Wrist-Worn Gesture Sensing With Wearable Intelligence

Xiangpeng Liang, *Student Member, IEEE*, Rami Ghannam, *Senior Member, IEEE*, and
Hadi Heidari , *Senior Member, IEEE*

Abstract—This paper presents an innovative wrist-worn device with machine learning capabilities and a wearable pressure sensor array. The device is used for monitoring different hand gestures by tracking tendon movements around the wrist. Thus, an array of PDMS-encapsulated capacitive pressure sensors is attached to the user to capture wrist movement. The sensors are embedded on a flexible substrate and their readout requires a reliable approach for measuring small changes in capacitance. This challenge was addressed by measuring the capacitance via the switched capacitor method. The values were processed using a programme on LabVIEW to visually reconstruct the gestures on a computer. In addition, to overcome limitations of tendo's uncertainty when the wristband is re-worn, or the user is changed, a calibration step based on the support vector machine (SVM) learning technique is implemented. Sequential minimal optimization algorithm is also applied in the system to generate SVM classifiers efficiently in real-time. The working principle and the performance of the SVM algorithms demonstrate through experiments. Three discriminated gestures have been clearly separated by SVM hyperplane and correctly classified with high accuracy (>90%) during real-time gesture recognition.

Index Terms—Support vector machine, sequential minimal optimization, gesture recognition, wearable intelligence, capacitance measurement.

I. INTRODUCTION

THE latest advances in artificial intelligence (AI) provide more possibilities for conventional wearable devices in the form of novel sensor systems and smart sensor arrays [1]–[3]. In the case of a Smart Wearable System (SWS), a number of sensors are attached to a user's body to collect data or receive commands from the user, which will be used to control a robot or a smart device to assist users in performing everyday tasks [1]–[5]. Various sensors can detect and measure the properties of the human body by different SWS. For example, gestures, electromyography (EMG), electroencephalography (EEG), sphygmus, eye movement, etc. are valuable data that can be collected by a SWS [2], [4], [6], [7].

The hand is a highly flexible part of the human body. Among the smart wearable applications, gesture recognition has been a centre of attention because of its flexibility and feasibility [6], [8]. It aims to measure the motions of the hand and is a typical multidisciplinary research. The conventional approaches can mainly be classified into two categories: movement-sensor-based and camera-based [8], [9], which will be discussed in Section II.

This paper introduces a gesture recognition method that requires less on-body attachments and has no limitation from camera [10], [11]. The principle assumes that gestures are based on tendon movement around the wrist. The pressure distribution around the wrist can be measured by a pressure sensor array embedded in a wristband. These pressure values are used to distinguish and reconstruct gestures on a computer after calibration [10]–[12]. As shown in Fig. 1, this work introduces a system based on flexible and thin capacitive pressure sensors, a measuring readout circuit, a flexible substrate wristband and machine learning algorithms. The gesture is reconstructed using the Support Vector Machine (SVM) learning algorithm to perform the corresponding control. SVM is a supervised learning algorithm for pattern recognition, data analysis and regression analysis. It is suitable at non-linear classification, requires fewer samples and enables high-dimensional pattern recognition [16]–[19]. It can drive the system to adapt to different situations by taking samples in the calibration step and optimising thresholds in the system [16]–[21].

Recent advances in flexible and wearable electronics as well as materials research allow this kind of device to be implemented in the form of a wristband [13]–[15]. After collecting the pressure data from the capacitive sensors, there should be a procedure to process these values and reconstruct gestures. However, the relatively small, uncertain and different tendon movements from person to person imply that there are challenges with this approach [10]. This issue could be solved by recording values, calculating the threshold and classification. Therefore, this algorithm must be automatically self-adapting to different gestures, users and working conditions.

In this paper, SVM will be used to create hyperplanes, which are classifiers according to the samples collected from the calibration step [22]. Afterwards, when a device is in use, the current data can be arranged according to these classifiers. The conventional SVM is too computationally intensive to apply on a real-time calculation [23], [24]. However, the gesture recognition of this device should respond rapidly and run
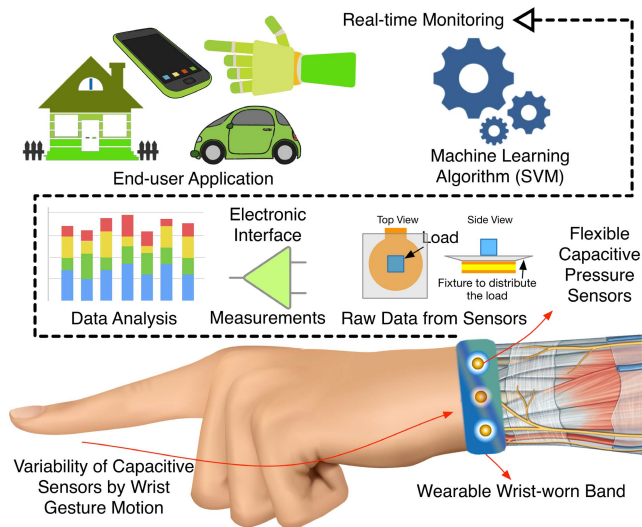
Fig. 1. The conceptual schematic of wrist-worn gesture sensing with machine learning for wearable intelligence.

in real-time continuously. Hence, we use Sequential Minimal Optimization (SMO) instead of the typical SVM procedure to optimize calculation procedures and achieve high speed training. SMO is a quadratic programming algorithm for efficiently solving the duality of Lagrange in SVM, which can reduce computing significantly [23], [24]. With the help of SMO, small-size real-time SVM system can be implemented at relatively low cost. Furthermore, it is worth noting that single SVM creates a binary classifier, while the number of recognizable gestures should be more than two which is multi-class classification. Therefore, we need more classifiers and a structure named Directed Acyclic Graph Support Vector Machines (DAG SVM) to organize their relationship [24], [25].

