Simposio CEA de Robótica, Bioingeniería, Visión Artificial y Automática Marina 2025

RBVM
Simposio CEA de Robótica, Bioingeniería, Visión por Computador y Automática Marina

Algebraic Machine Learning for Robotic Garment Unfolding

Naranjo-Campos, Francisco J.*; Victores, Juan G.; Balaguer, Carlos; Jardón Huete, Alberto

RoboticsLab, Systems and Automation Engineering Department, University Carlos III of Madrid, Leganés, 28911, Madrid, Spain

Abstract

The demand for robots capable of performing assistive tasks has increased due to the need to support people in various environments, such as healthcare and domestic settings. However, among all possible tasks, those involving the manipulation of deformable objects—such as fabrics—pose a greater challenge. This work presents an implementation of robotic cloth manipulation using the TIAGo++ robot and algebraic machine learning (AML) algorithms. These algorithms allow for the definition of rules that are constructed from the model's inputs and outputs during training. AML is applied as a regression problem to estimate the optimal pick and release points of the folded cloth. Then, using an RGB-D camera, the 3D positions of these points are obtained, and a manipulation routine is executed to unfold the cloth. The point estimation in the image has been evaluated by comparison with a standard convolutional neural network. Finally, experiments were conducted on the complete folding task—comprising both perception and manipulation—demonstrating the effectiveness of the proposed implementation.

Keywords: Perception and sensing, Autonomous robotic systems, Robots manipulators, Machine learning, Garment manipulation.

Aprendizaje máquina algebraico para desdoblado de prendas robótico

Resumen

La demanda de robots que realicen tareas de asistencia ha aumentado debido a la necesidad de ayudar a las personas en distintos entornos, como el sanitario y el doméstico. Sin embargo, de todas las tareas posibles, aquellas que implican la manipulación de objetos deformables, como los tejidos, representan un reto mayor. En este trabajo se presenta una implementación de manipulación robótica de telas utilizando el robot TIAGo++ y algoritmos de aprendizaje automático algebraicos (AML, por sus siglas en inglés). Estos algoritmos permiten la definición de reglas que se construyen a partir de las entradas y salidas del modelo durante el aprendizaje. De esta manera, AML se aplica a un problema de regresión para obtener los puntos óptimos de agarre y de soltado de la tela doblada. A continuación, mediante una cámara RGB-D, se obtiene la posición en el espacio de estos puntos y se lleva a cabo una rutina de manipulación para ejecutar el desdoblado de la tela. La estimación de los puntos en la imagen se ha evaluado mediante la comparación con una red convolucional estándar. Finalmente, se han realizado experimentos de la tarea completa de doblado, que incluye percepción y manipulación, mostrando la efectividad de la implementación.

Palabras clave: Percepción y sensorización, Sistemas robóticos autónomos, Manipuladores robóticos, Aprendizaje automático, Manipulación de prendas.

1. Introduction

The demand for robots to perform assistance tasks has increased due to the growing need to support people in various environments. As a result, multiple assistive robotics applications have been developed in clinical, domestic, and other operational settings, covering tasks such as object transportation (Naranjo-Campos et al., 2024b), cooking (Nilwong et al., 2023), coffee-shop service (Naranjo-Campos et al., 2024a), and ironing (Estevez et al., 2017b,a). However, among these tasks, the manipulation of deformable objects, such as fabrics, poses a greater challenge, particularly in processes such as folding textiles.

The field of fabric manipulation presents various challenges and has led to multiple approaches. A key aspect is the perception of the deformable object, which involves the parametriza-

*Corresponding author: frnaranj@ing.uc3m.es

tion and categorization of garments to design adapted manipulation routines (Yew et al., 2018).

Another critical component in fabric manipulation is the grasping and manipulation process. This area of research emphasizes optimizing grasp strategies by adapting the robot's end-effector or gripper to the specific properties of the material (Borràs et al., 2020; Lee et al., 2024). In parallel, precise motion planning of the robotic arm is essential to execute manipulation tasks effectively. To enhance adaptability and address the uncertainty arising from the malleable nature of fabrics, several planners incorporating object deformation models have been proposed (Luque et al., 2024; Wang et al., 2024). A widely adopted approach leverages machine learning techniques for fabric manipulation, unfolding, and grasping (Chen et al., 2023; Gu et al., 2024), often utilizing data derived from human demonstrations (Jia et al., 2019; Fu et al., 2023; Zhu et al., 2025).

