The Teenager's Problem: Efficient Garment Decluttering as Probabilistic Set Cover

Aviv Adler*[1], Ayah Ahmad*[1], Yulei Qiu*[2], Shengyin Wang[2], Wisdom C. Agboh[1,2], Edith Llontop[1], Tianshuang Qiu[1], Jeffrey Ichnowski[3], Thomas Kollar[4], Richard Cheng[4], Mehmet Dogar[2], and Ken Goldberg[1]

[1] University of California, Berkeley, USA
[2] University of Leeds, UK
[3] Carnegie Mellon University, USA
[4] Toyota Research Institute, USA

Abstract. This paper addresses the "Teenager's Problem": efficiently removing scattered garments from a planar surface into a basket. As grasping and transporting individual garments is highly inefficient, we propose policies to select grasp locations for multiple garments using an overhead camera. Our core approach is *segment-based*, which uses segmentation on the overhead RGB image of the scene. We propose a *Probabilistic Set Cover* formulation of the problem, aiming to minimize the number of grasps that clear all garments off the surface. Grasp efficiency is measured by *Objects per Transport* (OpT), which denotes the average number of objects removed per trip to the laundry basket. Additionally, we explore several *depth-based* methods, which use overhead depth data to find efficient grasps. Experiments suggest that our segment-based method increases OpT by 50% over a random baseline, whereas combined *hybrid* methods yield improvements of 33%. Finally, a method employing *consolidation* (with segmentation) is considered, which locally moves the garments on the work surface to increase OpT, when the distance to the basket is much greater than the local motion distances. This yields an improvement of 81% over the baseline.

Keywords: Multi-object grasping · Deformable objects

1 Introduction

We introduce the "Teenager's Problem": removing a large number of scattered garments from a surface (e.g. the floor of a teenager's room, or a work surface) in the shortest time. This problem has applications in hotels, retail dressing rooms, garment manufacturing, and other domains where heaps of garments must be manipulated efficiently. This general problem can also be applied to cases where other deformable objects must be removed, for instance in clearing litter or garden debris.

We first formalize the Teenager's Problem, then consider several methods to solve it. Consider Fig. 1 with multiple garments on a work surface. Given an overhead RGB or RGBD image, what robot pick-and-place actions would minimize the total time to move all of the garments to a laundry basket? Removing garments one-by-one would be

* Equal Contribution

Fig. 1: An instance of the *Teenager's Problem* in the experimental setup; the work surface is white and the basket is beige, a UR5 industrial robot with a Robotiq parallel-jaw gripper is used, with overhead cameras above. The scale automatically records weight data as experiments are run.

inefficient. Thus we propose that the robot should use the deformable nature of garments and grasp multiple garments at once.

Given a scene like the one in Fig. 1, one approach is to identify individual garments and optimize grasps to pick as many of these garments as possible. This motivates *segment-based* methods, i.e., methods that use the RGB image to segment the individual garments. We use the Segment Anything Model (SAM) [16] to approximately model individual garments. Then, given the set of garments (segments) and a candidate grasp point, our method predicts the probabilities that each garment will be picked by that grasp. For example, a grasp candidate that is close to the intersection of a set of garments would have a positive probability of picking each of those garments, while it would have near-zero probability of picking garments that are farther away. Given many such candidate grasps, and their predicted probabilities of picking each garment, we formulate the task of minimizing the number of grasps to pick all garments as a *Probabilistic Set Cover Problem* [3]. We then express this as a Mixed Integer Linear Program and solve it.

While the segment-based method is able to identify grasps that would pick multiple *visible* garments simultaneously, it can also miss some multi-garment grasps. In particular, since only the top surface of the garments is visible to the camera, garments that are fully occluded are ignored by the segment-based method.

A second approach to solving the Teenager's Problem is to treat the whole scene as a homogeneous volume to be removed. This motivates *depth-based* methods, i.e., methods that use the depth image to infer grasp points that would then remove as much volume as possible. We consider two depth-based methods in this paper. The first method uses *height* and grasps at the highest garment point in the scene. The second method estimates a *volume*, by integrating the depth data within a grasp radius, and grasps at the point in the scene that gives the largest estimated volume.

The depth-based methods can identify large heaps and grasp multiple garments at a time, even when some of these garments are completely occluded by others and not individually visible to the camera. However, the depth-based methods also miss some good grasps. Particularly, since they do not detect individual garment positions and boundaries, they miss grasp points that may pick multiple garments simultaneously but are at a lower height or volume, e.g., points where multiple garment boundaries meet.

We also consider a *hybrid* method to combine the complementary strengths of the segment- and depth-based approaches. The hybrid method chooses which method depending on the maximum height available.

Finally, we consider methods that make use of *consolidation* actions that move within the workspace to gather the garments into larger heaps, before removing the heap. This improves the efficiency of grasps to transport the objects to the basket at the cost of the time used in consolidation, which can improve efficiency in cases where the basket is located far from the work surface.

Experiments suggest that, the segment-based method significantly reduces robot trips to the basket by 50%. The *hybrid* methods yield improvements of 33%. Finally, Objects per Transport (OpT) can be further increased by 81% by using *consolidation* actions within the workspace to set up highly efficient transport actions.

We make the following contributions:

- A formalization of the Teenager's Problem as a Probabilistic Set Cover Problem.
- Five methods (two depth-based, one segment-based, and two hybrid) to generate effective multi-garment grasps.
- A method that uses heap consolidation along with the segment-based grasp generation method to efficiently solve the Teenager's Problem.
- Physical experiments and data from grasping 1750 garments, that compare the performance of the six methods against a random baseline.

