

A Tactile-based Fabric Learning and Classification Architecture

A.A. Khan¹, M. Khosravi¹, S. Denei¹, P. Maiolino², W. Kasprzak³, F. Mastrogiovanni¹, G. Cannata¹

Abstract—This paper proposes an architecture for tactile-based fabric learning and classification. The architecture is based on a number of SVM-based learning units, which we call fabric classification cores, specifically trained to discriminate between two fabrics. Each core is based on a specific subset of the fully available set of features, on the basis of their discriminative value, determined using the p-value. During fabric recognition, each core casts a vote. The architecture collects votes and provides an overall classification result. We tested seventeen different fabrics, and the result showed that classification errors are negligible.

I. INTRODUCTION

Humans deal with garments everyday: picking them from the wardrobe, folding them and making the laundry. These kinds of activities require the ability to perceive and recognize fabrics (e.g. the selection of the proper washing cycle according to the fabric type). Providing robot with the capability of manipulating and recognising fabric is still a challenging task. Indeed a huge effort has been done in literature in order to classify various kinds of materials (polycotton, nylon, silicone, brass, wood plastic, foam, and PVC to name, but few) using tactile sensors based on different transduction principles [1], [2], [3] but with respect to these examples, where materials have clear different geometric and mechanical characteristics, to classify fabrics is more challenging due to the high variability of existing types that in many cases could have really similar characteristics (i.e., consider for example a jumper that can be made from wool or acrylics). A multi-sensorial approach can be used for improving the fabric classification as in [4], where data coming from RGB-D, tactile, and photometric stereo sensors are used, but when only one sensor modality is available, the challenge is to find a fabric exploration technique that allow to detect all its discriminative characteristics and also to determine which are the sensor data features that are most effective for discriminating the different type of fabrics. Different approaches have been used to explore object properties, implementing different exploratory behaviours like tapping, sliding, pressing and rubbing [5], [6] showing that there is no predominant way to explore an object,

The research leading to these results has received funding from the European Community's Seventh Framework Program (FP7/2007-2013) under grant agreement n. 288553 (CloPeMa).

¹A.A.Khan, M. Khosravi, S. Denei, F. Mastrogiovanni and G. Cannata are with the Department of Informatics, Bioengineering, Robotics and Systems Engineering (DIBRIS), University of Genoa, Via Opera Pia 13, 16145, Genoa, Italy. Corresponding author's email: abdul.attayyab.khan@edu.unige.it.

²P. Maiolino is with the Computing Department, Goldsmiths College University of London, 8 Lewisham Way SE14 6NW, London, United Kingdom. P.Maiolino@gold.ac.uk.

³W. Kasprzak is with the Institute of Control and Computation Eng., Warsaw University of Technology, Nowowiejska 15/19, 00-665 Warszawa, Poland. W.Kasprzak@elka.pw.edu.pl.

and it usually depends on used sensing modality and on the features to extract. Object classification algorithms are based on a discriminative set of features defined in both time and frequency domains. Examples are mean, standard deviation, variance, kurtosis, skewness in the time domain [7], [8]. While in the frequency domain, the first three peaks of the Fourier components can form a useful feature vector [9], whereas dominant frequencies have been considered in [2]. Even in this case, there is no general agreement about which features to use.

Our work tries to investigate the above aspects in order to improve the classification of different types of fabrics using tactile sensing modality. In particular the major contribution of this paper is two-fold: (i) we propose a distributed fabric classification architecture where each component is specifically designed to discriminate between two fabrics, and different results are aggregated by a voting mechanism; (ii) we define, for each couple of fabrics, which are the best features (to be selected from a huge feature set, including both time and frequency domain features) to obtain good classification results. Each core is based on a support vector machine (SVM) classifier. The proposed approach selects the number of features using an active filter tuned during the training phase of the algorithm. Seventeen different kinds of fabrics are considered in the experiment. This paper is organised as follows. Section II describes the used tactile sensor. Data generation and feature extraction are discussed in Section III-B and Section III-C, respectively. Section III-D describes the classification approach. The experimental scenario and the results are discussed in Section IV. Conclusion follows.

