Visuotactile Affordances for Cloth Manipulation with Local Control

Neha Sunil[*1], **Shaoxiong Wang**[*1], **Yu She**[2], **Edward Adelson**[1], **Alberto Rodriguez**[1]
[1]Massachusetts Institute of Technology [2]Purdue University
<nsunil, wang_sx, albertor>@mit.edu shey@purdue.edu adelson@csail.mit.edu

Abstract: Cloth in the real world is often crumpled, self-occluded, or folded in on itself such that key regions, such as corners, are not directly graspable, making manipulation difficult. We propose a system that leverages visual and tactile perception to unfold the cloth via grasping and sliding on edges. By doing so, the robot is able to grasp two adjacent corners, enabling subsequent manipulation tasks like folding or hanging. As components of this system, we develop tactile perception networks that classify whether an edge is grasped and estimate the pose of the edge. We use the edge classification network to supervise a visuotactile edge grasp affordance network that can grasp edges with a 90% success rate. Once an edge is grasped, we demonstrate that the robot can slide along the cloth to the adjacent corner using tactile pose estimation/control in real time.

Keywords: Multi-modal learning, Cloth manipulation, Tactile control

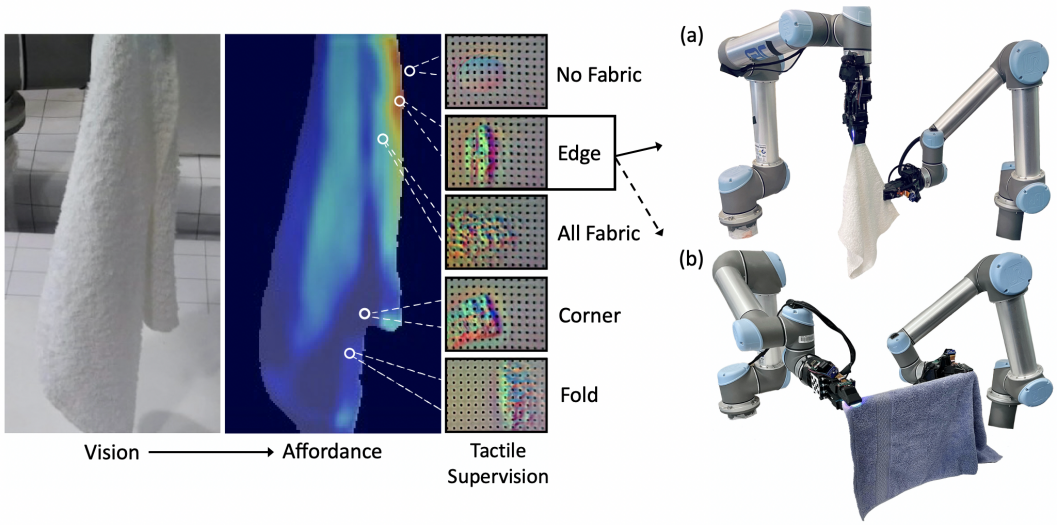

Figure 1: **Visuotactile affordance for sliding.** Using the visual depth image as input, our affordance network scores each pixel in terms of potential edge grasp success. The affordance image shown is trained first in simulation and then fine-tuned with tactile self-supervision from real grasps. In this particular cloth configuration, the adjacent corner is not directly visible or graspable, motivating first grasping an edge and then sliding. We demonstrate tactile sliding in two configurations: (a) sliding down with the cloth in a stable configuration and (b) horizontal sliding suitable for longer towels.

1 Introduction

The robotic manipulation of highly deformable objects is a growing research area due to its applicability in various fields such as assembling cable harnesses in factories, industrial garment manufacturing, and assisted dressing and laundry folding in the home and healthcare industry. Our goal is a

[*]Authors with equal contribution.
This work was done at MIT.

6th Conference on Robot Learning (CoRL 2022), Auckland, New Zealand.

framework that combines visual and tactile sensing of local features (specifically cloth corners and edges in our system) for perception and control of deformable objects. Within this paper, we focus on the manipulation of cloth.

Manipulating deformable objects is difficult due to their continuous and highly-underactuated nature. Applicable high-dimensional models are challenging to identify or simulate. Perception is another key challenge, given that the object is easily occluded by the robot's grippers and even self-occluded by the object itself. These issues pose an inherent difficulty in object representation since full state estimation and modeling is very challenging for real-time manipulation.

Most cloth manipulation studies address these challenges with visual perception [1]. A common first step is unfolding the cloth to make specific features visible by holding it up or flattening it on a table. Then, the robot can perceive and directly grasp features (like corners or collars) for subsequent manipulation tasks. However, these features may still not be directly graspable, as in the corner example in Fig. 1.

In contrast, we want to approach real-time, closed-loop manipulation that is closer to how humans manipulate deformable objects. Instead of directly looking for adjacent corners on a crumpled cloth to pick it up along an edge, we unfold the cloth while sliding until we feel a corner. As shown in Fig. 1, once an edge is successfully grasped, we can slide to the adjacent corner, which was not otherwise graspable. This bottom-up approach of using local features (specifically corners and edges) for perception and control, enables novel manipulation skills.

We leverage tactile perception, along with vision, to localize and manipulate local features. Vision provides global guidance to help with state identification and grasp point localization (in the style of visual affordance-based grasping [2, 3]). We introduce visuotactile affordances where we learn grasp affordances self-supervised with tactile feedback. Tactile feedback from the gripper aids in the classification and estimation of the state of local features. This is especially useful when vision is occluded by the gripper. Finally, we use tactile perception for grasp adjustments, local control, and grasp success confirmation. In this paper, the key technical contributions are:

- **Tactile Perception of Local Features.** We demonstrate a strategy for local feature pose estimation and classification where we show that it is possible to distinguish between edge, corner, fold, all fabric, and no fabric grasps with 92% accuracy (Sec. 4.1).
- **Visuotactile Grasp Affordances.** A grasp localization affordance network pre-trained in simulation and fine-tuned with tactile supervision with a 90% grasp success rate (Sec. 4.2).
- **Tactile Sliding.** Tactile based controllers to slide a gripper along the edge of cloth in two different configurations (Sec. 4.3).

Ultimately, our key experimental contribution is a demonstration of our framework picking a crumpled towel from the ground, grasping a corner using vision, grasping an edge using our visuotactile affordance network, and sliding to an adjacent corner, reaching a configuration which can be used to fold or unfold the towel.

2 Related Work

Perceiving the full state of cloth [4, 5, 6] or modeling its dynamics [7, 8, 9] are challenging and time-consuming approaches. In this section, we instead highlight works that grasp and manipulate local features for cloth folding and unfolding.

Manipulation of Local Features of Deformable Objects. Much of the work on cloth-centric perception for manipulation focuses on corner detection and ridge detection for determining grasp points for future manipulation tasks [10]. Both can be done using computer vision techniques, often using filters (e.g. Harris corner detection [11] and Gabor filters for wrinkles [12]). However, finding other more specific local features often requires first flattening the cloth [13, 14, 15, 16] or hanging it from specific grasp points [17, 18, 19] to prevent self-occlusion of the cloth.

Learning-based perception approaches enable more flexibility with different local features and starting configurations. However, determining ground truth labels is challenging for cloth. Qian et al. (2020) trains a network to segment edges and corners of cloth from a depth image by using a painted cloth as a ground truth label [20]. Simulators can also provide these labels [21, 22], but Sim2Real transfer is an issue, given the uncertain dynamics of cloth. In this work, we first train our edge grasp affordance network in simulation and fine-tune on the robot using tactile perception for supervision.

