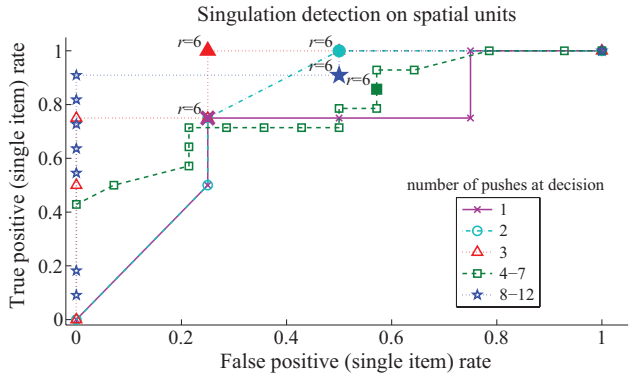


Fig. 7. Performance of singulation detection for different numbers of perturbation push actions. The point corresponding to the selected ratio threshold $r = 6$ is shown for all curves. This ratio threshold is used in the experiments described in Section VI.

The purpose is to identify a correspondence to the target of interest rather than evaluating the entire scene. We assume a stationary camera in relative to the scene and ignore any spatial unit containing points that remain unchanged (similar to a background subtraction scheme). If there are multiple non-stationary spatial units in the output state, this indicates a topology change in the scene state and is evidence that the original target was non-singulated. At this stage the perception module returns without further evaluation as the scene state has been reset.

C. Candidate rigid transforms

In the case where only a single spatial unit has changed, the target may still contain multiple items that were not sufficiently perturbed to spatially separate. The second portion of the perception module evaluates whether a single rigid motion is sufficient to describe the observed change (Fig. 6). Feature points in the before state and candidate matching points in the after state frame provide a set of sparse correspondences. Multiple candidate rigid motion transforms are determined from the sparse correspondences using a cascaded RANSAC method on the sparse feature points only.

Given a set of multiple rigid transforms (in our implementation, a maximum of 5 transforms), each transform is then evaluated against the dense point clouds representing the entire before and after spatial unit states. The strength of a single transform to describe the state change is measured by the percent of point matches relative to the number of points in the original target state. A point match occurs if the 3D transformation of a dense point results in a 3D location whose neighborhood includes a point in the second point cluster with a similar color. The single transform with the highest percentage match is retained and the others discarded.

D. Estimate of single likelihood ratios for singulation

The third portion of the perception module is evaluation of that the evidence from the single retained transform with respect to target singulation. We determine a likelihood ratio of a target being a single item or multiple items based on the magnitude of the transform motion and the percentage of dense point matches.

The model is derived from a small set of example pushes on objects not used in the experiments described in Section VI. Training data was captured from 8 sequences of 5-12 pushes of a pile that did not separate into multiple spatial units. Four sequences were of a single-item pile, and four sequences were of a multiple-item pile with 2-6 objects. The data was divided into small and large motions according to the translation magnitude of the best matching rigid transform. For small scale motions where the target moves very little, there is less distinction between single- and multiple-item piles since a near-null transform can describe the state change. Larger scale motions provide more differentiation because multiple items would tend to shift relative to each other.

The sorted examples provide the likelihood of observing a particular dense point match given the motion magnitude and the label of single or multiple objects. For a single push, this model gives the likelihood ratio that the target was singulated or consisted of multiple items. From a sequence of pushes, the likelihood ratio of single pushes is multiplicatively accumulated for a target spatial unit. This accumulation integrates observations from multiple interactions with a pile. It also allows for recovery from failed pushes where partial views change too dramatically to find sufficient sparse or dense point matches.

E. Classification of cumulative likelihood ratios

The action selection module depends on the results of the perception module to decide whether to remove or continue perturbing a target. This represents a classification on the value of the likelihood ratio between the two classes of single item or multiple items. Classification of the cumulative ratios from the 8 training data sequences was tested for a range of thresholds. Figure 7 shows the performance of the classifier as a set of ROC curves for different number of pushes. Note that the curve for a sequence of one push only has the worst performance or smallest area under the curve.

For the experiments described in Section VI, we chose a likelihood ratio of $r = 6$ as the threshold for sufficient confidence of target singulation. The points corresponding to this threshold, marked on Fig. 7, show a reasonable tradeoff being high true positive rate and low false positive rate relative to what is possible from the entire performance curve.