2 Related work

Our work is related to two lines of work: *manipulation of deformable objects* and *multi-object manipulation*.

2.1 Manipulation of Deformable Objects

Prior work on deformable object manipulation includes folding [4, 12], flinging [5, 11, 38], fabric smoothing [10, 26, 28], bed-making [27], untangling ropes [29], and singulating clothes from a heap [30, 35]. Several works aimed to detect specific features,

such as the corners and edges of fabrics, and to identify optimal grasp points [8, 19, 21]. Other techniques employ deep learning to identify successful grasps [7, 18]. Some studies have focused on determining optimal grasp points by considering not only the depth of the cloth but also targeting wrinkles as highly graspable regions [22–24, 34]. These prior works focus on manipulating a single deformable object at a time or singulating a deformable object from among others. Our work, on the other hand, concentrates on grasping multiple garments simultaneously. There also exist specialized grippers developed for garment grasping [9, 17]. While the strategies we propose can also be used with such specialized grippers, we experiment here with a general-purpose gripper, which increases the applicability of the results.

2.2 Multi-object Manipulation

Multi-object grasping can improve decluttering efficiency [1]. It has been studied, with analytic methods [36, 37], learning-based methods [2, 6], and with special gripper designs [20]. The focus, however, has remained on rigid objects. Instead, we consider the problem of grasping multiple deformable objects at a time, and using such grasps to efficiently clear a surface.

Multi-object manipulation scenarios can encompass cluttered [15] environments, which can include both deformable and rigid objects [33], however, their goal is to singulate the objects to grasp them individually. Prior work on manipulating multiple rigid objects used methods such as pushing, stacking, and destacking [1, 13, 25]. In cluttered scenes with multiple rigid objects, one method for determining how, or where, to grasp is by using image segmentation [31], detecting and isolating individual objects in the scene. We also use a segmentation approach but for deformable objects.

3 The Teenager's Problem

We formulate the *Teenager's Problem* as follows: deformable objects rest on a planar work surface. Given a fixed target basket, the goal is to transfer all the garments efficiently from the workspace to the basket with a minimum number of grasps (which naturally maximizes OpT).

3.1 Problem Statement

In the Teenager's Problem, there are m differently colored garments on a surface. We assume we are given n grasp candidates, where each grasp is a tuple $(x, y, \theta) \in \mathbb{R}^2 \times [-\pi/2, \pi/2]$ representing a top-down grasp, with θ representing the angle of the parallel-jaw gripper with respect to the workspace axes. (We explain how we generate such grasp candidates in Sec. 4.1.) We denote a *grasp plan* as $\boldsymbol{x} \in \{0, 1\}^n$, with x_i indicating whether grasp candidate i is in the grasp plan; i.e. $x_i = 1$ if it is in the plan and $x_i = 0$ otherwise.

For a grasp i and garment j, we denote by $p_{i,j}$ the probability that grasp i successfully grasps garment j. (We describe how we estimate $p_{i,j}$ in Sec 4.1.) For each garment

j, we denote by q_j the minimum probability that we want to have of removing garment j from the surface.

Then, the objective is to find a grasp plan $\boldsymbol{x}$ that minimizes $\mathbf{1}^\top \boldsymbol{x}$ (i.e. the number of grasps in the plan), such that for all garments $j \in \{1, 2, \ldots, m\}$, thus

$$\mathbb{P}[\text{remove garment } j \mid \boldsymbol{x}] \geq q_j. \tag{1}$$

The Teenager's Problem can thus be seen as a probabilistic variant of the classic Set Cover problem, since each grasp corresponds to the (weighted) set of garments it could potentially grasp, and the goal is to find a set of grasps whose 'union' encompasses all the garments. This also means that the benefit of a grasp depends on the set of other grasps that will also be taken—even if a grasp is likely to get several garments, it may be useless if those garments were already likely removed by other grasps in the set.

3.2 Mixed Integer Linear Program

We propose to solve the above problem by converting it to a Mixed Integer Linear Program (MILP).

Eq. 1 is equivalent to

$$\mathbb{P}[\text{fail to remove garment } j \mid \boldsymbol{x}] \leq 1 - q_j. \tag{2}$$

For any two grasps i, i' that do *not* overlap (i.e., grasps that are sufficiently away from each other), we assume the success probabilities are independent random events. Therefore, in any grasp plan without overlapping grasps, the success or failure of any grasp in getting any garment can be treated as independent from any other. We call grasp plan without overlapping grasps *valid*. We will restrict the planner to valid plans in order to prevent the planner from choosing grasps which are too close together and would likely interfere with each other.

Now we consider a valid grasp plan $\boldsymbol{x} = (x_1, \ldots, x_n) \in \{0, 1\}^n$. Then the probability of failing to get garment j is

$$\mathbb{P}[\text{fail to remove garment } j \mid \boldsymbol{x}] = \mathbb{P}[\text{no } i \text{ s.t. } x_i = 1 \text{ picks } j] \tag{3}$$

$$= \prod_{i : x_i = 1} (1 - p_{i,j}) = \prod_i (1 - p_{i,j})^{x_i}. \tag{4}$$

Therefore we want $\boldsymbol{x}$ satisfying the constraints such that

$$\prod_i (1 - p_{i,j})^{x_i} \leq 1 - q_j \tag{5}$$

$$\iff \sum_i x_i \log(1 - p_{i,j}) \leq \log(1 - q_j). \tag{6}$$

We can assume that $p_{i,j} < 1$ since no grasp is 100% certain of success, and $q_j < 1$ since we cannot get any garment with 100% certainty either, hence we can assume all the values above are finite.