Most work in fabric manipulation either uses incremental pick-and-place movements against a table [14, 23, 21] or holds up the cloth and regrasps while hanging [17, 18, 19]. However, the sliding skill simplifies the task of finding two adjacent corners in order to fold a piece of fabric, especially when the second corner is not immediately visible or graspable. Sahari et al. (2010) holds up a corner of the fabric and uses gravity to trace downward to find the second corner with low-resolution sensory feedback [24]. Similarly, Yuba et al. (2017) executes an open-loop "pinch and slide" motion along the top edge of a piece of fabric [25]. Zhou et al. (2020) use low resolution tactile sensing to learn a sliding policy in latent space [26]. Our high-resolution tactile sliding allows us to consistently follow longer towels while being robust to disturbances in both stable and horizontal configurations.

In previous work, Yu et al. (2019) slides along cables using GelSight [27] for tactile perception to estimate the pose of the cable in grasp while sliding [28]. Gelsight, and other vision-based tactile sensors, convert touch to vision by using a camera to visualize the deformation of the contact surface. We also approximate shear force by tracking the black markers on the sensor surface as a proxy for friction force while sliding.

Perception of Local Features. Visual grasp affordances are proposed for learning grasping configurations with data-driven methods [3]. The affordance is a metric to predict the grasp success rate under different configurations. Each configuration is represented as an individual pixel and a discrete rotation angle. Affordances can be labeled manually [3], but alternatively can be learned in a self-supervised manner [29, 2, 30, 16]. In these cases, the supervision for affordances comes from either simulation or the visual outcome on a real system, which usually requires the end-effector to move out of the camera frame for visual evaluation. In contrast, our visuotactile approach fine-tunes the affordance trained in simulation under visual occlusion. The learned visuotactile affordance allows for a smooth transition between grasping and sliding.

3 Cloth Manipulation

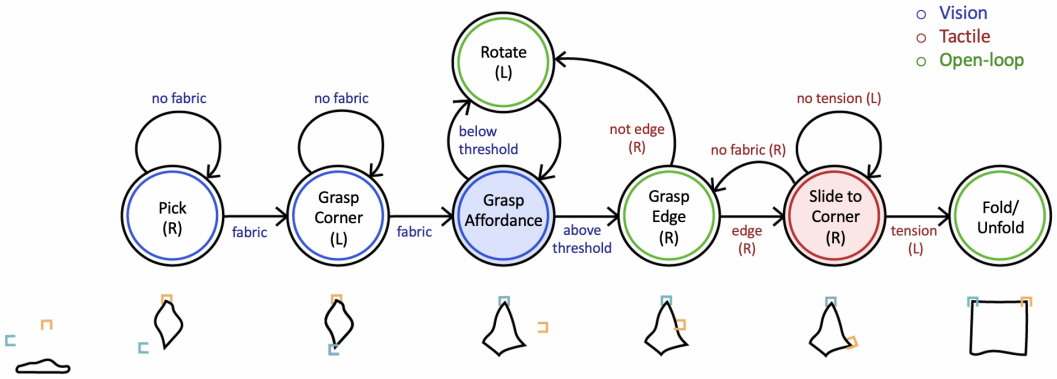

Figure 2: **State machine for cloth manipulation.** Colors in the state machine reflect sensing modalities required for each action and transition conditions between actions. The two key components, visuotactile grasp affordances (highlighted blue) and tactile sliding to corner (highlighted red), are discussed in Sec. 4.2 and Sec. 4.3 respectively. Below, we illustrate the cloth state with respect to each gripper.

In order to fold or unfold the cloth, we first pick the towel up from a surface, grasp a corner, grasp an edge adjacent to that corner, and slide to the adjacent corner (See Fig. 2). Once the robot is grasping two corners, it can complete the folding or unfolding task open-loop.

Tactile Sliding. In the cloth sliding task, two grippers start near each other, grasping the cloth edge. The goal is to traverse as much of the fabric edge as possible or to a specified goal condition (a corner, in this work) without losing the grasp, maintaining an estimate of the edge pose within the sliding gripper.

Sliding along the edge of short towels can be done with regrasps if the cloth is kept in a stable configuration under gravity. We use a simple control strategy for sliding along a cloth edge while hanging. However, a longer towel is more likely to be entangled, susceptible to disturbances, and partially resting on the table. In addition, the adjacent corner is likely self-occluded, making the sliding skill more useful. Thus, we extend the work of Yu et al. (2019) for sliding along cables to longer towels [28]. Here, a moving gripper with a fixed grip on the cloth pulls the cloth through a stationary gripper while attempting to keep the edge visible in the tactile image of the stationary gripper, as shown in Fig. 1. Both sliding strategies are discussed in Sec. 4.3. Perception of the cloth edge is more difficult than for cables. Therefore, we use a supervised data-driven method to estimate the pose of the fabric edge, explained in Sec. 4.1.

Grasp Affordance for Sliding. In order to grasp an edge in a way that allows for sliding, a single layer of fabric should be aligned within the gripper to avoid collisions. We train a visuotactile grasp affordance network that takes the depth image as input and outputs an affordance heatmap, indicating the grasp viability of each point on the image. This network is originally trained in simulation. Then, we fine-tune this network on a robot using the tactile information from the grasp attempt as supervision (Sec. 4.2).

4 Method

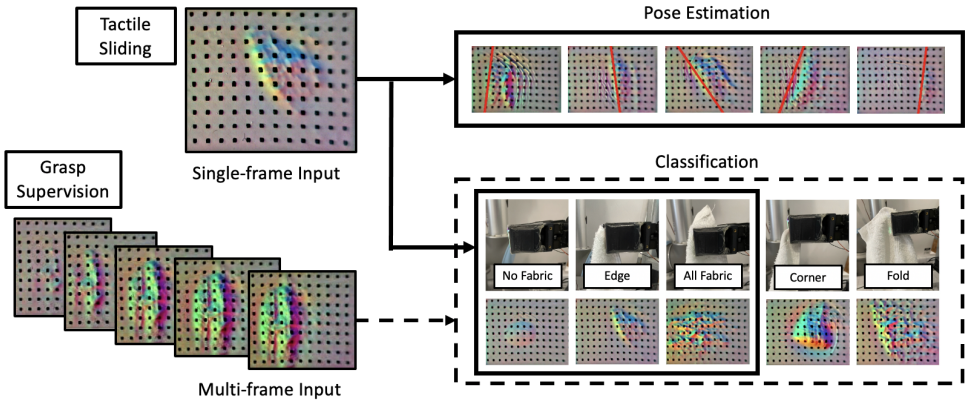

Figure 3: **Tactile perception.** The perception network used for sliding along the fabric edge takes a single-frame input and outputs the pose of the edge, including states where no edge is visible (the no fabric and all fabric cases). For our grasp supervision network, we sample multiple frames from the grasp attempt and classify between additional categories.

4.1 Tactile Perception of Local Cloth Features

Tactile perception is used for pose estimation and classification of local features within the grip. All of the following tactile perception networks share a similar architecture and strategies for data augmentation.

Pose Estimation. Applying the same cable pose estimation method as [28] to slide along the edge of soft cloths would be challenging because the fabric is relatively thin and covers a large area (see Fig. 3), unlike the cable which creates a distinct imprint. We found that traditional computer vision edge detection techniques are unsuccessful at consistently isolating the fabric edge due to the fabric texture, wrinkles in the sensor's gel surface, and noise in the tactile image.

Our pose network takes a down-sampled depth image as input and outputs the x and y coordinates of the center of the edge and the orientation of the edge, which can be directly used for tactile sliding.

The network has 3 convolutional layers followed by max pooling, and 3 fully connected layers. We choose a simple network architecture to optimize for speed to use in our sliding controller.

The dataset is generated from a small number of hand-labeled tactile images (frames from a video), and then augmenting the dataset by randomly varying the maximum depth threshold of each tactile image, and further augmenting the images by introducing a random translation and rotation (see Appendix for details). The human labeling process involves clicking two points to define the cloth edge if visible, or otherwise classifying the image as all fabric or no fabric. This labeling can then be transformed to all the synthetically augmented images.