II. THE REFERENCE TACTILE SENSOR

This work is based on the tactile sensor presented in [10] and shown in Figure 1a. The device has been developed to be integrated on an industrial gripper specifically designed for clothes manipulation and it embeds different sensing modalities, namely: a microphone, an ambient light, a proximity sensor and 16 capacitive pressure sensors constituted by circular pads of 2 mm diameter etched on the PCB with a 4 mm pitch (Figure 1b). The small dimension and the pitch of the pads have been selected in order to recognize small details of the manipulated garment. Over the PCB, a 100 μ m thick acetate sheet acts as a dielectric layer and over it, a ground plane, made by conductive fabric, covers all the pressure sensors completing the capacitor structure. Two capacitance to digital converters (AD7147 from Analog Device) encode the measured pressure in 16 bits values at a rate of 37 Hz. For the classification purpose, only the 16 pressure profiles measured by these capacitive sensors during fabric manipulation will be processed.

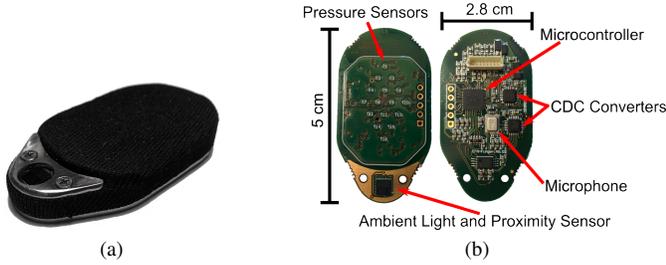


Fig. 1: (a) Top side of the reference tactile sensor (b) Top and bottom sides of the printed circuit board of the sensor.

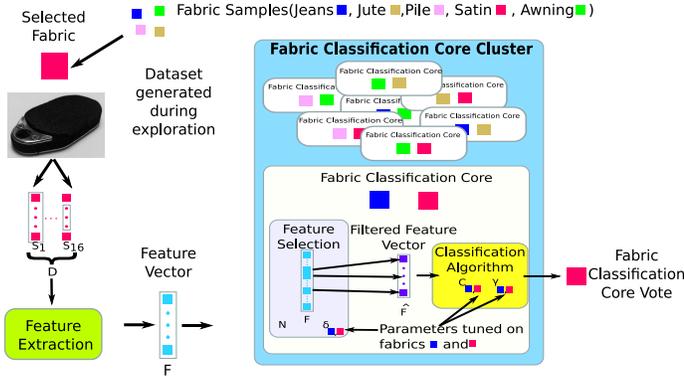


Fig. 2: The fabric classification architecture with three main components feature extraction, feature selection and classification algorithm.

III. THE FABRIC CLASSIFICATION ALGORITHM

A. Overview

The general structure of the fabric classification algorithm presented in this paper can be seen in Figure 2. It is constituted by three main components, which are: *feature extraction*, *feature selection* and the *classification algorithm*. The *feature extraction* component extracts from the raw tactile data, the feature vector that will be described in Section III-C.

Feature selection operates on the feature vector and selects the relevant features for the next stage according to a criteria that will be detailed in Section III-D. The *classification algorithm* component implements a soft margin support vector machine (SVM) with radial basis function kernel. During a *training* phase, the internal parameters for both *feature selection* and *classification algorithm* components are assessed and updated using a validation dataset (see Figure 3). After the training/validating phase, the testing phase, described in Section III-D, takes place in which the two components cooperate in order to classify a new feature vector using the learned parameters similarly to what happens in the validation phase that is shown in Figure 4. Within this architecture, the *feature selection* and *classification algorithm* components, constitute the *fabric classification core* (Figure 2). This classification core implements a *binary classifier*, thus operating on two different classes (i.e., the number of different fabrics we want

to classify). Only two different kinds of fabric can be classified using a single classification core. If the architecture has to operate on more than two classes, multiple core instances must be generated, each one operating on one of the possible 2-combinations on the set. For instance, to discriminate among three classes (e.g., pile, wool and velvet), three classifiers will be instantiated, namely pile vs. wool, wool vs. velvet and pile vs. velvet. Inside a single core, during the training phase, the *feature selection* component learns the best subset of relevant features $R_{i,j}$ from the feature vector for discriminating classes C_i and C_j , while the classification algorithm trains the SVM on a reduced set of features generated by the feature selection component. In the testing phase, *feature selection* component extracts the relevant features $R_{i,j}$, which will be used by the *classification algorithm* component for classification, using the support vectors learned during the training phase and, accordingly, express a *vote* on C_i or C_j . The final classification result is obtained by majority vote of all the fabric classification cores (see Figure 4).

