Model-free Approach to Garments Unfolding Based on Detection of Folded Layers

Jan Stria, Vladimír Petrík, Václav Hlaváč

***Abstract*— The proposed work deals with robotic unfolding of a garment that has been placed flat on a table and folded over a certain axis. The algorithm combines image and depth data to detect the bottom and top (folded) layer of the garment. The detection is formulated as a labeling of the garment surface and solved in an energy minimization framework. Once the garment pose is known, several candidate folding axes are generated and used to unfold the garment virtually. The correct folding axis is selected from these candidate axes. The method does not set any constraints on the garment shape; thus it can deal with various types of garments including jackets, pants, shorts, skirts or T-shirts of any sleeve lengths. The garment is unfolded by the dual-arm robot. One arm grasps boundary of the top layer and brings it over the estimated folding axis, while the second arm is holding the bottom layer to prevent the garment from slipping. The perception procedure was tested on the annotated dataset that we are making publicly available. The experimental evaluation of the robotic manipulation is also provided.**

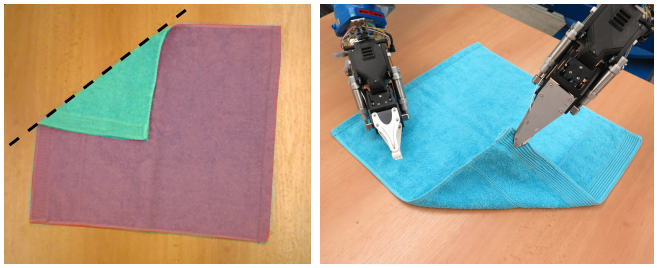
I. Introduction

Visual perception and robotic manipulation of garments is a challenging task because of significant variance in their appearance caused by *deformations*. The task has gained an increasing interest in cognitive robotics recently. The possible applications include housekeeping robots capable of laundering or helping the physically challenged people to get dressed. There are several industrial applications too, including fully automated sewing machines or placing the carpets and soft noise dampers on a car assembly line.

The presented work deals with robotic unfolding of a garment laid on a table. The type of the garment is not known in advance. It is assumed that the garment is posed in such configuration that can be achieved by grasping the boundary of the fully spread garment and pulling it over a certain *folding axis* (Fig. 1a). It is also assumed that the part of the garment folded over the axis does not cover the remaining part completely. Let us call the folded part a *top layer* and the remaining one a *bottom layer*, distinguished by their height above the table (Fig. 1a). Note that the majority of garments consist of a front and back side sewn together, i.e. two layers of material, but we consider them a single layer in this work. We allow the garment to be folded multiple times if the top folded layers do not overlap with each other (Fig. 6a).

The input of the method is formed by a *color image* (Fig. 2a) and a *depth map* (Fig. 3a). The goal is to *detect* both layers of the folded garment and estimate the folding axis. Note that the axis forms an approximate segment on the contour of the folded garment (Fig. 1a), which we utilize for its estimation. Once the pose of the garment is known, it is *unfolded* by coordinated manipulation of two robotic arms. The boundary of the top layer is grasped with a specialized gripper and brought back over the folding axis, while the other arm is pushing the bottom layer against the table in order to prevent the garment from slipping (Fig. 1b).

The authors are with the Czech Institute of Informatics, Robotics and Cybernetics at the Czech Technical University in Prague, {jan.stria, vladimir.petrik, vaclav.hlavac}@cvut.cz



(a) Folded garment (b) Robotic unfolding

Fig. 1: a) The garment folded over the axis (black dashed line). Its visible surface consists of the top (green) and bottom (red) layer. b) The top layer is grasped and unfolded over the axis, while the bottom layer is pushed against the table to prevent the garment from slipping.

The proposed task is motivated by a more general procedure for two-stage unfolding of a crumpled garment [1], where the garment is untangled and stretched out in the air at first and then the remaining fold is removed on the table. Another motivation is the improvement of the unfolding skill itself, which has been already studied previously [2], [3].

The *main contributions* of this work are:

- We formulate the detection of the stacked layers of the folded garment as a labeling problem. Is is solved in an energy minimization framework, fusing acquired image and depth information.
- The parameters of the energy function are estimated automatically from the currently observed data.
- We show how to estimate the folding axis by generating several candidate axes, unfolding the garment virtually and analyzing the unfolded shapes.
- The proposed method was evaluated experimentally on the dataset which we are making publicly available.
- The unfolding manipulation skill was implemented on our dual-arm robot, with both arms cooperating.

II. Related work

Visual perception and robotic manipulation of garments are closely related to their *automated laundering*. The manipulation starts with *grasping*. Common strategy is to grasp the central [4] or the highest observed point [5] of the garment. There are attempts to quantify the graspability of locations on

the garment surface [6] and selecting the best one. Another approach is grasping the detected garment boundary [7].