Classification. The tactile sliding perception network that outputs pose also classifies whether (1) there is no fabric on the sensor, (2) the sensor is covered in fabric, or (3) the sensor sees an edge (Fig. 3). We collect data from videos of sliding with different gripper widths. The final network we train has 150 raw training images which we augment to 30,000 depth images.

The tactile sliding network can classify between no fabric, edges, and all fabric given a single frame when the gripper is pressing against the towel. To supervise our edge grasp affordance network, we also need to distinguish between edges and folds. Fold classification is especially challenging because the final tactile image can look identical to other categories. Therefore, we use a sequence of images from the grasp attempt as input which introduces temporal information, instead of just the final grasp. The tactile image starts showing an imprint earlier with multiple layers of fabric.

We sample every 5 tactile depth images of the grasp attempt and choose the last 5 as input to the network. During the augmentation process, we shift the initial frame for sampling. We skip depth augmentation because timing is crucial to this network, but we continue to translate and rotate the images. We record 330 grasp attempts per category and augment this to over 6000 per category.

4.2 Visuotactile Grasp Affordances

In this work, there are three kinds of grasp points we are interested in: generic (for picking), corner, and edge. All three use vision to choose grasp points and can use either visual or tactile perception to verify grasp success (Fig. 2). We discuss the generic and corner grasps in Sec. 4.4.

Edge grasp point localization is more challenging than corners because we must also consider the direction of the edge. Furthermore, visual edge detection can be difficult since edges look similar to folds along the contour of the cloth. The cloth edge can also easily twist in on itself, but a successful grasp for sliding requires grasping a single layer. Moreover, aligning the edge within the gripper to avoid collision is crucial for grasp success. If the cloth collides with the gripper, it easily deforms, preventing a successful grasp. Given these additional considerations, we train a network that learns visuotactile edge grasp affordances.

We first train this affordance network in simulation. Then, we refine our fine-tuning technique while transferring between slightly different simulation environments. Finally, we fine-tune this network on a robot using the tactile information from the grasp attempt's success/failure as supervision.

Training in Simulation. We use the same Blender simulation setup as [21] for training the affordance network using U-Net [31], as shown in Fig. 4. The input is a depth image of the hanging cloth, and the output is the affordance heatmap. The ground-truth labels of the affordances in simulation are determined for each pixel by geometric computing, specifically (1) the percentage of neighboring cloth nodes that are classified as an edge, (2) whether the

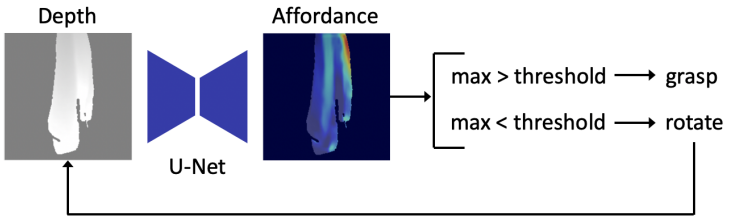

Figure 4: **Affordance architecture and rotation policy.**

gripper will collide with cloth, (3) whether the gripper would grasp a single or multiple layers of fabric, and (4) whether the point is reachable. All of these criteria can be determined in simulation because we have access to the full state of the cloth. The tactile sensor implicitly checks all of these criteria on the real system when it classifies a grasp as an edge or non-edge grasp. The simulated dataset contains 200 cloth configurations rotated in increments of $15°$, yielding 4800 depth images with corresponding affordance images.

Fine-tuning in Simulation. As a precursor to sim-to-real transfer, we first try sim-to-sim transfer to validate our network architecture and tune parameters. While the network in simulation has access to an entire labeled image during training, during fine-tuning, the robot can only test a single pixel at a time. We create a second simulated dataset with slightly different camera location, cloth size, and cloth stiffness. We change several network parameters (see Appendix for details) and ultimately find that fine-tuning with single pixel updates is feasible with an experience replay buffer.

Fine-tuning on Robot with Tactile Supervision. The real depth images on the robot have regions of zero-depth due to the offset in location between the projector and sensor of the depth camera. Also, items in the background affect the affordance. Therefore, we mask out all non-cloth regions in both simulation and deployment and add random black rectangles to the simulated depth images during training for robustness against regions with zero-depth (see Appendix for details).

During data collection, we grasp a corner, mask the reachable regions, and attempt to grasp the cloth at different points with fixed orientation before resetting and grabbing a new corner. The reachability mask avoids collisions between both grippers, chooses the side of the cloth closest to the grasping gripper, and filters out the bottom of the cloth to only grasp adjacent edges. The cloth is rotated in increments of $15°$ for a whole rotation with 3 grasp attempts per rotation. We implement a minimum affordance threshold to increase the number of positive examples (Fig. 4). Each grasp attempt is labeled with the tactile multi-frame classification network's confidence that the grasp is an edge. We collected around 3,000 grasps, which takes about 15 hours. We validate the network with 110 hand-labeled data collected offline.

4.3 Tactile Sliding

Sliding in Stable Configuration. Once one gripper holds a corner and the other holds an adjacent edge, we use the tactile signal to slide downward. The sliding gripper moves forward and back to keep the edge close to the target position within the grip until the robot reaches a corner. We use a proportional controller with respect to the fabric edge's offset from the target position to slide along the edge. We consider an increase in shear (as determined by GelSight marker displacement) above a certain threshold from the fixed gripper to indicate that the sliding gripper has reached the thicker corner.

Sliding along Long Towels. We follow the methods of [28] to slide along the edge of longer towels in a horizontal configuration. We learn a linear dynamics model that finds the relationship between state (defined by cloth edge position and orientation and robot position) and pulling angle. We use this model in an LQR controller that keeps the edge close to the target state as the cloth is pulled through (see Appendix for details).

4.4 Cloth Manipulation System

In order to complete the system, we also need to pick the towel up from the ground and then grasp a corner. Although the specific grasp location with respect to the rest of the cloth is unknown, this initial grasp is helpful for partially unfolding the towel using gravity. This allows for more of the towel to be visible for downstream tasks. For our specific grippers, the towel grasp point cannot be completely flat on the surface (wrinkles are much easier to grasp). Therefore, we sample from the highest points to choose a random grasp point.

Corner grasp points are common for flattening and folding tasks. However, corners are not always visible when the cloth is on a surface due to self-occlusions. Furthermore, visible corners on a table often lie flat on the surface, which would require modifications to our gripper design to grasp them. So, instead, we look for corners on the hanging cloth by choosing the lowest point.

For the hardware setup, we use dual UR5 robot arms, which are equipped with parallel-jaw grippers servoed with dynamixel motors (Dynamixel XM430-W210-T, Robotis). We mount two GelSight sensors [32], one on each gripper. We use a Kinect Azure camera to capture RGB-D images.

5 Results

Tactile Perception. Our pose estimation network for tactile sliding has a mean average error of 3.2 mm for the center position of the edge and $4.1°$ for the orientation of the edge. The overall sensing area is 30 mm wide. These error ranges are within the expected human labeling error and this performance is experimentally sufficient for tactile sliding. We compare this network to one trained on a dataset without augmentation, which has a similar position error, but a significantly larger orientation error of $10.2°$. This network runs at 33 Hz within the sliding controller.

The multi-frame classification network used for grasp affordance supervision has an overall classification accuracy of 92%. Since our affordance network only needs to distinguish edges from other grasps, we also report a non-edge classification accuracy of 98%.

Visuotactile Grasp Affordance. Our network trained in simulation successfully grasps edges in 21.3% of the given configurations, out of which only 39.7% have valid, accessible edge grasps. By introducing our threshold rotation policy (Fig. 4), our network successfully grasps an edge for 56.6% of initial configurations in which 95.0% of configurations had valid grasps. Rotations are crucial to finding and accessing valid edge grasps.