VI. Experimental validation

A. Strategy comparison: Experimental Set-up

We evaluate three singulation strategies for an item counting and arrangement task on the physical robot platform shown in Fig. 1. The manipulator is a PR2 robot (Willow Garage) whose base is stationary in the scene. A Kinect depth-camera (Microsoft Corporation) mounted on the PR2 head provides point cloud data for the perceptual evaluation of spatial units. In each strategy, an item is "counted" if closure of the fingers during the grasp attempt fails, based on the position sensors of the PR2 gripper.

The three strategies are designed to separately compare the value of pushing actions and the value of the proposed perception and decision module:

1) Grasping only strategy: Action types are limited to only grasp attempts on the target spatial unit. Target selection always returns the smallest spatial unit in the current scene. In this strategy, **ConfidentOfSingulation** always returns *True* to execute grasps only.

2) Fixed pushing and grasping strategy: Action types are grasp attempts and perturbation pushes. For a new target, two pushes are executed before the first grasp attempt. If no item is grasped/counted, subsequent grasp attempts occur after one additional push until the target is reset. **ConfidentOfSingulation** returns *True* if the *ActionHistory* includes at least two pushes.

3) Adaptive interaction: Action types are grasp attempts and perturbation pushes. The likelihood ratio of a target is accumulated while there are no topology changes in the target. For a new target, grasp attempts are only considered after two pushes based on whether the cumulative likelihood ratio of singulation exceeds the threshold $r = 6$. If no grasp is attempted or no item is grasped, subsequent grasp attempts are considered after each subsequent push until the target is reset. Thus, **ConfidentOfSingulation** always returns *True* if the updated state after **EvaluateAction** of the last push has a ratio exceeding the threshold $r = 6$ and the *ActionHistory* includes at least two pushes.

For all methods, the function **ContinueTestingLastTarget** returns false if there are more than 10 pushes in the *ActionHistory*, if the pile state has been "reset" due to lack of a matched spatial target, or if the last target was removed due to a *CompleteAction* grasp.

In the particular implementation of the perception module for the adaptive strategy, sparse feature points for initial target states are determined by Harris corner detection, and candidate correspondences in the outcome state after a push are determined from optical flow using the OpenCV library on the Kinect image data.

The strategies were tested on four initial pile conditions with different real-world items that were not part of the training data described in Section V. Grasp attempts were categorized according to the labels in Table I which are based on the actual number of items in the target spatial unit and the items physically grasped by the robot. Singulation success occurs when a single item is grasped. Successes are divided into single grasp success where the target only contained a single item and lucky grasps where the target contained multiple items. Singulation errors include grasps of multiple objects or no objects in an empty grasp. Another error outcome is the case of "lost items" that are removed from the initial scene but not delivered to the arrangement destination. This case can occur when there are only partial or insecure grasps that drag or prematurely release the object, failure to detect a grasped object in the gripper, or large perturbations of the pile that discard the object from view.

TABLE I
Outcomes for grasp attempts

Grasp attempts are categorized by target pile type and the final grasped items. Separately, items are lost (vii) if discarded from the workspace. No examples of (iii) cluttered grasps occurred in the experiments.

Grasped items	Number of items in target pile: 1, singulated item	$N>1$, multiple items
1 item	(i) Single grasp	(ii) Lucky grasp
$N>1$ items	(iii) Cluttered grasp	(iv) Multiple grasp
0, empty grasps	(v) Empty single	(vi) Empty multiple
Other	(vii) Lost items (discarded out of workspace)	

B. Strategy Comparison Results

The arrangement results of the three singulation strategies are shown in Fig. 8. Corresponding quantitative evaluation is provided in Table II. Example videos from the experiments are available in the paper's accompanying video and at the project webpage: `http://sensor.cs.washington.edu/PileSorting.html`.

All three strategies were able to count and arrange at least 70% of the items in the original pile scene. The count accuracy increased with the addition of pushing actions and the further addition of the perception module. The grasping only strategy resulted in a low grasp attempt efficiency due to several empty grasps or multiple-item grasps on non-singulated spatial units. The failed grasp attempts did, however, provide enough physical perturbation to eventually singulate many items from the piles.