Regarding fabric perception, machine learning algorithms are commonly used for wrinkle detection, grasp point identification, and task-specific goal determination. However, neural network-based methods often minimize training error, which in some cases leads to data memorization without achieving proper generalization (Goodfellow et al., 2016). To address this issue, regularization techniques and validation strategies have been proposed. An innovative alternative is Algebraic Machine Learning (Martin-Maroto and de Polavieja, 2018), which is based on an algebraic representation of the training data. This approach has proven effective in supervised learning tasks such as handwriting recognition, as well as in regression problems with real-world data.

In this context, we propose implementing a fabric folding task using perception and manipulation with the TIAGo++ mobile manipulator robot. The idea is to use Algebraic Machine Learning (AML) algorithms to determine the optimal pick-and-place points in 2D images of textiles, and to estimate their positions in space using the robot's RGB-D camera. Subsequently, a manipulation routine has been designed that incorporates force control to perform the unfolding motion.

2. Materials

In this section, we present the materials used in the implementation of the folding system. First, we describe the TIAGo++ robot, followed by an overview of the AML algorithm used for perception, as well as a convolutional neural network employed as a baseline.

2.1. TIAGo++

The robot used to perform the task is the TIAGo++ bi-manipulator mobile platform from PAL Robotics[1], shown in Figure 1. It features a modular design that allows adaptation to the specific requirements of the research task. In its default configuration, it consists of a sensorized mobile base for autonomous navigation, a prismatic torso, two 7-degree-of-freedom arms, two grippers as end-effectors, force sensors on the end-effectors, and an RGB-D camera mounted on a pan-and-tilt system.

On the other hand, the complete kinematic chain is available in URDF format, which enables the computation of both forward and inverse kinematics, and thus the achievement of desired end-effector poses when feasible. This is accomplished using the PyKDL[2] library, a tool designed for efficient kinematics calculations.

Figure 1: Mobile bi-manipulator robot TIAGo++ in its set up for the folding task.

2.2. Algorithms for perception

To implement the perception of pick and place points on the fabric image, two supervised learning algorithms have been tested: a Convolutional Neural Network (CNN) as a baseline, and the AML algorithm.

2.2.1. Convolutional Neural Network

The Convolutional Neural Network (CNN) architecture has been used as a reference, as it is a well-established technique that allows for performance evaluation.

The structure of the selected CNN, shown in Figure 2 and available as open source[3], has been applied to similar tasks, thus providing a reliable baseline. It was implemented in Keras and trained using the Adam optimizer, minimizing the mean squared error (MSE) loss function.

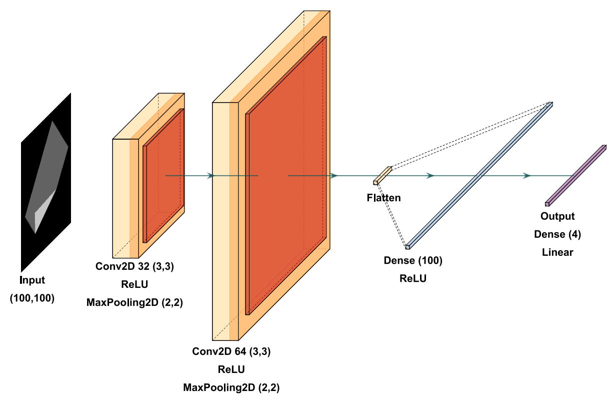

Figure 2: Architecture of the convolutional neural network used as a baseline.

[1]See https://blog.pal-robotics.com/tiago-bi-manual-robot-research/, accessed on March 19, 2025.
[2]See https://docs.ros.org/en/diamondback/api/kdl/html/python/, accessed on March 19, 2025.
[3]See https://github.com/roboticslab-uc3m/alma-dnn, accessed on May 5, 2025.