In majority of tasks, the type of the garment needs to be *classified* and its *pose estimated* to choose the proper manipulation strategy. The garment is usually grasped and lifted to a hanging position at first to reduce the space of its possible configurations [8]. Then one strategy is to match the acquired point cloud to virtual models of variously posed garments of different categories, select the best matching model and therefore solve the classification and pose estimation jointly [8], [9]. Different approach is to train a classifier, using either hand-crafted features [10], [11] or features learned automatically from data [12].

Folding of a spread garment is another broadly studied task. The pose of the spread garment is usually recognized by matching a polygonal model to its contour [13], [14]. The matching can be repeated after each fold to check the manipulation result [15]. There exist several strategies how to move the arm holding the garment to prevent it from slipping while folding it, including a simple triangular path [13] or more stable circular path [16]. More advanced approaches select the optimum trajectory based on the estimated material properties and physics-based simulation [17], [18].

The task related to the topic of this work is *flattening*. It is supposed that the garment is nearly spread on a table, but its surface is wrinkled [19], [20]. Size and orientation of the wrinkles are analyzed and the optimum strategy is planned how to pull the garment to side in order to remove them.

The proposed paper deals with *unfolding*. The majority of existing works unfold the garment while holding it hanging in the air [7], [11], [21], [22]. The garment is regrasped several times to be held at some predefined positions in the end, e.g. two corners of a towel. Different strategy is to unfold the garment only partially in the air and finish its unfolding on a table [1]. The partial unfolding is accomplished by detecting points on the garment outline, grasping them and stretching the garment out to a nearly planar state. The configuration of the garment laid on the table is then similar to our case.

The exact configuration and type of the partially unfolded garment are estimated by shape matching [1], [2]. The contour of the observed folded garment is partially matched to various templates of unfolded garments, giving a hypothesis on the location of the folding axis for each template. The observed contour is then unfolded virtually and registered to all templates, while allowing certain local deformations [2]. The best matching template determines both pose and type.

In [3], they acquire a depth map of a folded garment, segment the depths with the watershed algorithm and select the highest region for unfolding. The unfolding direction is determined by examining height changes on various paths from that region to the outer contour.

III. Preprocessing

The input of our method is formed by an RGB image I and depth map D. The image and depth map are equally sized and calibrated, i.e. the color $I(p)$ and depth $D(p)$ of the pixel p correspond to the same real world location.

(a) Input image (b) Segmentation (c) Erosion and dilation

Fig. 2: a) The input image is b) segmented. c) The segmentation mask P is eroded to P_e (removed green strip) and dilated to P_d (added red strip).

A. Segmentation

The image I is used to segment the observed garment from its background, which is a wooden table in our case. Fig. 2a shows a sample input. It is assumed that the background differs from the garment and does not change in the experiment. It is therefore possible to learn a probabilistic model of the background color in the form of the *Gaussian mixture model* (GMM) of RGB triples.

The segmentation consists of two phases. In the first phase, each pixel p is labeled based on the likelihood of its color $I(p)$ in the learned GMM. Less likely pixels are labeled as the foreground (garment), more likely pixels as the background (table) and the remaining ones as unknown. In the second phase, the foreground and background pixels are used to initialize the *GrabCut* [23] algorithm yielding the final segmentation mask (Fig. 2b). See [14], [15] for details.

Let us denote P the set of garment pixels from the mask. The mask is *morphologically eroded and dilated* [24] with a disc structuring element, the radius of which corresponds approximately to 2 cm. Let us denote the eroded pixels P_e and the dilated ones P_d. It holds $P_e \subset P \subset P_d$. Fig. 2c shows an example of the eroded and dilated mask.

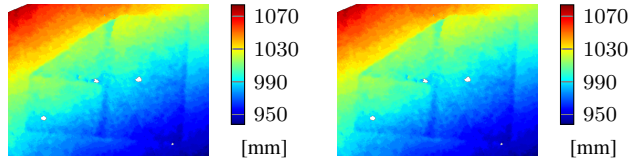
B. Height extraction

The depth map D is acquired by the ASUS Xtion sensor attached to the wrist of the robot. Although the arm points approximately downwards during the acquisition, the view direction is not perfectly perpendicular to the table surface. Another issue is that the depths are not known for all pixels due to the limitations of the *structured light* technology. See Fig. 3a for an example of the depth map.

The contour of the dilated mask P_d is extracted. All pixels on the contour should belong to the table. The depth values in the contour pixels are used to estimate the parameters of a plane approximating the table surface with RANSAC [25]. The input depth map D is then subtracted from that table plane. This results in a height map H, where $H(p)$ is a height of the pixel p above the table (Fig. 3b).

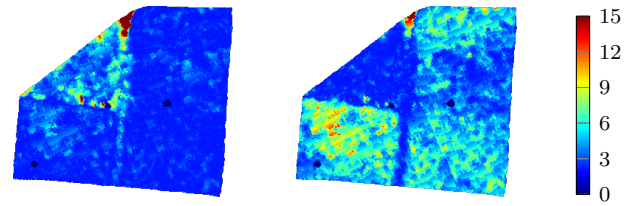
IV. Detection of Layers

The fold detection task, as defined in Sec. I, can be formulated as a *labeling* problem. Each garment pixel $p \in P$ needs to be assigned a label $z_p \in \{T, B\}$. Label T represents the top layer of the garment, B its bottom layer. The labeling relies on the presence of *edges* in the image I, which are coincident with the boundary of the layers, and on *different heights* of the top and bottom pixels in the height map H.