We compare our fine-tuned affordance network to three other baselines: (1) Segmentation-based: a visual edge segmentation network [20] combined with surface normal and right side-reachability cost metrics (see Appendix for details), (2) Sim2Real: the network trained in simulation that is directly transferred to the robot, and (3) Real2Real: a network trained directly on the collected robot data. We evaluate the networks offline using precision@k [33], using this metric due to the unbalanced nature of the evaluation set and because it does not need a set threshold. We found that Sim2Real performs the worst because of the differences between the simulated and real robot environments. However, the fine-tuned network outperforms the Real2Real network, showing that the network from simulation provides overall structure.

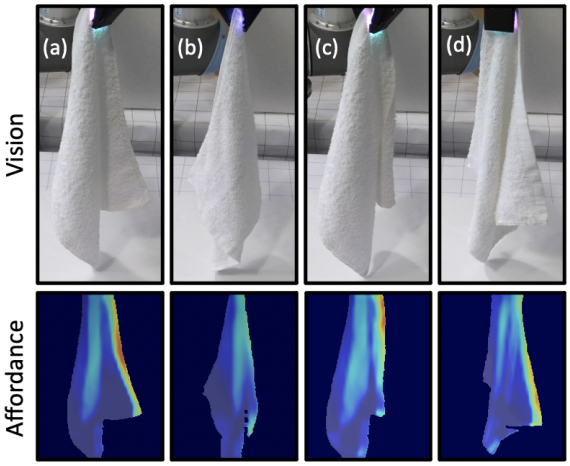

Figure 5: **Fine-tuned edge grasp affordance results.** The learned affordances effectively reflect the edge graspability in different configurations, where it is graspable for (a) the entire right edge and (b) no edge. The last two towels have edges that twist such that they are only accessible from (c) the top portion and (d) the bottom portion. The proximity to the rest of the fabric (c) or the misalignment of the surface normal (d) would cause collisions during the grasp attempt.

We further evaluate the networks in deployment by reporting the average number of grasp attempts before identifying an edge (as confirmed by tactile perception) and the overall success rate of true edge grasping. The threshold was chosen independently for each network and tuned to maximize performance on the offline dataset. The cloth is grasped by the corners and evaluated with 10 different configurations.

	Offline Evaluation	Real Robot Deployment	
	Precision@k	Average # attempts	Success Rate
Segmentation-based [20]	65.0%	4.2	65.0%
Sim2Real	45.0%	1.3	40.0%
Real2Real	62.5%	3.3	50.0%
Sim2Real + Finetune (Ours)	**80.0%**	1.6	**90.0%**

Table 1: Quantative results in real world for visual affordance learning with tactile supervision. We use $k = 40$ for precision@k, since there are around 40 positive samples out of 110 in the real-world data for offline evaluation. For evaluation during deployment, we compare methods with 10 different trials. Each model has similar initial cloth configurations for each trial.

Table 1 shows the experiment results. The failure modes include false-positive edge tactile classifications and rotating $360°$. In general, more grasp attempts increase the likelihood of a false positive edge classification. Our fine-tuned network outperforms both baselines with an overall success rate of 90%. The Sim2Real network often misses grasp opportunities and completes a full rotation without successful attempts. The Real2Real network is more likely to grab inside the cloth without the underlying structure from simulation. The robot often grabs an edge along with a fold, resulting in false positives from tactile classification. The segmentation-based network fails to identify grasps that would fail due to collisions with other regions of fabric. The fine-tuned network effectively identifies graspable edges, considering both global and local geometry and semantics for calculating affordance (Fig. 5).

Tactile Sliding. When sliding while the towel is in a stable configuration, we find that our sliding controller performs differently with the thin and thicker edges of the towel. The gripper traverses 100% of the thin edge (sliding all the way to the corner) no matter the initial edge position and 100% of both the thin and thicker edges if the initial edge pose is near the inner edge of the gripper (n = 5 trials). The sliding distance decreases (cloth slips from grip before reaching the corner) as the initial edge pose starts closer to the fingertip. At a centered initial pose, the sliding gripper travels an average of 95% of the towel length (14.0 cm for the thin edge and 12.7 cm for the thick edge).

For sliding in the horizontal configuration, we slide until the end of the robot workspace (smaller than the towel's total length). Similar to sliding in a stable configuration, we find that if the cloth edge starts near the gripper's inner edge, the robot can consistently slide for the entire length of the workspace in all trials. If the initial edge pose is in the center of the gripper, the robot travels an average of 76% of the workspace (43 cm out of 56 cm).

6 Discussion

We develop a robot system for cloth manipulation that uses both visual and tactile perception of local features for local control. Our tactile perception systems are used for both pose estimation for controlled tactile sliding and supervision for learning visuotactile grasp affordances. Both networks are trained with small amounts of real data by augmenting the tactile depth images. Our grasp affordance network learns global structure from simulation and is fine-tuned using tactile supervision during real robot grasp attempts. This network outperforms all of our baselines.

Limitations and Future Works. We use multiple neural networks for our local approach to perception and control, which lends itself to generalization because we do not rely on the global structure of the deformable object. However, while our pose network works on most towels, but the tactile classification network is specific to the training towel's thickness and texture and the fine-tuned affordance network is specific to the towel's dynamics and shape. These networks could be more generalizable if trained with a dataset with more variety. Our fine-tuning framework allows for introducing new cloth types without needing to train from scratch (given an applicable classification network). Our tactile approach allows certain modules to be independent of specific cloth parameters, for example, towel length or texture while sliding. As another limitation, tactile sliding success is dependent on the initial cloth edge position. Adding a tactile regrasp step to adjust the cloth edge position with tactile feedback before sliding could help address this issue.

Acknowledgments

We thank the anonymous reviewers for their helpful comments in revising the paper. Toyota Research Institute (TRI), Amazon Research Awards (ARA), the Google Faculty Research Awards, the Office of Naval Research (ONR) [N00014-18-1-2815], and GentleMAN and BiFrost projects of SINTEF provided funds to support this work. Neha Sunil is supported by the National Science Foundation Graduate Research Fellowship [NSF-1122374].

References

[1] P. Jiménez and C. Torras. Perception of cloth in assistive robotic manipulation tasks. *Natural Computing*, 19(2):409–431, 2020.

[2] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. *arXiv preprint arXiv:1703.09312*, 2017.

[3] A. Zeng, S. Song, K.-T. Yu, E. Donlon, F. R. Hogan, M. Bauza, D. Ma, O. Taylor, M. Liu, E. Romo, et al. Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 3750–3757. IEEE, 2018.

[4] C. Chi and D. Berenson. Occlusion-robust deformable object tracking without physics simulation. In *Intelligent Robots and Systems (IROS), 2019 IEEE International Conference on*. IEEE, 2019.

[5] 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 *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1046–1052. IEEE, 2014.

[6] C. Chi and S. Song. Garmentnets: Category-level pose estimation for garments via canonical space shape completion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3324–3333, 2021.

[7] W. Yan, A. Vangipuram, P. Abbeel, and L. Pinto. Learning predictive representations for deformable objects using contrastive estimation. *arXiv preprint arXiv:2003.05436*, 2020.

[8] R. Hoque, D. Seita, A. Balakrishna, A. Ganapathi, A. K. Tanwani, N. Jamali, K. Yamane, S. Iba, and K. Goldberg. Visuospatial foresight for multi-step, multi-task fabric manipulation. *arXiv preprint arXiv:2003.09044*, 2020.

[9] J. Wang, H. Gang, S. Ancha, Y.-T. Chen, and D. Held. Semi-supervised 3d object detection via temporal graph neural networks. In *2021 International Conference on 3D Vision (3DV)*, pages 413–422. IEEE, 2021.

[10] B. Willimon, S. Birchfield, and I. Walker. Model for unfolding laundry using interactive perception. In *IEEE International Conference on Intelligent Robots and Systems*, 2011. ISBN 9781612844541. doi:10.1109/IROS.2011.6048796.

