Cloth mechanical parameter estimation and simulation for optimized robotic manipulation

N. E. Anatoliotakis
Department of Electrical and Computer Engineering
Univeristy of Patras
n.anatoliotakis@ece.upatras.gr

P. Koustoumpardis
Department of Mechanical Engineering and Aeronautics
Univeristy of Patras
koust@upatras.gr

K. Moustakas
Department of Electrical and Computer Engineering
Univeristy of Patras
moustakas@ece.upatras.gr

Abstract

In this article a method for the estimation of cloth simulation parameters is presented. We propose a method, based on already published methods from different fields, that can successfully create the mechanical model of a cloth, based only on a single monocular video source of a cloth been held and moved in the air by two hands. We propose the use of a moving graph generation method using Scale Invariant Feature Transformation (SIFT). Having the moving graph of the real cloth as the goal, a method based on genetic algorithms was designed to produce the mechanical properties of the cloth's mechanical model. This way a simulated cloth with similar mechanical properties will be created. For our experiments a mechanical model based on Provot's mass-spring-damper (MSD) cloth model with adjustable springs and dampers was used. However, we present a method that can be easily adjusted to any particle-based cloth model. The method presented was designed to be easily applicable so as to enable the broader use of cloth models in robotized cloth manipulation tasks. The use of a cloth's digital twin, enables the major part of tuning of a robot controller to be made offline. This will significantly accelerate the tuning process, enabling the broader use of robots in more delicate cloth manipulation tasks. Finally, to prove the validity of our method we provide the results of experiments with cloths of different patterns and physical parameters.

1. Introduction

The textile industry has seen significant changes in terms of automation of tasks in the past few years. Many factories have now turned into robots to perform tasks that traditionally were performed by humans. However, this transition in most of the cases comes for tasks where the mechanical parameters of cloth can be surpassed. In these types of tasks, the cloth is considered to be rigid by the automated system. One way to achieve this, is to stiffen the cloth by stretching it and holding it in place. Another way to achieve stiffness is by changing temporarily the mechanical parameters of the cloth. A way of temporary stiffening is to dampen and then cool the cloth till it is rigid. However, most of cloth manipulation tasks require the cloth in its physical form. These tasks in a vast majority are still performed by humans. To properly automate these types of tasks, the mechanical model of cloth is needed. Given its mechanical model, the movement and the general behaviour of a cloth can be estimated. This information can then be used to tune a controller towards a desired cloth movement or towards minimizing the damage on the cloth.

Currently there are two notable methods that can accurately measure a cloth's mechanical parameters. The first is the Kawabata Evaluation System (KES) [13], which was the first method developed for this reason. Performing tensile, shear, pure bending, compression, surface friction and roughness tests, this method can measure a cloth's mechanical parameters with significant accuracy. The other method available is the Fabric Assurance by Simple Testing (FAST) method [7].This method applies a set of instruments and test procedures in a similar manner to KES, while also this method is cheaper and simpler to use compared to KES. These systems, due to their high price and their low accessibility, are not applicable for most cases of cloth manipulation outside large factories. In order to use the mechanical model of a cloth on an automated controller, we need simpler and more cost-efficient methods. This is the reason researchers in the past few years have developed methods of estimating a cloth's mechanical model that require less special equipment. Such research work is the one by P. Koustoumpardis et al. [14] who managed to perform automated tensile tests using a robotic arm and a force sensor. This research work made a part of the calculations needed for an accurate cloth simulation able to be performed directly by a robot without the need for special equipment. However, there is still room for improvement in terms of applicability and overall accuracy.

For the above-mentioned reasons, we present a novel method for the estimation of the mechanical parameters of a cloth, towards an accurate cloth simulation. This method is projected to be used for the creation of the digital twin of any cloth. This will enable the tuning of a robot controller for a cloth manipulation task, to be performed in its vast majority on a simulated environment. Thus many cloth manipulation tasks, requiring more accurate movement, will be able to be robotised. Our approach is based on graph matching to generate the moving graph of a cloth and then through a genetic algorithm it can estimate the corresponding mechanical model of the cloth. Contrary to most of the methods requiring to a greater or lesser extent special equipment, our method requires only a single web camera.

Our contribution can be summed up in three aspects:

1. We apply for the first time a graph matching method for the purpose of the estimation of a mechanical model of a cloth.
2. We present a non-model-depended method that can be applied with minor changes in any type of cloth model.
3. We present a method requiring the least amount of equipment, enabling its implementation outside of large factories or laboratories.

2. Related Work

The scientific challenges, faced by the proposed approach, rise from the research fields of *deformable object tracking*, *cloth modelling* and *genetic algorithms*.