Considering these aforementioned backgrounds, this paper presents a system that aims to reconstruct gesture by measuring a user's tendon movements. Moreover, we aim to demonstrate how the SVM algorithm can be tailored to a real-time sensing system to improve the performance and make the sensors smarter and self-adaptive. This paper is organized as follows: A brief state-of-the-art of the recently reported wearable intelligent techniques, especially gesture sensing, are discussed in Section II. System design and programming, including the key elements of the microcontroller, visual interface and hardware design are introduced in Section III. A detailed discussion of SVM, SMO, DAG and their implementation is presented in Section IV. Finally, the key outcomes of the prototype and SVM are provided in Section V.

II. STATE-OF-THE-ART

The idea of wearable intelligence has been widely explored in the past decades. Various emerging techniques and methodologies provide a lot of solutions to support people's living, especially for old and disable people [2], [4], [6], [21], [26]–[35]. Recent advances in sensing techniques for wearable intelligent devices are summarized in Table I. Below is a comparison between the different types:

(1) Electromyography (EMG) signals has been widely used in clinical diagnosis and biomedical applications. Recently,

it also has been developed robot-assisted purposes. The working principle is that an electrode array is attached to the arm as an armband to detect the electrical signal controlled by the nervous system in muscle [22], [26], [32], [33]. These signals contain information about muscle movement such as contraction and relaxation, which is valuable in wearable system.

(2) Another similar example is electroencephalograph (EEG). Instead, the measured signal is collected from brain and the electrodes are attached on the head or even implanted for better accuracy. The signal is more sophisticated and noisier. With this, people can control something by concentration or even thinking [30], [36].

(3) Eye movement is also detectable by computation vision. However, a wearable eye movement dictation requires a tiny size of the system including a camera. Therefore, the cost would be high in wearable domain [34].

(4) In recent years, the cost of pulse detection is acceptable compared with other advanced sensing technologies. Among the methods of pulse detection, Photoplethysmography has been widely used in health monitoring and robot-assisted living due to their low-cost optical techniques. The frequency and intensity of the pulse contains a lot of information about the user's physical health and activities [31], [37].

(5) Gesture recognition always plays an important role in the wearables field. Current approaches of gesture recognition can mainly be classified into two categories: movement-sensor-based and camera-based [8], [9]. Movement-sensor-based gesture recognition is concerned with attaching sensors to a user's body to measure the motion, while camera-based gesture recognition is based on the recent advances in computer vision that can extract gesture from the environment through image processing algorithms. With the various algorithm, it is also useful for understanding many kinds of human activities. A prime example is Microsoft Kinect [8], [38], [39]. This method is not wearable but more widely used because of its excellent features in flexibility and stability [8]. It is also easy to interact with games and other stationary machines to achieve a more intuitive interaction. However, an obvious weakness is that the working space is limited by the camera [8]. For wearable and portable gesture recognition, the traditional approach is a movement-sensors-based method which means that some sensors are attached to the user's fingers, which is low-cost and stable. It has not been widely adopted because the attachments are not convenient for the user [8], [9].

(6) Recent investigations have been exploring potential gesture sensing approaches [11]. WristFlex presents a resistive pressure sensor array embedded in a wristband for wrist-worn gesture sensing. This device performs a result model when the threshold of gesture is pre-recorded [11]. Their research indicated that machine learning is necessary for further improvement. Moreover, Digits tactfully sets up a tiny IR camera on a wristband to bring camera-based gesture recognition into wearable scenario, which is costly and computational intensive [40]. Xue *et al*. [41]'s research introduced machine learning into their capacitive-based gesture sensing glove. Their result shows the enhancement of the gesture sensing system after applying machine learning.

TABLE I

COMPARISON BETWEEN STATE-OF-THE-ART SMART WEARABLE SYSTEMS

| Measured Target	Wearable Package	Sensor Type	Stability	Size	Flexibility	Machine Learning	Cost	Reference
EMG	Armband	Electrode	Medium	Medium	Medium	Possible	Medium	[18, 22, 28, 29]
EEG	Headgear	Implanted Electrode	Low	Big	Low	Possible	Very High	[26, 32]
Eye Movement	Glass	Camera	Medium	Medium	Low	Less	Very High	[30]
Pulse	Watch or wristband	Optical Devices	Medium	Small	High	Less	Medium	[27, 33]
Gesture	Glove	Flex Sensor	High	Big	High	Less	Low	[7, 8]
	Camera (Not wearable)	Camera	Medium	Big	N/A	Less	High	[34, 35]
Gesture (up-to-date research in the labs)	Wristband (WristFlex)	Pressure sensor	Medium	small	High	SVM	Low	[10]
	Tiny camera (Digits)	Camera	Medium	small	Low	Less	High	[37]
	Glove	Capacitive pressure	High	Medium	High	Adaptive DAG	High	[38]
	Wristband	Capacitive pressure	Medium	small	High	SVM and DAG	Low	This work
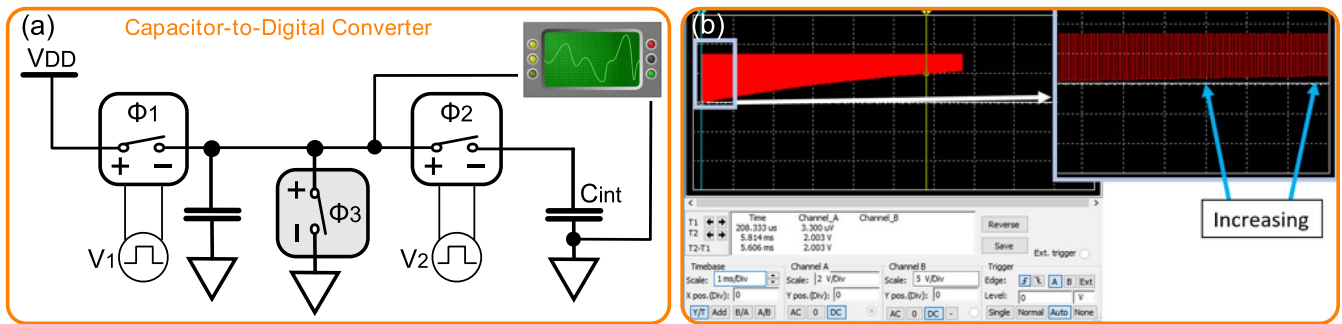
Fig. 2. (a) Equivalent circuit and simulation model of switched capacitor. (b) The change in V_{int} when $C_{sen} = 260$ pf at frequency of 10^5Hz for V4 and V1.