The strategy of fixed pushing and grasping, without the perception and decision modules of adaptive interaction, had decreased grasp errors due to the additional perturbation from the pushing actions. However, 15% of the singulation successes (5 of 33) were due to chance lucky grasps on targets that were in fact multiple-item piles. In addition, there were grasp errors from 2 multiple-item grasps and 12 empty grasps (including 4 lost items) out of the total of 52 grasp attempts.

In contrast, the fewest grasp errors occurred with the adaptive singulation strategy that included the perception of the pushing action and cumulative pile classification. The number of grasp attempts equaled the number of pile items, and overall for 46 attempts/items, with only 1 empty grasp and 1 lost item error (occurring in the same action).

The fourth test case (d) involving small toiletry items resulted in lost items for all three strategies. This was due to the small size of certain items (e.g. miniature dental floss

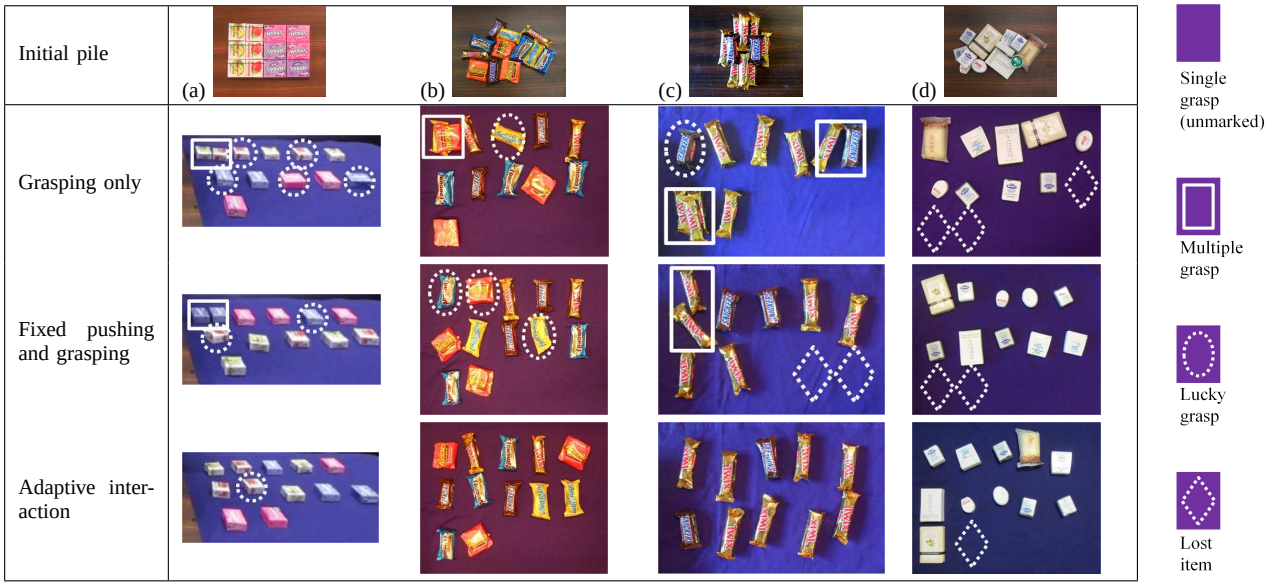


Fig. 8. Final singulation output for arrangement task.

TABLE II
EXPERIMENTAL RESULTS FROM COMPARISON OF SINGULATION STRATEGIES. SEE SECTION VI AND FIG. 8

The italicized column for fixed interaction strategy case (b) corresponds to the follow-up experiment in Section VI-C.