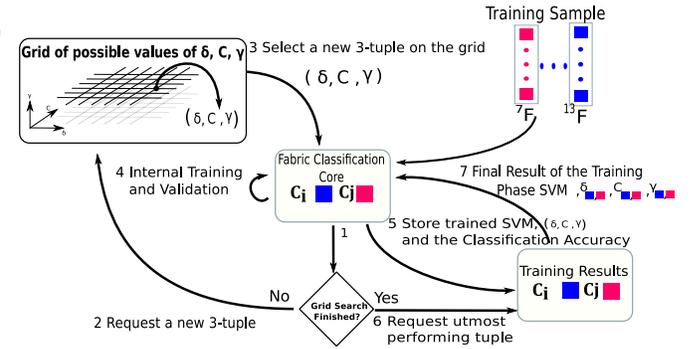


Fig. 3: Training phase of the fabric classification algorithm for classes C_i and C_j with tuning of parameters.

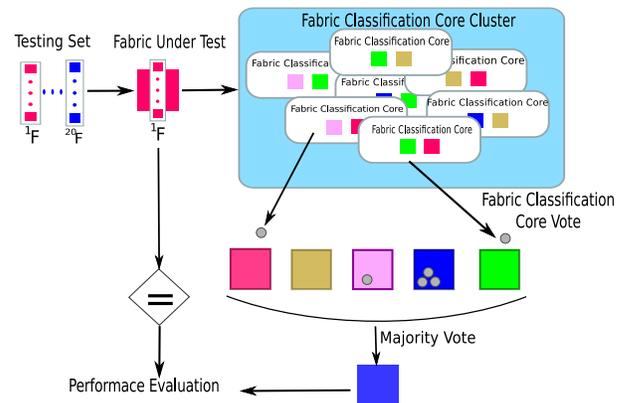


Fig. 4: Testing phase of the fabric classification algorithm with final classification result obtained through majority voting.

B. Data Generation

Depending on the knitting type, the fabric can produce different tactile sensor responses along different directions

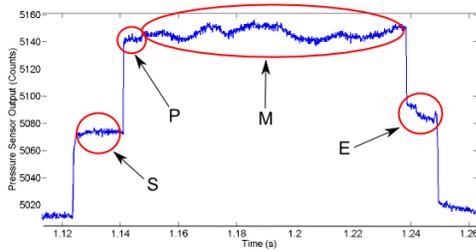


Fig. 5: The response of one of the pressure sensor in the different phases of the exploration.

of exploration, speed and exerted pressure. For this reason, the fabric exploration performed by the tactile sensor has been divided into a sequence of phases (Figure 5) each one interacting with the fabric in a different way:

- *Resting phase S*: the sensor statically touches the fabric but no force control is performed; the sensor response depends on the sensor weight, as exerted on the fabric.
- *Pressure phase P*: pressure control is activated, the sensor is not moving but a controlled pressure is exerted on the fabric.
- *Motion phase M*: the sensor is moving on the fabric applying a constant pressure.
- *Resting phase E*: the sensor stops and the pressure control is deactivated; similarly to phase *S*, the sensor response depends on its weight.