Thus, we define an MILP using matrix $\boldsymbol{A}$ where $A_{i,j} = \log(1 - p_{i,j})$ and $b_j = \log(1 - q_j)$. We want to minimize the number of grasps used, which gives the MILP:

$$\begin{aligned} &\text{minimize } \mathbf{1}^\top \boldsymbol{x} \text{ subject to} \\ &\qquad \boldsymbol{A}^\top \boldsymbol{x} \leq \boldsymbol{b} \\ &\qquad x_i + x_{i'} \leq 1 \text{ for all } i, i' \text{ which overlap} \\ &\qquad\qquad x_i \in \{0, 1\} \text{ for all } i\,, \end{aligned} \tag{7}$$

where the second constraint enforces that the grasp plans considered are valid.

In our implementation, we set $q_j = 0.7$ as the minimum probability of a garment being removed. We use the MILP solver [14] within the SciPy library [32].

3.3 Metrics

The primary metric for evaluating the methods is *Objects per Transport* (OpT), which denotes the average number of objects taken during each transport and measures the general effectiveness of the performed grasps. We use OpT, as opposed to Picks Per Hour (PPH), as OpT directly measures grasp quality and PPH depends heavily on implementation details (particularly concerning computation time) which reduces its reliability as a metric in this setting.

4 Teenager's Problem Methods

This study explores various strategies for efficiently grasping multiple garments concurrently, categorized broadly into two types: *segment-based* and *depth-based* approaches.

All the methods described below use a pre-processing of the RGB pixels to separate the background and the foreground, i.e. to determine the *garment points*, denoted $\mathcal{X}_g$. Since we assume the system knows the color and/or pattern of the background, this is achieved with color thresholding.

4.1 Segment-Based Methods

The segment-based approach divides the task into *cycles*: a segmentation is generated at the start of each cycle, along with candidate grasps. Using this segmentation and the grasp candidates, the MILP (Eq. 7) is solved, and the sequence of grasps is carried out based on that initial segmentation (however, an overhead RGB image is still taken between each grasp, as will be explained later). Each cycle relies on four subroutines: (1) Segmentation and cleanup; (2) prediction; (3) grasp selection; (4) execution. Once all the planned moves have been performed the next cycle begins. This cycle is repeated until the table is cleared. The steps of a cycle are detailed below.

Segmentation and cleanup The method uses Meta's Segment Anything Model (SAM) [16] with the *vit-b* weights, prompts the image as a whole, and generates segmentation masks for the image. However, the initial segmentation often contains multiple overlapping segments, gaps, and regions corresponding to the work surface. To address this, our

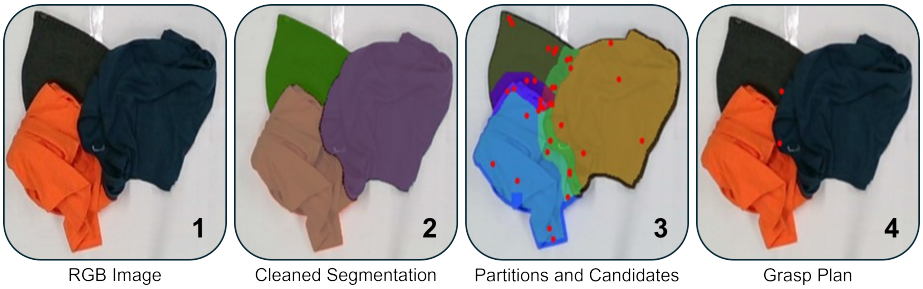

Fig. 2: An example of the segment-based grasp point selection algorithm (grasp orientations not shown). **From left to right:** *(1)* The original overhead RGB image. *(2)* The cleaned segmentation $\mathcal{M}$. *(3)* The partitions such that, each partition has the same 'nearby' set of segments; along with $k = 5$ grasp candidates sampled from each partition (red dots). *(4)* The two grasps (red dots) that constitute the grasp plan, found by solving the MILP.

method begins by applying color thresholding to remove the segments that represent the work surface. The remaining clothes and background regions are then converted into a binary cloth map. To effectively identify and fill/remove holes within the cloth map, we convolve this map twice using an all-ones kernel.

Let $\mathcal{M}$ denote the set of segments detected using this method. For that cycle, the number of garments is set to be the number of detected segments, i.e., $m = |\mathcal{M}|$, where each $M_j \in \mathcal{M}$ represents the segment belonging to garment j. Fig. 2-(2) shows an example output of this step.

Prediction The Teenager's Problem keeps the predictor function p general to accommodate a variety of different approaches, both analytic and learned, to estimating the effects of a grasp. In this work we use an analytic predictor which models the area under the gripper as an ellipse and the probability of a successful grasp as depending on the total area of the garment within the ellipse.

Specifically, given a grasp (x, y, θ), let $E(x, y, \theta)$ denote the ellipse centered at (x, y) whose major axis is oriented at angle θ with major axis length d_1 and minor axis length d_2. The axis lengths d_1 and d_2 are scaled to be the length and width of the parallel-jaw gripper. We estimate the probability of successfully grasping a garment j with grasp i, as

$$p_{i,j} = \frac{\text{area}(E(x, y, \theta) \cap M_j)}{\text{area}(E(x, y, \theta) \cap M_j) + b}$$

where (x, y, θ) represent the grasp i and $b > 0$ is a normalization constant. In our implementation, the area is measured in pixels and we use $b = 100$.

This predictor is intended to capture the following intuition: the gripper directly affects the area under it, and the more any segment falls in that area, the more likely it is to be removed by the grasp (but cannot have probability > 1 of being removed). Note

that this predictor can never be 100% certain that a given segment will be removed, which is realistic.

