

Bringing Clothing into Desired Configurations with Limited Perception

Marco Cusumano-Towner, Arjun Singh, Stephen Miller, James F. O'Brien, Pieter Abbeel



Fig. 1: The PR2 with a pair of pants in a crumpled initial configuration.

Abstract—We consider the problem of autonomously bringing an article of clothing into a desired configuration using a general-purpose two-armed robot. We propose a hidden Markov model (HMM) for estimating the identity of the article and tracking the article’s configuration throughout a specific sequence of manipulations and observations. At the end of this sequence, the article’s configuration is known, though not necessarily desired. The estimated identity and configuration of the article are then used to plan a second sequence of manipulations that brings the article into the desired configuration. We propose a relaxation of a strain-limiting finite element model for cloth simulation that can be solved via convex optimization; this serves as the basis of the transition and observation models of the HMM. The observation model uses simple perceptual cues consisting of the height of the article when held by a single gripper and the silhouette of the article when held by two grippers. The model accurately estimates the identity and configuration of clothing articles, enabling our procedure to autonomously bring a variety of articles into desired configurations that are useful for other tasks, such as folding.

I. INTRODUCTION

Due to their inherently high-dimensional configuration spaces, non-rigid objects pose a number of difficult challenges. This difficulty is exemplified by the state of the art in robotic laundry folding, where existing methods are far from being able to perform the task with general purpose manipulators. Perhaps the biggest challenge facing robotic laundry manipulation is how to bring a clothing article into a known configuration from an arbitrary initial state.

The authors are with the Department of Electrical Engineering and Computer Sciences, UC Berkeley, CA 94720, U.S.A. Contact Email: marcoct@berkeley.edu.



Fig. 2: The PR2 holding up the pair of pants after starting with the configuration shown in Figure 1.

We present an approach that enables a general purpose robot to bring a variety of clothing articles into desired configurations. The core of our approach is a hidden Markov model based on the behavior of cloth under certain simple manipulation strategies, and how it is perceived using basic computer vision primitives. Key contributions of this paper are:

- We propose a method for identifying an article of clothing and estimating its configuration with only simple manipulation and limited perception.
- We propose a convex relaxation of the isotropic strain-limiting model of Wang et. al [16] for cloth simulation.
- We present a planning algorithm for bringing an article from a known configuration into a desired configuration.
- We describe our implementation of an end-to-end system on the Willow Garage PR2 robotic platform. The system starts with crumpled articles and brings them into a desired configuration. We successfully tested our implementation on seven articles of various types. All of the parameters in our system were trained on a separate set of articles.

Videos of our experimental results are available at:

http://rll.berkeley.edu/ICRA_2011

II. RELATED WORK

Extensive work has been done on enabling specialized and general-purpose robots to manipulate laundry. To the best of our knowledge, however, with the exception of the towel folding capability demonstrated by Maitin-Shepard et al. [8], no prior work has reported successful completion of the full

end-to-end task of picking up an arbitrarily placed clothing article and bringing it into a neatly folded state. In this paper we focus on a key part of this end-to-end task; namely, bringing a clothing article from an unknown configuration into a desired configuration.

The work of Osawa et al. [12] and Kita et al. [3]–[6] is the most closely related to our approach. Osawa et al. use the idea of iteratively grasping the lowest-hanging point. They describe how this procedure leads to a relatively small number of fixed points. Once their perception unit recognizes that a corner has been grasped, their procedure compares the shape observed while pulling taut with pre-recorded template images. They reported recognition rates on seven different clothing categories. In contrast to our work, they require template images of the articles, they have a one-shot decision making process (rather than a probabilistic estimation framework), and their procedure only performs “lowest-hanging point” re-grasps. As a consequence of only re-grasping lowest points, their final configuration is not necessarily spread out. While they do not report success rates, they show successful manipulation of a long-sleeved shirt with a final configuration having it held by the ends of the two sleeves. Kita et al. consider a mass-spring model to simulate how clothing will hang. Their work shows the ability to use fits of these models to silhouettes and 3-D point clouds to extract the configuration of a clothing article held up by a single point with a good success rate. Their later work [4]–[6] shows the ability to identify and grasp a desired point with the other gripper. None of this prior work demonstrates the ability to generalize to previously unseen articles of clothing.

There is also a body of work on recognizing categories of clothing; some of this work includes manipulation to assist in the categorization. For example, Osawa and collaborators [12] and Hamajima and Kakikura [2] present approaches to spread out a piece of clothing using two robot arms and then categorize the clothing.

