Available online at www.sciencedirect.com

ScienceDirect

Procedia Technology 26 (2016) 43 – 50

Procedia Technology

3rd International Conference on System-integrated Intelligence: New Challenges for Product and Production Engineering, SysInt 2016

Embedded Electronic System based on Dedicated Hardware DSPs for Electronic Skin Implementation

Ali Ibrahim[a,b]*, Luca Noli[a], Paolo Gastaldo[a], Hussein Chible[b], Maurizio Valle[a]

[a]*Department of Electrical, Electronic and Telecommunication Engineering and Naval architecture (DITEN) - University of Genoa, via Opera Pia 11 – 16145 Genoa, Italy*

[b]*Microelectronics Research Lab, PhD School for Sciences and Technology (EDST)- Lebanese University, AL Hadath, Lebanon*

Abstract

The effort to develop an electronic skin is highly motivated by many application domains namely robotics, biomedical instrumentations, and replacement prosthetic devices. Several e-skin systems have been proposed recently and have demonstrated the need of an embedded electronic system for tactile data processing either to mimic the human skin or to respond to the application demands. Processing tactile data requires efficient methods to extract meaningful information from raw sensors data.

In this framework, our goal is the development of a dedicated embedded electronic system for electronic skin. The embedded electronic system has to acquire the tactile data, process and extract structured information. Machine Learning (ML) represents an effective method for data analysis in many domains: it has recently demonstrated its effectiveness in processing tactile sensors data.

This paper presents an embedded electronic system based on dedicated hardware implementation for electronic skin systems. It provides a Tensorial kernel function implementation for machine learning based on Tensorial kernel approach. Results assess the time latency and the hardware complexity for real time functionality. The implementation results highlight the high amount of power consumption needed for the input touch modalities classification task. Conclusions and future perspectives are also presented.

Peer-review under responsibility of the organizing committee of SysInt 2016

Keywords: Electronic Skin System; Dedicated Embedded Hardware Implementation; Digital Signal Processing; Real Time Functionality;

* Corresponding author. Tel.: +39 (010) 353 2288;
E-mail address: ali.ibrahim@edu.unige.it

Peer-review under responsibility of the organizing committee of SysInt 2016
doi:10.1016/j.protcy.2016.08.007

1. Introduction

Human skin is the physical barrier through which we interact with our surroundings, allowing us to perceive various shapes and textures, changes in temperature, and varying degrees of contact pressure. To achieve high sensing capabilities, several different types of highly specialized sense receptors are embedded within our skin. These receptors first transduce information generated by mechanical stimuli into electrical signals and then transmit it to the central nervous systems for more complex processing. The collected signals are eventually interpreted by the somatosensory cortex, [1] permitting us to perceive the sense of touch and to easily interact with our physical world.

The effort to create an electronic skin with human-like sensory capabilities is motivated by the possibility of being highly applicable for autonomous artificial intelligence (e.g., robots), biomedical instrumentations, and replacement prosthetic devices capable of providing the same level of sensory perception of the organic equivalent. Endowing appliances with the capability of sensing and processing touch enables tactile interaction between electronic devices and the environment.

Following the definition given by Dahiya et al. [2], tactile sensing involves the detection and measurement of contact parameters in a predetermined contact area and subsequent processing of the signals to extract structured and meaningful information which is subsequently transmitted to higher system levels for perceptual interpretation. The development of electronic skin starts from defining the system specifications, designing and fabricating the mechanical arrangement of the skin itself (i.e. soft or rigid mechanical support, structural and functional material layers, etc.) together with the electronic embedded system for tactile data processing. The different e-skin tasks are far from being properly addressed and are still in their infancy even if many research groups are addressing the topic with numerous different approaches at each level of the problem [3-9].