The system presented in this paper is a combination of sensing techniques and machine learning for wearable gesture recognition. Compared with the other wearable smart devices and gesture sensing system, this approach is low-cost, less-attachment and pint-sized with acceptable accuracy. It is also found to be more flexible and portable than camera-based approach. People can use it to control a robot or device by gesture [10]–[12], [40], [42]. For example, when a driver is driving a car, he can easily switch the music by showing a pre-recorded gesture.

III. SYSTEM DESIGN AND PROGRAMMING

To demonstrate the validity of the proposed models, we designed a wearable capacitive sensor array embedded in a PDMS based flexible and transparent substrate along with its capacitance readout circuit and visual interface. The sensors are commercial capacitive pressure sensors, whose capacitance value indicates the pressure applied on the surface. This section aims to introduce the system using the switched capacitor method to measure the capacitance values [10].

A. Wearable Sensor Array and Switched Capacitor

In this paper, we designed five capacitive sensors as a sensor array to measure the pressure distribution. The capacitive sensor is more precise and able to capture the small pressure difference [43]. However, since the changes in capacitance is

rather small, an accurate capacitance meter in the system is essential.

To achieve that, we applied the switched capacitor method on a high-speed microcontroller to acquire capacitance values. Fig. 2(a) is a typical equivalent circuit of switched capacitor in NI Multisim. C_{sen} is the capacitive sensor that needs to be measured; V_{ss} is the power source; C_{int} is a storage capacitor which should be much larger than C_{sen}. During each period, *1)* S1 is switched on firstly and C_{sen} is charged while C_{int} is disconnected, followed by *2)* a reversal in both S1 and S2. In this step, electricity is transferred from C_{sen} to C_{int} to equalize the voltage [44]. After the two steps, the voltage across the C_{int} (V_{int}) is slightly raised. The time of the two steps is called T_{sw}. The electricity is transferred during each time step, Q_s/T_{sw} and Q_{int}/T_{sw}, can be approximated the same as the currents I_s and I_{int}. Therefore, the switches and C_{sen} can be equivalent to a resistor (R_{sen}) [44]. We have:

$$I_s = I_{int} \tag{1}$$

$$\frac{Q_s}{T_{sw}} = \frac{Q_{int}}{T_{sw}} \tag{2}$$

$$\frac{(V_s - V_{sen})\, C_{sen}}{T_{sw}} = \frac{(V_{sen} - V_{int})\, C_{sen}}{T_{sw}} \tag{3}$$

$$(V_s - V_{sen}) = (V_{sen} - V_{int}) \tag{4}$$

Therefore,

$$C_{sen} = \frac{T_{sw}}{R_{sen}} \tag{5}$$

The equivalent circuit is an RC circuit. During the C_{int} and C_{sen} are charging and sharing, the V_{int} will exponentially reach V_{RH} which is a value between GND and V_{DD}, and here we take $0.68V_{DD}$. Eq. (8) shows the relationship between the current 'k' value and V_{int}[44].

$$V_{int} = V_{DD}(1 - e^{-(kT_{sw})/(T_{sw}/C_{sen})}) \tag{6}$$

Once the V_{int} reaches V_{RH}:

$$V_{RH} = V_{DD}(1 - e^{-(kT_{sw})/(T_{sw}/C_{sen})}) \tag{7}$$

$$C_{sen} = \frac{\ln(V_{DD}/(V_{DD} - V_{RH}))}{k/C_{int}} \tag{8}$$

As illustrated in Eq. (8), the C_{int}, V_{DD} and V_{RH} are constant while k is the measured value so that the C_{sen} can be calculated [44]. It is worth noting that the C_{sen} is not the actual value of the capacitive sensor. It consists of the capacitive sensor and parasitic capacitance (C_{par}) that are introduced by the circuit and microcontroller, which can be removed by offsetting a particular value of C_{par} at the testing stage. However, it would not influence the result since the C is the main variable indicating the change in force and the C_{par} is a constant [44].

The simulation result is shown in Fig. 2(b). The two pulse voltage generators are used to generate the square signals to control the two switches. Their configuration is a key to this simulation. They should have the same frequency and duty cycle but different phase by 180 degrees, to achieve taking turns to switch. As can be seen in Fig. 2(b), The V_{int} increases at each time step and the frequency is the same as the frequency of switches. By moving the nonius, we can get the time of raising the V_{int} to 2v (Tr) is 5.606ms. By changing the C_{sen} to 200 pF, the measured Tr' is 5.833ms.