2.1. Deformable Object Tracking

A study that was one of the first that presented a method for efficient deformable object tracking, was that of C. Bregler et al. [3].In this study, the 3D shape, in each frame of a video, was considered a linear combination of basis shapes. At the time the study published, it was the first to introduce a method for efficient 3D shape recovery without prior knowledge of a model. However, the results of their study were limited in human face and animal tracking. M. Salzmann et al. in their paper [23] presented a method for the construction of the 3D shape of a deformed surface by assembling linear local models of the surface. Using the SIFT (Scale Invariant Feature Transformation) method, they matched the 3D points to their corresponding points in the surface template image.

The above-mentioned studies were an inspiration for numerous future studies in this field. S. Vicente and L. Agapito [29] proposed a template-free method of surface reconstruction, based only on two monocular images. DT Ngo et al. [20] presented a method for 3D shape recovery using Laplacian meshes. This method reduced the dimensions of this problem and hence enabled real-time solutions. A study published later from A. Chhatkuli et al. [4] introduced a new method for 3D shape reconstruction from template based on the integration of first-order quantity estimations such as depth gradient and surface normal. Towards more efficient 3d reconstructions, S. Parashar et al. [21] developed a method that could find correspondences with greater efficiency than the methods previously presented.

One of the most recent studies to address this problem is the one of T. Wang et al. [31]. In that study authors presented a novel approach of 3D shape reconstruction by tracking a deformable surface using graph matching. By dividing the surface to equal parts and applying SIFT method to each of them, they managed to find correspondences with the template and by using the method they presented, they managed to track the surface with greater accuracy and speed.

2.2. Cloth Modelling

To be able to research a cloth's behavior, its mechanical model is needed. Thus, methods for cloth modelling were presented even when computers were not able to properly visualize them. Terzopoulos et al. [27] with their research work about the mechanical representation of a cloth laid the foundations for many future research works on this subject. One common method of cloth modelling is the mass-spring-damper (MSD) model. One of the first studies to present this model was the study of X. Provot [22].This method represents the cloth as a structure of interconnected nodes of definite mass. The connections between them can be modelled using springs and dampers. Another approach on cloth modelling is using finite element method (FEM). Based on thin shell theory, the cloth is able to be represented as discrete interpolated patches. FEM can produce greater results in terms of accuracy but requires significant computational time. JR.Collier et al. [6] simulated a cloth's behavior using this method, while O. Etzmuß et al. [8] presented a faster more efficient method that became the basis of many future research works. Particle-based system method is the third main type of cloth simulation methods. This method represents the cloth as a group of particles. A cloth's behavior can then be simulated by applying forces on them. D. Breen [2] presented a method based on this principle, while also that method enabled the direct use of KES measurements to produce the simulation of a cloth.

Except of the above mentioned methods, significant progress have been also made in the development of dynamic simulation methods. Towards a point based deformable object dynamic simulation method, K. Moustakas et al. [19] presented a framework which can achieve both real-time collision detection and haptic interaction using superquadric virtual object modeling. By creating an SQ-Map instead of using traditional computational methods, this method managed to accelerate the process by reducing the computational cost and the amount of data used. In the same sense, A. Vogiannou et al. [30] with their work, improved the performance of Bounding Volume Hierarchies method for collision detection. They presented an algorithm based on the concept of support planes that can outperform the typical Bounding Volumes method in terms of efficiency and speed.

Recent studies have managed to model a cloth at the yarn level. One of the first to present a cloth modelling solution at yarn level was JM. Kaldor et al. [12]. In their study focused on the modelling of knitted cloths, which due to their structure present different mechanical properties than those of woven cloths. Their model was consisted of inextensible yet flexible B-spline tubes. G. Cirio et al. [5] presented a yarn-level modelling method that can be applied both in woven and knitted cloths. It was mainly focused on the yarn-yarn contacts and treated them as persistent with a possibility of sliding. Thus, they managed to present a significantly faster method than the others available at the time. Z. Montazeri et al. in their study [18] presented a method to generate photorealistic and mechanically accurate cloth model at yarn level. They also introduced a method using deep neural networks to match yarn-level deformation to fiber reconfiguration.

2.3. Genetic Algorithms

Genetic algorithms firstly introduced by J. Holland [9]. This is a method mainly used in problems in which a deterministic approach is sincerely time-consuming. Following the principle of C. Darwin, this algorithm selects from each iteration the best candidates and then produces new candidates based on these. This algorithm has been used for numerous different problems. A method for image segmentation presented by B. Bhanu et al. [1], utilized the genetic algorithm method to adapt the segmentation process in changes caused by environmental conditions. P. R. Srivastava et al. [25] used genetic algorithms to identify the most error-prone parts of a program for software testing. Y. Sun et al. [26] developed a method that by using genetic algorithms can design a CNN structure for image segmentation tasks.