Some prior work assumes a known, spread-out, or partially spread-out configuration, and focuses on folding or completing other tasks. The work of Miller et al. [9], building on the work of van den Berg et al. [15], has demonstrated reliable folding of a wide range of articles. Paraschidis et al. [1] describe the isolated executions of grasping a laid-out material, folding a laid-out material, laying out a piece of material that

was already being held, and flattening wrinkles. Yamakazi and Inaba [17] present an algorithm that recognizes wrinkles in images, which in turn enables them to detect clothes laying around. Kobori et al. [7] have extended this work towards flattening and spreading clothing. They successfully spread out a towel.

III. OVERVIEW

A. Problem Definition

The problem we examine is defined as follows: we are presented with an unknown article of clothing and wish to identify it, work it into an identifiable configuration, and subsequently bring it into a desired configuration.

B. Notation

We consider articles of different *types* and *sizes*. These potential articles make up the set of articles (A) under consideration. For example, we may be considering two different pairs of pants and a t-shirt; in this case, we have $A = \{\text{pants}_1, \text{pants}_2, \text{t-shirt}\}$. We assume that each type of clothing can have its own desired configuration; for example, we choose to hold up pants by the waist.

We represent each potential article a (from the set A) via a triangulated mesh. For concreteness, let the mesh contain N points $\{v^1, \dots, v^N\}$. We work under the assumption that the robot may be grasping a single point or a pair of points on the mesh. Let g_t be the grasp state of the cloth at time t , where $g_t = (g_t^l, g_t^r)$ consists of the mesh point of the cloth in the robot’s left and right gripper respectively. More precisely, we have $g_t = (g_t^l, g_t^r) \in G = \{\emptyset, v^1, \dots, v^N\}^2$, where \emptyset denotes that the gripper does not contain any mesh point. The set G contains all possible grasp states of the cloth. The 3D coordinates of the left and right gripper at time t are denoted by \mathbf{x}_t^l and \mathbf{x}_t^r respectively. We denote the 3D coordinates of the N mesh points at time t as $\mathbf{X}_t = \{\mathbf{x}_t^1, \dots, \mathbf{x}_t^N\}$.

C. Outline of Our Approach

Our approach consists of two phases, as shown in Figure 3. First, we use a probabilistic model to determine the identity of the clothing article while bringing it into a known configuration through a sequence of manipulations and observations, which we refer to as the *disambiguation phase*. Second, we bring the article into the desired configuration through another sequence of manipulations and observations, which we call the *reconfiguration phase*.

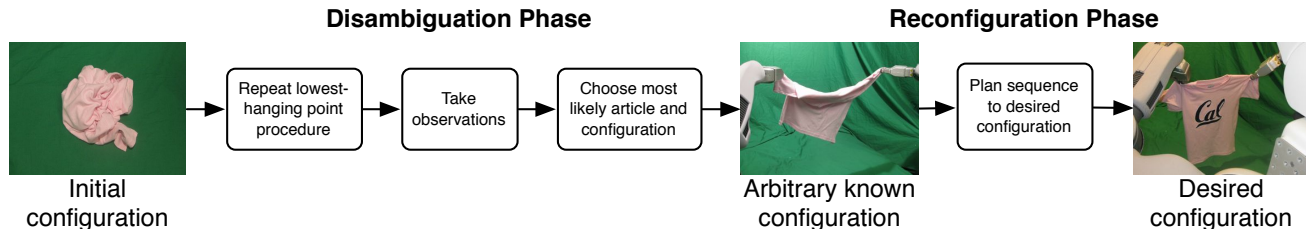


Fig. 3: Block diagram outlining our procedure. The t-shirt starts out in a crumpled state. We manipulate it with the lowest-hanging point procedure and take observations. We choose the most likely configuration and article and plan a sequence of manipulations to the desired configuration. The robot executes the sequence and grasps the t-shirt by the shoulders.

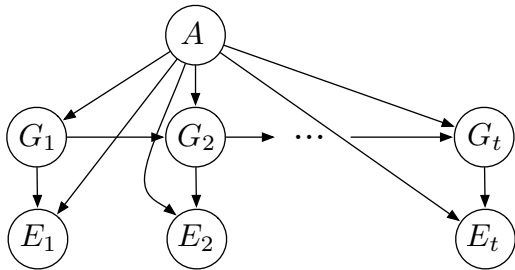


Fig. 4: Graphical representation of the hidden Markov model.

These phases use three major components: the hidden Markov model, our cloth simulator, and our algorithm for planning the manipulations for the reconfiguration phase.

1. Hidden Markov model. The hidden state of our model consists of the grasp state of the cloth (g_t) and the article which the robot is currently grasping (a). The HMM operates in the disambiguation phase, where the robot executes a sequence of manipulations consisting of repeatedly holding up the clothing article by one gripper under the influence of gravity and grasping the lowest-hanging point with the other gripper. The transition model of the HMM encodes how the grasped points change when the robot manipulates the cloth. This sequence quickly reduces uncertainty in the hidden state. After this manipulation sequence the HMM uses two observations, the height of the article when held by the last point in the sequence and the contour of the article when held by two points.