Significant progress in the development of e-skin has been achieved in recent years by the concentration on mimicking the mechanically compliant highly sensitive properties of human skin. For the sensing materials, stretchable electrodes for e-skin have been developed in [10], and the transformation of a typically brittle material, Si, into flexible, high-performance electronics by using ultrathin (100 nm) films connected by stretchable interconnects is presented in [11]. Someya et al. have fabricated flexible pentacene-based organic field-effect transistors (OFETs) for large-area integrated pressure-sensitive sheets with active matrix readout [12]. For the system implementation however, the design of a tactile sensor patch to cover large areas of robots and machines that interact with human beings is reported in [13]. The realizations are mostly custom-built and the sensor is implemented with commercial force sensors. This had the benefit of a more foreseeable response of the sensor if its behavior is understood as the aggregation of readings from all the individual force sensors in the array. [14] introduced a cheap, scalable, discrete force cell and integrated it, along with other (discrete) sensor devices, into a multi-modal artificial skin, based from hexagonal shaped, intelligent unit cells (i.e. PCBs). However, the huge amount of data, the complexity of data processing algorithms and the relevant amount of energy and area restrict the current implementations of e-skin systems to networked PCB systems.

In this paper, we present an embedded electronic system architecture and implementation for electronic skin systems. The design is based on machine learning based on Tensorial kernel approach. Results assure the feasibility of the approach despite the hardware complexity when real time functionality is aimed. Moreover, the paper highlights the high amount of power consumption needed for the input touch modalities classification task.

The rest of the paper is organized as follows: Section 2 describes the e-skin system defining the different structural components for the system development. Section 3 presents the computational architecture for the Tensorial kernel approach. It analyzes the computational load of the proposed approach and provides the dedicated hardware implementation results. A classification study based on hardware implementation results is elaborated in section 4, and finally conclusions and future perspectives are reported in section 5.

2. E-skin System

From a system perspective, e-skin is usually defined as a set of multiple sensing components including structural and functional materials, signal conditioning and acquisition, integrated with a dedicated sensor information processing embedded electronic system [15]. Figure 1 shows a general block diagram of the e-skin systems.

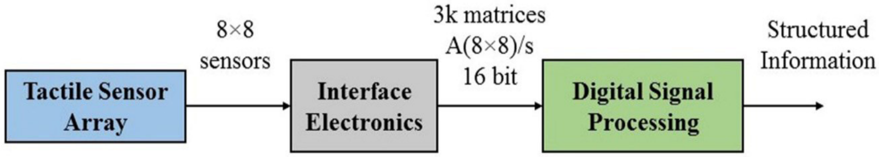

Fig. 1. General block diagram of the e-skin system.

Tactile sensor array is the first component to be addressed in e-skin development process. The adequate functional material enabling certain sensing capabilities should be identified. As the functional skin requirements are debatable and application dependent, piezoelectric polymer films of Polyvinylidene Fluoride (PVDF) [16] have been chosen as meeting the target requirements of mechanical flexibility, high sensitivity, detectability of dynamic touch (1Hz-1kHz frequency range) and robustness. The system consists of 8×8 tactile sensor array based on piezoelectric transducers [17].

The second block consists on the interface electronics which is in charge of signal conditioning and acquisition including the analog to digital conversion. The interface electronics needs a charge amplifier to collect the charge generated by the PVDF single taxel when stressed by tactile stimuli. The charge amplifier transfers the charge to a reference capacitor and produces an output voltage which is proportional to the charge on the reference capacitor and, respectively, to the input charge; hence the circuit acts as a charge-to-voltage converter. The charge amplifier may amplify the tactile stimuli in the frequency band of interest, which for our case is in the range from 1Hz to 1 kHz. The analog to digital converter scans the 8 × 8 sensor array at the rate of 2 kHz × (OSR) where 2 kHz is the input signal Nyquist frequency and OSR is the Over Sampling Ratio factor which must be larger than 2 (OSR has been set to 3 in the current setup) [18]. The sample rate at the output of the Interface Electronics block is consequently of 3k matrices A(8 × 8) /s with nominal data resolution (set by the Analog to Digital Converter) equals to 16 bits. Tactile sensors data have to be processed and structured information needs to be extracted and transmitted.

The Digital Signal Processing (DSP) block elaborates the tactile sensor signals using an embedded electronic system integrated together with sensing materials. Tactile data processing concerns different kinds of information which could be divided into two categories: 1) low level information such as contact location, area and duration, contact force intensity, direction and distribution, and temperature; 2) high level information for discrimination of the touch modality or the classification of attributes of the contacting objects e.g. roughness, textures, patterns, etc. In the present setup, the e-skin system deals with DSP block with high level information processing namely input touch modality classification. The classification uses Machine Learning based on tensorial kernel approach which has recently proven its effectiveness in processing tactile sensors data [19].