One property of this predictor is that, while it is important to choose grasps which get a large total area of the segments within the ellipse, having several different segments with nonzero area under the ellipse is generally preferable to having just one (even if the total area within the ellipse is the same), because the benefit of added area for one segment decreases as the area already captured increases. Thus, the best grasps will generally occur at or near the boundary between multiple segments.

Grasp selection With a clean segmentation and predictor, the method faces the task of determining which grasps to execute. We first generate a set of candidate grasp points and then use these candidates to solve the MILP (Eq. 7), to generate a grasp plan, $\boldsymbol{x}$.

The candidate grasp point generation method does the following (given a radius $r > 0$ in pixels, and segmentation $\mathcal{M}$):

- For each pixel $(x, y) \in \mathcal{X}_g$, determine the set $S(x, y) \subseteq \mathcal{M}$ of segments which are within distance r from (x, y).
- Partition $\mathcal{X}_g$ based on $S(x, y)$. In other words, two points $(x, y) \in \mathcal{X}_g$ and $(x', y') \in \mathcal{X}_g$ are considered in the same partition, if $S(x, y) = S(x', y')$.
- From each partition, randomly (uniform) sample k grasp points $\{(x_i, y_i)\}_{i=1}^{k}$. We used $k = 5$.

The rationale behind this procedure is to sample candidate grasp points from a variety of regions (partitions) with potential to grasp different garments. While a completely random sampling of $\mathcal{X}_g$ can miss important regions that have the potential to grasp multiple garments at a time, partitioning first, and then randomly sampling these partitions helps maintain a rich variety of candidate grasps. Fig. 2-(3) presents an example, where the partitions as well as the sampled grasp candidates are shown.

Then, for each point (x, y), we generate ℓ grasp candidates (x, y, θ), by enumerating a list of ℓ equally-spaced orientations in $[-\pi/2, \pi/2]$. For a balance between efficiency and thoroughness, we used $\ell = 6$.

Furthermore, if two grasps belong to the same partition, we treat them as *overlapping*, as discussed in Sec. 3.2 (second constraint of Eq. 7).

Fig. 2-(4) shows an example grasp plan (including only two grasps) as generated by the MILP solver.

Grasp execution The algorithm then attempts all grasps in the grasp plan $\boldsymbol{x}$ in sequence, in increasing order of distance to the basket; this is to prevent, as far as possible, dragging garments from disturbing the positions of the garments that remain (which may cause garments to fall off the work surface).

An RGB image is also captured after each transfer to the basket (when the arm is out of frame), although a new segmentation is *not* generated (until the next cycle). Instead, for each planned grasp remaining, the difference between its current state and the state at the beginning of the cycle (when the segmentation was generated) is estimated using the squared difference between the pixel values within a small square neighborhood around the grasp point; if the difference is too large, the grasp is deemed to be in a

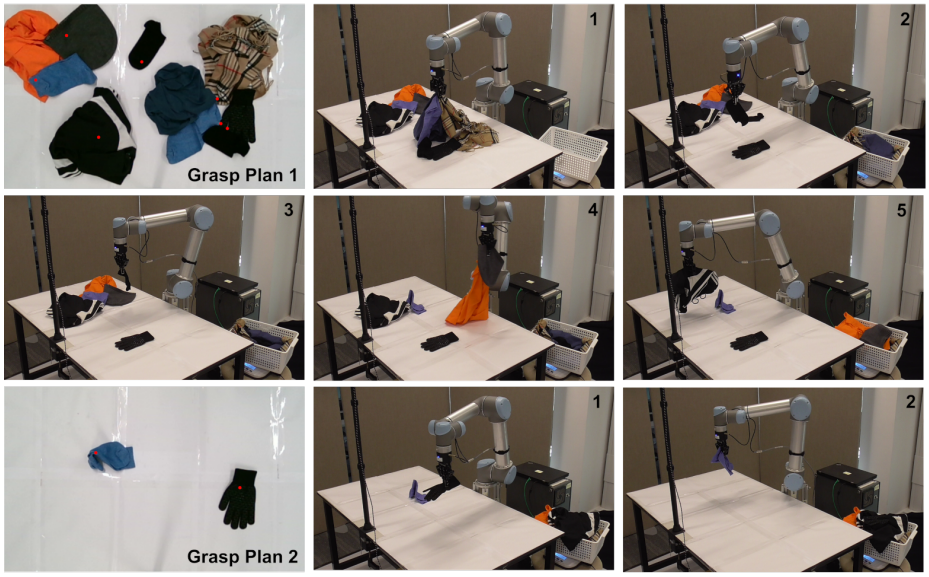

Fig. 3: An example execution using the segment-based method, where the table is cleared in seven grasps in total. Grasp Plan 1 is generated on the initial scene (red dots show planned grasp positions). Only five of these grasps are executed, before Grasp Plan 2 is generated with the remaining two grasps to clean the table.

different state from when it was planned and is not performed. This ensures that grasps are not performed unless the system knows that the local configuration of the garments is approximately the same as when it was planned.

This process is visualized in Fig. 3, where the system needs two cycles, and seven grasps, before clearing the table. Notice how the system starts from the grasp that is closest to the basket. However, since the first grasp changes the scene significantly in its neighborhood, the system does not execute the nearby two grasps. Similarly, after the grasp of the orange shirt and the hat (Grasp 5 in the figure), the grasp on the blue sock is not attempted. Instead, a new cycle begins requiring only two grasps.

4.2 Depth-Based Methods

Depth-based methods use the depth output of the RGBD overhead camera to select the next grasp. To solve the Teenager's Problem, these methods are used repetitively until the workspace is clear: at each step, we capture a new depth image of the scene, use one of the methods below to generate a new grasp, and then execute that grasp. We examine two variations:

Height This variation selects the highest point in the scene and chooses the orientation to be that of the major axis of a local principal component analysis (PCA) around the grasp point.