The Tr is related to the counter k in Eq. (8). As the period of the switches is 0.01ms, The k value:

$$k = \frac{Tr}{Period} = \frac{5.606ms}{0.01ms} = 560$$

$$k' = \frac{Tr'}{Period} = \frac{5.833ms}{0.01ms} = 588$$

According to the Eq. (8), we have:

$$\Delta C_{sen} = C_{sen} - C'_{sen} = \frac{\ln\left(\frac{V_{DD}}{V_{DD} - V_{RH}}\right)}{\frac{k}{C_{int}}} - \frac{\ln\left(\frac{V_{DD}}{V_{DD} - V_{RH}}\right)}{\frac{k'}{C_{int}}}$$

$$= 62pF$$

Where V_{DD} =3.3v; C_{int} = 1uF; V_{RH} =2V. Therefore, the simulating result is almost the same as the actual value.

To implement the switched capacitor in a system to measure the capacitive sensor array, a microcontroller is used to control the procedure we have described above. According to the requirements, the microcontroller should have a superior performance in terms of following parameters and they are mentioned as; *1)* fast system clock to control the charging and sharing, *2)* small input leakage current to minimize the effect

of unwanted charging and *3)* the number of analogue input pins. In addition, we will need at least two digital pins and one analogue pin for each sensor.

B. Hardware Design

The hardware design of this prototype mainly includes a wristband and circuit design. The wristband is vital for overall performance and stability [10]. The design of wristband should ensure the sensor array is properly attached to user's body and offer mechanical support to the sensors. The photo in Fig. 6(a) shows a Polydimethylsiloxane (PDMS) based wristband with five embedded sensors as the main body of the current prototype, which is low-cost, flexible, and commercially available. It is an implementation of the simplified switched capacitor circuit as an intermedia between microcontroller and sensor array on a PCB [14], [21], [45].

C. Software Design

The software design consists of microcontroller and visual interface. Due to the requirements of high-speed charging and sharing electricity between the C_{int} and C_{sen}, the microcontroller should be programmed by operating the register directly instead of the library function. The code implements the algorithm of the switched capacitor as described in Section II, which is shown in Table II.

The visual interface is built for debugging and visualizing the performance of the switched capacitor method and machine learning which is explained in detail in the next section. In the current version, the interface is developed on LabVIEW, a graphical programming platform from National Instrument Company [46]. The Fig. 3 shows the user interface which is developed under a Producer/Consumer structure that mainly consists of five parts: initialization, serial communication, data processing, capacitance calculation and gesture classification. The number of time step of charging and sharing of each capacitor is read out from the microcontroller through serial communication, followed by a mean filter before the values are used to calculate capacitance. Afterwards, SVM module will be called for gesture recognition. In order to train the system by SVM, a set of gesture data need to be stored as a labelled sample. Therefore, a calibration step where the user will be required to show a certain number of gestures is necessary to collect labelled samples before using it.

IV. MULTICLASS SUPPORT VECTOR MACHINE

After the system receives the values that represents the pressure level at which they are attached, finding the relation between the gesture and its corresponding pressure values is the last step. Due to the concern that the tendon movement is small and different from person to person, it is not practical to find a fixed model for every user [10]. Even the same user would have different result when the wristband is re-worn or slightly moved [10].

To address this issue, a calibration step is essential to adapt to different users and improve the accuracy. The recent development in machine learning provides some useful methods to train the system to classify the gestures. Support Vector Machine is especially good at limited samples,

TABLE II

ALGORITHM OF SWITCHED CAPACITOR FOR
CAPACITANCE MEASUREMENT

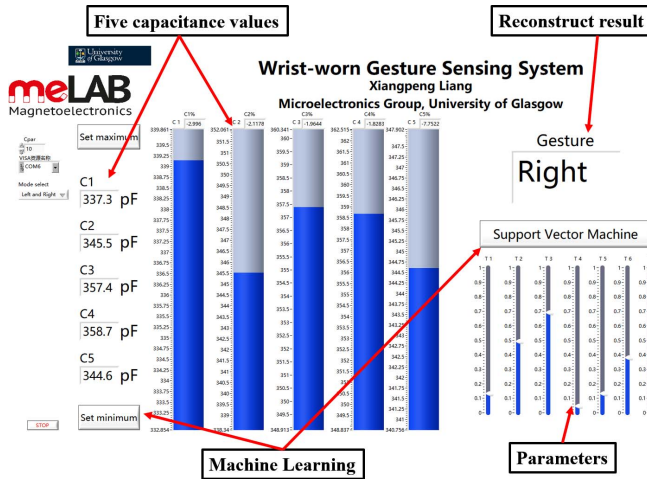| Circuit state	Pin states			Action
	D1	D2	AIN	
1. Initialize	---	---	---	Initiation
2.Discharge	LOW	LOW	input	Discharge
3. Charging	High-Z	High-Z	input	Avoid crowbar current
	HIGH	High-Z	input	Charge
4. Sharing	High-Z	High-Z	input	Avoid crowbar current
	High-Z	LOW	input	Charge Csen and Cint
5. Comparing	High-Z	High-Z	input	If Vint<V$_{RH}$, go to step 3
6. Calculate	LOW	LOW	input	Calculate Csen

Fig. 3.   Front panel of the user interface developed on LabVIEW.

nonlinear and high dimensional pattern recognition [16], [17], [24], [47]–[49]. This project applies SVM on binary gesture classification. Next, several binary SVM classifier is organized by DAG to fulfil real-time multiclass classification.

A. Support Vector Machine

SVM is a supervised machine learning model invented by Vapnik [17]. It is widely used and has good performance in less sample, non-linear and high-dimensional classification. Creating a hyperplane that separates samples into two group which are labelled as positive (+1) and negative (−1) according to their labels is its mechanism, at which the number of dimensions depends on the number of features of the target [17]. The main challenge is to find the support vectors (samples with nonzero Lagrange multiplier) and optimal the hyperplane and maximize the margin between the support vectors of the two classes [24]. A two-dimensional hyperplane and support vectors are shown in Fig. 4.