(a) Depths acquired by the sensor (b) Heights above the table

Fig. 3: a) Depths acquired by the range sensor are subtracted from the plane approximating the table surface to get b) heights above the table. The depths and heights are unknown in white areas.



(a) Bottom layer (b) Top layer

Fig. 4: Unary potentials of pixels being assigned to bottom and top layer of the folded garment, i.e. the potentials $U_p(B)$ and $U_p(T)$ for $p \in P_e$.

Since the outer contour of the garment is a source of undesirable edges in the image I, only the inner eroded pixels P_e (Fig. 2c) are labeled at first. The labeling is then extrapolated to the boundary pixels of the original mask P.

A. Energy minimization

The labeling of pixels $p \in P_e$ by labels $z_p \in \{T, B\}$ is formulated as the following *energy minimization* problem, which is used in many computer vision applications [26]:

$$Z^* = \underset{Z \in \{T,B\}^{|P_e|}}{\arg\min} \sum_{p \in P_e} U_p(z_p) + \sum_{\{p,q\} \in N_e} V_{p,q}(z_p, z_q) \quad (1)$$

The functions $U_p \colon \{T, B\} \to \mathbb{R}$ are called *unary potentials*. They express costs of assigning pixels to the particular layer. The functions $V_{p,q} \colon \{T, B\}^2 \to \mathbb{R}$ are *pairwise potentials*. They are defined for pairs of neighboring pixels $\{p, q\} \in N_e$, where $p, q \in P_e$. We use 8-connected neighborhood, i.e. all pairs of vertically, horizontally or diagonally adjacent pixels. The pairwise costs are used to align the boundary between the labeled layers with the edges observed in the image. They also make the boundary smooth.

The pairwise potentials, as defined in Sec. IV-C, are *regular* in the sense of [27], i.e. for each $\{p, q\} \in N_e$ it holds $V_{p,q}(T, T) + V_{p,q}(B, B) \leq V_{p,q}(T, B) + V_{p,q}(B, T)$. Therefore, the globally optimum labeling Z^* with respect to (1) can be found effectively by constructing a certain weighted graph and finding its minimum cut [26].

B. Unary potentials

The unary potential $U_p(z)$ for the pixel $p \in P_e$ and the label $z \in \{T, B\}$ is defined as follows:

$$U_p(z) = \begin{cases} -\log \mathcal{N}\big(H(p); \mu_z, \sigma^2\big), & H(p) \text{ is known} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

We suppose that heights of pixels above the table in one layer are distributed normally. Therefore, the potentials have the form of *negative log-likelihood*. The mean height of the top layer is μ_T and the bottom one is μ_B. The variance σ^2, shared by both distributions, is caused by wrinkled surface of the garment and by the noise present in the input depths. Sec. IV-D explains how μ_T, μ_B and σ^2 can be estimated from the observed heights. Fig. 4 shows an example of the unary potentials computed from Fig. 3b. The definition of unary potentials was inspired by [23], which uses distributions of foreground and background colors (instead of heights) for color-based segmentation of images.

C. Pairwise potentials

The pairwise potential $V_{p,q}(z_p, z_q)$ for two neighboring pixels $\{p, q\} \in N_e$ is defined similarly to [23]:

$$V_{p,q}(z_p, z_q) = \gamma_1 + \gamma_2 \frac{[\![z_p \neq z_q]\!]}{d(p, q)} \exp\left(-\frac{g(I, p, q)}{2E[g]}\right) \quad (3)$$

The term $d(p, q)$ denotes the *spatial distance* of the pixels p and q. It is equal to 1 for horizontally and vertically adjacent pixels and $\sqrt{2}$ for diagonal neighbors. The function g evaluates the *visual difference* of the pixels p and q in the image I. The term $E[g]$ denotes the mean value of the difference function g over all neighboring pixels. We set $\gamma_2 = 50$ as in [23] to balance the typical values of the unary and pairwise potentials. In addition to [23], we extend the potentials with the constant term $\gamma_1 = 1$ to prefer labelings with a *shorter boundary* between the top and bottom layer.

There are several options for choosing the function g. We smooth the image I by the *bilateral filter* [24], estimate the magnitudes of its gradient by convolving the image with the *Sobel filter* [24], and average these magnitudes among pairs of neighbors. Nevertheless, the experiments showed that the exact choice of g is not crucial, as long as it somehow expresses the presence of edges.