3. Tensorial Kernel Computational Architecture

Machine Learning (ML) methods have been increasingly used for the data analysis in many domains and have emphasized the need to take the structure of the original data into consideration. In [19], a ML based on Tensorial Kernel approach has been proposed to interpret touch modality in e-skin systems. The importance of this approach is that it preserves the inherent Tensorial structure of the signals provided by the sensing device. In this context, our goal is to implement a real time embedded electronic system based on dedicated hardware DSP of Tensorial Kernel approach for e-skin systems.

As shown in the block diagram of the Fig. 1, the input of the Digital Signal Processing block is 3k matrices A(8×8)/s which represents a data arrangement in terms of a time stream of arrays i.e. as a third order tensor £(8×8×3000) where the first two dimensions are defined by the geometry of the sensor array (8×8), while the time defines the third tensor dimension. Gastaldo et. al proposes a method to reduce the high amount of data contained in the tensor £(8×8×3000) in order to reduce the complexity of the computation [20]. Applying this method to the input tensor results a reduced tensor $\varphi(8\times8\times20)$ as described in [17].

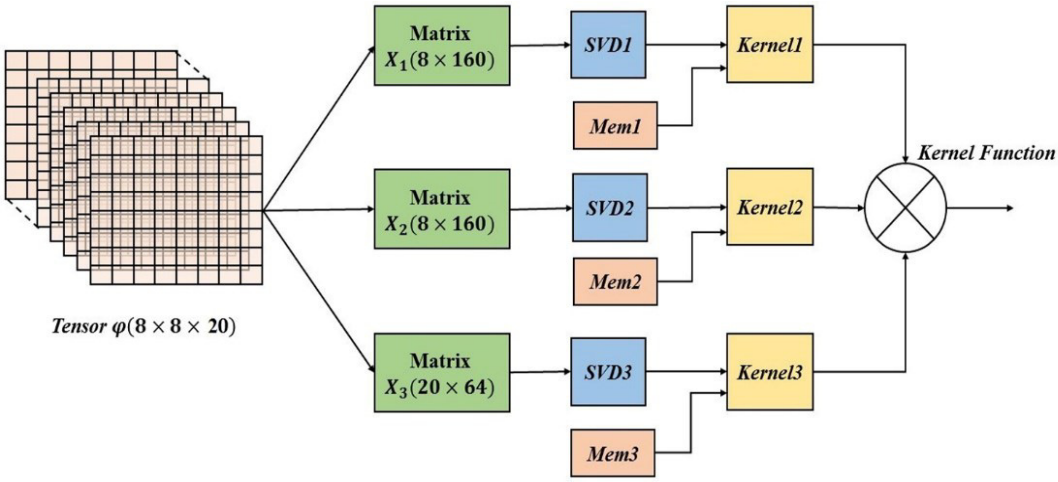

Fig. 2. Indicative block diagram for a single Tensorial kernel implementation.

Figure 2 illustrates the different computation steps needed to work out the Tensorial kernel function. The first step consists on tensor unfolding i.e. a matrix representation of $\varphi(8\times8\times20)$ where all the column (row) vectors are stacked one after the other [21]. Three matrices (X_1 (8×160), X_2 (8×160), X_3 (20×64)) are obtained by unfolding applying. The SVD blocks compute the singular value decomposition which transforms the unfolded matrices into the product of three matrices e.g. $X_1 = U_1S_1V_1^T$ where U_1 is an orthogonal matrix containing the Eigen vectors of $X_1X_1^T$, and V_1 is an orthogonal matrix containing the Eigen vectors of $X_1^TX_1$. The S_1 matrix is a diagonal matrix diag(σ_0,…….,σ_{n-1}), where the σi are the singular values of X_1 (i.e. the square roots of the Eigenvalues), being arranged in descending order.

The computation of the kernel factor comes into effect after the SVD computation. Kernel factor is computed by using the singular vectors (V_i) of the input tensor and the singular vectors of the training tensors for the different classes. In order to reduce the online computation, the singular vectors of the training tensors are computed offline and memorized in *Mem* blocks. Finally, the kernel function is obtained by multiplying the resulted kernel factors for the three unfolded matrices.