In this prototype, the number of dimensions is five since there are five sensors attached. The classification of gesture can be considered as a collection of binary classification of each two gestures. Assuming that the training data set has

Fig. 4.   Two-dimensional SVM and its hyperplane.

P=50 labelled samples for five gestures (10 for each) and each sample is donated by a vector $\vec{X}_i$ (i = 1, ..., P) which consists of 5 sensor values. As an example of 5 recognizable gestures, we have a training data sets $(\vec{X}_i, Y_i)$ where Y_i is their labels $Y_i \in (1, ..., 5)$. When we are creating a classifier that separates 'class 1' and 'class 2', named 'Classifier1V2' by SVM, the two classes firstly should be assigned into two poles, for example 'class 1' to '+1' and 'class 2' to '−1', then we have $(\vec{x}_i, y_i)$ where $\vec{x}_i \in (1, ..., p)$ and $y_i \in (+1, -1)$. Here we take p = 20 since there 2 set of gesture. Meanwhile, suppose the hyperplane is:

$$W^T \vec{x} + b = 0 \tag{9}$$

And the classification function is:

$$f(\vec{x}) = \text{sgn}(W^T \vec{x} + b = 0) \tag{10}$$

where $\vec{x}$ is the coming data in real-time operation, W^T is the normal and b is a bias. They should also satisfy the condition [17], [24]:

$$1 - y_i(W^T X + b_0) \leq 0 \tag{11}$$

Therefore, we need to find a W and its corresponding b maximizing the margin ρ in Fig. 4 [17], [24]. Assuming the optimal W and b is W$_o$ and b$_o$, the margin between the support vectors and hyperplane is:

$$\rho = 2r = \frac{2}{||W_o||} \tag{12}$$

Thus, the aim now is to minimize Wo under the constrain of (7). Minimising ||W|| is equivalent to minimise 1/2 ||W||^2. Create an equation:

$$\theta(W) = \frac{1}{2}||W||^2 = \frac{1}{2}W^T W \tag{13}$$

The constrain (11) is remained. To deal with the optimization under a certain constrain, Lagrangian multiplier is introduced to this equation:

$$L(W, b, \alpha) = \frac{1}{2}W^T W - W^T \sum_{i=1}^{p} \alpha_i y_i \vec{x}_i + \sum_{i=1}^{p} \alpha_i \tag{14}$$
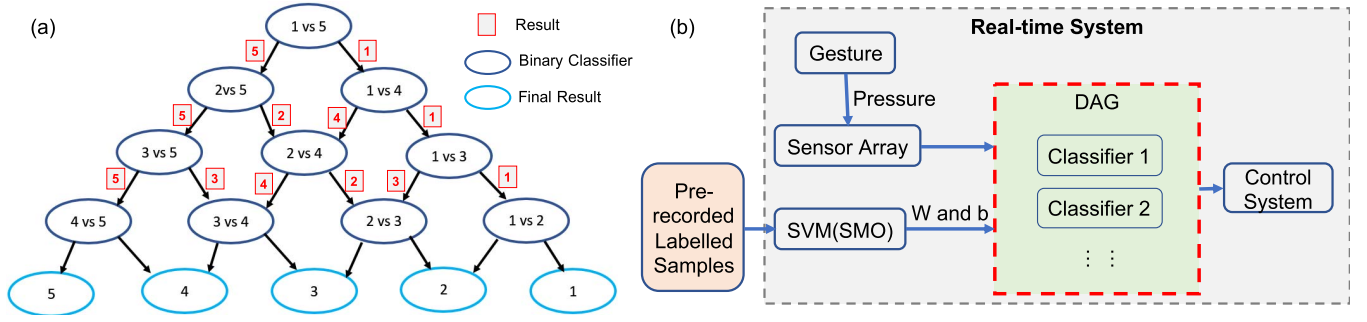
Fig. 5. (a) Flowchart of DAG SVM for 5 classes classification. Each dark circle represents a binary classifier created by SVM. The classification result will be obtained by this procedure, which is part of (b) the data flowchart for implementing SVM in a real-time system.

Partially derivative the Eq. (14) with respect to W and b, and equal to 0, we have:

$$W = \sum_{i=1}^{p} \alpha_i y_i \vec{x}_i \quad (15)$$

$$\sum_{i=1}^{p} \alpha_i y_i = 0 \quad (16)$$

Then:

$$\begin{aligned} W(\alpha) &= L(W, b, \alpha) \\ &= \sum_{i=1}^{p} \alpha_i - \frac{1}{2} \sum_{i=1}^{p} \sum_{j=1}^{p} \alpha_i \alpha_j y_i y_j (\vec{x}_i)^T \vec{x}_j \end{aligned} \quad (17)$$

Here we also need to take Karush-Kuhn-Tucker (KKT) condition into consideration and introduce slack variables ξ_i and regularization parameter C to weight the effect of those support vectors that may be error and cannot satisfy the KKT conditions [17], [18], [24]. The constrain now are:

$$0 \leq \alpha_i \leq C \quad (18)$$

$$\sum_{i=1}^{p} \alpha_i y_i = 0 \quad (19)$$

Therefore, the problem now is to find the optimal α_i to minimize the $W(\alpha)$.