2. Cloth simulator. We simulate articles using triangulated meshes in which each triangle element is strain-limited and bending energy and collisions are ignored. This model has a unique minimum-energy configuration (\mathbf{X}_t) when some points in the mesh are fixed at the locations (g_t^l, g_t^r).

3. Planning algorithm. To generate the plan for the reconfiguration phase, our planning algorithm generates a sequence of manipulations in which the robot repeatedly grasps two points on the cloth, places the cloth onto the table, and grasps two other points. The planning algorithm assumes that the most likely model and state reported by the HMM in the disambiguation phase are correct.

IV. HIDDEN MARKOV MODEL

We have the discrete random variables A and G_t , where G_t takes on values from the set G of all possible grasp states of the cloth (as defined in Section III-B). The model estimates the probability $P(A = a, G_t = g_t | E_{1:t} = e_{1:t})$, where $E_{1:t}$ is the set of all observations through time t . The robot's gripper locations (\mathbf{x}_t^l and \mathbf{x}_t^r) are assumed to be deterministic. The graphical model for our problem is shown in Figure 4.

As described above, the 3D coordinates of the mesh points at time t (\mathbf{X}_t) are uniquely determined from the article, grasp state of the cloth, and the locations of the grippers (a, g_t, \mathbf{x}_t^l , and \mathbf{x}_t^r , respectively). Using G_t as the state rather than

\mathbf{X}_t reduces the state space to a tractable size on the order of N^2 , where N is the number points in a mesh.

Without loss of generality, we assume that the robot first picks up the article with its left gripper. Intuitively, the initial probability distribution over models and grasp states should be zero for any state in which the right gripper is grasping the cloth and uniform over all states in which the left gripper is grasping the cloth. Therefore the initial distribution is:

$$P(a, g_0) = \begin{cases} \frac{1}{N|A|} & g_0^r = \emptyset, g_0^l \neq \emptyset \\ 0 & \text{otherwise.} \end{cases}$$

Recall that the disambiguation phase consists of repeatedly holding up the article (under the influence of gravity) by one gripper and grasping the lowest-hanging point with the other gripper. Let g_{t-1} be the grasp state of the robot at the end of this process; the robot is only holding the article in one gripper. Next, we take an observation of the height (h_{t-1}) of the article in this grasp state (g_{t-1}). Afterwards, we have the free gripper grasp the lowest-hanging point, bringing us into the grasp state g_t , in which both grippers are grasping the article. We then move the grippers such that they are at an equal height and separated by a distance of roughly h_{t-1} . Therefore the gripper locations are now

$$\mathbf{x}_t^l = \left(x, \frac{h_{t-1}}{2}, z \right), \quad \mathbf{x}_t^r = \left(x, \frac{-h_{t-1}}{2}, z \right)$$

where the exact values of x and z are unimportant. We then take an observation of the contour of the article against the background.

Together, the lowest-hanging point sequence along with the two observations compose all of the information obtained about the article during the disambiguation sequence; the details of the probabilistic updates for the transitions and observations are explained below.

A. Transition Model

A transition in the HMM occurs after holding the cloth up with one gripper, grasping the lowest-hanging point with the free gripper, and then releasing the topmost point. Without loss of generality, let the cloth be grasped by only the left gripper at time t . Specifically, the grasp state g_t is (g_t^l, \emptyset) . This implies $g_{t+1} = (\emptyset, g_{t+1}^r)$, where g_{t+1}^r is the lowest-hanging point at time t . The transition model gives the probability that each mesh point hangs lowest and is therefore grasped by the right gripper. In particular,

$$P(g_{t+1}|a, g_t) = \begin{cases} P(g_{t+1}^r \text{ is lowest} | a, g_t^l \text{ is held}) & \text{if } g_{t+1}^l = \emptyset \\ 0 & \text{if } g_{t+1}^l \neq \emptyset. \end{cases}$$

The transition model assumes the robot has successfully grasped a point with the right gripper. When we simulate an article held at a single point v^i , the resulting configuration \mathbf{X} is a straight line down from v^i ; the probability of point v^j hanging lowest when the cloth is held by v^i depends on \mathbf{X} . Let d_{ij} be the vertical distance from v^i to v^j in this configuration. The probability of a point hanging lowest is

based on d_{ij} :

$$P(v^j \text{ is lowest} \mid a, v^i \text{ is held}) = \frac{e^{\lambda d_{ij}}}{\sum_{k=1}^N e^{\lambda d_{ik}}}.$$

This expression is a soft-max function, resulting in a distribution in which points that hang lower in the simulated configuration are more likely to be the lowest-hanging point in reality. The parameter λ expresses how well the simulated configuration reflects reality.

