Petrik, Vladimir; Kyrki, Ville

Feedback-based Fabric Strip Folding

Feedback-based Fabric Strip Folding

Vladimír Petrík and Ville Kyrki, *Senior Member, IEEE*

Abstract— Accurate manipulation of a deformable body such as a piece of fabric is difficult because of its many degrees of freedom and unobservable properties affecting its dynamics. To alleviate these challenges, we propose the application of feedback-based control to robotic fabric strip folding. The feedback is computed from the low dimensional state extracted from a camera image. We trained the controller using reinforcement learning in simulation which was calibrated to cover the real fabric strip behaviors. The proposed feedback-based folding was experimentally compared to two state-of-the-art folding methods and our method outperformed both of them in terms of accuracy.

I. INTRODUCTION

Feed-forward robotic manipulation of an object requires accurate prediction of the object state evolution under the given robot actions. This prediction is difficult especially for deformable/soft objects because they have a high degree of freedom and the deformation usually depends on many directly unobservable object properties. If reliable prediction is not available, feedback-based control can be used to increase accuracy.

Robotic strip folding visualized in Fig. 1 is an example of soft material manipulation. In the folding, a robot grasps one edge of the strip and follows the folding path. The shape of the folding path affects the accuracy of the folding mainly for two reasons: (1) the gripper can cause the slipping of the strip on the desk, (2) the relative position of the upper and lower layers is hard to change after first touch because of friction.

Several approaches for designing folding paths have been proposed in the literature. Two of them [1], [2] are based on unrealistic assumptions to design paths which are independent of material properties. They are easy to compute but ignoring the material properties leads to inaccurate folds. Other approaches [3], [4], [5] take the material properties into account but assume the properties are known in advance or rely on special hardware to estimate the properties in a course of folding and are computationally expensive [6].

In this paper, we propose to use a feedback-based controller to perform the folding in a closed loop. The input to the controller is a low dimensional state computed from an image obtained from a calibrated camera. The controller policy is determined by reinforcement learning (RL) in simulation using domain randomization to account for unobservable properties. The controller is trained to minimize the misalignment of layers after the touch. The controller is

Authors are with Aalto University, Finland {vladimir.petrik, ville.kyrki}@aalto.fi.
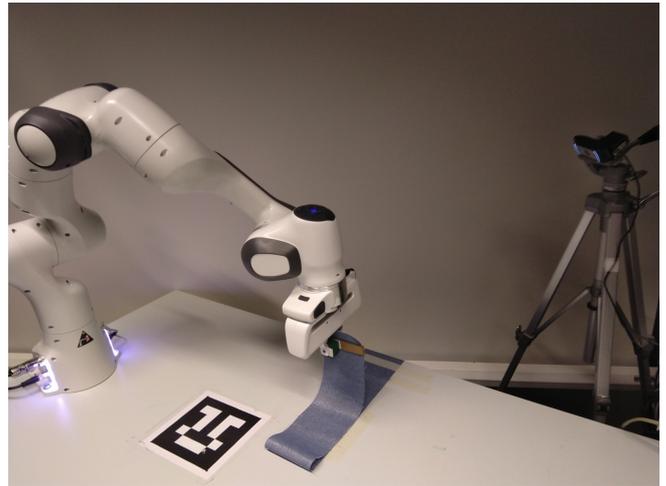
Fig. 1: The *Franka Emika Panda* robot used in the experiment to perform folding of a strip. The folding proposed in this paper is based on feedback computed from the images obtained from the calibrated camera.

experimentally evaluated by real robotic folding for various strip materials.

The contributions of the paper are: (1) a vision feedback-based controller for fabric strip folding, (2) calibration of a simulation model to match the behavior of real fabric strips, (3) the controller parameters training in simulation by RL algorithm, (4) experimental comparison of the controller performance w.r.t. the state-of-the-art folding methods.