B. Sequential Minimal Optimization

SMO is an efficient algorithm to solve the duality of Lagrange [23], [24]. As known from above, after each iteration we will have an improved hyperplane and according to KKT conditions, three kinds of $\vec{x}_i$ which are (1) samples that have been correctly classified by the current hyperplane ($\alpha_i = 0$ and $y_i(W^T X + b_0) > 1$), (2) support vectors ($0 \leq \alpha_i \leq C$ and $y_i(W^T X + b_0) = 1$) and (3) samples between the two borders ($\alpha_i = C$ and $y_i(W^T X + b_0) = 1$). However, there might be some samples do not satisfy the KKT conditions, which means they need to be adjusted. There might be many α_i need to be adjusted. The idea of SMO is that select two of them (α_1 and α_2) by heuristic selection and adjust them according to their old value and margin, whilst the other α_i remain unchanged [23], [24]. From Eq. (16) we can obtain the relationship between the new and old values:

$$a_1^{new} y_1 + a_2^{new} y_2 = a_1^{old} y_1 + a_2^{old} y_2 = € \quad (20)$$

Where € is a constant. We can replace a_1^{new} with a_2^{new} in $W(\alpha)$, followed by taking the derivative of a_2^{new} and equal to 0. We can find the optimized a_2^{new}:

$$a_2^{new} = a_2^{old} - \frac{y_2(E_1 - E_2)}{\eta} \quad (21)$$

Where $E_i = W^T X + b_{old} - y_i$ and $\eta = 2\vec{x}_1 \vec{x}_2 - \vec{x}_1^2 - \vec{x}_2^2$
The range of a_2^{new} is:

$$\begin{cases} if \ y_1 \neq y_2 \quad L = \max\left(0, a_2^{old} - a_1^{old}\right), \\ \qquad H = \min(C, C + a_2^{old} - a_1^{old}) \\ if \ y_1 = y_2 \quad L = \max\left(0, a_2^{old} + a_1^{old} - C\right), \\ \qquad H = \min(C, a_2^{old} + a_1^{old}) \end{cases} \quad (22)$$

After getting a_2^{new}, a_1^{new} can be calculated easily without considering the range which has been done for a_2^{new}. The next step is to update the normal W with Eq. (15). However, we have two new 'b' because there are two new KKT constrains with updated a. Respectively, we choose (1) b_1 if $0 \leq a_1^{new} \leq C$ and (2) b_2 if $0 \leq a_2^{new} \leq C$. Otherwise take their average. So far, a SMO cycle is completed. Keep running it until all samples satisfy KKT. The outcome of the SMO operation is a W and b. Put them into Eq. (10) we will have a classifier [23], [24].

Additionally, it is worth noting that we always mention Kernel function and non-linear classification in SVM [17], [18], [24]. SVM has a good performance in non-linear classification by mapping the non-linear date into high-dimensional feature space, where they can be linearly separated, using suitable kernel function such as polynomial kernels ($K(\vec{x}_i, \vec{x}_j) = (\overrightarrow{\vec{x}_i \vec{x}_j + 1})^d$) and the Gaussian radial basis function ($K(\vec{x}_i, \vec{x}_j) = \exp(-\frac{||\vec{x}_i, -\vec{x}_j||^2}{2\sigma^2})$) [24]. To implement it, we can replace each multiplying x vector, such as $\vec{x}_1 \vec{x}_2$, with kernel function, instead [23]. However, the experiment shows that the wrist-worn gesture data can be classified linearly in their space without mapping, and extra computation is required for kernel.

C. DAG SVM

A single SVM is only able to do binary classification. However, in this project, we need multi-class classifier since there are a certain number of gestures that need to be recognized. The general idea is to create several binary classifiers and realise multi-class classification by organizing their relation.

This section introduces a solution named Directed Acyclic Graph Support Vector Machines (DAG SVM) [24], [25]. Fig. 5(a) illustrates the fundamental structure of DAG for 5 gestures classification. Each circle represents a classifier created by SVM, and it is a Boolean function indicating the closest side of the current data is closer. When the system is doing classification, the route depends on the last result. The number of classifiers is (N(N-1)/2) and (N-1) of them have been called, where N is the number of categories [25]. This method is fast and has no overlap and unclassified situation. The disadvantage is the incorrect classification cannot be corrected. For example, if the first classifier does not work properly, the route will never go back, and this mistake will be continued [25].

D. Implementation

Fig. 5(b) illustrates the procedure and structure of machine learning in the current system. Firstly, the pre-recorded labelled samples are collected in calibration step at which the user will be required to show a certain number of gestures and system will store the corresponding data. In the current prototype with five attached sensors, the maximum amount of recognizable gesture is five whilst three is much more stable and accurate. Put these samples into SVM algorithm and calculate a W and b for every two gestures and use Eq. (10) to create decision hyperplane which is a classifier. The DAG consists of (N-1)/2 classifiers from SVM. When there is a new $\vec{x}$ representing the real-time value comes to the DAG flowchart, it will be classified to its most likely type. With this, the gesture is reconstructed and can be used to control something like prosthesis hand in this paper.