2.2.2. Algebraic Machine Learning

Algebraic Machine Learning (AML) (Martin-Maroto and de Polavieja, 2018) is a novel approach that hybridizes determinism and probabilism. It enables the explicit embedding of rules (deterministic) while simultaneously incorporating multiple data inputs (probabilistic). Furthermore, AML allows uncertainties to be explicitly represented within specific applications; however, this still relies on expert human knowledge (Fox and Victores, 2024). The core AML algorithms check that the human-defined constraints are maintained in the subsequent relationships between inputs and outputs that are learned during training (Martin-Maroto and de Polavieja, 2018). Human-defined rules are enacted through explicit instances of the “$<$” inclusion relationship operator, which is the same operator used to aggregate large quantities of data to the model and can be used at any time. The basic building blocks are AML “constants”. They are called constants because they are always present in an AML model, i.e., they are present in the model constantly. Constants can be expressed in natural language so they can be understandable for model users. At the same time, constants are the primitives used by the algebra and can be combined within the model via the “$\odot$” merge operator to be sets of constants called “atoms”. Training algorithms lead to the “freest model” of an algebra, which accurately describes the system through the minimum set of rules (through the reduction of “cardinality”). This is achieved by the trace-invariant “crossing” of atoms derived from the original constants. The “full crossing” training algorithm involves internally checking all the combinations of atoms of a training batch to establish an algebra, while the “sparse crossing” algorithm includes heuristics for the efficiency of these operations without the loss of generality.

The implementation code is provided in an open-source repository[4], that contains the applied algorithm.

3. Methodology

Two different environments have been developed for this task, both implemented as Gymnasium[5] (formerly OpenAI Gym) environments. Gymnasium provides a lightweight abstraction layer that cleanly decouples environments from perception algorithms in the code. This makes it possible to switch between environments when testing an algorithm, or to test different algorithms on the same environment. To improve code and algorithm reusability, all environments have been designed to share common observation and action spaces. They have been released and open-sourced under a common repository[6].

The following sections detail the perception process and describe both environments.

3.1. Perception of pick and place points

The input to the CNN or the AML consists of a 100x100 image, while the output consists of four values representing the coordinates of the pick and place pixels. Each input image is a three-channel representation of a rectangular sheet of cloth, which can vary in size, proportions, angular orientation, and may be folded along arbitrary lines.

For the input image, we leverage from previous developments (Estevez et al., 2020), which provide an image that has been transformed from a depth point cloud in color in camera coordinates to a very simplified flat matrix, the rectified depth-map.

Figure 3 illustrates the part of the pipeline of transition from full depth point cloud to the flat segmented garment, where height clusters are obtained via the watershed algorithm[7].

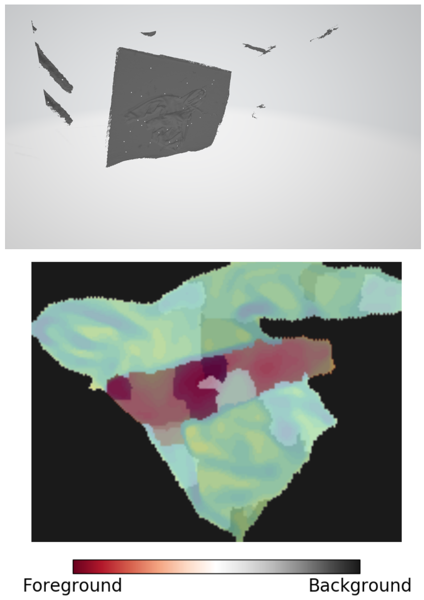

Figure 3: Transformation from full depth point cloud to the segmented garment (background removal) laying flat with obtained height clusters.

The garment surface is provided as a flat discrete image, where each pixel corresponds to a discrete value that marks to which layer (merged similar height clusters) it belongs (lowest is without garment, first height involves one layer of cloth, second height involves two layers of cloth e.g. due to a fold).

On the other hand, for the use of the AML algorithm, images were encoded using three “constants” for each of the 10,000 pixels. Each of the four output fields was modeled as the idempotent summation of a constant in an ascending chain and another constant in a descending chain.