II. STATE-OF-THE-ART

The robotic folding of garments is usually decomposed into individual folds [7], [8]. The accuracy of each fold is crucial for obtaining an accurately folded garment in the end because the errors accumulate with each fold. We study the accuracy of the single fold when applied to one-dimensional single layer fabric strips.

Triangular paths have been proposed for folding [1], [7], [8]. This path was derived based on assumptions of infinitely flexible strip and an infinite friction between the desk and the garment. A violation of these assumptions results in an inaccurate fold as was experimentally shown e.g. in [4] and theoretically analyzed for continuum model of fabric in [9]. Another unrealistic assumption of a rigid body with flexibility only in the folding line was used in [2]. This assumption results in a circular path which is complementary to the triangular path. The triangular and circular paths do not depend on the fabric material and are easy to compute but provide inaccurate alignment in the end.

More complex methods for the folding path design are specific to the fabric material and assume that the material properties have been identified prior to folding. Several methods have been proposed, differing in the type of used model and in the simulation utilized to find the folding path. A mass-spring model in *Autodesk Maya* simulation was proposed in [3], a mechanical physics-based model derived from the *Euler-Bernoulli* beam theory and boundary value problem solver in [4], and finite element simulation of a model derived from the *Kirchoff-Love* shell theory in [5]. All these methods obtained superior accuracy compared to the triangular and circular paths but they require the material properties to be known before the folding starts.

The previous folding path generation methods do not use feedback in the folding process. Recent work [6] uses a laser range finder to measure the shape of the strip while folding. The parameters of a finite-element model [5] are then perturbed until the model shape matches the measurement. The shape fitting is performed after each robot motion to avoid the strip slipping on the desk and to improve the estimation while lifting the strip upward, which requires solving the finite element simulation many times taking several minutes for a single fit. After the strip is in the upper position the last estimate is used to design the rest of the folding path which is then followed in a feed-forward manner. The method proposed in this paper differs from [6] by: (1) we use a standard camera instead of specialized accurate laser range finder which simplifies the calibration process, (2) we use the feedback during the whole motion, (3) our method produces a controller which computes the action for the robot quickly and can be used in real time.

Recently, several learning-based approaches to robotic folding were proposed, e.g. [10], [11], [12], [13], [14]. However, these methods have been tested only on a single fabric material and the accuracy has not been measured.

III. FABRIC STRIP MODEL

To learn a folding policy in simulation we require models of strips with realistic dynamics. In this section, we first describe the phenomena of real fabric strips which affect the folding accuracy. We then propose a model for simulating these phenomena. The treatment is restricted to single layer strips.

A. Real Fabric Strip Behavior

A shape of a folded fabric strip is visualized in Fig. 2. The maximum height of the folded strip (denoted h hereinafter) depends on the fabric material. Height h was used in [4] to estimate *weight-to-stiffness-ratio* for continuum model of the strip. It was then used to compute the folding path for the particular strip material. Therefore, the folding path depends on h.

The accuracy of strip folding is affected mainly by the strip slipping and by the position of the first touch of the strip with itself. The slipping occurs if the horizontal force created by the gripper is larger than the frictional force. Therefore, it can be avoided by minimizing horizontal force as shown

Fig. 2: The folded shape of the strip. The maximum height of the folded strip h depends on the fabric material.

in [4]. Alternatively, the folding path can be adjusted after the slipping occurs if the slipping is measured online.

The first touch of fabric with itself is shown in Fig. 4. The position of the first touch measures the alignment of individual layers. The friction coefficient between the parts of the fabric is usually higher than the friction coefficient between the desk and fabric. Therefore, after the layers are in contact it is difficult to change the position of the touch without lifting the strip upward and breaking contact.

Manipulating a strip slowly results in a statically stable equilibrium of the strip in most of the work space. However, there is a static instability immediately before the touch occurs as documented in [9]. This instability results from a dynamic snap-through behavior known from mechanics. During the dynamic motion, there is another property of fabric material which affects the position of the first touch and therefore the accuracy. This property was not modeled in the continuum models used for the folding path generation in [4]. The effect of this property on the first touch position for various gripper paths was measured experimentally in [9]. We plot these data in Fig. 3.