D. Estimating parameters of unary potentials

We show how to estimate the expected height of the top μ_T and bottom μ_B layer above the table as well as their variance σ^2. They are used to compute the pairwise potentials from Sec. IV-B. We model the heights of pixels from P_e using the *Gaussian mixture model* (GMM) with two components, each of them corresponding to a single layer. The components are weighted by *priors* π_T, π_B that are also unknown. Let us denote $\theta = (\mu_T, \mu_B, \sigma^2, \pi_T, \pi_B)$ all unknown parameters of the mixture. The likelihood of the pixel p having the height $H(p)$ is given by:

$$\mathcal{L}(\theta; H(p)) = \sum_{z \in \{T,B\}} \pi_z \, \mathcal{N}\big(H(p); \mu_z, \sigma^2\big) \quad (4)$$

The unknown parameters θ are estimated by the *expectation-maximization* (EM) algorithm [28]. It is based on the incremental refinement $\theta^{(t)}$ of their initial estimate $\theta^{(0)}$ for $t = 1, 2, \ldots$ The EM algorithm alters between the expectation and maximization steps, while increasing the lower bound on the likelihood from (4). Our algorithm is similar to estimation of the standard GMM with independent

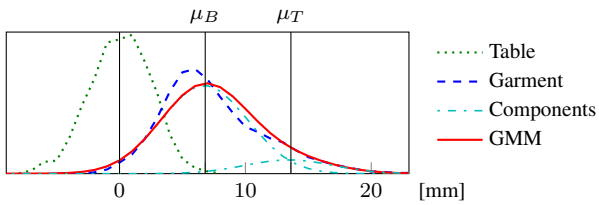


Fig. 5: Normalized histograms of table heights and garment heights. The garment heights are fitted GMM with two components by EM algorithm to estimate μ_T, μ_B and σ^2.

components [29]. However, the components are not independent in our case. They share the variance σ^2. Moreover, denoting Δ_μ the unknown thickness of one layer of the garment, the mean heights are linked as follows:

$$\mu_B = \Delta_\mu \qquad \mu_T = 2\Delta_\mu \tag{5}$$

The initial values of the parameters $\theta^{(0)}$ are estimated by applying the *k-means* algorithm [30] on the heights of eroded pixels P_e for $k = 2$. It results in a cluster of bottom layer pixels and a cluster of top pixels. The thickness $\Delta_\mu^{(0)}$ is computed based on (5) from the heights corresponding to the centers of clusters. The priors $\pi_T^{(0)}$ and $\pi_B^{(0)}$ are initialized to the relative sizes of clusters and $\sigma^{(0)}$ is the population value of the standard deviation.

In the *estimation* (E) step, a *variational distribution* $Q_p^{(t)}(z)$ is built for each pixel $p \in P_e$ using $\theta^{(t)}$:

$$Q_p^{(t)}(z) = \frac{\pi_z^{(t)} \mathcal{N}\left(H(p); \mu_z^{(t)}, (\sigma^{(t)})^2\right)}{\sum\limits_{z' \in \{T,B\}} \pi_{z'}^{(t)} \mathcal{N}\left(H(p); \mu_{z'}^{(t)}, (\sigma^{(t)})^2\right)} \tag{6}$$

In the *maximization* (M) step, the variational distributions $Q_p^{(t)}(z)$ are used to estimate the new values of parameters $\theta^{(t+1)}$. The thickness of the layer $\Delta_\mu^{(t+1)}$ is estimated at first:

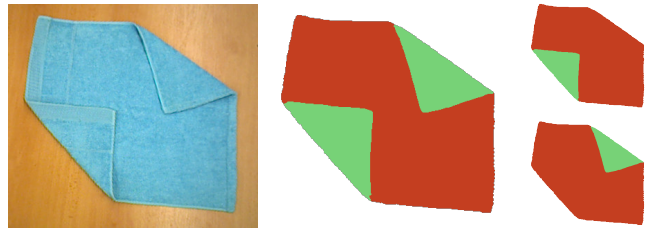
$$\Delta_\mu^{(t+1)} = \frac{\sum\limits_{p \in P_e} \left(2Q_p^{(t)}(T) + Q_p^{(t)}(B)\right) H(p)}{\sum\limits_{p \in P_e} 4Q_p^{(t)}(T) + Q_p^{(t)}(B)} \tag{7}$$

Using (5) to compute $\mu_T^{(t+1)}$, $\mu_B^{(t+1)}$, the variance and priors are estimated as in the standard EM for GMM [29]:

$$(\sigma^{(t+1)})^2 = \frac{1}{|P_e|} \sum_{p \in P_e} \sum_{z \in \{T,B\}} Q_p^{(t)}(z) \left(H(p) - \mu_z^{(t+1)}\right)^2 \tag{8}$$

$$\pi_z^{(t+1)} = \frac{1}{|P_e|} \sum_{p \in P_e} Q_p^{(t)}(z) \tag{9}$$

The EM algorithm is guaranteed to converge to a *local optimum* [28], which takes 30–50 iterations in our case. Fig. 5 shows an example of the final GMM estimated from the height map H shown in Fig. 3b.



(a) Input image (b) Detected layers (c) Single fold

Fig. 6: a) Twice folded garment with top layers not overlapping. b) Result of layers detection. Both top folded layers (green) were detected. c) Two modified labelings where only a single top layer is forced.