3.1 Computational Load Analysis

Besides the huge amount of tactile data to be processed in real time, the computation complexity poses a tough challenge in the development of the embedded electronic system. Computational requirements depend on the overall number operations (mainly arithmetic) that the Tensorial kernel approach must perform and on the real-time operation.

In order to assess the computational load, a case study [17] has been considered: the given task is to classify a touch interaction among Nc=3 touch modalities (i.e. paintbrush brushing; finger sliding; washer rolling) in 1 second; here Nc is the number of classification classes and the number Nt of training data is set to 100.

As described in the Fig. 2 the approach consists first on computing the singular value decomposition (SVD) [22] of the unfolded matrix. The analysis of the computational requirements for the SVD is based on the One sided Jacobi algorithm which provides high accuracy and convergence in about K = 5:10 iterations. Following step is the computation of the kernel factor for a couple of SVDs, the first corresponding to the tensor input and the second to

TABLE 1: FLOATING POINT OPERATIONS PER SECOND (FLOPS)

	Number of operations
Addition/Subtraction	1.56×10^{10}
Multiplication	1.58×10^{10}
Division	6.48×10^{5}
Square root	4.32×10^{5}
Total FLOPS	3.14×10^{10}
Power Consumption [W] [23]	1.04×10

the tensor representing a predefined class extracted from the training data. Table 1 shows the number of operations and flops per second needed to implement the Tensorial Kernel approach. The power consumption of the resulted total FLOPS number has been estimated according to [23].

Following estimations presented in Table 1, about 31 GFLOPS (Giga Floating Point Operations per Second) are needed for real-time single touch classification. These requirements for the data processing unit are very challenging: an appropriate data processing unit need to be carefully selected in order to meet the target requirements.

Embedded DSP microprocessors for instance, perform their arithmetic operations via software; this can give the flexibility in design, allowing late design changes. For example, let us consider the very well-known ARM Cortex processor family [24]: Cortex-R7 can achieve 6 GFLOPS, which are lower than the target requirements highlighted by Table 1. Moreover, power consumption is not compatible with the target application requirements.

A possible approach to tackle this issue could be to design dedicated application specific integrated circuit (ASIC) on a standard cell technology; To this end, our approach is to use the field programmable gate array (FPGA) which represents an efficient solution combining the strengths of hardware and software. Moreover, prototyping ASIC designs in FPGAs is an effective and economical method of verification.

3.2 Dedicated Hardware Implementation Results

The computational load study results the SVD as the most computational expensive algorithm of the Tensorial kernel approach: it represents about 70% of the computational complexity of the overall approach [17]. For this reason, methods and architectures for the hardware implementation of the SVD have to be well studied and assessed in order to select an appropriate architecture suitable for the targeted application. In this perspective, three different hardware implementations for the SVD have been presented and assessed in [25], and an implementation suitable for embedded real time processing has been selected. The SVD computation results based on the selected hardware implementation using a Virtex-5 XC5VLX330T FPGA device are shown in Table 2.

The computation of the kernel function presented here has been pursued using the selected SVD implementation. Table 3 shows the implementation results of the kernel function using a Virtex-5 XC5VLX330T FPGA device. These results correspond to one kernel function computed for an input tensor compared with an only one training tensor belongs to one class.

TABLE 2. SVD IMPLEMENTATION RESULTS FOR VIRTEX 5 XC5VLX330T

Matrix size	160×8
Time Latency (ms)	0.42
Percentage Occupied Area (%)	18
Nb. of Slice Registers	28101
Nb. of Slice LUT’s	22076
Dynamic Power Consumption (W)	0.948

TABLE 3. KERNEL FUNCTION IMPLEMENTATION RESULTS FOR VIRTEX 5 XC5VLX330T

Matrix size	160×8
Time Latency (ms)	1.59
Percentage Occupied Area (%)	74
Nb. of Slice Registers	97761
Nb. of Slice LUT’s	70529
Dynamic Power Consumption (W)	2.709