B. MuJoCo Simulation

We use a MuJoCo [15] environment to simulate fabric strips. The strip is modeled as a one-dimensional sequence of
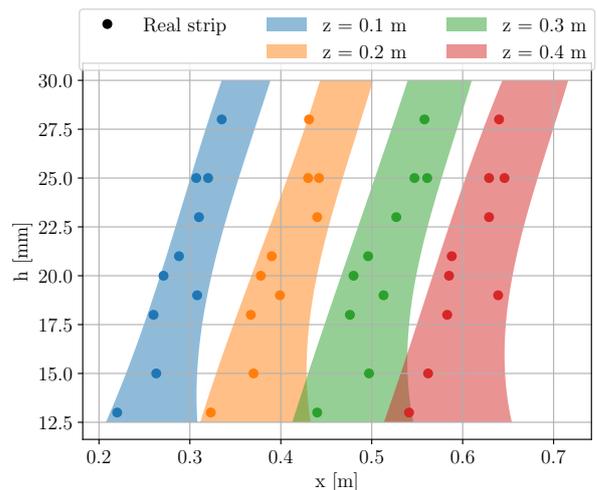
Fig. 3: The position of the gripper x for various gripper heights z at which the first touch of fabric layers occurred. In the experiment, the x-position was changing continuously until the touch occurred. Individual colors correspond to different gripper heights. The shaded areas correspond to values obtained from the MuJoCo simulation by continuously changing strip properties. The maximum height of the folded strip is denoted by h.

equally spaced and weighted spheres connected by rotational joints. The joints allow motion of the strip in xz-plane only.

We used two parameters of joints which affect the behavior of simulated strip: stiffness and damping. Other parameters were fixed to constant values and the number of the spheres was fixed. We assume homogeneity of the fabric material and use the same values for all joints. Continuous change of the stiffness and damping parameters can cover both phenomena: different folded state height as well as different first touch position after passing instability. The coverage of the phenomena is shown in Fig. 3. It can be seen, that our model is able to cover all real measurements. The simulation parameters required to replicate these results are shown in Appendix I.

Slipping is not measured in simulation directly. Instead, we fix the non-grasped edge of the strip and measure the force required to hold the edge fixed. This force is then minimized to avoid slipping similarly to [4]. This simplifies the simulation because in addition to gripper force the amount of slipping depends on friction coefficient. Therefore, various friction models would need to be simulated to get the slippage for various real-life conditions. The example of the model states for one set of strip parameters is shown in Fig. 4. It should be noted that the force measurement is only used to learn the control policy in simulation and thus not needed in the physical system.

The current structure of the strip model does not simulate the extension of the strip, i.e. the strip length is constant independent of the stretching force. This is in correspondence with the fact that strips extend negligibly under their own weight as reported in [6]. However, if an extension is required then the model could be enriched by prismatic joints located between the spheres. The stiffness of the joints would control the extensibility of the strip.

IV. Feedback-based Controller

To control the folding we use a feedback-based controller in a form:

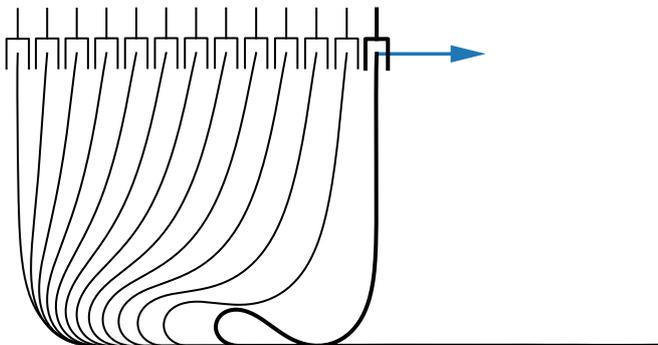$$\varphi = \pi_{\boldsymbol{w}}(\boldsymbol{x}), \qquad (1)$$

Fig. 4: An evolution of strip states caused by the gripper horizontal motion as visualised by the blue arrow. The bold state represents the first state at which the touch of the layers occurs.