In our experiments, motion phase *M* has been realized by two basic patterns. The first pattern, M_1 , is a linear motion along one direction. The second pattern, M_2 , is an L-shaped motion. The two patterns have been selected to explore the warps and wefts of the fabric. During the experiments, the pressure profiles of all the 16 tactile elements are recorded and later segmented depending on the exploration phase. As a result, the vector \mathbf{s}_c of pressure measurements of a single sensor c can be expressed as the concatenation of different vectors \mathbf{s}_c^Φ where $\Phi \in \{S, P, M_1, M_2, E\}$ is the exploration phase:

$$\mathbf{s}_c = \mathbf{s}_c^S || \mathbf{s}_c^P || \mathbf{s}_c^{[M_1, M_2]} || \mathbf{s}_c^E \quad (1)$$

In (1), the term $||$ indicates a concatenation of vectors while the term $[M_1, M_2]$ indicates an alternate selection of pattern motions. In fact, as we will see in Section IV-A, M_1 and M_2 belong to a different set of experiments. Therefore, the dataset \mathbf{s}_c can be generated only by using M_1 or as an alternative M_2 . Finally, the complete dataset D of a single experiment can be defined as:

$$D = \{\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_{16}\} \quad (2)$$

C. Feature Extraction

In this paper, the widely used set of features introduced by Weiss *et al.* [11], plus additional new ones, are used. In all the experiments, the 195 features in Table I are computed on the dataset D . Moreover, features 4 to 163 are time domain features, while features 164 to 195 are frequency domain features. The feature extraction phase is performed by a set

TABLE I: List of features extracted from the sensor signals.

No.	Notation	Description
1	F_{mean}^P	Average value of 16 sensors in phase <i>P</i> .
2	$F_{diff}^{S,P}$	Difference of means in phases <i>S</i> and <i>P</i> .
3	$F_{diff}^{P,E}$	Difference of means in phases <i>P</i> and <i>E</i> .
4-19	F_{max}^M	Maximum magnitude in phase <i>M</i> , one feature for each sensor.
20-35	F_{min}^M	Minimum magnitude in phase <i>M</i> , one feature for each sensor.
36-51	F_{mean}^M	Mean value of magnitudes in phase <i>M</i> , one feature for each sensor.
52-67	F_{var}^M	Variance of magnitudes in phase <i>M</i> , one feature for each sensor.
68-83	F_{norm}^M	Norm value of magnitudes in phase <i>M</i> , one feature for each sensor.
84-99	F_{kurt}^M	Kurtosis of the signal in phase <i>M</i> , one feature for each sensor.
100-115	F_{skew}^M	Skewness of the signal peaks in phase <i>M</i> , one feature for each sensor.
116-131	F_{range}^M	Range of magnitude in phase <i>M</i> , one feature for each sensor.
132-147	F_{mode}^M	Most frequent magnitude in phase <i>M</i> , one feature for each sensor.
148-163	F_{5th}^M	Fifth moment of the signal in phase <i>M</i> , one feature for each sensor.
164-179	$F_{H.frq}^M$	Maximum rubbing magnitude in frequency domain in phase <i>M</i> , one feature for each sensor.
180-195	$F_{S.h}^M$	Sum of higher half of spectrum amplitudes in phase <i>M</i> , one feature for each sensor.

of functions f_k each one implementing the extraction of the feature k from the dataset D . Therefore, for each experiment, we can define the *feature vector* \mathbf{F} as the vector of the outcomes of the functions f_k computed on D , that in our case is equal to:

$$\begin{aligned} f_k(D) &= F_k, \forall k \in \{1, 2, \dots, 195\} \\ \mathbf{F} &= (F_1, F_2, \dots, F_{195}) \end{aligned} \quad (3)$$

For each raw data eD acquired during an experiment e , we can compute a feature vector ${}^e\mathbf{F}$. As we will see in the next paragraphs, a collection of ${}^e\mathbf{F}$ vectors is required for training and validating the classification algorithm in the training/validating phase while a single ${}^e\mathbf{F}$ vector is used in the testing phase to perform the classification of the currently explored fabric.