(a) Candidate axes (b) True axis (c) False axes

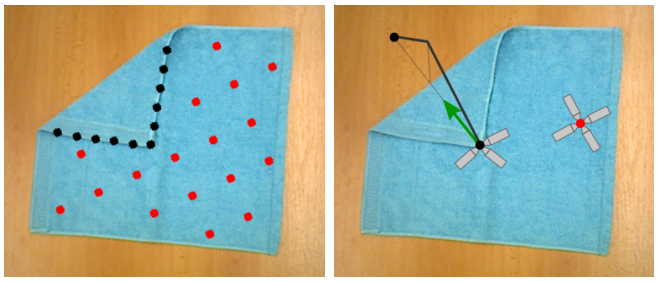
Fig. 7: a) Outer contour of the top layer (green) approximated with 4 candidates for folding axis (various colors). b) Unfolding over the true axis (cyan). c) Unfolding over the false candidate axes leads to overlap (red) or complex unfolded contour (yellow).

V. Estimation of the Folding Axis

Once the layers detection is finished, each pixel $p \in P$ is assigned either to the top or bottom layer. As stated in Sec. I, we allow the garment to be *folded multiple times*, as long as the top layers do not overlap (Fig. 6a). Each top layer then forms a *connected component* in the computed labeling (green components in Fig. 6b). The top layer to be unfolded in the next step is chosen in a greedy manner. A modified labeling is constructed for each component, in which only the pixels from that component are labeled as top and all other pixels are forced to be bottom (Fig. 6c). The cost of each modified labeling with respect to the energy function (1) is computed and the minimum is chosen for unfolding.

As stated in Sec. I, the *folding axis* forms an *approximate segment* on the outer contour of the top layer (Fig. 1a). In order to estimate the folding axis, the outer contour of the top layer is approximated with a *polyline* [31]. If the polyline consists of a single segment (Fig. 1a), the axis is estimated from it. Otherwise, each segment forms a candidate for the axis (variously colored segments in Fig. 7a). To estimate the folding axis from the segment, a line is fitted to the contour points adjacent to the segment by a robust *M-estimator* [32].

The detected top layer (green polygon in Fig. 7a) is then unfolded virtually by *reflecting* it over each candidate axis (Fig. 7b, 7c). The candidate axes, for which the reflected top layer overlaps the bottom layer, are rejected from further processing (red unfolded layer in Fig. 7c). Otherwise, the contour of the unfolded garment is estimated for the candidate axis (cyan contour in Fig. 7b and yellow one in Fig. 7c). Since the true shape of the garment below the top layer is not known, the contour of the bottom layer is simply connected



(a) Grasping and holding candidates (b) Unfolding path

Fig. 8: a) The grasping candidates on the inner contour of the top layer (black dots) and holding candidates on the surface of the bottom layer (red dots). b) The unfolding path (black arrow) for the most prioritized grasping point and for different orientations of the grippers. The sliding direction (green arrow) is same for all orientations.

(a) Testing garments (b) Gripper

Fig. 9: a) Garments used for experimental evaluation, including jacket, jeans, shorts, 2 skirts, 2 sweaters, sweatshirt, towel and 4 T-shirts of various sleeve lengths. b) Gripper designed for grasping of textiles and ASUS Xtion sensor.

to the contour of the unfolded top layer with two straight segments. The candidate axis giving the shortest unfolded contour is chosen as the true folding axis (cyan contour in Fig. 7b is shorter than yellow one in Fig. 7c).

VI. Unfolding

The robotic unfolding uses two cooperated arms. The first arm grasps the inner boundary of the top layer and follows a *triangular unfolding path* afterwards. The second arm holds the bottom layer to prevent the garment from slipping during unfolding. Several grasping and holding position candidates are sampled to increase the robustness of the method, which is affected mainly by the robot kinematic restrictions.

The *grasping* position candidates are sampled uniformly on the inner boundary of the top layer (Fig. 8a). Several orientations of the gripper, limited by shape of the top layer and by the gripper mechanics, are sampled for each candidate grasping point to find such configuration that allows the gripper to slide under the top layer while grasping it.

To prevent the garment from slipping while sliding the gripper under the top layer, the second arm pushes the garment against the table. The *holding* position candidates are sampled uniformly on the bottom layer (Fig. 8a), with the orientations allowed by the gripper mechanics.

The *planning algorithm* repeatedly selects a pair of grasping and holding candidates and tries to follow the triangular unfolding path (Fig. 8b). If the unfolding path is feasible for the robot, the planning ends. Otherwise, the next grasping and holding pair is selected and checked. The grasping positions are selected based on the priority queue, with higher priority given to the positions further away from the folding axis. The grasping orientations and holding positions are checked in an arbitrary order for each grasping position.