However, genetic algorithms are also widely used for robotic tasks. Hu Yanrong [10] presented a knowledge-based genetic algorithm for the path planning of mobile robot. This method can be used both on static and dynamic environment, while it can also be used for multi-robot path planning by considering other robots as moving obstacles. In the same sense, Ismail [11] developed another method based on genetic algorithms for mobile robot path planning. A method presented by CC Tsai [28] used two parallel elite genetic algorithms (EGA) and a genetic operator. This method can produce a feasible path that is then smoothed towards an real world application of autonomous driving.

2.4. Cloth Simulation for Robotic Manipulation

Due to the importance of an accurate simulation for robotized manipulation of a deformable object, several studies were conducted, with lots of them presenting significant results. Y. Li et al. [15] presented a method for the classification of different types of cloths and the estimation of their poses. Their method using the simulation of each cloth can produce large sets of depth images that they then use to retrieve useful information for each cloth. These data are then used to identify the cloth and estimate its pose. This method was tested in robotized manipulation tasks such as regrasping and folding. In an other study, Y. Li et al. [16], further improved the aforementioned method. They reduced the cost by using a low-cost depth sensor, while also they further improved the accuracy of their method by applying different processes for feature extraction. Similar to the two previously referred works, I. Mariolis et al. [17] presented a method based on deep convolutional neural networks to identify the type of a cloth and estimate its pose. They evaluated their method using both real and synthetic depth images. The experiments of these studies prove the potential of the use of cloth simulation in robotized cloth manipulation tasks.

3. Overview

The method we propose 1 consists of two steps. We first obtain a video of a cloth been manipulated. As a video capturing device, we use a single typical 720p web camera. The manipulation of the cloth can be performed either by a human or by cooperative robotic devices. From the monocular video we then extract the state of the cloth on each time frame. The state of the cloth can be represented by its moving graph. So, we generate a moving graph for each time frame of the manipulation process, using a simplified version of a method [31] presented by T. Wang et al. that used this method to track a deformed surface in a video. Thus,

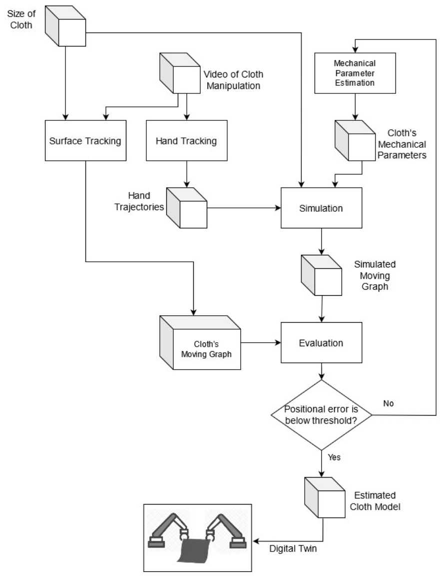

Figure 1. General workflow of the system proposed

using the template of a cloth and a Scale Invariant Feature Transformation (SIFT) process, the position of the cloth can be retrieved. A SIFT with the whole template of the cloth has to be performed to identify the cloth in the initial video frame and generate the initial state of the moving graph. Then, for each of the following frames, the template of the cloth divided in equal parts and the same process is performed for each of the generated nodes with its according part of the template. To properly run the simulation, the trajectories of the points in the camera space from which the cloth is held are also needed. In case of robotized manipulation, these trajectories can be directly retrieved from the trajectory planning of robot's end effector. However, when the cloth is manipulated by a human user, the recovery of these trajectories is achieved by tracking user's hands and in particular the two points from which he holds the cloth. This is achieved using the method presented by T. Simon et al. [24] which can track up to 20 key points on a hand and hence give a good approximation of its pose.

The system can evaluate its estimation by applying them directly to a simulated mechanical model of a cloth. This model based on a mass-spring-damper (MSD) model will undergo the same exact manipulation as the real one had. As depicted in 1 the moving graph of the simulated model will then be compared to the moving graph generated from the video and their positional distance will serve as the fitness function of our genetic algorithm method. Once the whole process is completed, a digital twin of the cloth will have been created. Then using a reinforcement learning tuning process, a robot controller can be properly tuned. This controller will be specially designed to perform a task with as much accuracy as possible for the corresponding cloth.

4. Graph matching method for cloth tracking

To effectively track the cloth in each frame we used a simplified method of the one presented by T. Wang [31].

The cloth can be tracked using its template of shape S_T . This known template of shape S_T can be represented as a graph of $M \cdot N$ interconnected nodes $\{n_{i_j} = [x_{i_j}, y_{i_j}], 1 \leq i \leq N, 1 \leq j \leq M\}$. To track the cloth, we use a video of N_F frames. We will refer to the shape of the deformed cloth in each frame as $\{S_{F_t}, 1 \leq t \leq N_F\}$.