		Grasping only				**Fixed pushing and grasping**					**Adaptive interaction**			
	Initial pile (see Fig. 8)	a	b	c	d	a	b	c	d		a	b	c	d
Item counts	Actual number in pile	12	12	10	12	12	12	*12*	10	12	12	12	10	12
	Items counted by robot	11	11	7	9	11	12	*9*	7	10	12	12	10	11
	Count accuracy	.92	.92	.70	.75	.92	1.00	*.83*	.70	.83	1.00	1.00	1.00	.92
Action counts	Grasp attempts	26	15	13	19	11	15	*12*	12	14	12	12	10	12
	Perturbation pushes	–	–	–	–	22	27	*60*	19	26	61	57	91	67
Grasp attempt outcomes (Table I, Fig. 8)	Success (i) Single grasp	5	9	4	7	8	9	*7*	6	10	11	12	10	11
	(ii) Lucky grasp	5	1	1	2	2	3	*1*	-	-	1	-	-	-
	Errors (iv) Multiple grasp	1	1	2	-	1	-	*2*	1	-	-	-	-	-
	(v) Empty single	-	-	-	4	-	-	-	1	1	-	-	-	1
	(vi) Empty multiple	15	4	6	6	-	3	*3*	4	3	-	-	-	-
	(vii) Lost items	-	-	-	3	-	-	*1*	2	2	-	-	-	1
Singulation rate	Single grasps/Attempts	.19	.60	.31	.37	.73	.60	*.58*	.50	.71	.92	1.00	1.00	.92
	(Single+Lucky)/Attempts	.38	.67	.38	.47	.91	.80	*.67*	.50	.71	1.00	1.00	1.00	.92
Action efficiency	Grasp attempts/Counted	2.4	1.4	1.9	2.1	1.0	1.3	*1.3*	1.7	1.4	1.0	1.0	1.0	1.1
	Pushes/Counted	–	–	–	–	2.0	2.3	*6.6*	2.7	2.6	5.1	4.8	9.1	6.1
	Time[minutes]/Counted	1.4	1.1	1.0	1.4	1.5	1.6	*2.6*	2.1	2.2	2.5	2.3	3.6	2.9

dispensers) that became discarded due to perturbation outside the observable workspace or failure to detect grasps.

C. Matched pushes for fixed interaction strategy

In the previous results (Table II), interactive singulation with perceptual accumulation used more pushes per item than the fixed pushing and grasping strategy. A follow-up experiment on the second test case (b) repeated the fixed interaction strategy with five instead of two pushes before every grasp attempt, to approximately match the average 4.8 pushes per item of the adaptive singulation strategy.

Due to the grasp errors, the modified fixed strategy resulted in an average of 6.6 pushes per item (see highlighted column in Table II). Even with the increased number of pushes, the fixed strategy still resulted in more grasp errors than the adaptive strategy, and also more than the fixed strategy trial with two pushes per grasp. This example illustrates that the perception and decision modules allow for the adaptation of the manipulation actions to a pile's singulation state before grasping. With unpredictable item compositions, a pile may require more pushes than a set number before an item is singulated.

D. Alternative shape-based perception module

For objects with little texture, finding candidate rigid transforms by the approach in Section V-C may fail due to lack of visual features. The framework can accommodate an alternative perception module to avoid this limitation. We implement a shape-based registration method which first computes and then aligns the centroid and principal axes of the before and after state point clouds. Perturbations from this reference alignment are generated by applying small random rotations. Evaluation of these shape-based transform candidates allows the framework to singulate difficult piles,

(a) Initial pile, top and side views (b) Singulated items

Fig. 9. For items without strong features, shape-based matching generates candidate rigid transforms. (a) Initial pile with several completely green items. (b) Singulated green items after interaction.

such as the group of all-green objects in Fig. 9.

VII. Discussion

This work presents a framework for representing interaction strategies with piles of cluttered objects. We have developed a specific set of strategies for singulation in a item arrangement task, including a perceptual module for accumulation of singulation evidence. Experimental evaluation demonstrated that the perceptual module reduces the grasp attempt errors due to the conservative estimation of the target state as a singulated or non-singulated spatial unit before grasping.

A limitation of the presented singulation strategy with perceptual accumulation is the efficiency of the pushes in introducing large enough motion to split the target state topology into multiple spatial units. In this work, small scale pushes of about 1-4 cm motions depended on the distance between the target footprint centroid and the boundary point in the selected push direction. The small scale pushes were intentional in order to retain similar view points of the target to facilitate the sparse feature matching. Even with the limited scale of pushes, there were several instances of rotated views of the target that resulted in low percentage matches due to the change in partial surface views. Thus a large number of perturbations were necessary to eventually accumulate evidence of singulation that exceeded the threshold.

One direction for further investigation is the evaluation of the sensitivity of the performance to changes in parameters such as the number of pushes before considering grasp attempts, the scale of the pushing actions, and the matching criteria for sparse and dense points. Our selected parameter values were based on the performance of the initial training data sequences on different objects from the test conditions. In general these parameters may be difficult to tune for unknown objects without modeling prior expectations of the range of object sizes, weights, and surface properties to predict the resulting motion of the perturbations.