Repeated application of the lowest-hanging point primitive causes the grasp state to converge to one of a small number of regions on the clothing article. For example, if a pair of pants is initially grasped by any point and held up under gravity, the lowest-hanging point will likely be on the rim of a pant-leg. Once the rim of the pant-leg is grasped and held up, the rim of the other pant-leg will likely contain the lowest-hanging point. Continuing this procedure typically cycles between the two pant-leg rims. In practice, the clothing articles we consider converge to a small number of points within two or three repetitions of the lowest-hanging point primitive. The HMM transition model captures this converging behavior, thereby significantly reducing uncertainty in the estimate of the article’s grasp state (g_t). Note that this transition model does not change the marginal probabilities of the articles ($P(a|e_{1:t})$).

B. Height Observation

When the article is held up by a single gripper, the minimum-energy configuration provided by our cloth simulator is a straight line, as described in Section V. Although this provides no silhouette to compare against, the length of this line (h_{sim}) is a good approximation to the article’s actual height (h_t). The uncertainty in this height is modeled with a normal distribution:

$$P(h_t|g_t, a) \sim \mathcal{N}(h_{\text{sim}} + \mu, \sigma^2)$$

where μ is the mean difference between the actual height and the simulated height.

This update makes configurations of incorrect sizes less likely. For example, if we measure that the article has a height of 70 cm, it is highly unlikely that the article is in a configuration with a simulated height of 40 cm.

C. Contour Observation

When the cloth is held up by two grippers, the contour of the simulated configuration is a good approximation to the actual contour, as seen in Figure 5. The predicted contours for each pair of grasp states and articles (g_t, a) are computed from the mesh coordinates (\mathbf{X}_t) generated by the cloth simulator, which is detailed in Section V. Next, the dynamic time warping algorithm is used to find the best alignment of each predicted contour to the actual contour. The score associated with each alignment is then used to update the belief $p(g_t, a|e_{1:t})$.

Although the general shape of the simulated contour and the actual cloth contour are similar, the amount of overlap between them can vary greatly between different trials due

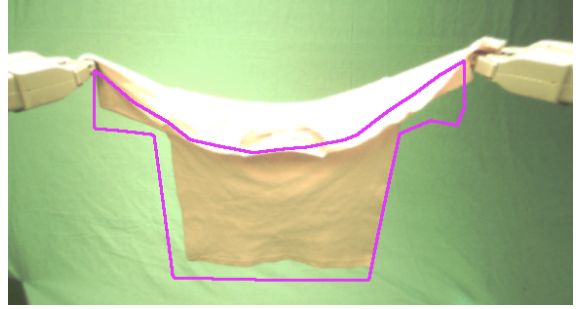


Fig. 5: The simulated contour (pink) is overlaid on the actual cloth image.

to inconsistency in grasping and other unmodeled factors. To account for this, we use a dynamic programming algorithm known as dynamic time warping (DTW) in the speech-recognition literature [13] and the Needleman-Wunsch algorithm in the biological sequence alignment literature [11]. Dynamic time warping is generally used to align two sequences and/or to calculate a similarity metric for sequences.

In order to closely match the key features of clothing articles, such as corners, collars, etc., we choose a cost function of the form

$$\phi(p_i, p_j) = \|\theta(p_i) - \theta(p_j)\|$$

where θ extracts the weighted features of each pixel p_i . Our features are the (x, y) pixel coordinates of the contour points and the first and second derivatives with respect to the arc length s , $(\frac{dx}{ds}, \frac{dy}{ds})$ and $(\frac{d^2x}{ds^2}, \frac{d^2y}{ds^2})$. The derivative terms force corners and other salient points to align to each other. An example where the amount of overlap is a poor measure of similarity but DTW returns a reasonable alignment is shown in Figure 6.

Let the dynamic time warping cost for each article and grasp state pair (a, g_t) be denoted c_{a, g_t} . We found that using an exponential distribution with the maximum-likelihood estimate was too generous to costs associated with incorrect configurations. Based on inspection of empirically collected

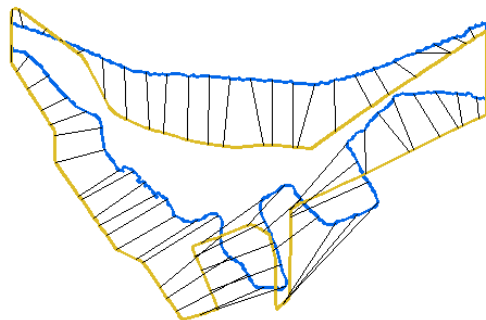


Fig. 6: An example of a challenging alignment where a simple overlap metric would perform poorly. The dynamic time warping algorithm matches the salient features of simulated contour (yellow) to the actual contour (blue) well.