D. Feature Selection and Classification Algorithm

Machine learning applications are usually characterized by very high dimensional feature spaces that need to be reduced to lower dimensions. Reduction in dimension can sometimes lead to overfitting and low classification accuracy. To overcome this problem, feature space can be compressed to a lower dimensional one on the basis of information criteria (e.g., Principal Component Analysis and similar techniques [12]) or a suitable subset of all the features can be found [13]. In this work, the subset selection approach is adopted. The selection criteria is based on the concept of *p-value* [14]. Let us assume

that n experiments have been performed on each of the two classes C_1 and C_2 , and we want to find out whether the feature F_{mean}^P is a discriminative feature for the two classes. We can re-define the single feature value as ${}^e F_{mean_c}^P$, where c identifies the belonging class and e the experiment. In turn, we can rearrange all the feature values in two different vectors defined as:

$$\begin{aligned} \mathbf{x} &= ({}^1 F_{mean_1}^P, {}^2 F_{mean_1}^P, \dots, {}^n F_{mean_1}^P) \\ \mathbf{y} &= ({}^{n+1} F_{mean_2}^P, {}^{n+2} F_{mean_2}^P, \dots, {}^{2n} F_{mean_2}^P). \end{aligned} \quad (4)$$

The values in (4) are outcomes of experiments whose results are not known in advance. Therefore, they can be considered as random variates of two random variables, one associated with experiments on class C_1 , the other with experiments on C_2 . Values in (4) characterize the two fabrics only from a statistical point of view. In this way, for example, we can understand whether the feature F_{mean}^P is a discriminative feature by comparing the average outcomes of the experiments for the two classes. In the case they are equal, we can conclude that, with respect to feature F_{mean}^P , both fabrics respond on average in the same way, thus the feature F_{mean}^P is not discriminative. Otherwise, we have to analyse the difference with respect to the associated variances to conclude whether it is a discriminative feature or not. We can relate our problem to a *two-sample t-test* [15], assuming the following null hypothesis: *data in vectors \mathbf{x} and \mathbf{y} originate from independent random samples from normal distributions with equal means and equal but unknown variances*, i.e., feature F_{mean}^P is not informative in discriminating the two classes. The outcome of this test shows whether we must reject the null hypothesis or not. If the null hypothesis is to be rejected, we have separable and independent data sets and accordingly we can conclude that F_{mean}^P is a discriminative feature for classes C_1 and C_2 . The strength of the rejection evidence is called p-value and can be computed from the outcome of the t-test [15].

A small p-value (typically less than 0.05) indicates strong evidence against the null hypothesis, so hypothesis must be rejected, while a large p-value (greater than 0.05) indicates weak evidence against the null hypothesis which should not be rejected. This suggests that the p-value can be used for comparing different features with respect to their ability to discriminate between two kinds of classes. In our approach, at the beginning of the training phase, the *feature selection* component inside the *fabric classification core* of classes C_i and C_j performs a t-test on the related training set and computes for each feature the related p-value. The result is a vector of p-values defined as:

$$\mathbf{p}_{i,j} = (p_1, p_2, \dots, p_k, \dots, p_{195}) \quad (5)$$

where p_k is the p-value related to the k -th feature and subscripts i and j refer to classes C_i and C_j . It is now straightforward to define a function $P_{i,j}(k)$ that maps to each feature k its p-value as:

$$P_{i,j}(k) = p_k, \forall k \in \{1, 2, \dots, 195\}$$

Using the information in the $\mathbf{p}_{i,j}$ vector, the *fabric selection* component filters the relevant features by imposing a threshold $\delta_{i,j}$ on the p-values, thereby selecting the most relevant ones for the classification of C_i and C_j . The relevant feature set can be defined as:

$$R_{i,j} = \{k : P_{i,j}(k) < \delta_{i,j}, \forall k \in \{1, 2, \dots, 195\}\}$$

It must be noted that each feature selection operates on p-values computed from the training set generated by a specific pair of classes, e.g., vectors \mathbf{x} and \mathbf{y} in (4). As a result, each *feature selection* component composes a different relevant feature set with different size. Such an approach specializes the features selection on a specific couple of classes of the *fabric classification core* instead of finding a general feature vector for all the available classes. The threshold value is not arbitrarily selected. On the contrary, it is determined during the training phase. In fact, $\delta_{i,j}$ is trained along with the parameters of the *classification algorithm* component C and γ of the radial basis function, that are tuned using K-fold cross validation. Thus a unique set of parameters (C , γ , δ) is obtained using grid search (as shown in Figure 3). At the end of the training phase, the value of $\delta_{i,j}$ with the lowest misclassification error is selected.