According to this embedding method for real numbers, a value x is represented as:

$$g_x \odot l_x,$$

where l_x is a constant in an ascending chain (i.e., $l_x \leq l_{x+1}$) and g_x is a constant in a descending chain (i.e., $g_{x+1} \leq g_x$). Since there are 4 output fields, 8 chains are required.

Training was conducted in a supervised manner, and an algebraic model M was derived. The output for an image term I

[4] `https://github.com/roboticslab-uc3m/alma-aml`, accessed on May 5, 2025.
[5] `https://github.com/Farama-Foundation/Gymnasium`, accessed on March 19, 2025.
[6] `https://github.com/roboticslab-uc3m/gymnasium-alma`, accessed on May 5, 2025.
[7] `https://scikit-image.org/docs/0.24.x/auto_examples/segmentation/plot_watershed.html`, accessed on March 19, 2025.

was computed as follows; for each of the four fields, we calculated:

$$x = argmin_i |dis_M(g_i \odot l_i, I)|.$$

3.2. Folding 2D

The environment named `FakeFolding-v0`, provides testing against a ground truth labeled pairs dataset (Victores and Jardon Huete, 2025). Upon use, it loads a random image from the test dataset, as well as its corresponding ground truth. The environment evaluates the accuracy of estimated pick-and-place points with the perception algorithms over images from the labeled dataset. Observations are 100×100 grayscale images (0–255); actions are four coordinates. The reward is the squared error vs. ground truth.

3.3. Folding with TIAGo++

The `TiagoFolding-v0` builds upon `FakeFolding-v0`, implementing a cloth folding task using the TIAGo++ robot. It can operate both to the simulation environment (see Figure 4) and with the real robot (see previous Figure 1).

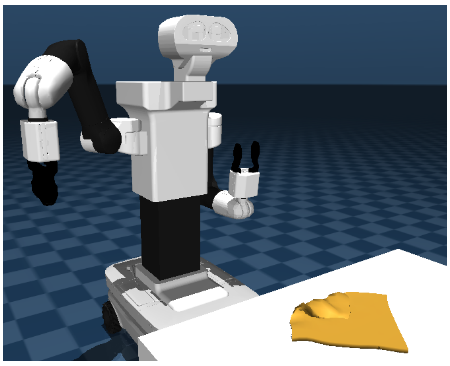

Figure 4: Screenshot of `TiagoFolding-v0` connected to TIAGo++ robot simulator.

In the setup, MuJoCo[8] is used to simulate the TIAGo++ robot[9], which interacts with a deformable cloth object. However, in both the simulation and real robot implementations, movements are executed via joint position commands communicated through ROS.

The observation space is defined as a Box of 100x100 representing grayscale pixel values in the range 0-255. The action space is also a box containing four values corresponding to the pick-and-place coordinates. Then, it initializes by taking a grayscale image of the cloth object using the camera of the TIAGo++ rectified and adapted as described at the beginning of this Section.

The step function receives the pick-and-place points detected from the image using the presented solvers algorithms and initiates the folding procedure. The first step involves determining the 3D positions of these points using depth information from the RGB-D camera and its intrinsic/extrinsic parameters and robot kinematics.

After transforming the image coordinates into 3D space, four key poses—pre-pick, pick, pre-place, and place—are computed using inverse kinematics. To achieve this, PyKDL, along with the TIAGo++ robot's kinematic model, enables the specification of end-effector poses. The positions of the pick and place poses correspond to the previously computed points, while the pre-pick and pre-place positions are set 0.15m above them. Then, the orientation for these poses are defined as follows:

- Roll: Set to π to ensure the gripper aperture remains horizontal.
- Pitch: Set to $\pi/2$ so that the fingertips make contact with the fabric.
- Yaw: Adjusted to align perpendicularly with the line connecting the pick and place points, ensuring the fabric corner is covered, as it is shown in the Figure 5.