At first, a SIFT algorithm is applied to detect correspondences between the slightly deformed shape S_{F_0}, in the initial frame space F and the shape S_T in template space T. Since the deformation of cloth can be changed from frame to frame, it can be represented as an equation mapping each point P_T of space T to a point P_{F_t} of space F: $d, \mathbb{R}^2 \rightarrow \mathbb{R}^w$. Since our final goal is only to compare the moving graph of the simulated cloth to the moving graph of the real cloth so as to provide the necessary feedback to the reinforcement learning system, there is no need to convert our 2D points to 3D.

The m detected correspondences between space T and space F can be stored as a set of points, $C : \{P_{T_i}, P_{F_{1_i}}\}_{i=1}^{m}$. Given that the user holds the cloth still and slightly stretched, we can assume that the cloth deformation at the initial frame is negligible. So, we consider S_{F_0} to be equal to S_T for the initial frame. Then from the detected correspondences we can estimate the homography between space T and space F. Homography will estimate the position of the shape S_T in space F. Then, by dividing S_T to $(M-1) \cdot (N-1)$ equal parts, we generate the graph representing the shape of the cloth. The initial position of each node n will be set so as to be evenly placed across the shape S_{F_0}.

To effectively generate the moving graph of the cloth in all frames we will need to use a new moving graph of $(M-1) \cdot (N-1)$ nodes. From now on we will refer to the nodes of this moving graph as $\{w_{i,j_t} = [x_{i,j_t}, y_{i,j_t}], 1 \leq i \leq N-1, 1 \leq j \leq M-1, 1 \leq t \leq N_F\}$. The position of each node will be the center of each of the equal parts of S_{F_0}. This representation has been selected instead of the original moving graph of $(M-1) \cdot (N-1)$ nodes, due to the way the graph matching algorithm we used works. This algorithm requires a rectangular part, consisting of four nodes, to be able to produce a new position for w_{i,j_t}.

For each of the following frames a SIFT algorithm will be applied to find the correspondences between a rectangular window around each w_{i,j_t} and its corresponding part of the template S_T. We will refer to the rectangle of each node w_{i,j_t} as $\{r_(i,j), 1 \leq i \leq N-1, 1 \leq j \leq M-1\}$. Let the distance between two consecutive nodes of the same line be l, the size of the rectangle window can be represented as $\{d \cdot d, l \leq d \leq 2l\}$.

This time the SIFT algorithm, unlike its application in the initial frame where we retrieved all the correspondences, it will only keep the position in S_{F_t} where the best correspondence occurred. This position will then serve as the position of w_{i,j_t}. This happens because, by dividing the cloth in equal parts we significantly reduce the number of correspondences corresponding to each part. Hence, depending on cloth's pattern, it will not always be possible to calculate homography for every part. For groups that no correspondences can be found in a specific frame, we will assume that $w_{i,j_t} = w_{i,j_{t-1}}$. This is inevitable; hence this method only depends on SIFT algorithm. Such phenomena mostly occur when the part of a cloth does not have a distinct pattern. However, because they are only observed in specific groups, it will not have a significant effect on our result.

5. Estimation of the mechanical parameters of a cloth

To effectively estimate the parameters of the cloth, we have designed a method based on genetic algorithms. The estimation sequence can be divided into two parts; the algorithm used for the parameter estimation and the simulation of the cloth that serves as the evaluation of the algorithm's results.

5.1. Estimation method based on genetic algorithms

For the estimation of the mechanical parameters needed for simulating the cloth, a method based on genetic algorithms was selected. This method can provide high-quality estimations of the parameters in lower time in comparison with other similar methods. In addition, due to its design 2 it can explore different solutions, even while converging to one. Thus, minimizing the possibility of converging in a local ultimo.

The values to be estimated are the spring constant of Structural and Shear springs, the spring constant of Flexion springs and the general damper value between two interconnected nodes of the cloth. Each of these values, given upper and lower bounds, can be represented by a 16-bit number. A set of these values serves as a chromosome in the genetic algorithm. Thus, each of the algorithm's chromosomes consists of 48-bits. Starting, the algorithm generates 20 individual 48-bit strings representing a set of values. By using the fitness function, which in our case stands for the simulation of the cloth, a score can be calculated for each set of values. The algorithm tries to maximize this score by generating better sets of values. The generation of each set of values is based on three main genetic operators: *Selection*, *Crossover* and *Mutation*. Our method directly adds the single best scoring set of values in the next batch of 20 48-bit strings. A product of *Mutation* applied to the best scoring 48-bit string, is also added to the next batch of strings. The other 18 strings will be added after applying all three abovementioned genetic operators.