4. Classification Study based on Hardware Implementation

Two well-established learning paradigms in the class of regularized kernel methods, namely, Support Vector Machines (SVMs) and kernel-based Extreme Learning Machines (K-ELMs) have been computed for the Tensorial kernel approach assessment in [19]. In order to compute the classification function, it is needed first to compute the kernel function corresponding to the input tensor with respect to the memorized training tensors. The number training data for the Tensorial kernel approach varies roughly between a minimum of Nt = 100 and a maximum of Nt = 1000 training tensors [17]. According to the dedicated hardware implementation results, two cases study have been assessed: 1) Classification of three input touch modalities with a number minimum of training data (Nc = 3 and Nt = 100) which represents the study case presented in section 3.1; and 2) Classification of five input touch modalities with an average number of training data (Nc = 5 and Nt = 500).

- Case 1: Nc = 3 and Nt = 100

Let us define the real time functionality as a time latency less than 1 second so that the system should figure out one classification per second. This case deals with a total of 300 training tensors, so 300 kernel functions must be computed with a time latency less than 1 second. However, the SVD for the input tensor is computed only for the first kernel function, then memorized and used for the remaining kernel function computations the fact which reduces the time latency of the overall system. According to Tables 2, and 3 these computations are done in a time latency equals to $1.59 + (1.59-0.42)\times 299=351$ ms < 1 s. So, using the computational architecture presented in Figure 3 assure the real time functionality for 3 input touch modalities with the minimum number of training tensors. Moreover, hardware complexity and power consumption remain unchanged. Thus, the hardware complexity and the power consumption presented in Table 3 are needed to classify 3 different input touch modalities with 100 training tensors.

- Case 2: Nc = 5 and Nt = 500

A total of 2500 kernel functions need to be computed in this case. The corresponding time latency is equals to $1.59 + (1.59 - 0.42) \times 2499 = 2900$ ms > 1 s. So, the computational requirements presented in this case don’t satisfy the real time functionality. Time latency should be three times reduced to be less than 1 s. This issue can be tackled by implementing a parallel hardware architecture providing 3 parallelism levels. Fig. 3 shows the hardware architecture of the kernel function computation providing 3 parallelism levels of the computational steps presented by Fig. 2. Using this architecture the time latency will be given by $1.59 + (1.59 - 0.42) \times 2499/3 = 976$ ms < 1 s, and so the real time functionality is assured. However, although the parallel architecture assure the real time classification, it increases the hardware complexity and the power consumption. Table 4 show the requirements for input touch modalities classification for the two studied cases.

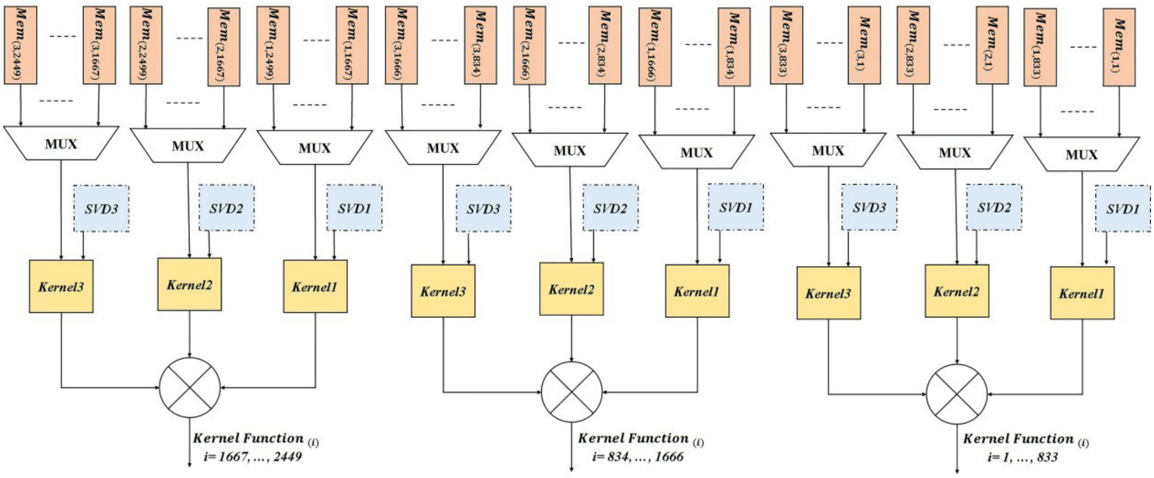

Fig. 3. Parallel architecture for the Tensorial kernel function hardware implementation for Nc = 5 and Nt = 500.