[11] C. Harris and M. Stephens. A Combined Corner and Edge Detector. 2013. doi:10.5244/c.2.23.

[12] K. Yamazaki, K. Nagahama, and M. Inaba. Daily clothes observation from visible surfaces based on wrinkle and cloth-overlap detection. In *MVA*, pages 275–278, 2011.

[13] P. Sundaresan, J. Grannen, B. Thananjeyan, A. Balakrishna, M. Laskey, K. Stone, J. E. Gonzalez, and K. Goldberg. Learning rope manipulation policies using dense object descriptors trained on synthetic depth data. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9411–9418. IEEE, 2020.

[14] Y. Wu, W. Yan, T. Kurutach, L. Pinto, and P. Abbeel. Learning to manipulate deformable objects without demonstrations, 2019.

[15] R. Hoque, K. Shivakumar, S. Aeron, G. Deza, A. Ganapathi, A. Wong, J. Lee, A. Zeng, V. Vanhoucke, and K. Goldberg. Learning to fold real garments with one arm: A case study in cloud-based robotics research. *arXiv preprint arXiv:2204.10297*, 2022.

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

[17] 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 *2014 IEEE international conference on robotics and automation (ICRA)*, pages 987–993. IEEE, 2014.

[18] M. Cusumano-Towner, A. Singh, S. Miller, J. F. O'Brien, and P. Abbeel. Bringing clothing into desired configurations with limited perception. In *2011 IEEE international conference on robotics and automation*, pages 3893–3900. IEEE, 2011.

[19] 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 *2010 IEEE International Conference on Robotics and Automation*, pages 2308–2315. IEEE, 2010.

[20] J. Qian, T. Weng, L. Zhang, B. Okorn, and D. Held. Cloth region segmentation for robust grasp selection. In *IEEE International Conference on Intelligent Robots and Systems*, 2020. ISBN 9781728162126. doi:10.1109/IROS45743.2020.9341121.

[21] A. Ganapathi, P. Sundaresan, B. Thananjeyan, A. Balakrishna, D. Seita, J. Grannen, M. Hwang, R. Hoque, J. E. Gonzalez, N. Jamali, K. Yamane, S. Iba, and K. Goldberg. Learning to smooth and fold real fabric using dense object descriptors trained on synthetic color images. *arXiv*, 2020.

[22] E. Corona, G. Alenya, A. Gabas, and C. Torras. Active garment recognition and target grasping point detection using deep learning. *Pattern Recognition*, 74:629–641, 2018.

[23] R. Hoque, D. Seita, A. Balakrishna, A. Ganapathi, A. K. Tanwani, N. Jamali, K. Yamane, S. Iba, and K. Goldberg. VisuoSpatial foresight for multi-step, multi-task fabric manipulation, 2020.

[24] K. S. M. Sahari, H. Seki, Y. Kamiya, and M. Hikizu. Edge tracing manipulation of clothes based on different gripper types. *Journal of Computer Science*, 2010. ISSN 15493636. doi:10.3844/jcssp.2010.872.879.

[25] H. Yuba, S. Arnold, and K. Yamazaki. Unfolding of a rectangular cloth from unarranged starting shapes by a Dual-Armed robot with a mechanism for managing recognition error and uncertainty. *Advanced Robotics*, 2017. ISSN 15685535. doi:10.1080/01691864.2017.1285722.

[26] W. Zhou, S. Bajracharya, and D. Held. Plas: Latent action space for offline reinforcement learning. *arXiv preprint arXiv:2011.07213*, 2020.

[27] W. Yuan, S. Dong, and E. Adelson. Gelsight: High-resolution robot tactile sensors for estimating geometry and force. *Sensors*, 17(12):2762, 2017.

[28] Y. She, S. Wang, S. Dong, N. Sunil, A. Rodriguez, and E. Adelson. Cable manipulation with a tactile-reactive gripper. *arXiv preprint arXiv:1910.02860*, 2019.

[29] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser. Learning synergies between pushing and grasping with self-supervised deep reinforcement learning. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4238–4245. IEEE, 2018.

[30] A. Zeng, S. Song, J. Lee, A. Rodriguez, and T. Funkhouser. Tossingbot: Learning to throw arbitrary objects with residual physics. *IEEE Transactions on Robotics*, 36(4):1307–1319, 2020.

[31] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[32] S. Wang, Y. She, B. Romero, and E. Adelson. Gelsight wedge: Measuring high-resolution 3d contact geometry with a compact robot finger. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6468–6475. IEEE, 2021.

[33] M. Sanderson. *Test collection based evaluation of information retrieval systems*. Now Publishers Inc, 2010.

[34] F. R. Hogan, M. Bauza, O. Canal, E. Donlon, and A. Rodriguez. Tactile regrasp: Grasp adjustments via simulated tactile transformations. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2963–2970. IEEE, 2018.

[35] Y. Hou, Z. Jia, A. M. Johnson, and M. T. Mason. Robust planar dynamic pivoting by regulating inertial and grip forces. In *Algorithmic Foundations of Robotics XII*, pages 464–479. Springer, 2020.

[36] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[37] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

7 Appendix

7.1 System Design Choices

One of the main intentions of this paper is to show the capabilities of local visuotactile perception and control for deformable object manipulation, rather than overall task success or efficiency. While both edge grasping and tactile sliding have high success rates if initialized properly, the transitions require more engineering to make smooth. We choose to use a state machine such that whenever an action fails, our visual and tactile checks (if accurate) can restart the system where appropriate.

State Machine. In the state machine (See Fig. 2), the robot tries to pick up the towel from the ground with its right gripper, then grasp a corner (the lowest point of the towel) with its left gripper. After each of these steps, visual perception is used to ensure that the grasp is successful, meaning that there is cloth in the expected region of the depth image. If at any point the towel is on the ground, then the state machine restarts from the initial pick. Once a corner is grasped, the robot uses our grasp affordance network to find an edge grasp whose affordance value is above a certain threshold. If no grasp is found, the towel is rotated. If a full rotation is reached, then the towel is released to the floor and the state machine restarts. The gripper approaches the edge with a fixed orientation. After grasping an edge (confirmed using tactile), the gripper slides to the adjacent corner, where the end condition is determined from tactile. From this position, the robot can unfold or fold the towel using open-loop commands because we use a towel of a fixed size.

Motivation for Hanging. While ideally we would grasp edges or corners directly from the table-top if visible, these grasps are usually unavailable for our hardware. Other papers use a compliant surface or smaller, tweezer-like grippers that can more easily slide under a layer of cloth in order to grasp an edge [20]. Our tactile grippers, while unable to directly grasp a single layered edge from the ground due to their larger size, enable other skills (sliding, tensioning). Picking the towel off the table is an important step so that edge grasps are exposed.

Corner Detection. We originally intended to use corner detection in both tactile networks as an end condition for tactile sliding and to confirm direct corner grasps. However, we found that sliding over the thicker corner required a wider gripper opening that compromised tactile perception of the edge during the rest of sliding. Therefore, we chose to use shear in the tactile signal of the stationary gripper as the indicator of reaching the opposite corner. Corner detection accuracy in the grasp network was not sufficiently high for direct corner grasping for our task (Fig. 7). The requirements for that initial pick of a corner are less stringent, since it is not necessary to grasp on a "clean" corner. For example the robot can pick up a corner folded in on itself and still successfully complete the task. Therefore, we found picking the lowest point and using vision to confirm that the towel is being held was sufficient for our task.

Individual Action Evaluation. We evaluate the success of individual actions because our system can sense and handle failures of actions, but some transitions would require further engineering to smooth out. We evaluate our edge grasp policy starting from a corner grasp and our tactile sliding controller starting from an edge grasp. We vary initial configurations across trials (cloth configuration for edge grasping and edge position for tactile sliding).