Fig. 5: The visualisation of the state which is used to compute the action φ shown by blue arrow. The state consists of x and z position of the grasped edge of the strip (origin of the blue arrow) and of x position of the first touch of the strip with the desk (cross symbol). The black dot visualises the fixed point which is used to prevent slipping in simulation.

where φ is a one-dimensional action which represents angle between the x-axis and relative gripper motion, and $\boldsymbol{x}$ is a three-dimensional state containing the gripper position and x-coordinate of the first contact of fabric with the desk. The state and action are visualised in Fig. 5. The policy $\pi_{\boldsymbol{w}}$ computes the action φ based on the state $\boldsymbol{x}$. The policy is a function approximator parametrized by vector $\boldsymbol{w}$. In our setup, we used a neural network with one hidden layer and 20 hidden neurons to represent the policy. The network uses hyperbolic tangent as an activation function. The weights of the network parametrize the policy. First, we will describe how to find parameters $\boldsymbol{w}$ for a strip with fixed fabric dynamics parameters. Then, we extend the formulation to generalize the policy across varying parameter values.

A. Single Strip Folding

The parameters $\boldsymbol{w}$ are trained in simulation using reinforcement learning. Thus, we maximize total rewards with respect to policy parameters,

$$\boldsymbol{w} = \arg\max_{\boldsymbol{w}} R(\boldsymbol{\theta}), \qquad (2)$$

where $\boldsymbol{\theta}$ represents the strip properties (constant in a single strip folding) and $R(\cdot)$ is the total reward computed from the strip states obtained from the simulation by applying the policy $\pi_{\boldsymbol{w}}(\cdot)$. The total reward can be decomposed into the intermediate reward for i-th step r_i and terminal reward r_f:

$$R(\boldsymbol{\theta}) = \sum_{i=0}^{N} r_i(\boldsymbol{\theta}) + r_f(\boldsymbol{\theta}), \qquad (3)$$

where N is the number of simulation steps. Both the intermediate as well as terminal reward depend on the strip properties $\boldsymbol{\theta}$.

In our scenario, the simulation stops after the touch of layers occurs or after a maximum number of steps H. Note, that N equals to H only in the case when the layers did not touch. The terminal reward is then computed as

$$r_f = \begin{cases} -|d|, & \text{if touch occurred,} \\ -\infty, & \text{otherwise,} \end{cases} \qquad (4)$$

where d is the misalignment of the layers as visualised in Fig. 6. The intermediate rewards represent the force f_x

required to hold the strip in order to prevent slipping:

$$r_i = -a\,\frac{|f_x|}{N}\,, \qquad (5)$$

where a is a scale factor manually selected to give higher importance to the terminal reward.

The infinite value of terminal reward in Eq. (4) can be used if policy initialization ends in the state where layers touch. Otherwise, reward function gradient does not provide information for the optimization of Eq. (2). In our implementation, we used reward shaping to guide the optimization towards the touch by enforcing horizontal gripper motion as visualised in Fig. 4.

For solving the optimization in Eq. (2), we used a modified version of BlackDROPS [16] where we removed the Gaussian process model fitting and optimized the reward function in simulation directly with CMA-ES [17]. The optimized policy is visualized in Fig. 7.