DTW data, we propose the following distribution for the dynamic-time-warping costs:

$$P(c_{a,g_t}|a, g_t) = \begin{cases} \frac{1}{f+\frac{1}{d}} & \text{if } c_{a,g_t} < f \\ \frac{1}{f+\frac{1}{d}} e^{-dc_{a,g_t}} & \text{if } c_{a,g_t} \geq f. \end{cases}$$

This distribution, shown in Figure 7, is uniform for costs below a certain threshold and quickly drops off as the cost increases past the threshold.

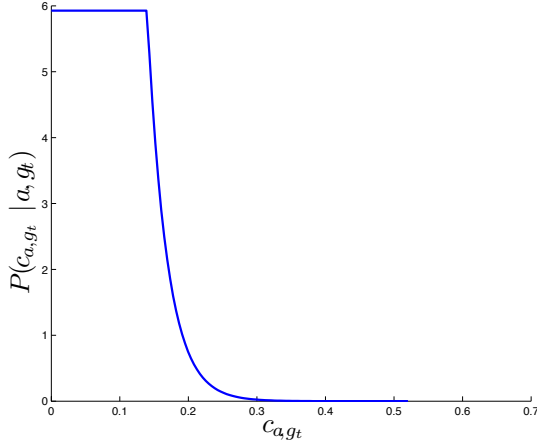


Fig. 7: Our probability distribution over DTW costs for the correct grasp state and article.

V. CLOTH SIMULATOR

To go from the grasp state (g_t) and article (a) to the simulated 3D coordinates of each mesh point ($\mathbf{X}_{\text{sim}} = \{\mathbf{x}_{\text{sim}}^1, \dots, \mathbf{x}_{\text{sim}}^N\}$), we minimize the gravitational potential energy of all mesh points subject to two sets of constraints. Our choice of constraints and energy function make this a convex optimization problem with a unique solution.

The first set of constraints represents the cloth’s grasp state as equality constraints on the simulated mesh configuration \mathbf{X}_{sim} . If the grasp state $(g_t^l, g_t^r) = (v^a, v^b)$, then the equality constraints are $\mathbf{x}_{\text{sim}}^a = \mathbf{x}_t^l$ and $\mathbf{x}_{\text{sim}}^b = \mathbf{x}_t^r$.

The second set of constraints limits the extensibility of the cloth. We use the isotropic strain-limiting model introduced by Wang et al. [16]. This model limits the strain of each triangle element $e \in E \subset \{v^1 \dots v^N\}$ ³ by restricting the singular values of the deformation gradient F_e^1 . Wang et al. restrict the minimum and maximum strain of each triangle element. We restrict only the maximum strain²; therefore our constraints are expressed as

$$\max \text{SingularValue}(F_e(\mathbf{X}_{\text{sim}})) \leq \sigma \text{ for all } e \in E,$$

where σ is the extensibility of the mesh surface, with $\sigma = 1$ indicating the cloth cannot be stretched at all³. This can

¹See Wang et al. for details [16].

²The minimum strain constraint is non-convex. We also ignore bending energy and collisions because they too are non-convex.

³We found that $\sigma = 1.03$ works well.

also be expressed as the following semidefinite programming (SDP) constraint:

$$\begin{bmatrix} \sigma^2 I_3 & F_e(\mathbf{X}_{\text{sim}}) \\ F_e^\top(\mathbf{X}_{\text{sim}}) & I_2 \end{bmatrix} \succeq 0 \text{ for all } e \in E. \quad (1)$$

In summary, our optimization problem becomes:

$$\begin{aligned} \min_{\mathbf{X}_{\text{sim}}} \quad & U(\mathbf{X}_{\text{sim}}) = \sum_{i=1}^N z^i \\ \text{s.t.} \quad & \mathbf{x}_{\text{sim}}^a = \mathbf{x}_t^l, \mathbf{x}_{\text{sim}}^b = \mathbf{x}_t^r \\ & \mathbf{X}_{\text{sim}} \text{ satisfies Equation (1)} \end{aligned}$$

where z^i is the z -coordinate of the i th mesh point. We feed this problem into the SDP solver SDPA [18] to simulate the article’s configuration⁴.

Despite the strong assumptions, this model predicts configurations whose contours are realistic for the case of two fixed points. The predicted contours are used in the observation update described in Section IV-C.

When the robot is only grasping the cloth with one gripper, the simulated configuration is a straight line down from the fixed point. Although this predicted configuration is visually unrealistic, the resulting height of each point is a good approximation. These height values are used by the HMM in the transition model and the height observation as described in Sections IV-A and IV-B respectively.

VI. PLANNING ALGORITHM

Our planning algorithm generates the sequence of manipulations to be carried out during the reconfiguration phase. We assume that the disambiguation phase correctly identifies the article and grasp state of the cloth.