Volume This variation considers the total volume of garments in a disc of radius R around a candidate grasp point (x, y), which is estimated by summing the heights of all the pixels within that radius, and then selects the point with the largest total volume. As in the Height method, orientation is selected using a local PCA.

4.3 Hybrid methods

One experimental observation was that depth-based methods and segment-based methods have different strengths—in particular, depth-based methods excel at picking occluded garments (which generally result in taller piles) while segment-based methods excel at simultaneously picking adjacent garments. This motivates considering *hybrid* methods which make use of both depth data and segmentation data.

To take advantage of how these methods complement each other, hybrid methods do the following: given a height threshold (we use 0.1 m), if the tallest pile is taller than the threshold, execute a (single) grasp as given by the depth-based method; if all piles are below the threshold, execute one cycle of the segment-based method.

4.4 Segment-based method with consolidation

Another avenue to improving OpT is to first consolidate the garments into large piles for transport to the basket; this can improve overall efficiency in cases where the basket is located at some distance from the work surface, making transports costly relative to manipulations within the workspace. An efficient primitive for consolidation is the *grasp sequence* where each pick-and-place movement picks up where the last one placed; this both saves on robot movement time (no travel distance to the next pick point) and, ideally, allows the robot to accumulate more garments as it goes before depositing them in the basket.

An extension of the segment-based method above to include consolidations follows:

1. generate the grasp plan as in the no-consolidation segment-based method (i.e., as described in Sec. 4.1), and sort them in decreasing distance to the basket (i.e., the first grasp is the furthest one to the basket);
2. for the next grasp in the plan, i, estimate the *expected area* of grasped garments, using the formula $\sum_j p_{i,j} M_j$;
3. if the *expected area* for this next grasp will *not* make the *total expected area* exceed a pre-determined *grasp area threshold*, then execute the next grasp, and add the *expected area* to the *total expected area*. If the expected area for the next grasp *will* make the *total expected area* exceed the threshold, then go back to step 2 above;
4. if no such grasp exists, transport the currently-held garments to the basket.

Going from the furthest grasp point to the closest follows the intuition that a method using consolidation should consolidate towards the basket since this will always shorten the distance between the grasped garments and the basket, even if some are dropped along the way—and this will tend to compress them into a smaller space, facilitating later multi-object grasps.

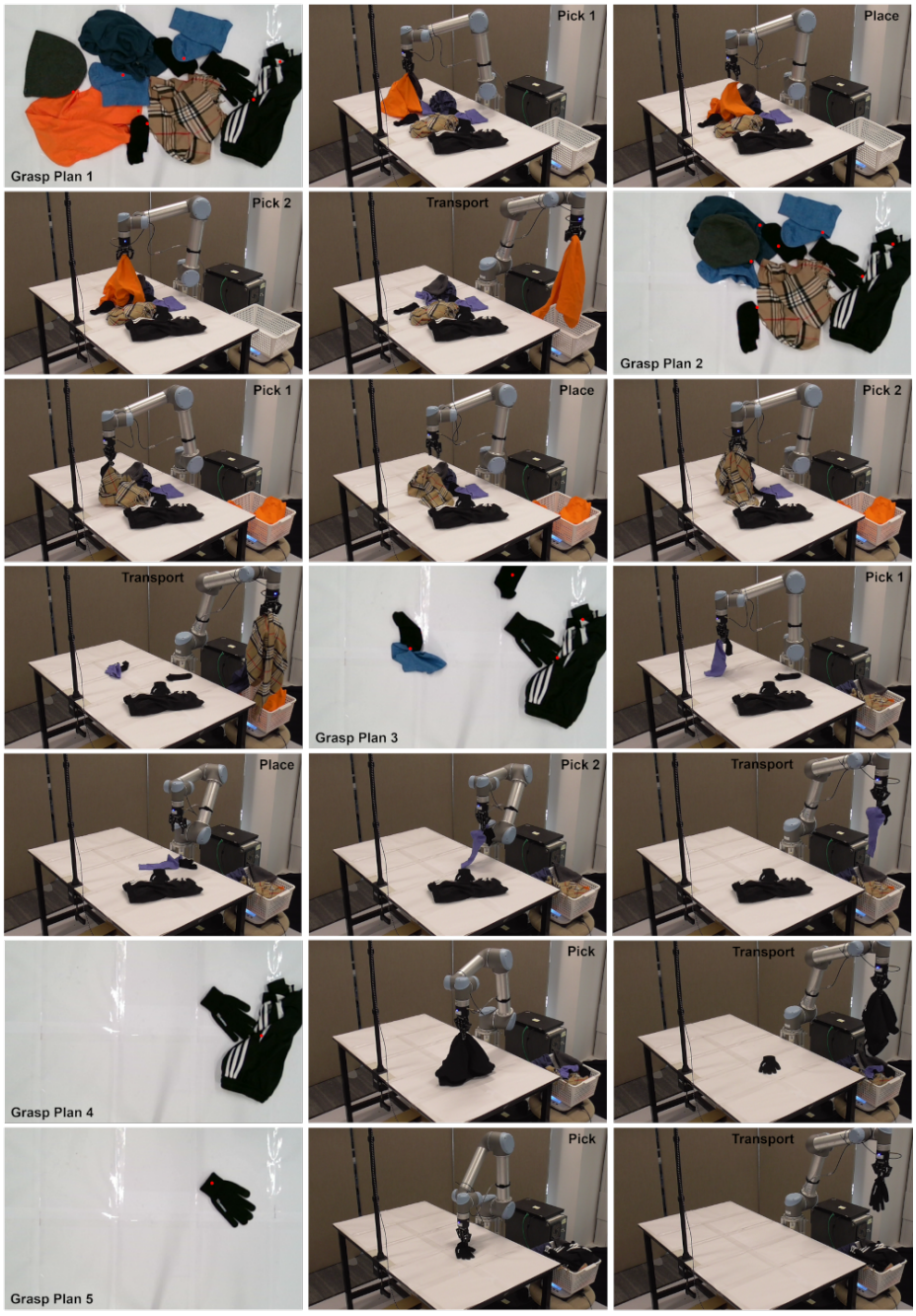

Fig. 4: An example execution using the consolidation method. The robot performs sequence of pick-and-place actions on the work surface to consolidate garments. It takes only five transport actions to clear all ten garments off the table.