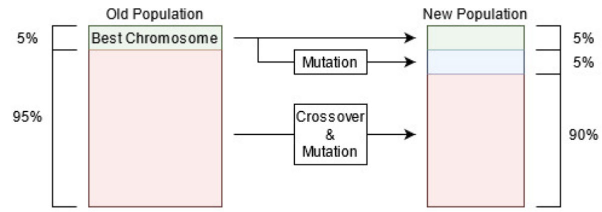

Figure 2. Reproduction strategy of our genetic algorithm method

The main goal of *Selection* is to generate sets of values based on the sets with the best score from the previous iteration. The method used for selecting the best values is Tournament *Selection*. The way it works is that it runs a tournament among several individual chromosomes from the previous iteration and selects those with the best values from each tournament. The selected 48-bit strings will then proceed to the *Crossover* and *Mutation* stages. This way the method will exclude the bad scoring sets from the production of the chromosomes of the next iteration.

Moving to the *Crossover* module the 48-bit strings are divided into pairs. From each pair two new chromosomes will then be generated. There are lots of ways of generating new chromosomes based on two older chromosomes, such as *Single Point*, *Two Point* and *Uniform Crossover*. In our method we used the *Single Point Crossover*, based on its speed advantage compared to the others. The Single Point crossover method selects a random crossover point from which each of the 48-bit strings will be divided. So, each string is divided into two pieces and then the opposite pieces from each string will then be merged to produce the two new 48-bit strings. This method is also designed not to perform crossover in a median rate of 10% and to pass the previous strings to the next generation without any further change. This is performed to further ensure the diversity of the chromosomes.

The last phase before generating the 18+1 new strings for the next iteration is the *Mutation* stage. *Mutation* stage is a decisive part towards the genetic diversity across iterations. In this stage, one or more bits of each chromosome string can be changed based on the *Mutation* probability we have set. By applying *Mutation* to a 48-bit string, its corresponding value can be significantly changed. Thus further reducing the probability of converging in a local ultimo.

When the score of the estimated values must be calculated, a transition has to be made. Thus, the 48-bit string is divided in 16-bit strings that will then be mapped, based on the bounds we have set, to real values. The new values will be used by the fitness function to calculate the score. In our case the simulation of the cloth serves as the fitness function.

5.2. Evaluation of the estimated parameters using a cloth's simulation

To be able to evaluate the output of our genetic algorithm we designed a cloth simulator. This simulator can effectively simulate the behavior of a cloth with defined mechanical parameters at a given environmental state.

For the purposes of our genetic algorithm sequence, a simulation of the cloth under the same manipulation is needed. Thus, an effective transition of the user's movement is mandatory. If the cloth is manipulated by a robot, then by tracking the robot's movements, we can easily obtain the position of its manipulators in each frame. When a user manipulates the cloth in the video, a hand keypoint detection method [24] was used. Hence, we can effectively track the points from which the user picks the cloth and then transfer them to our simulation environment.

Regarding the cloth model that we used to simulate the cloth; Provot's mass-spring-damper (MSD) [22] cloth model was used. This model consists of nodes of exact mass, equally distributed across the rectangular surface of the cloth, forming rows and columns. These nodes are connected with each other with mass-spring systems. Based on the nodes they are connecting there are three different types of springs.

1. *Structural Springs*, which connect nodes $[i, j]$ with $[i, j+1]$ and $[i, j]$ with $[i+1, j]$
2. *Shear Springs*, which connect nodes $[i, j]$ with $[i+1, j+1]$ and $[i+1, j]$ with $[i, j+1]$
3. Flexion Springs, which connect nodes $[i, j]$ with $[i, j+2]$ and $[i, j]$ with $[i+2, j+2]$

The spring constant k and the damping ratio ζ are the values of each connection that can be adjusted to model the behavior of the real cloth. In our experiments we considered Structural and Shear springs to have the same spring constant k_1, while Flexion springs had a different constant k_2. The damping ratio ζ considered to be the same for all the different springs. Hence, the coefficients that the genetic algorithm have to produce are the two different spring coefficients k_1, k_2 and the damping ratio ζ.

The fitness function of our genetic algorithm derives from the positional difference of the moving graph of the simulated cloth and the moving graph of the real cloth. Specifically, the positional error is calculated by calculating the positional errors of each node in every frame. Then, we calculate the mean error for each node and subsequently we calculate the mean of the mean errors of all nodes. This way we create a robust metric that will not be significantly affected by temporarily poorly tracked nodes. We will refer to the position of each node of the simulation in each time frame as $\{r_{i,j_t} = [x_{i,j_t}, y_{i,jt}, 1 \leq i \leq N-1, 1 \leq j \leq M-1, 1 \leq t \leq N_F\}$. The position of each node of the moving graph can be described as $\{w_{i,j_t} = [x_{i,j_t}, y_{i,j_t}], 1 \leq i \leq N-1, 1 \leq j \leq M-1, 1 \leq t \leq N_F\}$. Given the above, the mean accumulative positional error *mpe* serving as the evaluation metric, results by dividing the sum of the error of each time frame sd with N_F and can be described by the following equation:

$$sd = \sum_{i=1}^{N-1} \sum_{j=1}^{M-1} \sum_{t=1}^{N_F} \frac{|r_{i,j_t} - w_{i,j_t}|}{(N-1) \cdot (M-1)}$$

$$mpe = \frac{sd}{N_F}$$

6. Results

In order to prove the efficiency of the proposed system, several experiments were conducted. In particular, two experiments are presented in this section to both prove the system's stability and denote the condition in which system's results lack accuracy.