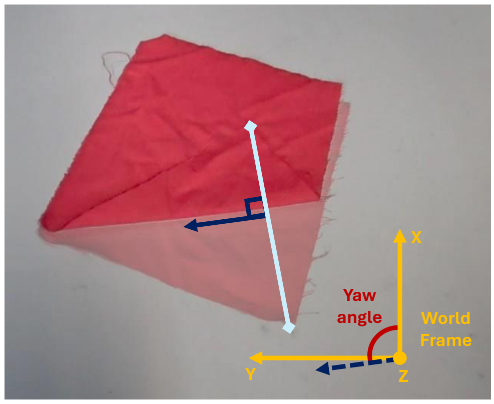

Figure 5: Yaw angle estimation for orientation in folding poses.

Since achieving the exact desired position and orientation may not always be feasible, a relaxation mechanism is applied. A tolerance of ±10 degrees is considered for pitch and yaw angles. If the mentioned orientation is not feasible, inverse kinematics is computed for alternative angles within this range until four valid poses are found or the maximum number of attempts is reached.

Once the valid poses are determined, the robot proceeds with the pick-and-place operation, as illustrated in the motion sequence in Figure 6, following these steps:

1. Moves to the pre-pick pose using the joint values obtained from inverse kinematics.
2. Executes a cartesian descent along the vertical axis until a variation in the vertical axis force is detected, indicating contact with the fabric.
3. Closes the gripper to grasp the fabric corner.
4. Returns to the pre-pick pose and transitions to the pre-place pose.

[8] See `https://mujoco.readthedocs.io/en/stable/overview.html`, accessed on March 19, 2025.

[9] See `https://github.com/pal-robotics-forks/mujoco_menagerie/tree/140ae8d30b430d9d8d8f0c42e031b93b59cb2968/pal_tiago_dual`, accessed on March 19, 2025.

5. Performs another Cartesian descent until again a variation in the vertical axis force is detected, which means a contact with the placement surface.
6. Opens the gripper to release the fabric.
7. Returns to the pre-place pose and then to its initial configuration, completing the folding task.

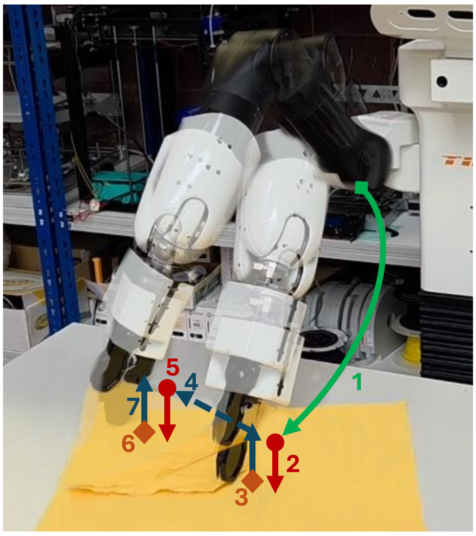

Figure 6: Pick-and-place routine: (1) move to pre-pick, (2) descend, (3) grasp, (4) move to pre-place, (5) descend, (6) release, (7) return to initial pose..

This structured approach ensure precise execution of the folding procedure while maintaining adaptability to variations in the workspace (handled by a proper workspace calibration).

Finally, the reward is computed based on the error measurement using the concept of Rectangularity (feature-based learning as performed in Victores (2014)), calculated as the ratio between the area of the contour and the area of the minimum bounding box.

4. Results

This section presents the experiments conducted, which are divided into two categories. First, by the use of the `FakeFolding-v0` environment, the AML and CNN algorithms are evaluated on a dataset of images with known ground truth, and their performance is assessed by measuring the prediction error. Second, the complete folding task is executed using the AML algorithm and the `TiagoFolding-v0` environment, evaluating rectangularity reward outcome.

4.1. AML and CNN evaluation

The experiments in this environment involved testing various images from a dataset with known ground truth. Pick and place pixel points were obtained using both the CNN and AML solvers, and the error was evaluated by comparing the solver outputs to the ground truth. This error was computed over 10 runs, and the resulting mean and standard deviation are presented in Table 1. The “misses” value given for the AML results can be interpreted as an uncertainty metric of the output.

Table 1: Comparison of errors between Ground Truth and solver results over 10 runs using dataset images. Mean and standard deviation (std) of the errors are calculated.