The grasp area threshold corresponds to the intuition that the gripper has a limit to the amount of fabric it can hold and thus trying to accumulate more than that limit in one grasp is counterproductive.

We show an example execution using the consolidation method in Fig. 4. The trips to the basket are marked as "Transport." The consolidation actions result in multiple garments being picked simultaneously, and significantly reduce the number of transport actions.

4.5 Baseline

Finally, as a baseline, we use the *random* method, which uniformly randomly selects a garment point $(x, y) \in \mathcal{X}_g$ with a uniformly random orientation $\theta \in [-\pi/2, \pi/2]$, accounting for the gripper's symmetry.

Table 1: OpT (Objects per Transport) by grasping method

	Depth-Based Methods			Hybrid Methods		
Random	Volume	Height	Segment	Volume	Height	Consolidation
1.07 ± 0.07	1.27 ± 0.11	1.22 ± 0.07	1.55 ± 0.11	1.34 ± 0.10	1.45 ± 0.14	$\mathbf{1.94 \pm 0.15}$

5 Experiments

We tested all algorithms on the test set of 10 garments (see Fig. 5) with 25 sample runs. Each sample run begins with a randomized scene containing all 10 test set garments, and ends when the workspace is cleared of garments.

5.1 Data collection pipeline

To run the experiments, we used a semi-autonomous data collection pipeline, in which experimental scene reset, randomization and data recording are done automatically, with the experimenter only needing to correct problems when they arise (for instance, if a garment falls off the work surface, the experimenter must return it for the next sample). The system uses the recorded weight data to automatically notify the experimenter when such a problem occurs, to minimize the amount of human attention necessary for data collection.

The scene is automatically reset in the following way:

1. The robot grasps the basket and empties it over the work surface to deposit the garments, then places the basket back to its original position.
2. The robot executes a sequence of random pick-and-place actions on the surface to shuffle the garments. Each pick position is randomly (uniform) sampled from the foreground cloth points $\mathcal{X}_g$, and each place position is sampled from the full rectangular workspace (a rectangular region of 1 m $\times$ 0.6 m). In our experiments, 10 such moves were performed for every scene reset

Then the experiment is performed with the selected method, recording at each step the overhead RGBD output, grasp location and orientation, and weight of the garments in the basket. The experiment is paused for 3 seconds after every transport to allow the scale's output to settle. Full dataset is available at `https://sites.google.com/view/TeenSP2024`.

5.2 Metrics evaluated

Our metric for evaluating the methods is *Objects per Transport* (OpT), which denotes the average number of objects taken during each transport and measures the general effectiveness of the performed grasps. We use OpT, as opposed to Picks Per Hour (PPH), as OpT directly measures grasp quality and PPH depends heavily on implementation details (particularly concerning computation time) which reduces its reliability as a metric in this setting. Additionally, maximizing OpT is a good strategy particularly when the target basket is far and consolidation is important.

For each algorithm tested, OpT was evaluated on all 20 sample runs, which were then averaged to yield the final result and 95% confidence bounds.

6 Results

The results of the experiments are given in Table 1, showing that both depth-based and segment-based methods yield clear improvements over the random baseline. In particular, the segment-based method provides additional 50% OpT, achieving 1.57 objects per transport to the basket. Although hybrid methods achieve better OpT compared to depth-based methods, they cannot beat the segment-based approach. Finally, at the cost of both computational overhead and additional physical actions, the segmentation with

Fig. 5: The test set of 10 garments, representing a variety of different sizes, weights, textures, colors, patterns, flexibility, and garment classes. Some garments have similar colors to present a challenge for segmentation. We also use long garments, such as the scarf, that present a challenge for grasping.

consolidation method drastically improves OpT, achieving 1.91 objects per transport, 81% over the baseline.

Note that while grasp quality is the focus and OpT is the most meaningful metric for this work, another important metric for pick efficiency is PPH. The depth-based methods, which do not perform significant computations to find grasps, improve PPH from 125 for the baseline to 147 and 145 for max-volume and max-height grasps, respectively. While the segment-based method requires more computation, it still registers a higher PPH of 174. However, compared to the segment-based method, the consolidation approach registers a slight decrease in PPH (to 160), even if it has a higher OpT score. This is due to the extra actions required by the consolidation method, and shows that its efficacy relative to the segment-based method depends on the basket being further from the workspace.

6.1 Comparison of Methods

What are the inherent advantages and disadvantages of the two approaches outlined above?

- Depth-based methods require both RGB and depth images to compute grasps. In contrast, segment-based methods only rely on RGB images, rendering them suitable for systems lacking depth cameras.
- Segment-based methods often require more computational resources, as they involve neural network-driven segmentation and subsequent cleanup. To ensure efficient computation without compromising speed, it may be necessary to deploy GPUs or opt for segmentation methods optimized for CPU processing.
- A notable challenge faced by segment-based methods is their limited ability to detect grasps that remove occluded garments. In contrast, depth-based methods more often grasp over occluded garments due to their utilization of depth information, which provides an enhanced perception of garment depth.
- Conversely, segment-based methods explicitly choose grasps to simultaneously capture garments situated closely together, whereas depth-based methods cannot determine which points are in proximity to multiple visible garments.