In these experiments, several different types of cloths were used. The cloths selected, differ both in size, pattern, color and mechanical properties. For each of the experiments, the same workflow was applied. At first a short monocular video was taken. The video depicts a man stretching and unstretching a cloth while moving it at the same time. From this video, using the algorithms we presented in previous chapters we are able to extract the trajectories of the points from which the user holds the cloth, while also the trajectories of the nodes which constitute the model of the cloth. The trajectories of the manipulation points will be used from the simulation program so that to reenact the movements of the cloth. The node trajectories will only be used by the end of the simulation for evaluation purposes. While the initial mechanical properties of the model of the cloth are randomly selected, for each other iteration these properties result from the genetic algorithm process. The genetic algorithm runs for 50 iterations and its output is the set of properties with the least positional error from the measurements of the real cloth.

First, the results of the initial cloth tracking are presented. The SIFT algorithm managed to efficiently track the cloth with significant accuracy. In order to be able to identify many keypoints from cloth's template, we used an image of greater resolution in comparison with the video frames. After matching all the keypoints, the system creates an outline of the cloth. The cloth in the initial frame is held in a straight position to be able to be outlined with the least amount of line segments. By the time the outline is designed 3, the area of the cloth is divided into equal $M \cdot N$ parts and the nodes are designed. The specific clothes presented are of size $60cm \times 30cm$. Hence we divide it in $20 \cdot 10$ equal rectangular parts. Each of those parts will be of size

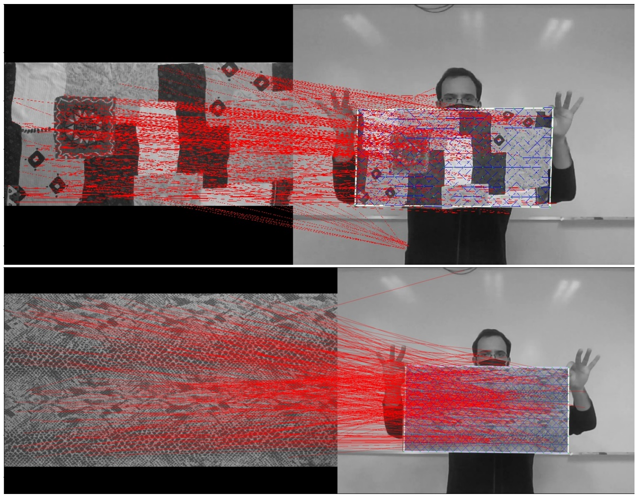

Figure 3. SIFT algorithm results and graph initialization on two different clothes

$3cm \times 3cm$. In the experimental results presented in 3, the correspondences between the template of the cloth (left) and the initial frame of cloth manipulation (right) are depicted in red line segments, while the nodes and the springs of the constructed graph are depicted using blue dots and blue line segments.

After the initial generation of nodes and connections, the graph matching method is applied to track the movements of the cloth and generate the moving graph. This moving graph will be the example with which the results of the simulation will be compared. Due to the method's unreliability in cases with no distinct patterns, the evaluation will only take into consideration the nodes for which it has high reliability. This means that, at any case the method cannot produce an accurate position about a node, the position of this node at that exact time frame will not be considered for the evaluation.

The next step of the overall method is the calculation of the mechanical parameters of the mechanical model of the cloth using genetic algorithms. In each iteration the genetic algorithm produces two spring coefficients k_1, k_2 and the damping ratio ζ. These values will be given as an input in the simulation and a moving graph of the simulated model will then be produced. The genetic algorithm will run for 50 iterations with a population of 20 different sets of values.

Due to an unavailability of the proper equipment, a proper measure of the mechanical properties of the real cloth was not possible, hence the evaluation of the results is only based on the difference between the real cloth and its simulation at an exact time frame. Regarding the experiments presented, first cloth was less stiff than the second one. This difference can be recognized both on the values of the mechanical properties but mainly on the resulted simulations. For the first more flexible cloth, the resulted simulation can depict the overall shape on each movement but fails to accurately depict the slight fold on the upper side of the second image 4. The *mpe* for the first cloth is 2.5 cm.

Simulation Parameters of the 1st cloth.
$k_1 = 4.258$ $k_2 = 16.065$ $\zeta = 0.049$
Mean Accumulative Positional Error
2.5 cm

Table 1. Results of the experiment with a flexible cloth.