7.2 Failure Modes

We summarize several major failure modes during different phases of the system.

Corner Grasping. The main failure modes of corner grasping are slightly missing the towel (be-

cause of movement while sensing or errors from the camera) or the lowest point not being a corner (if the towel is partially folded). This can be improved with modified tactile verification.

Edge Grasping. Edge grasping fails if the tactile classifier falsely determines an edge is grasped (especially if grasping a compressed fold). Another common failure mode is that the towel is rotated fully without a viable edge grasp. Sometimes, the edge is visible but sticks to another layer of cloth. This can be improved with more dexterous grippers or manipulation skills to separate an edge from other parts of the cloth with vision and tactile feedback.

Grasping and Sliding. The edge grasping to sliding transition is difficult because some successful edge grasps can be challenging for sliding. The position of the edge is not a consideration for the edge grasp network, while it is a factor that highly influences sliding success. For example, when the edge is too close to the tip of the gripper, it can be difficult to adjust back. Additionally, the tactile sliding network performs better on the thin edges of the towel while the edge grasp affordance network tends to give thicker edges higher affordances. These can be improved with a tactile edge adjustment module, which adjusts the edge pose with tactile regrasps [34] before sliding, and dynamic manipulation [35, 16] to unfurl the cloth during sliding.

7.3 Network Architectures and Training Details

Tactile Perception Networks. We choose a small network architecture for our tactile sliding network to optimize for speed. Specifically, our base structure is C6-C16-C32-MP-F512-F128, where C represents convolutional layers (with 5x5 kernel sizes), MP represents max-pooling layers (with kernel size of 2x2), and F represents fully-connected layers. We use ReLU as our activation function. We add separate final fully-connected layers for pose estimation, and classification. Pose estimation uses a loss function of Mean Square Error (MSE) and classification uses cross-entropy loss. We use the Adam [36] optimizer. For handling video sequences, we use a similar structure, with stacked frames in the color channel, following [37]. In our experiments, we stack 5 frames in the end of the video, with interval of 5 frames (e.g. frames 10, 15, 20, 25, 30). We train our model with NVIDIA Titan X Pascal GPU, and the training time for our tactile perception networks is about 5 hours.

Visuotactile Networks. We choose U-Net[1] as our network for training visuotactile affordances. We use MSE loss for the whole image to train affordance in simulation and use MSE loss for individual pixels for fine-tuning with real grasps. For fine-tuning, we also include the loss of the pixels around the selected pixel (with distance in 3 pixels) to improve training efficiency. For better generalization and preventing over-fitting, we reduce the number of channels by half in each layer, and remove the skip connection in the last two Up-sampling layers. Based on our experiments, this modification produces more generalized and smoother affordance. During experiments, we also test fully convolutional networks [29]. In comparison, we find that U-Net provides more resolution, which is crucial for successful edge grasps. We train our models with NVIDIA Titan X Pascal GPU. The training time for our visuotactile affordance networks in simulation is about 8 hours, and the training time for fine-tuning is about 1 hour.

7.4 Data Augmentation for Tactile Perception

For both the tactile sliding and grasp supervision networks described in Sec. 4.1, we augment the data we collect and label using the process shown in Fig. 6. We randomly vary the maximum depth threshold and further augment the images by introducing a random translation and rotation. For the tactile sliding pose network, the hand-labeled edge is transformed as well. Different categories have their own transformation parameters. For example, no fabric images have minimal transformations because the imprint in the depth image is caused by the convexity of the gel surface, which is always in the same position.

[1] See https://github.com/milesial/Pytorch-UNet for details.

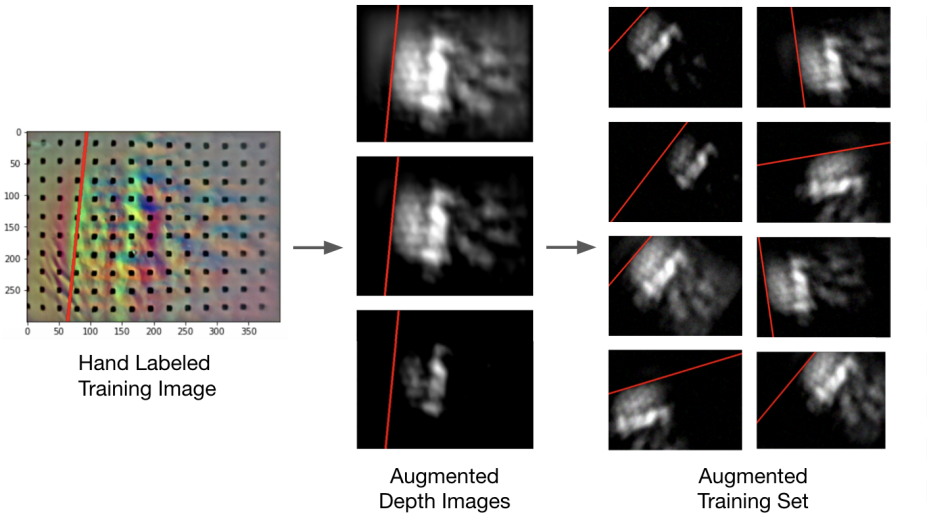

Figure 6: **Dataset augmentation for tactile perception.** We augment the data by varying the depth scale and transformation to improve results with a limited amount of labeled data.

7.5 Tactile Grasp Supervision Network Results

Tactile classification accuracy is evaluated on a held out test set evenly distributed across classes. We use a network that best differentiates between edge and non-edge grasps. The network performs well on most categories except for folds (See Fig. 7), which can look like most other categories. However, the network is sufficiently capable of distinguishing edges and folds for the purposes of our grasp supervision network.

7.6 Training Affordance Network in Simulation

The ground-truth labels of the affordances in simulation are determined for each pixel by geometric computing. Specifically, for each scene, we extract the depth image, the point cloud of the visible and the whole cloth, and the undeformed coordinates of each point in the local cloth frame. The parallel-jaw gripper is represented as a 3D box. Given a grasping point, we calculate the ground-truth affordance using the following criteria:

- **Edge percentage.** The percentage of neighboring cloth nodes that are classified as an edge, in the gripper box. Computationally, the edge is determined based on the distance to the cloth boundary, in the undeformed local cloth frame.
- **No collision.** Whether the gripper will collide with the cloth before grasping. This checks whether the orientation of the local cloth is aligned with the gripper, to grasp without the cloth being pushed away. Computationally, we assign some thickness to each finger of the gripper in the box region to simulate the real hardware. We check whether surrounding points collide with these finger regions.
- **Single layer.** Whether the gripper would grasp a single layer of fabric. A single layer of cloth is desired to start sliding. Computationally, we cast rays from one finger to the other finger and check whether each ray collides with a single layer of points. For points colliding with each ray, we set a threshold based on the grid size to determine whether these points are adjacent or apart in the undeformed coordinates in the local cloth frame.
- **Reachability.** Whether the point is reachable from the robot arm on the right side without colliding with other parts of the cloth. Computationally, we check the sweeping region from the right side of the candidate gripper box, to determine whether there are cloth points in collision.