V. RESULTS AND DISCUSSION

Experiments were carried out in terms of the performance of pressure sensing and machine learning. We implemented the switched capacitor method on ATSAM3X8E to measure the capacitance. The fastest charging and sharing frequency it can achieve is 10^5 Hz with register programming, while 26,315Hz with Arduino library. The error regarding a change in capacitance can be kept within 0.3 pF at static state for a 230 pF capacitor with 25Hz sampling rate. We attached five capacitive pressure sensors on the wrist and applied pressure. The five pressure sensors are commercially available, with the range of 230 pF - 235.5 pf. However, the indicators in the system show that the values and ranges are much larger than the rated value in the manual when they are attached to a wrist and applied pressure (around 320 pF –360 pF). The reasons are sphygmus, small hand movement and the parasitic capacitor from the circuit and human skin. This issue does not have unsatisfied effect as long as the stability and repeatability of the sensors' output is acceptable because the SVM algorithm will adjust itself to the new range automatically. Subsequently, we can see a clear difference in the five capacitance values between each gesture (Fig. 6(b) and Fig. 6(c)). Following that we put these data into SVM along with their labels. The samples are five-dimensional since there are five sensors, but the five-dimensional data is non-displayable in the three-dimensional world. Therefore,
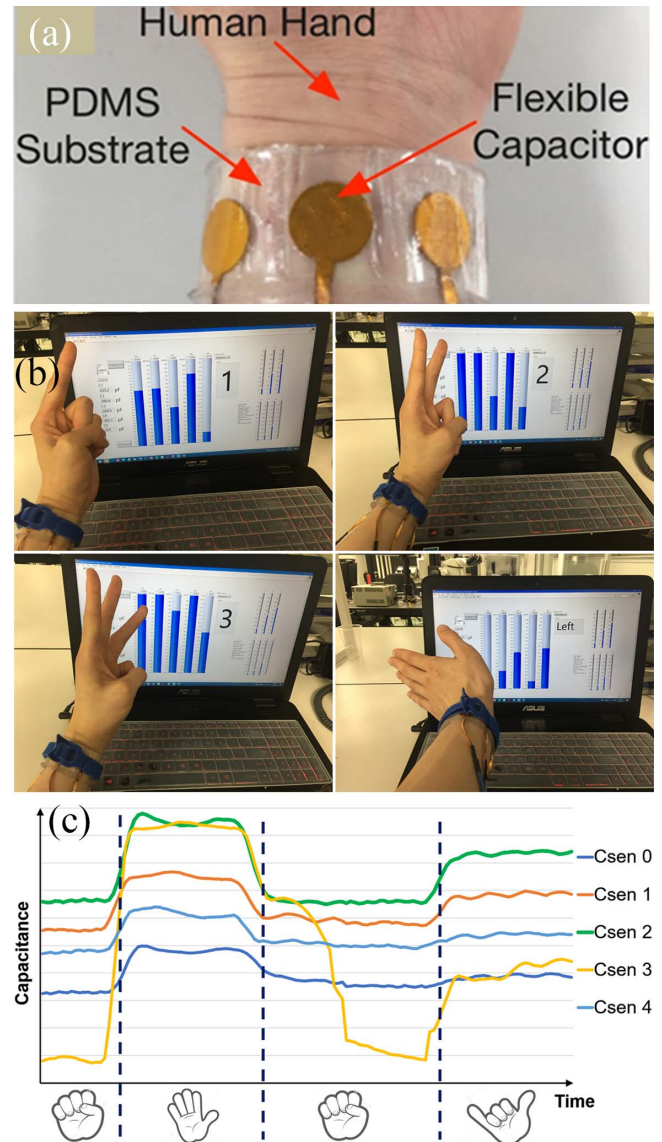

Fig. 6. (a) The five capacitive sensors on the human body, and (b) The five capacitance values change by changing gesture over time. (c) The change in gesture results in the capacitances variations over time.

we map them to three-dimension space to display them in the figures. Fig. 7(a) shows a classifier created by SVM algorithm and the two set of labelled samples. The gap between the two sample groups is considerably large for hyperplane to ideally separate them, which means the theoretical accuracy can reach 100%. Based on this, three classifiers are created for every two gestures, as shown in Fig. 7(b). Subsequently, the logical relationship of the three classifier and data flow has been organised according to the DAG structure (Fig. 5(a)).

With the current proof-of-concept prototype and the selected gestures, the accuracy of classification is approximately 90% with 25Hz refresh rate for three gestures classification. As can be seen in the Fig. 6(c), the first half of the third gesture (rock) performs a relatively slow drop and leads to a short misclassification due to the retraction after stretch of PDMS substrate, which is a shortcoming of the porotype.
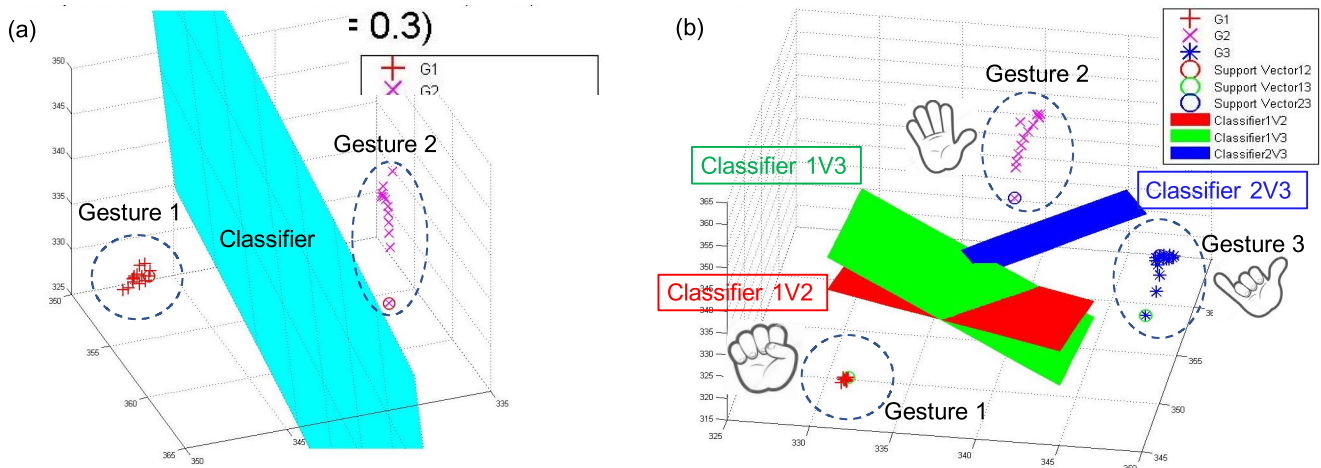
Fig. 7. By SVM algorithm, (a) the system can create a decision hyperplane that separates the different gestures. Under the DAG structure, we need 3 hyperplanes for 3 gestures recognition. (b) The 3 gestures data set are ideally separated by the 3 hyperplanes.