Simulation Parameters of the 2nd cloth.
$k_1 = 4.258$ $k_2 = 16.065$ $\zeta = 0.038$
Mean Accumulative Positional Error
3.8 cm

Table 2. Results of the experiment with the stiffer cloth.

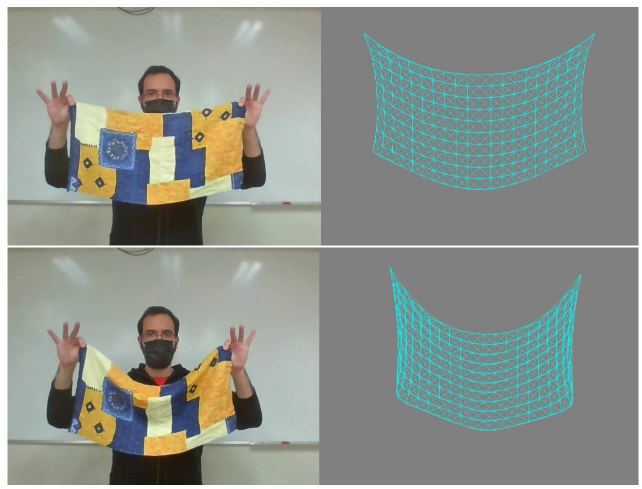

Figure 4. Two-frame comparison of the results for a flexible cloth.

The second cloth was significantly stiffer and while the simulation shares lots of similarities on the behaviour with the real cloth, fails to forecast the non-uniform folds 5. As expected, the *mpe* is higher than the *mpe* of the previous experiment, at 3.8 cm. This result proves the disadvantage of our MSD model to properly visualise the non-uniform folds of this cloth.

7. Conclusion

In this work, we proposed a complete method of estimating the mechanical model of the cloth. This work differs from others due to its applicability. The use of a monocular

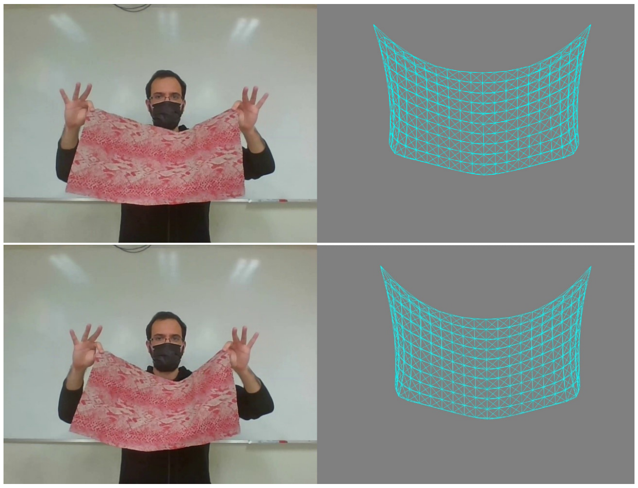

Figure 5. Two-frame comparison of the results for a stiffer cloth.

video source along with the low cost of the equipment, creates a reliable and affordable solution for cloth modelling applications. Our method can be applied only in fabrics with distinct pattern, because of the graph matching method used, that relies only in optical data. This work was also designed to make the use of a reliable cloth model easier for robotics applications. In the future, this method is projected to be used to create fast and reliable cloth models, in order to be used in offline tuning of automated controllers, designed for cloth manipulation tasks. Making accurate estimation of a cloth's mechanical parameters will accelerate the training of robots on cloth manipulation tasks, by enabling a significant amount of tuning to be performed offline. Hence, the traditional control methods based on reinforcement learning could be able to be performed in a simulated environment with accurate cloth and robot models. This will not only accelerate the overall process but also will make the training process safer by using the simulated rather than the real robot in the training phase.

References

[1] Bir Bhanu, Sungkee Lee, and John Ming. Adaptive image segmentation using a genetic algorithm. *IEEE Transactions on systems, man, and cybernetics*, 25(12):1543–1567, 1995.

[2] David E Breen, Donald H House, and Michael J Wozny. Predicting the drape of woven cloth using interacting particles. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 365–372, 1994.

[3] Christoph Bregler, Aaron Hertzmann, and Henning Biermann. Recovering non-rigid 3d shape from image streams. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 2, pages 690–696. IEEE, 2000.

[4] Ajad Chhatkuli, Daniel Pizarro, Adrien Bartoli, and Toby Collins. A stable analytical framework for isometric shape-from-template by surface integration. *IEEE transactions on pattern analysis and machine intelligence*, 39(5):833–850, 2016.

[5] Gabriel Cirio, Jorge Lopez-Moreno, and Miguel A Otaduy. Yarn-level cloth simulation with sliding persistent contacts. *IEEE transactions on visualization and computer graphics*, 23(2):1152–1162, 2016.

[6] Joan R Collier, Billie J Collier, Gina O'Toole, and SM Sargand. Drape prediction by means of finite-element analysis. *Journal of the Textile Institute*, 82(1):96–107, 1991.