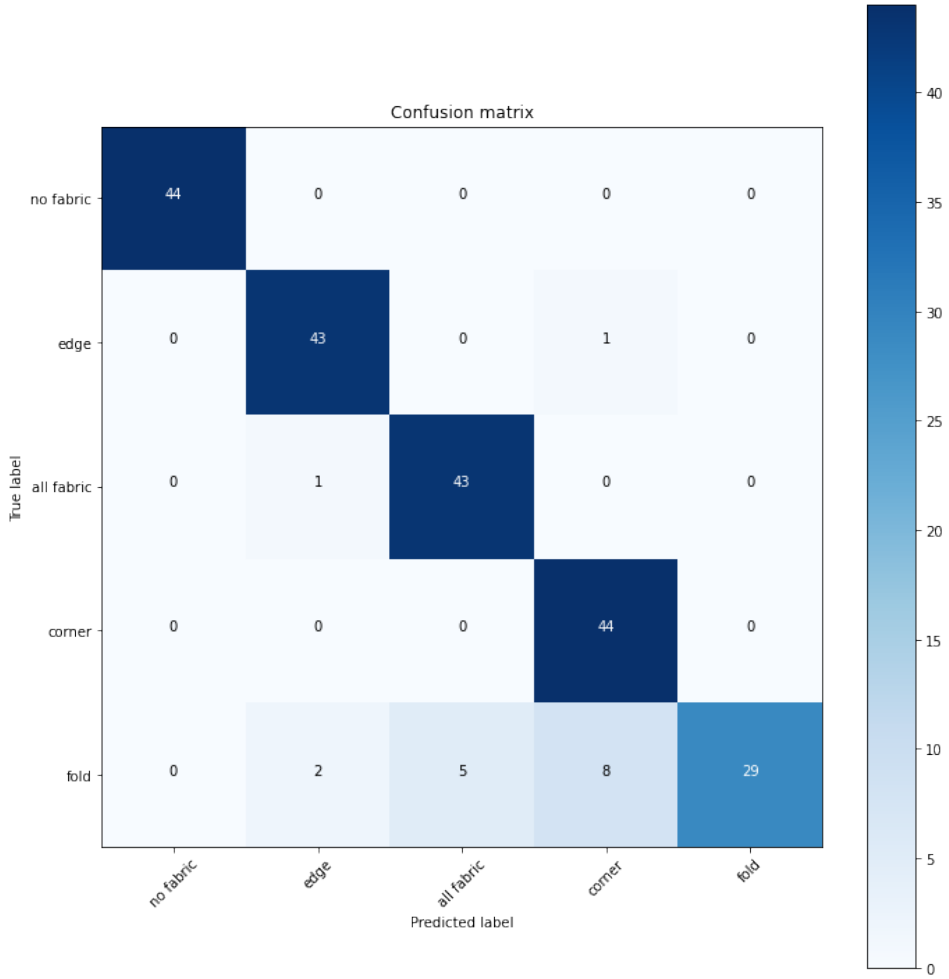

Figure 7: **Confusion matrix for tactile grasp supervision.**

An example is shown in Fig. 8 for visualization. All of these criteria can be determined in simulation because we have access to the full state of the cloth. The tactile sensor implicitly checks all of these criteria on the real system when it classifies a grasp as an edge or non-edge grasp. The simulated dataset contains 200 cloth configurations rotated in increments of $15°$, yielding 4800 depth images with corresponding affordance images (Fig. 9).

To test the feasibility of fine-tuning on the real robot system where we can only test one pixel at a time during training, we try fine-tuning our network in a slightly different simulation environment. The second simulated dataset has slightly different camera location, cloth size, and cloth stiffness. We test several different network parameters including the addition of a replay buffer, which layers are frozen, balanced sampling from the replay buffer, adding random crops for data augmentation, and updating multiple nearby pixels at once during training. Our key finding was that fine-tuning with single pixel updates is feasible with an experience replay buffer. Without the replay buffer, training converges with 88,000 grasps. With the replay buffer, we achieve comparable performance with 3,000 grasps.

7.7 Sim-to-Real Transfer

The real depth images on the robot have regions of zero-depth due to the offset in location between the projector and sensor of the depth camera. Also, items in the background affect the affordance.

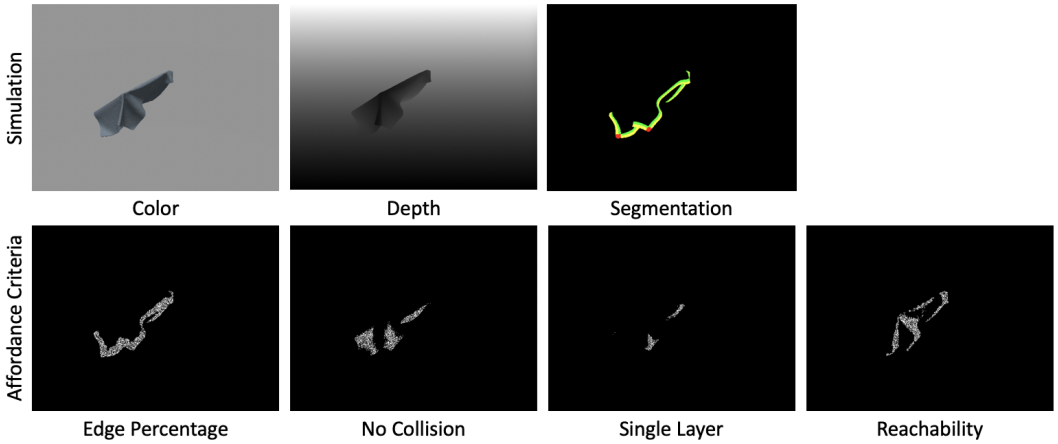

Figure 8: **Affordance calculation in simulation.** *Top*: the color image, depth image, and segmentation of a piece of cloth from the simulated environment. The segmentation is visualized with corners in red, outer edges in yellow, inner edges in green, following the visualization in [20]. *Bottom*: the criteria used to calculate the affordance. The final affordance combines these together.

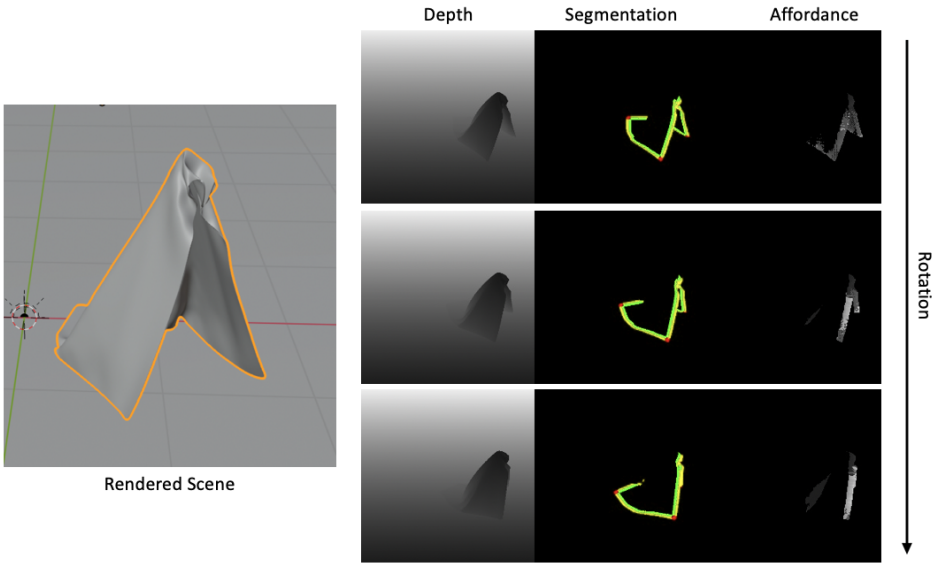

Figure 9: **Simulated dataset for affordance network.** With rotation, the affordance becomes higher when the edge is aligned with gripper, and can be grasped with no collision.

We found that the original network tends to assign high affordances to the zero-depth regions (Fig. 10). Therefore, we mask out all non-cloth regions in both simulation and deployment and add random black rectangles with different sizes and orientations to the simulated depth images during training for robustness against regions with zero-depth (Fig. 11).

7.8 Robot Implementation and Control

Grasping with the UR5s uses position-control while tactile sliding uses velocity-control for smooth behavior, which runs at around 30 Hz. For the grippers, we use Dynamixel motors. They are position-controlled with current limits to protect the sensor. For tactile classification with a multi-frame input, we sent decreasing width commands over time to gradually close the gripper, and collect the tactile signals under different gripper widths.