VII. Experimental evaluation

A. Dataset of folded garments

We have acquired a *dataset* of folded garments that we are making publicly available[1]. It contains manually annotated images and depth maps of garments posed in various folded configurations. To the best of our knowledge, it is the first such dataset available. The dataset contains 13 garments of 8 categories (Fig. 9a). They are made of different *materials*, e.g. cotton, polyester, denim or leather. Therefore they vary significantly in thickness, stiffness or friction. They are also variously *colored*. Each garment was placed on the table and posed into 15 different folded configurations, which gives 195 data items in total.

The images and depth maps were acquired by the ASUS Xtion sensor. They have the same resolution 640×480 pixels. The sensor was calibrated properly. Therefore the images and depths are *registered*. The acquired data were *annotated* manually by drawing a polyline between the folded layers observed in the image and also by denoting the folding axis.

B. Testbed description

The real world experiments were performed on our dual-arm robot that was developed by the consortium of CloPeMa (Clothes Perception and Manipulation) project[2]. The robot is composed mainly of *standard industrial components*, including two Motoman MA1400 arms mounted to R750 turntable and controlled by two DX100 units. The left arm is attached the custom-made *jaw-like gripper* [33] designed specifically for grasping of textiles (Fig. 9b). Its wrist is mounted the ASUS Xtion sensor that is used for perception.

C. Folds detection

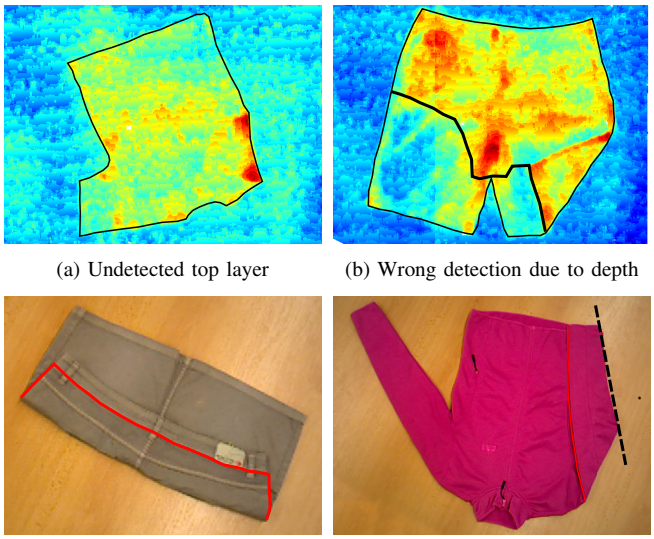
The dataset was used to evaluate the methods for layers detection (Sec. IV) and folding axis estimation (Sec. V). The results are summarized in Tab. I. The first column states the type of the garment and eventually the number of items of that type. The second column presents the ratios of the correctly recognized folded configurations. The next two columns analyze the counts of failures caused either by the layers detection or the folding axis estimation. The overall success rate is 87 %. It does not vary significantly for various garments, which proves the generality of the proposed method. The layers detection mechanism tends to be more error-prone, having 11 % failure rate.

[1] Annotated dataset of folded garments, detailed experimental results and videos: `http://cmp.felk.cvut.cz/~striajan/iros2017`

[2] CloPeMa project: `http://clopema.eu`

Garment	Success	Failure		Displac. [mm]	
		Layers	Axis	Mean	Stdev.
Jacket	14 / 15	1	0	6.4	8.6
Jeans	12 / 15	3	0	3.4	4.5
Shorts	14 / 15	1	0	3.5	3.4
Skirt (2)	25 / 30	5	0	4.1	8.9
Sweater (2)	26 / 30	2	2	4.2	4.8
Sweatshirt	14 / 15	0	1	2.7	3.2
Towel	14 / 15	1	0	3.4	2.4
T-shirt (4)	51 / 60	9	0	4.2	3.8
Total	170 / 195 87 %	22 11 %	3 2 %	4.0	5.2

TABLE I: Performance evaluation of layers detection and folding axis estimation on various garments included in the published dataset.



(a) Undetected top layer (b) Wrong detection due to depth

(c) Wrong detection due to edges (d) Wrong folding axis

Fig. 10: Analysis of observed failures: a) All pixels assigned to a single layer. b) Wrongly detected layers due to misleading depth information or c) misleading edges in the image. d) Wrongly selected folding axis.

The last two columns of Tab. I provide a quantitative evaluation of the layers detection. For each of 173 configurations, in which the layers were detected correctly, we compute the displacement of the detected and annotated boundary between the layers. Namely, for each point of the correctly detected boundary, we find its closest point on the manually annotated boundary. The usual displacement is several millimeters which is more than sufficient for reliable grasping, considering the size of the gripper. The detailed results for all configurations are provided on our website[1].

The observed failures can be split into three categories. The first is the failure in the layers detection when all pixels are assigned to a single layer. It appears when the heights of the layers differ insignificantly and their boundary is not clearly visible in the image (Fig. 10a). The second category corresponds to the wrong detection of the layers. It can be caused by misleading depth information (Fig. 10c) or by presence of strong image edges not adjacent to the boundary between the layers (Fig. 10c). The last and rarest type of failure is a wrongly chosen folding axis. It is caused by the employed heuristics on choosing the candidate axis with the shortest unfolded contour (Fig. 10d).