Another refinement to the method is to adaptively change the scale or direction of the push based on the action history and cumulative likelihood ratio. In our method, the planned pushing actions serve both actions without differentiation of the action type. A more complex method may include separate push planners that have different purposes of either validating a target that is already likely singulated or introducing a large perturbation to split a target that has low evidence of being a single object. Further considerations including recording the direction of the push as part of the action history in order to select new directions for perturbation. In our evaluation, there were a few cases where pushes were repeated along similar directions and failed to separate the objects. However, in most cases, the planned pushes changed direction due to the constraints from surrounding clutter.

Overall, we believe that the proposed interaction framework encompasses a large set of interesting pile interaction strategies relevant to multiple manipulation tasks. This work introduces the framework and provides an evaluation of strategies for singulation in an arrangement task. This framework may be used to describe, evaluate, and learn alternative strategies that allow a robot to use integrated perception and manipulation to interact with cluttered environments.

Acknowledgment

The authors thank the Robots & State Estimation Lab and the Sensor Systems Lab for discussion .

References

[1] R. B. Kelley, H. A. S. Martins, J. R. Birk, and J.-D. Dessimoz, "Three vision algorithms for acquiring workpieces from bins," vol. 71, no. 7, pp. 803–820, 1983.

[2] J.-D. Dessimoz, J. R. Birk, R. B. Kelley, H. A. S. Martins, and C. Lin, "Matched filters for bin picking," *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, no. 6, pp. 686–697, 1984.

[3] E. Klingbeil, D. Drao, B. Carpenter, V. Ganapathi, O. Khatib, and A. Y. Ng., "Grasping with application to an autonomous checkout robot," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2011.

[4] L. Jacobson and H. Wechsler, "Invariant image representation: A path toward solving the bin-picking problem," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, vol. 1, 1984, pp. 190–199.

[5] B. K. P. Horn and K. Ikeuchi, "The mechanical manipulation of randomly oriented parts," *Scientific American*, vol. 251, no. 2, pp. 100–111, Aug. 1984.

[6] H. Yang and A. Kak, "Determination of the identity, position and orientation of the topmost object in a pile: Some further experiments," in $IEEE_ICRA$, vol. 3, 1986, pp. 293–298.

[7] J. Sinapov, T. Bergquist, C. Schenck, U. Ohiri, S. Griffith, and A. Stoytchev, "Interactive object recognition using proprioceptive and auditory feedback," *IJRR*, 2011.

[8] P. Fitzpatrick, "First contact: an active vision approach to segmentation," in *Proc. IEEE/RSJ Conf. Intelligent Robots and Systems (IROS)*, vol. 3, 2003, pp. 2161–2166.

[9] J. Kenney, T. Buckley, and O. Brock, "Interactive segmentation for manipulation in unstructured environments," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2009, pp. 1377–1382.

[10] D. Katz and O. Brock, "Manipulating articulated objects with interactive perception," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2008, pp. 272–277.

[11] ——, "Interactive segmentation of articulated objects in 3d," in *Workshop on Mobile Manipulation: Integrating Perception and Manipulation, IEEE Int. Conf. Robotics and Automation (ICRA)*, 2011.

[12] W. H. Li and L. Kleeman, "Segmentation and modeling of visually symmetric objects by robot actions," *IJRR*, 2011.

[13] M. Krainin, P. Henry, X. Ren, and D. Fox, "Manipulator and object tracking for in-hand 3d object modeling," *IJRR*, 2011.

[14] S. Kriegel, T. Bodenmuller, M. Suppa, and G. Hirzinger, "A surface-based next-best-view approach for automated 3d model completion of unknown objects," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2011, pp. 4869–4874.

[15] M. Kopicki, S. Zurek, R. Stolkin, T. Morwald, and J. Wyatt, "Learning to predict how rigid objects behave under simple manipulation," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2011, pp. 5722–5729.

[16] M. Dogar and S. Srinivasa, "A framework for push-grasping in clutter," in *Robotics: Science and Systems VII*, 2011.

[17] D. Kappler, L. Y. Chang, N. S. Pollard, T. Asfour, and R. Dillmann, "Templates for pre-grasp sliding interactions," *Robotics and Autonomous Systems*, vol. 60, no. 3, pp. 411 – 423, 2012, Autonomous Grasping.