TABLE 4. REQUIREMENTS FOR INPUT TOUCH MODALITIES CLASSIFICATION

	Nc = 3 and Nt = 100	**Nc = 5 and Nt = 500**
Time Latency (s)	0.35	0.97
Dynamic Power Consumption (W)	2.7	6.2
Nb. of Slice Registers	97761	150604 (estimated)
Nb. of Slice LUT's	70529	108652 (estimated)

5. Conclusion and Future Perspectives

When processing tactile sensors data, the electronic system embedded into e-skin has to comply with severe constraints imposed by the application, e.g. real time response, low power consumption and small size. In this paper we presented an embedded electronic system based on dedicated hardware implementation for electronic skin systems. It provided a Tensorial kernel function implementation for machine learning based on Tensorial kernel approach. Time latency, power consumption, and hardware complexity for real time functionality have been assessed. The implementation results highlight the hardware complexity and the high amount of power consumption needed which represent the main issues for the system development.

The requirements related to the development of embedded data processing unit for e-skin are still far from being achieved with the current methods. Methods and techniques to reduce hardware complexity and power consumption of the embedded electronic system should be investigated. Approximate computing has recently emerged as a promising approach to energy efficient design of digital systems. Using such method could provide a solution to reduce the hardware complexity and the power consumption of the desired embedded electronic system. Another possible solution could be by designing the embedded electronic system with reconfigurability feature i.e. making it possible to modify the system components at run-time which could be used to reduce the power consumption of the overall system.

References

[1] M. L. Hammock , A. Chortos , B. C.-K. Tee, J. B.-H. Tok , and Z. Bao, 25th Anniversary Article: The Evolution of Electronic Skin (E-Skin): A Brief History, Design Considerations, and Recent Progress, Adv.Mater., Vol. 25, pp. 5997–6038, 2013.

[2] Dahiya Ravinder S, Metta G, Valle M, Sandini G. Tactile sensing – from humans to humanoids. IEEE Trans Rob, 1552-3098 2010. http://dx.doi.org/10.1109/TRO.2009.2033627.

[3] Silvera-Tawil D, Rye D, Velonaki M. Artificial skin and tactile sensing for socially interactive robots: a review. Robot Auton Syst 2015;63(3):230–43. http://dx.doi.org/10.1016/j.robot.2014.09.008.

[4] Hammock ML, Chortos A, Tee BC-K, Tok JB-H, Bao Z. 25th anniversary article: the evolution of electronic skin (e-skin): a brief history, design considerations, and recent progress. Adv. Mater. 2013;25:5997–6038 [Invited Review].

[5] Someya Takao. Building bionic skin: how flexible electronics can provide eskins for humans. IEEE Spectr 2013:50–6.

[6] Chortos Alex, Bao Zhenan. Skin-inspired electronic devices. Mater Today, 1369-7021 2014;17(7):321–31. http://dx.doi.org/10.1016/ j.mattod.2014.05.006.

[7] Tiwana MI, Redmond SJ, Lovell NH. A review of tactile sensing technologies with applications in biomedical engineering. Sens Actuators A, 0924-4247 2012;179(June):17–31. http://dx.doi.org/10.1016/j.sna.2012.02.051.

[8] Kampmann, Kirchner F. A tactile sensing system for underwater manipulation. In: Proceedings of the workshop on: advances in tactile sensing and touch based human-robot interaction to be held in conjunction with the 7th ACM/ IEEE international conference on human–robot interaction (HRI 2012), March 5–8, Boston (Massachussets), USA; 2012.

[9] Fondahl K, Kuehn D, Beinersdorf F, Bernhard F, Grimminger F, Schilling M, Stark T, Kirchner F. An adaptive sensor foot for a bipedal and quadrupedal robot. In: Proceedings of IEEE international conference on biomedical robotics and biomechatronics, (BioRob-12), June 24–27, Rome, Italy; 2012.