Garment	Success	Reason of failure		
		Detection	Planning	Execution
Shorts	3 / 5	1	1	0
Sweatshirt	4 / 5	1	0	0
Towel	5 / 5	0	0	0
T-shirt	5 / 5	0	0	0
Total	17 / 20	2	1	0

TABLE II: Performance evaluation and failure analysis of unfolding.

The proposed perception method was tested on a notebook with Intel i7-3740QM 2.7 GHz processor and 8 GB memory. The preprocessing stage (Sec. III) takes 3.1 seconds on the average, spent mostly by the segmentation. Time spent by the detection of layers (Sec. IV) varies from 0.4 to 1.8 seconds. It is directly proportional to the size of the observed garment, as it is solved by labeling each pixel. Estimation of the folding axis (Sec. V) takes 0.1 seconds at most.

D. Robotic unfolding

The described robot was used to test the proposed unfolding procedure (Sec. VI). The manipulation performance is affected by two factors. First, the space of positions reachable by the robot is limited. We are able to unfold rather small garments placed close to the robot. Second, only the left arm is equipped with the griper suitable for grasping of garments (Sec. VII-B). We are therefore able to unfold only such configurations, where the folding axis is located on the left side (Fig. 8a). This issue could be solved by equipping the right arm with the specialized griper as well and trying to plan the manipulation for both possible combinations of the grasping and holding arm. Since both limiting factors are rather technical than methodological, we tend to place the garments into the suitable configurations in our experiments.

Tab. II summarizes the experimental results. Each garment was placed into 5 different folded configurations. All trials for each garment were performed in a continuous series and video-recorded[1]. The garments were unfolded successfully in 17 attempts out of 20. Two failures were caused by the wrong detection of folds. In one case, the garment was placed in such configuration that it was impossible to plan its unfolding. The robotic manipulation itself was always successful in the experiment.

Selection of the grasping and holding position and planning of the folding trajectory takes 1–15 seconds, depending strongly on the ranking of the successfully planned candidate in the priority queue (Fig. 8a). The unfolding manipulation takes usually 40–50 seconds, with the robot starting and finishing in the initial position and moving rather slowly due to the safety reasons[1].

VIII. Conclusion

We proposed a novel method for recognizing the configuration of an unknown garment that was folded once or more times over an unknown folding axis. The algorithm is based on the detection of the bottom and top layer. The detection of the layers is formulated as an optimization problem in which information coming from camera and range sensor is fused effectively. The correct folding axis is selected by unfolding

the garment virtually over all candidate folding axes and checking the unfolded states. Once the garment configuration is known, it is unfolded by a cooperative manipulation with two robotic arms. The performed experiments show a promising 85–87 % success rate on a variety of garments.

The proposed method does not utilize any *model* of the garment shape and therefore it is usable for all garment types without modifications. Such model could, however, improve the robustness for highly textured garments made of thin textile, where the edge and depth information is unreliable. Another possible improvement is to *generalize* the method to deal with several overlapping folded layers. It is, in principle, possible to extend the labeling task to three or more layers. The emerged task is NP hard, but it can be solved effectively using the available approximation methods.

Acknowledgment

The authors of this work were supported by the European Regional Development Fund under the project IMPACT no. CZ.02.1.01/0.0/0.0/15_003/0000468, by the European Commission under the projects TRADR no. FP7-ICT-609763 and RadioRoSo (part of Echord++ no. FP7-ICT-601116), and by the Grant Agency of the Czech Technical University in Prague under the projects no. SGS15/204/OHK3/3T/13 and SGS15/203/OHK3/3T/13.

References

[1] D. Triantafyllou, I. Mariolis, A. Kargakos, S. Malassiotis, and N. Aspragathos, "A geometric approach to robotic unfolding of garments," *Robotics and Autonomous Systems*, vol. 75, pp. 233–243, 2016.

[2] I. Mariolis and S. Malassiotis, "Modelling folded garments by fitting foldable templates," *Machine Vision Applications*, vol. 26, no. 4, pp. 549–560, 2015.

[3] D. Estevez, J. G. Victores, S. Morante, and C. Balaguer, "Towards robotic garment folding: A vision approach for fold detection," in *Proc. Int. Conf. on Autonomous Robot Systems and Competitions (ICARSC)*, 2016, pp. 188–192.

[4] 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. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2010, pp. 2308–2315.

[5] B. Willimon, S. Birchfield, and I. Walker, "Classification of clothing using interactive perception," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2011, pp. 1862–1868.

[6] A. Ramisa, G. Alenyà, F. Moreno-Noguer, and C. Torras, "Using depth and appearance features for informed robot grasping of highly wrinkled clothes," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2012, pp. 1703–1708.

[7] M. Cusumano-Towner, A. Singh, S. Miller, J. F. O'Brien, and P. Abbeel, "Bringing clothing into desired configurations with limited perception," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2011, pp. 3893–3900.