For each type of clothing (e.g., shirts, pants, etc.), the user specifies a desired configuration, determined by the pair of points (v^i, v^j) that the robot should grasp. This determines a desired grasp state $g_d = (v^i, v^j)$. For example, the user could select that the robot hold a t-shirt by the shoulders or a pair of pants by the hips.

Our algorithm plans a sequence of grasp states, where each state has both grippers holding the cloth, to get from the initial grasp state g_i (obtained from the disambiguation phase) to the desired grasp state g_d . The sequence of manipulations to get from one grasp state to the next consists of laying the cloth on the table, opening both grippers, and picking up the cloth by a new pair of points.

The appropriate sequence of grasp states is generated by building the directed *graspability graph*, which indicates which other grasp states can be reached from each grasp state. To build this graph the article is simulated for all grasp states, and the resulting configurations are analyzed for graspability. To ensure that our assumption that the robot fixes a single point with each gripper is reasonable, we say that a point v^i is graspable in a given configuration \mathbf{X} when

⁴On a dual-core 2.0 GHz processor, SDPA can run roughly four simulations per second when we use about 300 triangle elements per mesh. We find that increasing the number of elements past this point does not make the simulations significantly more realistic.

only points in the local neighborhood⁵ of v^i on the mesh surface are close to v^i in terms of Euclidean distance in \mathbf{X} . Note that the local neighborhood on the mesh surface is a property of the mesh and does not change with the configuration \mathbf{X} . For example, consider grasping the corner of a sleeve of a t-shirt. When the sleeve is folded onto the chest, the mesh points on the chest are close to the corner of the sleeve in Euclidean distance but not local to each other on the surface. We say that the sleeve cannot be grasped because the robot would likely fix points on the chest in addition to the sleeve, resulting in a very different configuration from the one where only the sleeve is grasped. The complete planning algorithm is given in Algorithm 1.

Algorithm 1 Planning algorithm

Input: geodesic distances D , start pair (v^a, v^b)
desired end pair (v^x, v^y)
initialize graspability graph G with no edges
for all pairs (v^p, v^q) **do**
 $\mathbf{X} \leftarrow \text{Simulate}(v^p, v^q)$
for all pairs (v^s, v^t) **do**
if $\text{Graspable}(D, \mathbf{X}, v^s)$ and $\text{Graspable}(D, \mathbf{X}, v^t)$
then
add edge $((v^p, v^q) \rightarrow (v^s, v^t))$ to G
end if
end for
end for
 $\text{path} \leftarrow \text{Dijkstra}(G, (v^a, v^b), (v^x, v^y))$

Algorithm 2 Graspable

Input: geodesic distances D , configuration \mathbf{X} , point v^p
Parameters: geodesic thresh. d_1 , Euclidean thresh. d_2
for $i = 1$ to N **do**
if $D(v^i, v^p) > d_1$ and $\|\mathbf{x}^i - \mathbf{x}^p\| < d_2$ **then**
return false
end if
end for
return true

To go from a grasp state g_a to another grasp state g_b , the robot drags the cloth onto the table in a way that preserves the configuration that was present when the cloth was held in the air under gravity. The friction force of the table on the cloth acts similarly to gravity. We then simulate the minimum-energy configuration for the grasp state g_a and extract the predicted contour. An example contour is given in Figure 8. We then align this predicted contour to the actual cloth contour using the dynamic time warping method described above in Section IV-C. We then find the two pixels in the predicted contour corresponding to points g_b^l and g_b^r . Next, we follow the alignment to get the corresponding pixels in the actual contour and determine the 3D coordinates of points g_b^l and g_b^r . The desired grasp state for a pair of pants is shown in Figure 8.

⁵The local neighborhood is determined by the geodesic distances from v^i to the other points. We use the algorithm by Mitchell et al. [10] to find the geodesic distances in the mesh.

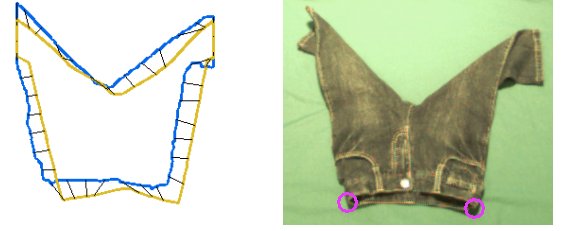


Fig. 8: The grasp points, shown in pink, are identified by following the alignment from the simulated contour (yellow) to the actual contour (blue).

VII. EXPERIMENTS

We assess our system’s performance on two tasks. The first is the identification of the article and its grasp state after the end of the disambiguation phase. The second task adds the reconfiguration phase; together, the two phases compose the end-to-end task of bringing articles into a desired configuration.