Error Type	**DNN**	**AML**
Pick X Error	3.1 ± 1.3	2.3 ± 1.4 (misses 1.2 ± 0.7)
Pick Y Error	1.7 ± 0.8	4.1 ± 1.9 (misses 0.9 ± 0.5)
Place X Error	2.2 ± 1.2	5.2 ± 2.0 (misses 1.5 ± 0.9)
Place Y Error	0.3 ± 0.9	8.2 ± 2.1 (misses 2.4 ± 1.1)s

4.2. Folding task performance

Experiments on folding with TIAGo++ were conducted with the real robot, with pick-and-place pixel points estimated using the AML algorithm. The manipulation process is illustrated in Figure 7, where the resulting Rectangularity value over 10 runs was 97 ± 2%.

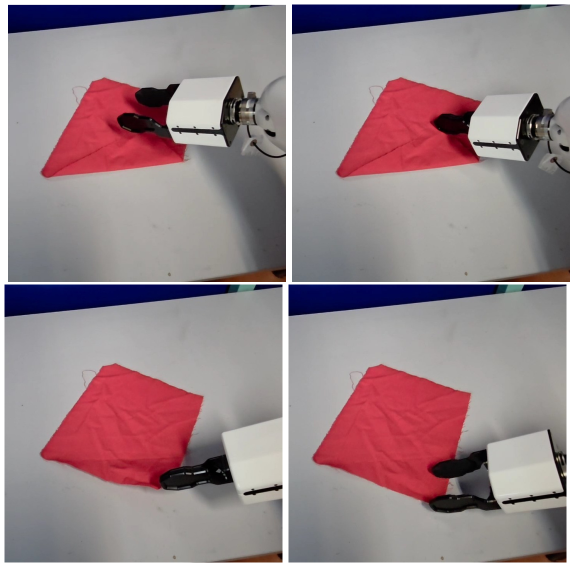

Figure 7: Sequence of manipulation process.

5. Conclusions

In summary, this work presents a functional minimum viable demonstration of cloth folding using the TIAGo++ bimanipulator robot. A novel Algebraic Machine Learning algorithm has been integrated and applied to a regression task to estimate pick and place points, which are then used in a robotic manipulation routine. Both the perception algorithm and the complete folding task (perception and manipulation) have been evaluated through a series of experiments, showing positive results.

Acknowledgments

This research has received funding from the project “ALMA: Human Centric Algebraic Machine Learning”, funded from the European Union’s Horizon 2020 research and innovation program, under grant agreement 952091; it has received funding from ”iRoboCity2030-CM Cont” (TEC 2024/TEC-62) of Comunidad de Madrid, ”iREHAB” (DTS22/00105) of Instituto de Salud Carlos III, and EU structural funds; and from the Carlos III University of Madrid (Escuela de Doctorado).

References

Borràs, J., Alenyà, G., Torras, C., 6 2020. A grasping-centered analysis for cloth manipulation. IEEE Transactions on Robotics 36, 924–936.
DOI: 10.1109/TRO.2020.2986921

Chen, W., Lee, D., Chappell, D., Rojas, N., 2023. Learning to grasp clothing structural regions for garment manipulation tasks. IEEE International Conference on Intelligent Robots and Systems, 4889–4895.
DOI: 10.1109/IROS55552.2023.10342086

Estevez, D., Fernandez-Fernandez, R., Victores, J. G., Balaguer, C., 6 2017a. Robotic ironing with a humanoid robot using human tools. 2017 IEEE International Conference on Autonomous Robot Systems and Competitions, ICARSC 2017, 134–139.
DOI: 10.1109/ICARSC.2017.7964065

Estevez, D., Victores, J. G., Fernandez-Fernandez, R., Balaguer, C., 12 2017b. Robotic ironing with 3d perception and force/torque feedback in household environments. IEEE International Conference on Intelligent Robots and Systems 2017-September, 6484–6489.
DOI: 10.1109/IROS.2017.8206556

Estevez, D., Victores, J. G., Fernandez-Fernandez, R., Balaguer, C., 1 2020. Enabling garment-agnostic laundry tasks for a robot household companion. Robotics and Autonomous Systems 123, 103330.
DOI: 10.1016/J.ROBOT.2019.103330