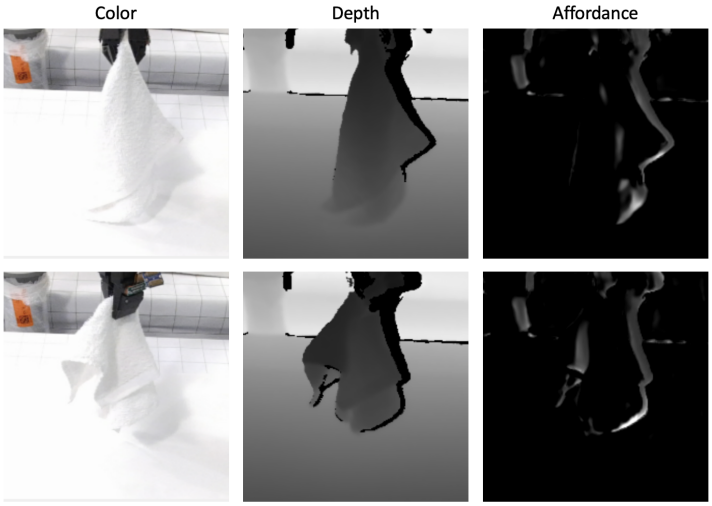

Figure 10: **Direct sim2real transfer without additional modifications.** Without additional modification, the Sim2Real network attends to the black region caused by the offset between the projector and the sensor of the depth camera.

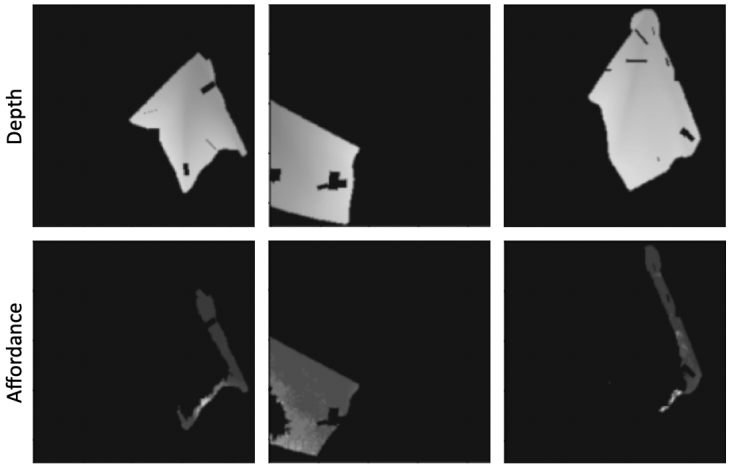

Figure 11: **Additional modification with mask and black blocks for Sim2real transfer.** With the added cloth mask and random black blocks, the network can predicate the affordance as desired.

7.9 Segmentation-Based Affordance Baseline

We train a network based on [20] that takes a visual depth image as input and outputs the segmentations of corners, outer edges, and inner edges. We collected 10,000 data points on our system, with a painted cloth to label different segmentation to provide ground-truth labels [20]. To assign affordances to each point, we have a cost metric that includes edge direction uncertainty, right side reachability, and collision-free surface normals.

7.10 Vertical Tactile Sliding

To test the robustness of vertical tactile sliding, we varied the initial edge position across experiments with 5 trials per initial condition (Table 2). The controller was able to consistently follow the entire length of the thin edge of the towel, but sliding success along the thicker edge was dependent on starting pose. The closer the edge was to the tip of the gripper (less initial cloth coverage), the more likely the cloth was to slip out before the gripper slid to the edge.

Initial Cloth Coverage of Tactile Sensor (%)	Distance Traveled (%) Thin Edge	Thick Edge
100%	100%	100%
75%	100%	92.1%
50%	100%	90.4%
25%	100%	82.9%

Table 2: Vertical tactile sliding results given different starting configurations (n = 5 trials/configuration). Initial edge position indicates the fraction along tactile sensor that is covered in fabric. With an initial cloth coverage of 100%, the cloth edge is aligned with the sensor. The cloth edge is centered within the sensor at 50%.

7.11 Horizontal Tactile Sliding

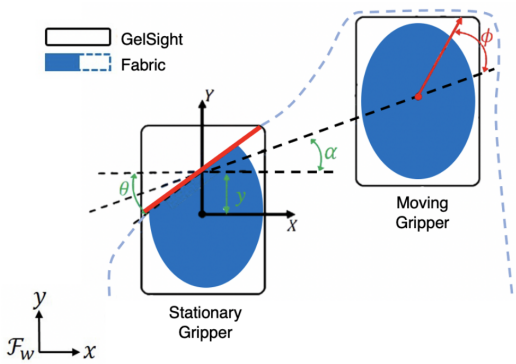

Figure 12: **Cloth sliding dynamics.** Diagram of the cloth sliding modeling.

We use horizontal sliding to slide along longer towels. Horizontal sliding has more complex dynamics than vertical sliding because the towel is more likely to slip out of the grip due to gravity, and the controller needs to compensate. We model the dynamics as shown in Fig 12. Similar to the cable following system from [28], the system's state is $\mathbf{x} = [y\ \theta\ \alpha]^T$, where y is the fabric edge height from the target edge position on the gripper, θ is the fabric edge angle with respect to the gripper, and α is the angle between the two grippers (Fig. 12). The control input $\mathbf{u} = [\phi]$ is the pulling angle. The linear dynamic model is of the form:

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}, \tag{1}$$

where A and B are the linear coefficients of the model.

To learn a dynamics model, we collect data using a simple proportional controller ($\phi = -Ky$) supplemented with uniform noise. The dataset has 7100 observations from 30 different following runs with a variety of starting states and pulling angles. We perform both linear regressions and sparse Gaussian Process (GP) regressions. From initial testing, we were not sure if a linear model could capture the complex dynamics of cloth sliding, so we tested controllers based on the non-linear GP dynamics model as well.

We compare the two best performing controllers for cable following for cloth edge following. The first baseline controller is Linear–quadratic Regulator (LQR) using the linear dynamics model. The second time-varying LQR (tvLQR) controller linearizes the GP model about the current state. For the LQR parameters, we use $\mathbf{Q} = [100000, 1, 0.1]$ and $\mathbf{R} = [0.1]$, to prioritize regulating y and θ (so the cloth does not fall).

When we tested these controllers on the physical system, we found that both controllers performed comparably. However, The tvLQR network with a linear kernel for the $\dot{y}$ model had similar commanded actions to LQR based on the linear model, but it runs slower due to the time required to calculate the predictive gradient for the GP. Therefore, we found the linear model was favorable.

Horizontal sliding experiments are initialized by hand-feeding the towel to the grippers. With an optimal starting position, the LQR controller based on the linear model consistently travels to the end of the workspace in all five trials. We adjust the y setpoint to be much closer to the inner edge than with cable following. When the initial position is in the center of the gripper (closer to the tip compared to the setpoint), the gripper traverses an average of 76% of the workspace (43 cm out of 56 cm).

7.12 Potential for Generalization

In this work, we generate local techniques for perception and control more tenable for generalization across types of cloth because they do not rely on global structure like most previous works. However, in order to apply the same framework to a different towel, individual modules might need to be updated to handle different size, dynamics, texture, etc. (Sec. 6). Greater variety in the training set could improve generality. This approach focusing on local features can also be useful for more complex clothing items like shirts or pants. Tactile supervision in our framework only works if different categories (edge, corner, fold, ...) can be reliably differentiated. If a feature is unique, like the hem of a shoulder strap, a zipper, or a shirt collar, then we could similarly use tactile feedback to confirm grasps of local features. However, it would be difficult to differentiate, for example, a sleeve hem and a waist hem on a T-shirt, which might be necessary for certain manipulation tasks. A reasonable way to approach this more general problem would be to use a visual prior to inform of the global structure of the cloth [21] and use tactile sensing to confirm a grasp on a hem.