B. Domain Randomization

The previous approach requires the strip properties $\boldsymbol{\theta}$ to be known or directly observable. However, our goal is to find a policy which is independent on the strip properties. One possibility of finding such a policy is to use domain randomization [18], [19]. Domain randomization for our task may be achieved by modifying the optimization (2) into:

$$\boldsymbol{w} = \arg\max_{\boldsymbol{w}} E_{\boldsymbol{\theta}\sim p(\boldsymbol{\theta})}\left[R(\boldsymbol{\theta})\right]\,, \qquad (6)$$

where E stands for the expectation computed over the parameters given by the prior distribution $p(\boldsymbol{\theta})$. The prior distribution encodes the strip properties used to generate shaded areas in Fig. 3. By solving Eq. (6) we obtain the policy which is independent of the strip properties. The resulting policy adapts to different strips as visualized in Fig. 8.
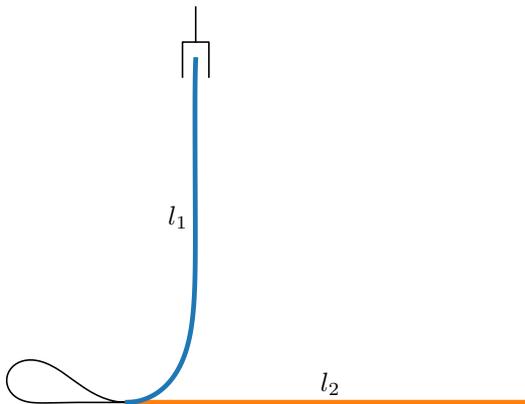

Fig. 6: The displacement of the layers is measured after the strip touches itself. The symbol l_1 denotes the length of the hanging part of the strip (shown in blue). The length of the laying part of the strip measured between the point of layers contact and the fixed end of the strip is denoted by symbol l_2 and shown in orange. The oriented displacement is computed by $d = l_1 - l_2$.
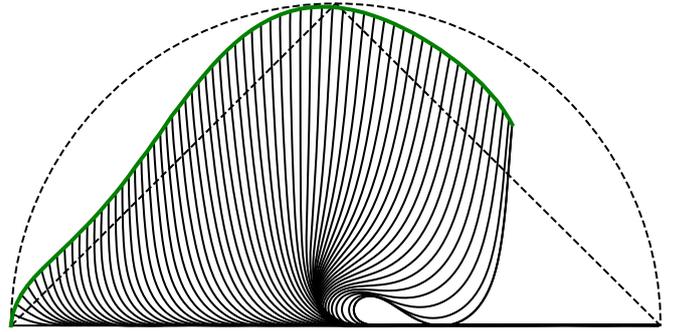

Fig. 7: Evolution of strip state (solid lines) obtained by following the learned control policy. The green curve shows the followed path. The triangular and circular paths (dashed lines) are shown as a reference.

V. EXPERIMENTS

A. Testbed Description

To perform real experiments we used the *Franka Emika Panda* robot shown in Fig. 1 and a camera to measure the state. We estimated a homography [20] between the undistorted image plane and the xz-plane aligned with the edge of the strip. The homography was used to project detected position of the first touch between the strip and the desk into the robot coordinate system. An example of image taken from the camera is shown in Fig. 9.

For the detection of the point at which the strip touches the desk, we performed a line search on the line which corresponds to the strip in the unfolded state. The colors on the line were compared to the desk color to detect the touch position. Because of the camera position, it is guaranteed that no other part of the strip will occlude the line in image space before the point of contact.
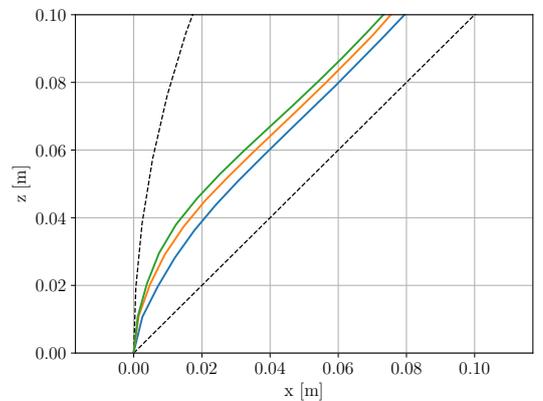

Fig. 8: The policy adapts the folding path based on the feedback in order to minimize the horizontal force. Different colors represent paths obtained for different values of stiffness s.t. blue was obtained for the lowest stiffness and the green for the highest stiffness value.