A. Setup

We use the PR2 robotic platform, manufactured by Willow Garage, to perform all manipulations. The robot is entirely autonomous throughout the procedure. We used a compliant foam working surface to allow the PR2’s grippers to slide underneath the cloth when grasping. We use a pair of two 640x480 color cameras mounted on the PR2 for all visual perception tasks. We use a green-colored background to simplify the task of image segmentation of the clothing article from the background.

The system requires mesh models of all clothing articles under consideration. Models of new clothing articles are generated by a nearly-autonomous process. The user measures three dimensions on the garment, identifies the clothing type (i.e. shirt, pants, skirt, etc.), and an appropriate cloth model is generated. Next, the mesh-generation software Triangle [14] produces an initial planar mesh with triangle areas constrained to 15 cm². This mesh is then converted into a full two-sided 3D mesh. The final 3D mesh contains approximately 300 triangle elements per clothing article.

We use an approximation in the contour observation update of the HMM. Simulating all N^2 possible grasp states can be done in around two hours per article. Although this could be done offline, we opted to only consider a subset of the pairs. In our experiments we tracked the probabilities associated with about 15 possible grasp points. We also prune simulations that are obviously infeasible, resulting in a contour observation runtime of one or two minutes when considering five articles.

All experimental runs start with the clothing article in a random configuration on the table in front of the robot, as shown in Figure 1. The test set for our experiments, shown in Figure 9, consists of one short-sleeve shirt, one long-sleeve shirt, one pair of pants, one skirt, one towel, and two infant clothing articles meant to judge how well our



Fig. 9: Our test set of clothing articles.

algorithm generalizes. We found the maximum likelihood estimates of our parameters using data from a separate training set of articles; the only prior interaction with the test set was measuring the dimensions needed to generate the candidate article meshes. We conducted 30 full end-to-end experiments the test set. We also conducted 10 more experiments consisting only of the disambiguation phase. The data collected in the end-to-end runs was also used to calculate the disambiguation results.

B. Disambiguation Experiments

We tested the disambiguation phase and probabilistic model under three settings of the candidate article set A :

- 1) In the first experiment, A only includes the correct article’s mesh. In this test, we measure the accuracy of the grasp state reported by the HMM. We declare success when the most likely grasp state $g_t = (g_t^l, g_t^r)$ has both g_t^l and g_t^r within 5 cm of the points on the cloth that the robot is actually holding.
- 2) In the second experiment, A includes the correct article a as well as four extra articles which are transformed versions of a where each differs from a by 10 cm in a single dimension. We record whether the grasp state and article (g_t, a) reported by the HMM has the correct article a . This experiment assesses the ability of the algorithm to determine an article’s size to a resolution of 10 cm. We also check if the reported grasp points are within 5 cm of the actual grasp points, as above.
- 3) In the third experiment, A includes the correct models of all seven articles in our test set. We assess whether the reported grasp state and article (g_t, a) has the correct article a , and whether the grasp state is accurate to within 5 cm.

The results for the disambiguation experiments are shown in Table I and detailed below.

Experiment 1: During the first experiment, 38 out of 40 runs were successful. One failure occurred because that article had relatively weak shape features for distinguishing between grasp states. In the other failure, the article was flipped over itself, as shown in Figure 10. Therefore it was not in the minimum-energy configuration predicted by the simulator.

Experiment 2: In the second experiment, the correctly sized article was chosen in 26 out of 40 trials. Because there were five choices (the actual article and the four transformed articles), this is significantly better than a random selection. Eight of the fourteen errors were from two articles that lack strong shape features; adding 10 cm to a single dimension of these articles does not significantly change the shape of the contour or the height when held by a single point. However, in the same test, the grasp state of the articles was still accurately determined in 38 out of 40 trials (even if the incorrectly sized article was reported).

Experiment 3: In the third experiment, in which all seven articles from the test set were considered (with no transformed versions), the correct article was chosen in 36 out of 40 trials. One of the errors was due to the robot grasping two points with one gripper that were not in the same local neighborhood on the mesh surface (the sleeve and the waist of a t-shirt); because our grasp state model assumes each gripper only grasps a single point, this configuration was not considered by the HMM. In one of the other failures, the robot was grasping too much fabric, which violates this same assumption but in a less drastic manner. Of the cases where the article was correctly identified, the correct grasp state was estimated in all but one.

Candidate article set	Correct article	Correct grasp (5 cm)
Correct article only	—	94.87%
With transformed articles	64.10%	94.87%
All test articles	92.31%	89.74%

TABLE I: Results for the disambiguation experiments, in which the identity and grasp state of the clothing articles are estimated. See Section VII-B for details.

Overall success rate	20/30
Failures	
— Robot could not reach point	9/30
— Incorrect estimated grasp state	1/30

TABLE II: Results for the full end-to-end task. Note that the majority of failures were due to the robot not being able to reach a target grasp point in the planned sequence.