6.2 How our system scales with the number of garments

We performed additional experiments to measure how long it takes (i) for the segment-based method to identify garments, and (ii) to solve the resulting MILP, as we increased the number of garments. We present the results in Table 2. In the experiment, we incrementally added five items at a time to the scene, ensuring they remain non-overlapping to only measure the effect of number of garments on the performance. Still, the detected number of segments did not exactly match the number of garments, which are also shown in the table. While the segmentation time was not affected by the number of garments, the MILP solution time showed a faster than linear increase.

We also performed an experiment where we maintained a constant number of garments (35) but increased the number of overlapping groups, where all objects within a group were in contact, to test how the system scaled with the degree of overlap of the

Table 2: Segmentation and MILP solution time with increasing number of garments

# garments	# segments	Seg. time (s)	MILP soln. time (s)
5	4	19.66	1.43
10	9	18.86	3.88
15	14	18.74	6.73
20	19	18.73	9.95
25	24	18.72	13.42
30	29	18.81	16.86
35	35	18.70	22.27

garments in the scene. The results did not show an obvious relationship between the degree of overlap and segmentation or MILP solution time.

7 Conclusion

In this work, we tackle the challenging problem of robotic garment decluttering, by formalizing the Teenager's Problem and developing both depth- and segment-based methods to solve it. We use recent advances in image segmentation [16] to explore an approach that uses it to distinguish garments in the image and find grasps that are likely to capture as many as possible.

7.1 Dirty Laundry and Future Work

However, this work has certain limitations and leaves a number of areas open for improvement:

- All the proposed methods rely on accurately separating the garments (the foreground) from the table surface (the background) using the RGB image, which is done here via color thresholding. While this was reliable in our experimental setup, a different system may be needed if any garments are the same color with the background.
- While all the methods considered here grasp at a fixed height above the work surface with a perfectly vertical gripper, the most efficient grasp may not share those characteristics. Additional improvements might be obtained by optimizing the grasp height or angle.
- While the 81% OpT increase from the segmentation with consolidation method is large, an average of roughly 2.1 rearrangement actions were performed to consolidate prior to each transport saved over the baseline. This will increase efficiency in cases where transports are relatively costly, e.g. when the target basket is far from the workspace.

In future work, we can explore extensions such as sorting of clothes (for instance, separating clothing by type or color). Additionally, although we present only the analytic grasp predictor, the segment-based method described in Section 4.1 is compatible with any grasp predictor, which could be improved using self-supervised data collection.

Acknowledgments

M. Dogar was supported by the UK Engineering and Physical Sciences Research Council [EP/V052659/1].

References

[1] W. C. Agboh, J. Ichnowski, K. Goldberg, and M. R. Dogar, "Multi-object grasping in the plane," in *International Symposium on Robotics Research (ISRR)*, 2022.

[2] W. C. Agboh, S. Sharma, K. Srinivas, M. Parulekar, G. Datta, T. Qiu, J. Ichnowski, E. Solowjow, M. Dogar, and K. Goldberg, *Learning to efficiently plan robust frictional multi-object grasps*, 2022.

[3] S. Ahmed and D. J. Papageorgiou, "Probabilistic set covering with correlations," *Operations Research*, vol. 61, no. 2, pp. 438–452, 2013.

[4] Y. Avigal, L. Berscheid, T. Asfour, T. Kröger, and K. Goldberg, "Speedfolding: Learning efficient bimanual folding of garments," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 1–8.

[5] L. Y. Chen, H. Huang, E. Novoseller, D. Seita, J. Ichnowski, M. Laskey, R. Cheng, T. Kollar, and K. Goldberg, "Efficiently learning single-arm fling motions to smooth garments," in *The International Symposium of Robotics Research*, Springer, 2022, pp. 36–51.

[6] S. Chen and Y. Zhu, "Grasping objects in clutter with deep learning and grasping affordance prediction," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[7] F.-J. Chu, R. Xu, and P. A. Vela, "Real-world multi-object, multi-grasp detection," in *IEEE Robotics and Automation Letters*, 2018.

[8] Y. Deng, C. Xia, X. Wang, and L. Chen, "Graph-transporter: A graph-based learning method for goal-conditioned deformable object rearranging task," in *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2022, pp. 1910–1916.

[9] M. Dragusanu, S. Marullo, M. Malvezzi, G. M. Achilli, M. C. Valigi, D. Prattichizzo, and G. Salvietti, "The dressgripper: A collaborative gripper with electromagnetic fingertips for dressing assistance," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7479–7486, 2022.

[10] A. Ganapathi, P. Sundaresan, B. Thananjeyan, A. Balakrishna, D. Seita, J. Grannen, M. Hwang, R. Hoque, J. E. Gonzalez, N. Jamali, K. Yamane, S. Iba, and K. Goldberg, "Learning dense visual correspondences in simulation to smooth and fold real fabrics," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 11 515–11 522.

[11] H. Ha and S. Song, "Flingbot: The unreasonable effectiveness of dynamic manipulation for cloth unfolding," in *Conference on Robot Learning*, PMLR, 2022, pp. 24–33.

[12] R. Hoque, K. Shivakumar, S. Aeron, G. Deza, A. Ganapathi, A. Wong, J. Lee, A. Zeng, V. Vanhoucke, and K. Goldberg, "Learning to fold real garments with one arm: A case study in cloud-based robotics research," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 251–257.