B. Strip Folding

We measured the misalignment of layers after passing through the instability [9]. The non-grasped edge of the strip was fixed to discard the contribution of slipping to the measurement. The strip length measured between the grasping point and the fixed point was 0.6 m. We started folding from the 10 cm height and gripper orientation was fixed s.t. the gripper pointed downward (Fig. 9). The folding was stopped manually after the touch of layers occurred (i.e. after passing through the instability). In addition to the proposed feedback-based folding, we measured the displacement for triangular and circular paths for ten different fabric materials. The measured displacements are shown in Fig. 10.

The measured oriented displacement for the triangular path is negative for all materials with average displacement of -36.8 mm, such that during the folding the strip touched itself sooner than desired. This is in alignment with predictions made in [9]. If the folding continues after the touch occurs, the upper layer can slide on the bottom layer or the whole strip will slip on the desk depending on the friction coefficients. The displacement for the circular path is positive (average displacement 62.7 mm) for all materials which is also in alignment with theoretical predictions [9].

The feedback-based controller outperformed the triangular and circular paths. The measured displacement was under 20 mm for all materials without knowing the folded strip properties in advance. This is inferior to performance of methods that assume the strip properties are known. For the same fabric materials, the performance obtained in [21] was under 10 mm. On the other hand, our method does not require manual estimation of the properties as it adjusts the path automatically based on the measured feedback.

The displacements in Fig. 10 were measured once per
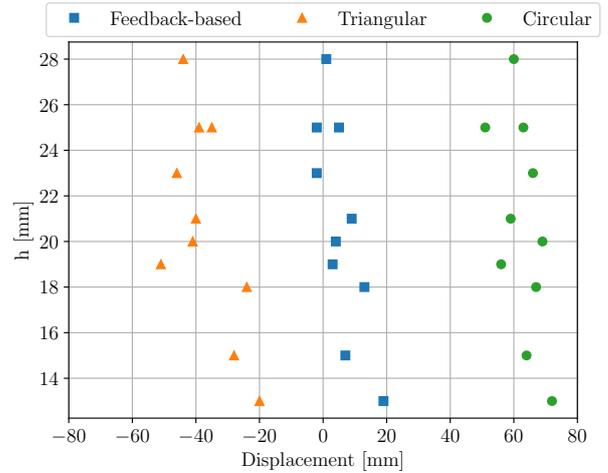
Fig. 10: The measured displacement of the layers after the touch occured. The zero displacement corresponds to aligned layers which is the goal of folding.

strip. We tested the repeatability of the folding process for one selected strip which we folded 5 times per folding method. The differences between individual measurements were maximally 1 mm for all tested methods: the feedback-based, triangular, and circular. The measured differences are lower than the expected accuracy of the whole testbed. Therefore, we decided to perform one experiment per strip only.

C. Simulated Displacement

To analyze why the feedback-based folding does not perform as well as methods tuned to a particular fabric, we performed the folding in simulation for various strip parameters and measured the displacement. The obtained results compared to real measurements are shown in Fig. 11. It can be seen that zero displacements were not achieved for all strips in the simulation. This can be caused by the policy representation not being sufficiently complex or by the measured strip state not being sufficiently informative. Analysis of this issue is the subject of our future work. All except one real measurements lie in the displacement obtained in simulation. That indicates that the range of simulated properties needs to be expanded during the training to cover the behavior of that strip.