In addition, a size limiting parameter N is introduced to fix the maximum amount of features that can be used. This parameter has been added mainly to limit the computational requirements for training. The outcome of the *feature selection* component of the *feature classification core* specialized for classes C_i and C_j is a reduced feature vector $\hat{\mathbf{F}}_e$ containing feature values of vector \mathbf{F}_e for all the features $k \in R_{i,j}$. The reduced feature vector $\hat{\mathbf{F}}_e$ is used by the *classification algorithm* component for discriminating between the fabric C_i and C_j . In particular, we use one vs one binary soft margin SVM with radial basis function kernel that discriminates C_i from C_j by finding the hyperplane in the features space that separates features generated by C_i from features generated by C_j . The idea behind using one vs one is to avoid the problem of imbalanced training set which results in losing the symmetry of the original problem in the case of one vs all [16].

IV. EXPERIMENTAL VALIDATION

A. Experimental Setup

In order to assess the overall performance of the classification architecture, a number of classification tests have been performed with the tactile sensor presented in Section II and a workbench set-up. The used setup is shown in Figure 6. It consists of a Cartesian robot equipped with a force/torque sensor used to measure the pressure exerted on fabrics, which are located on a horizontal plate. The Cartesian robot is a 5 DoF system with independent x , y and z -linear axis, as well as two rotating angles along the x and y -axis, which are actuated using high-precision motorised stages from ThorLabs¹. The x and y -axis, as well as one rotational motor, move an instrumented platform where fabrics are fixed for

¹See the official web page at <http://www.thorlabs.de/>

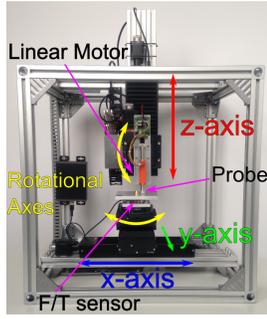
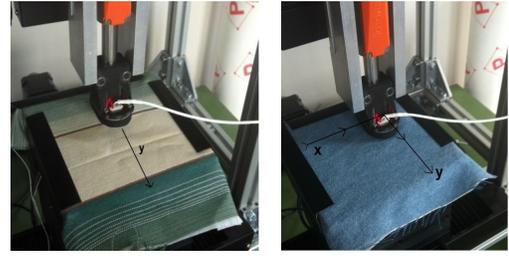


Fig. 6: Experimental setup.



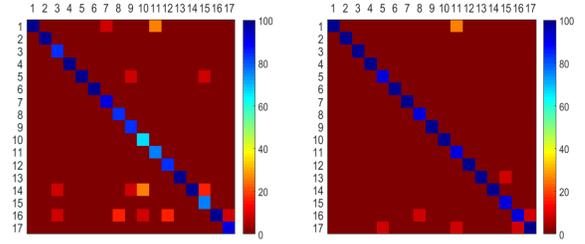
Fig. 7: Seventeen samples of fabrics.

all the experiments. Under the platform, an ATIMini40 6-axes force/torque sensor is able to measure exerted forces and torques. The remaining rotational joint and a linear actuator along the z -axis move and rotate a probe, on which the tactile sensor is mounted. The linear actuator installed on the z -axis regulates the exerted force (in the range $0 - 10\text{ N}$) using the feedback provided by the force/torque sensor. The probe can be moved with a precision of $1\ \mu\text{m}$ and a speed up to 4 mm/s in any direction. The working area of the Cartesian robot is about $150 \times 150 \times 150\text{ mm}$. A robot controller has been developed to plan a sequence of motions and to impose force profiles. The classification architecture algorithm has been tested on the set of fabrics shown in Figure 7. This peculiar set of fabrics has been selected because they constitute a realistic set of fabrics with different characteristics as far as elasticity and softness are concerned. A preliminary set of experiments has been conducted to identify the proper force and speed values associated with tactile exploration in order to obtain the best dynamic response from the tactile sensors. These experiments consist of a number of simple one-way rubbing movements at different pressures and speeds. As a result of such initial tests, a 5 N force and a 2 mm/s speed have been selected as process parameter values in all the experiments. As already anticipated in Section III, two kinds of motion patterns have been defined in the experiments for motion phase M . The former, referred to as M_1 (Figure 8a), is a single direction motion for the tactile sensor along the y -axis for about 50 mm . The latter referred to as M_2 (Figure 8b), is an L-shaped motion,