Fox, S., Victores, J. G., 5 2024. Safety of human–artificial intelligence systems: Applying safety science to analyze loopholes in interactions between human organizations, artificial intelligence, and individual people. Informatics 2024, Vol. 11, Page 36 11, 36.
DOI: 10.3390/INFORMATICS11020036

Fu, T., Bai, Y., Li, C., Li, F., Wang, C., Song, R., 12 2023. Human-robot deformation manipulation skill transfer: Sequential fabric unfolding method for robots. IEEE Robotics and Automation Letters 8, 8454–8461.
DOI: 10.1109/LRA.2023.3329768

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep learning, 1–23.

Gu, N., He, R., Yu, L., 2 2024. Learning to unfold garment effectively into oriented direction. IEEE Robotics and Automation Letters 9, 1051–1058.
DOI: 10.1109/LRA.2023.3341763

Jia, B., Pan, Z., Hu, Z., Pan, J., Manocha, D., 4 2019. Cloth manipulation using random-forest-based imitation learning. IEEE Robotics and Automation Letters 4, 2086–2093.
DOI: 10.1109/LRA.2019.2897370

Lee, D., Chen, W., Chen, X., Rojas, N., 3 2024. G.o.g: A versatile gripper-on-gripper design for bimanual cloth manipulation with a single robotic arm. IEEE Robotics and Automation Letters 9, 2415–2422.
DOI: 10.1109/LRA.2024.3357028

Luque, A., Parent, D., Colome, A., Ocampo-Martinez, C., Torras, C., 7 2024. Model predictive control for dynamic cloth manipulation: Parameter learning and experimental validation. IEEE Transactions on Control Systems Technology 32, 1254–1270.
DOI: 10.1109/TCST.2024.3362514

Martin-Maroto, F., de Polavieja, G. G., 3 2018. Algebraic machine learning.
DOI: 10.48550/arXiv.1803.05252

Naranjo-Campos, F. J., De Matías-Martínez, A., Victores, J. G., Gutiérrez Dueñas, J. A., Alcaide, A., Balaguer, C., 2024a. Robot tiago para servicio en cafetería. Jornadas de Automática (45).

Naranjo-Campos, F. J., Matías-Martínez, A. D., Victores, J. G., Dueñas, J. A. G., Alcaide, A., Balaguer, C., 8 2024b. Assistance in picking up and delivering objects for individuals with reduced mobility using the tiago robot. Applied Sciences 2024, Vol. 14, Page 7536 14, 7536.
DOI: 10.3390/APP14177536

Nilwong, S., Butdee, P., Pongthanisorn, G., Capi, G., 2023. Vision-based yakiniku recognition system for autonomous cooking robot. IEEE International Symposium on Robotic and Sensors Environments, ROSE 2023 - Proceedings.
DOI: 10.1109/ROSE60297.2023.10410807

Victores, J. G., 2014. Robot imagination system. Ph.D. thesis, Universidad Carlos III de Madrid.
DOI: 10016/19834

Victores, J. G., Jardon Huete, A., Feb. 2025. Alma wp7 folding.
DOI: 10.5281/zenodo.14864393

Wang, Y., Yang, L., Zhou, P., Qi, J., Lu, L., Zhu, J., Pan, J., 2024. Efficient planar fabric repositioning: Deformation-aware rrt for non-prehensile fabric manipulation. IEEE Robotics and Automation Letters.
DOI: 10.1109/LRA.2024.3484131

Yew, C. H., Sahari, K. S. M., Neoh, T. H. D., 7 2018. Parametrized 2d/3d garment model for dexterous robotic manipulation. Proceedings - 2018 Joint 10th International Conference on Soft Computing and Intelligent Systems and 19th International Symposium on Advanced Intelligent Systems, SCIS-ISIS 2018, 287–291.
DOI: 10.1109/SCIS-ISIS.2018.00058

Zhu, Z., Guo, Z., Hu, Q., Zhou, Y., He, B., 2025. Dual-arm fabric manipulation learning with grasp pose constraints. IEEE Robotics and Automation Letters.
DOI: 10.1109/LRA.2025.3541911