VI. CONCLUSIONS

We proposed a feedback-based approach for robotic strip folding. In our approach, the low dimensional feedback is measured in an image obtained from a calibrated camera. We optimized a controller in the calibrated simulation using reinforcement learning. The displacement of strip layers and the horizontal force was used in a reward function. We experimentally verified the controller in real robotic folding for 10 different fabric strips. Our feedback-based approach outperformed existing approaches that are agnostic to fabric type.
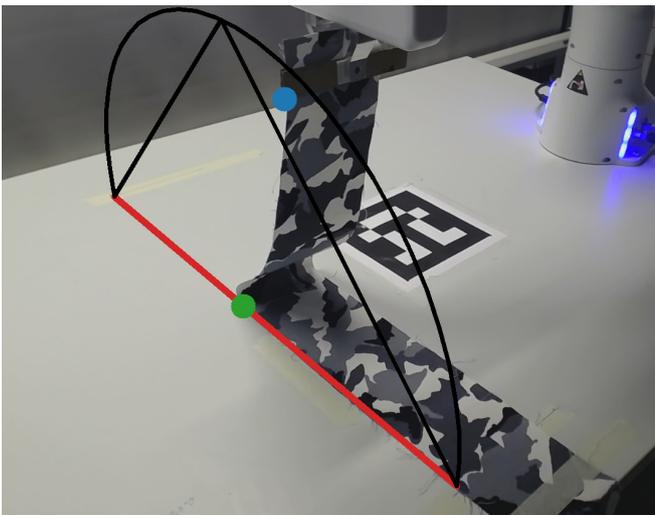

Fig. 9: Image of the fabric used for the feedback control. The red line is used for line search of the first touch between the strip and the desk. The first touch is shown by the green dot. The blue dot shows the projected gripper position. The black curves are projected shapes of triangular/circular paths.
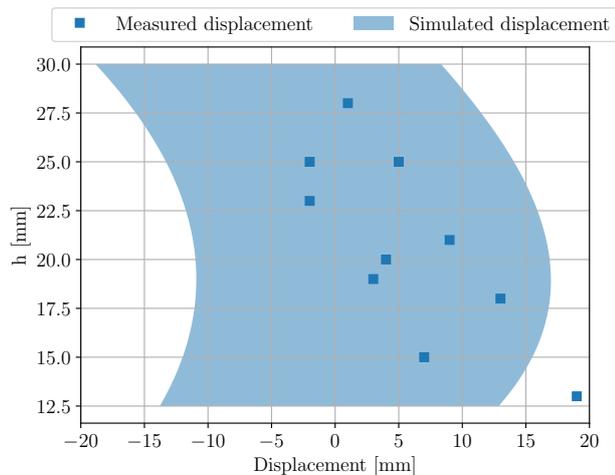
Fig. 11: The comparison of measured displacement (squares) for fabric strips with the simulated displacement for the whole range of simulated strip properties (shaded area).

In comparison to the state-of-the-art method for strip property estimation [6], our approach is based on easily available monocular camera instead of on an accurate laser range finder. Using the standard camera simplifies the calibration of the testbed and we showed that it can be used to improve the accuracy of folding. Moreover, the controller is capable of online computation while the state-of-the-art method requires solving an optimization problem in the finite element simulation which takes several minutes.

The measured absolute displacements for the feedback-based folding were under 20 mm for all tested strips. This performance was obtained without transfer learning by training in simulation using domain randomization technique. In our future work, we will study how the transfer learning and different policy/state representations could be used to reduce the displacement.

Appendix I

The following parameters were used in MuJoCo simulation [15] to simulate strips: distance between spheres: 2 mm, spheres diameter: 1 mm, spheres mass: 3.3 g, strip length: 0.6 m, simulation step: 5 ms. We used Newton solver with maximum number of 100 iterations. The contacts computation between spheres were dissabled. The joints stiffness value k lies in range: $k \in \langle 0.02, 0.3 \rangle$. The minimum damping is stiffness dependent and computed by: $b_{\min} = 3 \cdot 10^{-3} k + 3.5 \cdot 10^{-4}$. The maximum damping is 50-times larger than minimum damping for each stiffness value.

References

[1] J. van den Berg, S. Miller, K. Y. Goldberg, and P. Abbeel, "Gravity-based robotic cloth folding." in *Int. Workshop on the Algorithmic Foundations of Robotics (WAFR)*, 2010, pp. 409–424.