[8] Y. Kita, T. Ueshiba, E. S. Neo, and N. Kita, "Clothes state recognition using 3d observed data," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2009, pp. 1220–1225.

[9] Y. Li, Y. Wang, M. Case, S.-F. Chang, and P. K. Allen, "Real-time pose estimation of deformable objects using a volumetric approach," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2014, pp. 1046–1052.

[10] A. Doumanoglou, A. Kargakos, T.-K. Kim, and S. Malassiotis, "Autonomous active recognition and unfolding of clothes using random decision forests and probabilistic planning," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2014, pp. 987–993.

[11] A. Doumanoglou, T.-K. Kim, X. Zhao, and S. Malassiotis, "Active random forests: An application to autonomous unfolding of clothes," in *Proc. European Conf. on Computer Vision (ECCV)*, 2014, pp. 644–658.

[12] I. Mariolis, G. Peleka, A. Kargakos, and S. Malassiotis, "Pose and category recognition of highly deformable objects using deep learning," in *Proc. Int. Conf. on Advanced Robotics (ICAR)*, 2015, pp. 655–662.

[13] J. van den Berg, S. Miller, K. Goldberg, and P. Abbeel, "Gravity-based robotic cloth folding," in *Proc. Int. Workshop on Algorithmic Foundations of Robotics (WAFR)*, 2011, pp. 409–424.

[14] J. Stria, D. Průša, and V. Hlaváč, "Polygonal models for clothing," in *Proc. Conf. Towards Autonomous Robotic Systems (TAROS)*, 2014, pp. 173–184.

[15] J. Stria, D. Průša, V. Hlaváč, L. Wagner, V. Petrík, P. Krsek, and V. Smutný, "Garment perception and its folding using a dual-arm robot," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2014, pp. 64–67.

[16] V. Petrík, V. Smutný, P. Krsek, and V. Hlaváč, "Robotic garment folding: Precision improvement and workspace enlargement," in *Proc. Conf. Towards Autonomous Robotic Systems (TAROS)*, 2015, pp. 204–215.

[17] Y. Li, Y. Yue, D. Xu, E. Grinspun, and P. K. Allen, "Folding deformable objects using predictive simulation and trajectory optimization," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2015, pp. 6000–6006.

[18] V. Petrík, V. Smutný, P. Krsek, and V. Hlaváč, "Physics-based model of a rectangular garment for robotic folding," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2016, pp. 951–956.

[19] L. Sun, G. Aragon-Camarasa, S. Rogers, and J. P. Siebert, "Accurate garment surface analysis using an active stereo robot head with application to dual-arm flattening," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2015, pp. 185–192.

[20] L. Sun, G. A. Camarasa, A. Khan, S. Rogers, and P. Siebert, "A precise method for cloth configuration parsing applied to single-arm flattening," *Int. J. of Advanced Robotic Systems*, vol. 13, no. 2, p. 70, 2016.

[21] Y. Li, D. Xu, Y. Yue, Y. Wang, S.-F. Chang, E. Grinspun, and P. K. Allen, "Regrasping and unfolding of garments using predictive thin shell modeling," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2015, pp. 1382–1388.

[22] A. Doumanoglou, J. Stria, G. Peleka, I. Mariolis, V. Petrík, A. Kargakos, L. Wagner, V. Hlaváč, T.-K. Kim, and S. Malassiotis, "Folding clothes autonomously: A complete pipeline," *IEEE Trans. on Robotics*, vol. 32, no. 6, pp. 1461–1478, 2016.

[23] C. Rother, V. Kolmogorov, and A. Blake, "GrabCut – interactive foreground extraction using iterated graph cuts," *ACM Trans. on Graphics*, vol. 23, no. 3, pp. 309–314, 2004.

[24] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, *Digital Image Processing Using MATLAB, 2nd ed.* Gatesmark, 2009.

[25] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[26] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, 2004.

[27] V. Kolmogorov and R. Zabin, "What energy functions can be minimized via graph cuts?" *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 147–159, 2004.

[28] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. of the Royal Statistical Society*, pp. 1–38, 1977.

[29] J. A. Bilmes *et al.*, "A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models," *Int. Comput. Sci. Inst.*, vol. 4, no. 510, p. 126, 1998.

[30] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification, 2nd ed.* Wiley, 2000.

[31] J.-C. Perez and E. Vidal, "Optimum polygonal approximation of digitized curves," *Pattern Recognition Letters*, vol. 15, no. 8, pp. 743–750, 1994.

[32] P. J. Huber, "Robust regression: asymptotics, conjectures and Monte Carlo," *The Annals of Statistics*, pp. 799–821, 1973.

[33] T.-H.-L. Le, M. Jilich, A. Landini, M. Zoppi, D. Zlatanov, and R. Molfino, "On the development of a specialized flexible gripper for garment handling," *Int. J. of Automation and Control Engineering*, vol. 1, no. 2, pp. 255–259, 2013.