[10] M. Park, J. Im , M. Shin , Y. Min , J. Park , H. Cho , S. Park , M.-B. Shim, S. Jeon, D.-Y. Chung , J. Bae , J. Park , U. Jeong , K. Kim ,” Highly stretchable electric circuits from a composite material of silver nanoparticles and elastomeric fibres”, Nat. Nanotechnol. 2012, 7, 803 .

[11] L. Hu , M. Pasta , F. La Mantia , L. Cui , S. Jeong , H. D. Deshazer , J. W. Choi , S. M. Han , Y. Cui , “Stretchable, Porous, and Conductive Energy Textiles”, Nano Lett. 2010, 10, 708.

[12] C. Reese, W.-J. Chung, M.-M. Ling, M. Roberts, Z. Bao, “High-performance microscale single-crystal transistors by lithography on an elastomer dielectric”, Appl. Phys. Lett. 200 , 89, 20, 2108.

[13] Van F. Vidal-Verdu, M. J. Barquero, J. Castellanos-Ramos, R. Navas-Gonzalez, J. A. Sanchez, J. Seron, A. Garcia-Cerezo, “A Large Area Tactile Sensor Patch Based on Commercial Force Sensors,” Sensors, Vol. 11, No. 5, pp. 5489-5507, 2011.

[14] Mittendorfer, P.; Yoshida E.; Cheng, G., "Realizing whole-body tactile interactions with a self-organizing, multi-modal artificial skin on a humanoid robot,"Advanced Robotics, Vol. 29, No. 1, 51–6, 2015.

[15] Dahiya, R.S.; Mittendorfer, P.; Valle, M.; Cheng, G.; Lumelsky, V.J., "Directions Toward Effective Utilization of Tactile Skin: A Review,"in Sensors Journal, IEEE, vol.13, no.11, pp.4121-4138, Nov. 2013. doi: 10.1109/JSEN.2013.2279056.

[16] Seminara, L.; Pinna, L.; Valle, M.; Basirico, L.; Loi, A.; Cosseddu, P.; Bonfiglio, A.; Ascia, A.; Biso, M.; Ansaldo, A.; Ricci, D.; Metta, G., "Piezoelectric polymer transducer arrays for flexible tactile sensors," in Sensors, 2012 IEEE , vol., no., pp.1-4, 28-31 Oct. 2012.

[17] L. Seminara, L. Pinna, A. Ibrahim, L. Noli, S. Caviglia, P. Gastaldo, M. Valle, “Towards integrating intelligence in electronic skin”, Special Issue on System-Integrated Intelligence, Mechatronics, Available online 29 April 2015.

[18] Ibrahim A., “Architectures and Hardware Implementation of Embedded Electronic Systems for Tactile Information Processing,” Ph.D. Thesis University of Genova, Italy, February 2016.

[19] Gastaldo P, Pinna L, Seminara L, Valle M, Zunino R. A tensor-based pattern recognition framework for the interpretation of touch modality in artificial skin systems. IEEE Sens 2014;14(7):2216–25.

[20] P. Gastaldo, L. Pinna, L. Seminara, M. Valle, R. Zunino, A tensor-based approach to touch modality classification by using machine learning, Robotics and Autonomous Systems, Volume 63, Part 3, January 2015, Pages 268-278, ISSN 0921-8890.http://dx.doi.org/10.1016/j.robot.2014.09.022.

[21] De Lathauwer L, De Moor B, Vandewalle J. A multilinear singular value decomposition. SIAM J Matrix Anal Appl 2006;21(4):1253–78.

[22] Ibrahim A., Valle M., Noli L., Chible H.,”Singular Value Decomposition FPGA Implementation for Tactile Data Processing”. IEEE NEWCAS June 2015. DOI: 10.1109/NEWCAS.2015.7182094.

[23] D. Schneider, "Could supercomputing turn to signal processor (again)?," in Spectrum, IEEE , vol.49, no.10, pp.13-14, 2012.

[24] http://www.arm.com/products/processors/cortex-r/index.php.

[25] Ibrahim A., Valle M., Noli L., Chible H., ”Assessment of FPGA Implementations of One Sided Jacobi Algorithm for Singular Value Decomposition”. IEEE ISVLSI 2015, July 2015.