[2] V. Petrík, V. Smutný, P. Krsek, and V. Hlaváč, "Robotic Garment Folding: Precision Improvement and Workspace Enlargement," in *Annu. Conf. Towards Autonomous Robotic Systems (TAROS)*, Liverpool, United Kingdom, 2015, pp. 204–215.

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

[4] V. Petrík, V. Smutný, P. Krsek, and V. Hlaváč, "Physics-Based Model of Rectangular Garment for Robotic Folding," in *Proc. Int. Conf. on Intelligent Robots and Systems (IROS)*, Daejeon, Korea, 2016, pp. 951–956.

[5] V. Petrík, V. Smutný, P. Krsek, and V. Hlaváč, "Single arm robotic garment folding path generation," *Advanced Robotics*, vol. 31, no. 24-24, pp. 1325–1337, 2017. [Online]. Available: http://dx.doi.org/10.1080/01691864.2017.1367325

[6] V. Petrík, J. Cmíral, V. Smutný, P. Krsek, and V. Hlaváč, "Automatic Material Properties Estimation for the Physics-Based Robotic Garment Folding," in *Proc. Int. Conf. on Robotics and Automation (ICRA)*, Brisbane, Australia, 2018.

[7] S. Miller, J. van den Berg, M. Fritz, T. Darrell, K. Goldberg, and P. Abbeel, "A geometric approach to robotic laundry folding," *Int. J. Robotics Research (IJRR)*, vol. 31, no. 2, pp. 249–267, 2012.

[8] 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 Transactions on Robotics*, vol. 32, no. 6, pp. 1461–1478, 2016.

[9] V. Petrík, V. Smutný, and V. Kyrki, "Static stability of robotic fabric strip folding," *arXiv preprint arXiv:1902.11021*, 2019.

[10] B. Jia, Z. Hu, Z. Pan, D. Manocha, and J. Pan, "Learning-based feedback controller for deformable object manipulation," *arXiv preprint arXiv:1806.09618*, 2018.

[11] J. Matas, S. James, and A. J. Davison, "Sim-to-real reinforcement learning for deformable object manipulation," *arXiv preprint arXiv:1806.07851*, 2018.

[12] B. Sannapaneni, M. Shaswat, and A. N. Kumaar, "Learning from demonstration algorithm for cloth folding manipulator," in *Advances in Computing, Communications and Informatics (ICACCI), 2017 International Conference on*. IEEE, 2017, pp. 1339–1398.

[13] A. Colomé and C. Torras, "Dimensionality reduction for dynamic movement primitives and application to bimanual manipulation of clothes," *IEEE Transactions on Robotics*, 2018.

[14] P.-C. Yang, K. Sasaki, K. Suzuki, K. Kase, S. Sugano, and T. Ogata, "Repeatable folding task by humanoid robot worker using deep learning," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 397–403, 2017.

[15] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 5026–5033.

[16] K. Chatzilygeroudis, R. Rama, R. Kaushik, D. Goepp, V. Vassiliades, and J.-B. Mouret, "Black-Box Data-efficient Policy Search for Robotics," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2017.

[17] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evol. Comput.*, vol. 9, no. 2, pp. 159–195, June 2001. [Online]. Available: http://dx.doi.org/10.1162/106365601750190398

[18] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," *arXiv preprint arXiv:1710.06537*, 2017.

[19] X. Chen, A. Ghadirzadeh, J. Folkesson, and P. Jensfelt, "Deep reinforcement learning to acquire navigation skills for wheel-legged robots in complex environments," *arXiv preprint arXiv:1804.10500*, 2018.

[20] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[21] V. Petrík, V. Smutný, P. Krsek, and V. Hlaváč, "Accuracy of Robotic Elastic Object Manipulation as a Function of Material Properties," in *Int. Workshop on Modelling and Simulation for Autonomous Systems (MESAS)*, Rome, Italy, 2016, pp. 384–395.