[7] A De Boss. The fast system for objective measurement of fabric properties, operation, interpretation and application, 1991.

[8] Olaf Etzmuß, Michael Keckeisen, and Wolfgang Straßer. A fast finite element solution for cloth modelling. In *11th Pacific Conference onComputer Graphics and Applications, 2003. Proceedings.*, pages 244–251. IEEE, 2003.

[9] John H Holland. Genetic algorithms. *Scientific american*, 267(1):66–73, 1992.

[10] Yanrong Hu and Simon X Yang. A knowledge based genetic algorithm for path planning of a mobile robot. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, volume 5, pages 4350–4355. IEEE, 2004.

[11] AT Ismail, Alaa Sheta, and Mohammed Al-Weshah. A mobile robot path planning using genetic algorithm in static environment. *Journal of Computer Science*, 4(4):341–344, 2008.

[12] Jonathan M Kaldor, Doug L James, and Steve Marschner. Simulating knitted cloth at the yarn level. In *ACM SIGGRAPH 2008 papers*. SIGGRAPH, 2008.

[13] S Kawabatra. " the standardization and analysis of hand evaluation". *The Textile Machinery Society Japan*, 1980.

[14] Panagiotis N Koustoumpardis, John S Fourkiotis, and Nikos A Aspragathos. Intelligent evaluation of fabrics' extensibility from robotized tensile test. *International Journal of Clothing Science and Technology*, 2007.

[15] Yinxiao Li, Chih-Fan Chen, and Peter K Allen. Recognition of deformable object category and pose. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 5558–5564. IEEE, 2014.

[16] Yinxiao Li, Yan Wang, Michael Case, Shih-Fu Chang, and Peter 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.

[17] Ioannis Mariolis, Georgia Peleka, Andreas Kargakos, and Sotiris Malassiotis. Pose and category recognition of highly deformable objects using deep learning. In *2015 International conference on advanced robotics (ICAR)*, pages 655–662. IEEE, 2015.

[18] Zahra Montazeri, Chang Xiao, Yun Fei, Changxi Zheng, and Shuang Zhao. Mechanics-aware modeling of cloth appearance. *IEEE transactions on visualization and computer graphics*, 27(1):137–150, 2019.

[19] Konstantinos Moustakas, Dimitrios Tzovaras, and Michael Gerassimos Strintzis. Sq-map: Efficient layered collision detection and haptic rendering. *IEEE*

Transactions on Visualization and Computer Graphics, 13(1):80–93, 2006.

[20] Dat Tien Ngo, Jonas Östlund, and Pascal Fua. Template-based monocular 3d shape recovery using laplacian meshes. *IEEE transactions on pattern analysis and machine intelligence*, 38(1):172–187, 2015.

[21] Shaifali Parashar, Daniel Pizarro, and Adrien Bartoli. Robust isometric non-rigidstructure-from-motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[22] Xavier Provot et al. Deformation constraints in a mass-spring model to describe rigid cloth behaviour. In *Graphics interface*, pages 147–147. Canadian Information Processing Society, 1995.

[23] Mathieu Salzmann and Pascal Fua. Linear local models for monocular reconstruction of deformable surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):931–944, 2010.

[24] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1145–1153, 2017.

[25] Praveen Ranjan Srivastava and Tai-hoon Kim. Application of genetic algorithm in software testing. *International Journal of software Engineering and its Applications*, 3(4):87–96, 2009.

[26] Yanan Sun, Bing Xue, Mengjie Zhang, Gary G Yen, and Jiancheng Lv. Automatically designing cnn architectures using the genetic algorithm for image classification. *IEEE transactions on cybernetics*, 50(9):3840–3854, 2020.

[27] Demetri Terzopoulos, John Platt, Alan Barr, and Kurt Fleischer. Elastically deformable models. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 205–214, 1987.

[28] Ching-Chih Tsai, Hsu-Chih Huang, and Cheng-Kai Chan. Parallel elite genetic algorithm and its application to global path planning for autonomous robot navigation. *IEEE Transactions on Industrial Electronics*, 58(10):4813–4821, 2011.

[29] Sara Vicente and Lourdes Agapito. Soft inextensibility constraints for template-free non-rigid reconstruction. In *European conference on computer vision*, pages 426–440. Springer, 2012.

[30] Athanasios Vogiannou, Konstantinos Moustakas, Dimitrios Tzovaras, and Michael G Strintzis. Enhancing bounding volumes using support plane mappings for collision detection. In *Computer Graphics Forum*, volume 29, pages 1595–1604. Wiley Online Library, 2010.

[31] Tao Wang, Haibin Ling, Congyan Lang, Songhe Feng, and Xiaohui Hou. Deformable surface tracking by graph matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 901–910, 2019.