Fig. 10: A cloth configuration in which the article is flipped over itself and therefore not in the minimum-energy configuration that is predicted by our simulator.

C. End-to-End Task

On the end-to-end task of bringing articles into a desired configuration, our system successfully brought the article into the desired configuration in 20 out of 30 trials. Of the ten failures, nine were because the robot could not reach a grasp point. The other failure was because the disambiguation procedure reported an incorrect grasp state. The results are summarized in Table II.

VIII. CONCLUSIONS AND FUTURE WORK

We proposed a method, involving only simple manipulations and basic computer vision, for bringing an article from an unknown configuration into a desired configuration while simultaneously determining its identity. This process involved two phases. We first identify the article while bringing it into an arbitrary known configuration. We have shown that our method is highly successful at identifying the type and grasp state of the article, with a success rate of about 90%. The second phase involves taking it from the known configuration and bringing it into a desired configuration. Our results here are promising and demonstrate the first reliable completions of this task.

The disambiguation phase failed when the robot fixed multiple points on the cloth or the article's shape was uninformative. Our model could easily be extended to account for both of these. For example, one could add additional observations into the HMM; adding image features such as buttons could help distinguish between grasp states with similar contours. Furthermore, one could model the potential for the robot to fix multiple points on the cloth with a single gripper by fixing additional points in the simulator. The majority of failures in the end-to-end procedure were due to the robot not being able to reach a target point on the cloth. We believe that integrating a motion planner that considers both base and arm motion into our system would eliminate these failures.

ACKNOWLEDGEMENTS

This work was supported in part by NSF under award IIS-0904672 and by Willow Garage's PR2 Beta Program.

REFERENCES

- [1] N. Fahantidis, K. Paraschidis, V. Petridis, Z. Doulgeri, L. Petrou, and G. Hasapis. Robot handling of flat textile materials. *Robotics & Automation Magazine, IEEE*, 4(1):34–41, Mar 1997.
- [2] K. Hamajima and M. Kakikura. Planning strategy for task of unfolding clothes. In *Int. Conf. on Robotics and Automation*, volume 32, pages 145 – 152, 2000.
- [3] Y. Kita and N. Kita. A model-driven method of estimating the state of clothes for manipulating it. In *Proc. of 6th IEEE Workshop on Applications of Computer Vision*, 2002.
- [4] Y. Kita, F. Saito, and N. Kita. A deformable model driven visual method for handling clothes. In *Proc. ICRA*, 2004.
- [5] Y. Kita, T. Ueshiba, E. S. Neo, and N. Kita. Clothes state recognition using 3D observed data. In *Proc. ICRA*, 2009.
- [6] Y. Kita, T. Ueshiba, E. S. Neo, and N. Kita. A method for handling a specific part of clothing by dual arms. In *Proc. IROS*, 2009.
- [7] H. Kobori, Y. Kakiuchi, K. Okada, and M. Inaba. Recognition and motion primitives for autonomous clothes unfolding of humanoid robot. In *Proc. IAS*, 2010.
- [8] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei, and P. Abbeel. Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding. In *Proc. ICRA*, 2010.
- [9] S. Miller, M. Fritz, T. Darrell, and P. Abbeel. Parameterized shape models for clothing. In *Proc. ICRA*, 2011.
- [10] J. S. B. Mitchell, D. M. Mount, and C. H. Papadimitriou. The discrete geodesic problem. *SIAM J. on Computing*, 16(4):647668, 1987.
- [11] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. of Molecular Biology*, 48(3):443 – 453, 1970.
- [12] F. Osawa, H. Seki, and Y. Kamiya. Unfolding of massive laundry and classification types by dual manipulator. *JACIII*, 11(5):457–463, 2007.
- [13] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49, 1978.
- [14] J. Shewchuk. Triangle: Engineering a 2D quality mesh generator and Delaunay triangulator. *Applied Computational Geometry Towards Geometric Engineering*, pages 203–222, 1996.
- [15] J. van den Berg, S. Miller, K. Goldberg, and P. Abbeel. Gravity-based robotic cloth folding. In *Proc. WAFR*, 2010.
- [16] H. Wang, J. O'Brien, and R. Ramamoorthi. Multi-resolution isotropic strain limiting. In *Proc. ACM SIGGRAPH Asia*, 2010.
- [17] K. Yamakazi and M. Inaba. A cloth detection method based on image wrinkle feature for daily assistive robots. In *IAPR Conf. on Machine Vision Applications*, 2009.
- [18] M. Yamashita, K. Fujisawa, and M. Kojima. Implementation and Evaluation of SDPA 6.0 (Semidefinite Programming Algorithm 6.0). *Optimization Methods and Software*, 18(4):491–505, 2003.