(a) Pattern for M_1 (b) Pattern for M_2

Fig. 8: Motion patterns for M_1 (straight line segment) and M_2 (L-shaped).



(a) Confusion matrix for M_1 (b) Confusion matrix for M_2

Fig. 9: Confusion matrices for motion patterns M_1 and M_2 using p-value where the numbers correspond with respect to the fabrics mention in Figure 7

again of 50 mm length in both directions. L-shaped motion has been selected in order to mimick real-world tactile explorations along non-preferred directions. In this case, it corresponds to the fabric warp and weft. The fastening of fabric samples has been performed accordingly, with the warp placed along the x -axis and the weft along the y -axis. In order to generate the dataset for the evaluation of the classification architecture, 20 repetitions have been performed for each fabric in the set and for the two different motion patterns, thereby obtaining a total of 680 experiments.

B. Results

The performance of the classification architecture has been validated using the trained models or classifiers. As a reminder, the estimation of the tuning parameters (C and γ) for training were selected by using grid search with 5-fold cross validation. In our experiments, the training set is 40% of the whole dataset. Table II shows various results of computation time during classification for each sample. The overall computation time is obtained by averaging the classification time of 30 different trials for a sample. Figure 9a reports a confusion matrix that refers to classification performed on the M_1 pattern, by considering all the feature set (i.e., no threshold on the p-value is posed). The average classification rate is 89%. The above results show that selecting the proper subset of features, for each fabric classification core, leads to better results compared to consider all the features extracted by the

TABLE II: Computational time for each sample in classification.

Feature extraction time per sample (sec)	0.017
Total training time (sec)	0.566
Testing time per sample (sec)	0.045
Classification results (%)	98.87

TABLE III: Result of the training phase of feature selection components with $N = 20$. In the middle column, the top four features are reported. The number in the subscript of each feature identify the pressure sensor related to that feature. In the right column, the total number of selected features for the classification.

Core	Top Four	Total
Knitwear1 Vs. Wool	$F_{range.2}^M, F_{range.4}^M$	2
Knitwear1 Vs. Pile	$F_{min.1}^M, F_{min.2}^M, F_{min.4}^M, F_{min.7}^M$	20
Flax Vs. Pile	$F_{norm.13}^M, F_{norm.15}^M, F_{mean.15}^M, F_{H.frq.15}^M$	18

feature extraction component. It is noteworthy that each fabric classification core has its peculiar subset of features, as can be seen in Table III, ranging from just 2 features for the core knitwear1 vs. wool to 20 features for the core knitwear1 vs. pile. Finally, in order to compare our approach to another well-known feature reduction technique, the same previously described experiments have been performed using principal component analysis in the place of p-value thresholding. The results, reported in Table IV, show that p-value performs better compared to traditional PCA. Whereas showing comparable results for M_2 pattern.

V. CONCLUSION

This paper introduces an architecture for tactile-based fabric classification which is based on the concept of *fabric classification core*, a classification unit devoted to discriminate between two fabrics only. A voting mechanism is used to compile the overall classification scores. The classification core is based on the SVM approach and performs a selection of which features are better suited to discriminate between two fabrics by reasoning on the p-value of each feature. The features with a p-value higher than threshold are considered to be higher than the threshold. We validated the proposed architecture using seventeen common fabrics and two different tactile exploration behaviours. Results show very good performance values when performing p-value based feature selection. Current work is focused on extending the number of considered fabrics and to include additional motion pattern by exploring the fabric along the diagonals. It is noteworthy