[13] H. Huang, L. Fu, M. Danielczuk, C. M. Kim, Z. Tam, J. Ichnowski, A. Angelova, B. Ichter, and K. Goldberg, "Mechanical search on shelves with efficient stacking and destacking of objects," in *Int. S. Robotics Research (ISRR)*, 2022, pp. 1–16.

[14] Q. Huangfu and J. A. J. Hall, "Parallelizing the dual revised simplex method," *Mathematical Programming Computation*, vol. 10, no. 1, pp. 119–142, 2018.

[15] H. Kasaei, M. Kasaei, G. Tziafas, S. Luo, and R. Sasso, "Simultaneous multi-view object recognition and grasping in open-ended domains," 2021, pp. 8295–8300.

[16] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W. Lo, P. Dollár, and R. Girshick, "Segment anything," *arXiv:2304.02643*, 2023.

[17] L. Le, M. Zoppi, M. Jilich, R. Camoriano, D. Zlatanov, and R. Molfino, "Development and analysis of a new specialized gripper mechanism for garment handling," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers, vol. 55942, 2013, V06BT07A013.

[18] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," in *Robotics: Science and Systems (RSS)*, 2013.

[19] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei, and P. Abbeel, "Cloth grasp point detection basedon multiple-view geometric cues with application to robotic towel folding," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2010.

[20] P. V. Nguyen, P. N. Nguyen, T. Nguyen, and T. L. Le, "Hybrid robot hand for stably manipulating one group objects," *Archive of Mechanical Engineering*, vol. vol. 69, no. No 3, pp. 375–391, 2022.

[21] J. Qian, T. Weng, L. Zhang, B. Okorn, and D. Held, "Cloth region segmentation for robust grasp selection," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 9553–9560.

[22] Y. Qiu, J. Zhu, C. Della Santina, M. Gienger, and J. Kober, "Robotic fabric flattening with wrinkle direction detection," *arXiv preprint arXiv:2303.04909*, 2023.

[23] A. Ramisa, G. Aleny`, F. Moreno-Noguer, and C. Torras, "Using depth and appearance features for informed robot grasping of highly wrinkled clothes," in *IEEE International Conference on Robotics and Automation*, 2012, pp. 1–6.

[24] A. Ramisa, G. Alenyà, F. Moreno-Noguer, and C. Torras, "Determining where to grasp cloth using depth information," in *International Conference of the Catalan Association for Artificial Intelligence*, 2011.

[25] T. Sakamoto, W. Wan, T. Nishi, and K. Harada, "Efficient picking by considering simultaneous two-object grasping," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021.

[26] D. Seita, A. Ganapathi, R. Hoque, M. Hwang, E. Cen, A. K. Tanwani, A. Balakrishna, B. Thananjeyan, J. Ichnowski, N. Jamali, K. Yamane, S. Iba, J. Canny, and K. Goldberg, "Deep imitation learning of sequential fabric smoothing from an algorithmic supervisor," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 9651–9658.

[27] D. Seita, N. Jamali, M. Laskey, A. K. Tanwani, R. Berenstein, P. Baskaran, S. Iba, J. F. Canny, and K. Goldberg, "Deep transfer learning of pick points on fabric for robot bed-making," in *International Symposium of Robotics Research*, 2018.

[28] S. Sharma, E. Novoseller, V. Viswanath, Z. Javed, R. Parikh, R. Hoque, A. Balakrishna, D. S. Brown, and K. Goldberg, "Learning switching criteria for sim2real transfer of robotic fabric manipulation policies," in *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*, 2022, pp. 1116–1123.

[29] P. Sundaresan, J. Grannen, B. Thananjeyan, A. Balakrishna, J. Ichnowski, E. Novoseller, M. Hwang, M. Laskey, J. Gonzalez, and K. Goldberg, "Untangling Dense Non-Planar Knots by Learning Manipulation Features and Recovery Policies," in *Proceedings of Robotics: Science and Systems*, Virtual, Jul. 2021.

[30] S. Tirumala, T. Weng, D. Seita, O. Kroemer, Z. Temel, and D. Held, "Learning to singulate layers of cloth using tactile feedback," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 7773–7780.

[31] K. M. Varadarajan and M. Vincze, "Object part segmentation and classification in range images for grasping," in *2011 15th International Conference on Advanced Robotics (ICAR)*, 2011, pp. 21–27.

[32] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[33] X. Wang, X. Jiang, J. Zhao, S. Wang, and Y.-H. Liu, "Grasping objects mixed with towels," *IEEE Access*, vol. 8, pp. 129 338–129 346, 2020.

[34] X. Wang, X. Jiang, J. Zhao, S. Wang, and Y.-H. Liu, "Picking towels in point clouds," vol. 8, 2020, pp. 129 338–129 346.

[35] B. Willimon, S. Birchfield, and I. Walker, "Classification of clothing using interactive perception," in *Int. S. Robotics Research (ISRR)*, 2011, pp. 1–7.

[36] T. Yamada and H. Yamamoto, "Static grasp stability analysis of multiple spatial objects," *Journal of Control Science and Engineering*, vol. 3, pp. 118–139, 2015.

[37] K. Yao and A. Billard, "Exploiting kinematic redundancy for robotic grasping of multiple objects," *IEEE Transactions on Robotics*, 2023.

[38] H. Zhang, J. Ichnowski, D. Seita, J. Wang, H. Huang, and K. Goldberg, "Robots of the lost arc: Self-supervised learning to dynamically manipulate fixed-endpoint

cables," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2021, pp. 4560–4567.