TABLE IV: Comparison between p-value and PCA for M_1 and M_2

Motion pattern	p-value		PCA	
	Classification accuracy (%)	Training time	Classification accuracy (%)	Training time
M1	98	0.525	89	0.720
M2	99	0.566	99	0.956

that one vs one approach suffers from dimensionality curse as the number of cores grows quadratically with the number of classes. However, this effect can be mitigated by parallelization since each core is computationally independent from the others. Nevertheless, given the testing time reported in Table II for 17 different fabrics, the number of classes becomes an issue at a point where it is not immediate to foresee a practical application where there is the need to distinguish so much different fabrics. Moreover, techniques exist to reduce this dependency to a linear one [17].

REFERENCES

- [1] H. Muhammad, C. Recchiuto, C. Oddo, L. Beccai, C. Anthony, M. Adams, M. Carrozza, and M. Ward, "A capacitive tactile sensor array for surface texture discrimination," *Microelectronic Engineering*, vol. 88, no. 8, pp. 1811–1813, 2011.
- [2] A. Drimus, G. Kootstra, A. Bilberg, and D. Kragic, "Classification of rigid and deformable objects using a novel tactile sensor," in *Advanced Robotics (ICAR), 2011 15th International Conference on*. IEEE, 2011, pp. 427–434.
- [3] T. Bhattacharjee, J. M. Rehg, and C. C. Kemp, "Haptic classification and recognition of objects using a tactile sensing forearm," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 4090–4097.
- [4] C. Kampouris, I. Mariolis, G. Peleka, E. Skartados, A. Kargakos, D. Triantafyllou, and S. Malassiotis, "Multi-sensorial and explorative recognition of garments and their material properties in unconstrained environment," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1656–1663.
- [5] Y. Tada, K. Hosoda, Y. Yamasaki, and M. Asada, "Sensing the texture of surfaces by anthropomorphic soft fingertips with multi-modal sensors," in *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 1. IEEE, 2003, pp. 31–35.
- [6] J. Sinapov, V. Sukhoy, R. Sahai, and A. Stoytchev, "Vibrotactile recognition and categorization of surfaces by a humanoid robot," *Robotics, IEEE Transactions on*, vol. 27, no. 3, pp. 488–497, 2011.
- [7] F. De Boissieu, C. Godin, B. Guilhamat, D. David, C. Serviere, and D. Baudois, "Tactile texture recognition with a 3-axial force mems integrated artificial finger," in *Robotics: Science and Systems*. Seattle, WA, 2009, pp. 49–56.
- [8] P. Dallaire, P. Giguère, D. Émond, and B. Chaib-Draa, "Autonomous tactile perception: a combined improved sensing and bayesian nonparametric approach," *Robotics and Autonomous Systems*, vol. 62, no. 4, pp. 422–435, 2014.
- [9] N. Jamali and C. Sammut, "Majority voting: material classification by tactile sensing using surface texture," *Robotics, IEEE Transactions on*, vol. 27, no. 3, pp. 508–521, 2011.
- [10] S. Denei, P. Maiolino, E. Baglini, and G. Cannata, "On the development of a tactile sensor for fabric manipulation and classification for industrial applications," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 5081–5086.
- [11] C. Weiss, H. Frohlich, and A. Zell, "Vibration-based terrain classification using support vector machines," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. IEEE, 2006, pp. 4429–4434.
- [12] I. T. Jolliffe, *Principal Component Analysis*, ser. Springer Series in Statistics. Heidelberg, Germany: Springer, 2002.
- [13] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 97, no. 12, pp. 273 – 324, 1997, relevance. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S000437029700043X>
- [14] R. A. Fisher, *The design of experiments*, 9th ed. London, United Kingdom: Macmillan Book Co, 1971.
- [15] J. Rice, *Mathematical statistics and data analysis*. Cengage Learning, 2006.
- [16] G. G. Yunqian Ma, *Support vector machines applications*. Switzerland: Springer, 2014.
- [17] J. C. Platt, N. Cristianini, and J. Shawe-Taylor, "Large margin dags for multiclass classification," in *nips*, vol. 12, 1999, pp. 547–553.