Both of the related works, WristFlex in [11] and Xue *et al.* [41]'s glove in, perform high accuracy (>80% and >95% respectively) while this prototype reaches 90%. Under the acceptable accuracy, the significant merit of this work is the combination of less on-body attachment and machine learning. Moreover, Xue *et al.*'s device with intensive signal processing, data fusion and adaptive DAG comprises of a glove, wristband, EMG device etc. The main difference between this work and WristFlex is the capacitive pressure sensor that is more sensitive to human skin and real-time DAG are applied in this work. On the other hand, the drawback of the current prototype is the limited number of recognizable gestures and the design of mechanical support (wristband) compared with the related works.

VI. Conclusion and Outlook

This work has successfully shown a complete system with accurate capacitance measurement and real-time multiclass classification. This prototype has unique potential in the wearable system domain. It also demonstrates how recent advances in machine learning make a sensing system smarter and provide further functionality in comparison to traditional sensing systems. It could be a good interface between humans and smart devices such as smartphones, watches, smart homes, high-tech sports equipment, etc.

However, the main drawback is that the performance highly relies on the stability and repeatability of the wristband which is easy to break. Therefore, future work is required to explore different materials for flexible pressure sensors, which could be embedded into a flexible substrate along with their circuit. On the other hand, we will also investigate applying Principal Component Analysis (PCA) algorithms on the labelled samples to remove the less-relevant features in order to reduce the samples' dimension and save computing resource further.

References

[1] Y. Li, H. Li, Y. Zhang, and D. Qiao, "Packet transmission policies for battery operated wireless sensor networks," *Frontiers Comput. Sci.*, vol. 4, pp. 365–375, Sep. 2010.

[2] N. Raveendranathan *et al.*, "From modeling to implementation of virtual sensors in body sensor networks," *IEEE Sensors J.*, vol. 12, no. 3, pp. 583–593, Mar. 2012.

[3] G. Fortino, G. Di Fatta, M. Pathan, and A. V. Vasilakos, "Cloud-assisted body area networks: State-of-the-art and future challenges," *Wireless Netw.*, vol. 20, no. 7, pp. 1925–1938, 2014.

[4] C. Zhu and W. Sheng, "Wearable sensor-based hand gesture and daily activity recognition for robot-assisted living," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 41, no. 3, pp. 569–573, May 2011.

[5] M. Kaboli, K. Yao, D. Feng, and G. Cheng, "Tactile-based active object discrimination and target object search in an unknown workspace," *Auton. Robots*, vol. 43, no. 3, pp. 1–30, 2018.

[6] N. Hu, "Human activity understanding for robot-assisted living," M.S. thesis, Univ. van Amsterdam, Amsterdam, The Netherlands, 2016. Accessed: Jun. 2018. [Online]. Available: http://www.uva.nl/binaries/content/assets/uva/nl/onderzoek/promoveren/samenvattingen/201/11/samenvatting-hu.pdf?2957735233497

[7] G. Fortino, R. Giannantonio, R. Gravina, P. Kuryloski, and R. Jafari, "Enabling effective programming and flexible management of efficient body sensor network applications," *IEEE Trans. Human-Mach. Syst.*, vol. 43, no. 1, pp. 115–133, Jan. 2013.

[8] S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 37, no. 3, pp. 311–324, May 2007.

[9] J. Kela *et al.*, "Accelerometer-based gesture control for a design environment," *Pers. Ubiquitous Comput.*, vol. 10, pp. 285–299, Aug. 2006.

[10] X. Liang, H. Heidari, and R. Dahiya, "Wearable capacitive-based wrist-worn gesture sensing system," in *Proc. New Gener. Circuits Syst. Conf. (NGCAS)*, Sep. 2017, pp. 181–184.

[11] A. Dementyev and J. A. Paradiso, "WristFlex: Low-power gesture input with wrist-worn pressure sensors," in *Proc. ACM Symp. User Interface Softw. Technol.*, 2014, pp. 161–166.

[12] Y. Zhang and C. Harrison, "Tomo: Wearable, low-cost electrical impedance tomography for hand gesture recognition," in *Proc. ACM Symp. User Interface Softw. Technol.*, 2015, pp. 167–173.

[13] N. Yogeswaran *et al.*, "New materials and advances in making electronic skin for interactive robots," *Adv. Robot.*, vol. 29, no. 21, pp. 1359–1373, 2015.

[14] H. Heidari, N. Wacker, and R. Dahiya, "Bending induced electrical response variations in ultra-thin flexible chips and device modeling," *Appl. Phys. Rev.*, vol. 4, no. 3, pp. 1603–1607, 2017.

[15] C.-Y. Chen, C.-L. Chang, T.-F. Chien, and C.-H. Luo, "Flexible PDMS electrode for one-point wearable wireless bio-potential acquisition," *Sens. Actuators A, Phys.*, vol. 203, pp. 20–28, Dec. 2013.

[16] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[17] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY, USA: Springer-Verlag, 1995.

[18] A. J. Smola and B. Schölkopf, *A Tutorial on Support Vector Regression*. Norwell, MA, USA: Kluwer, 2004.

[19] K. J. Kubota, J. A. Chen, and M. A. Little, "Machine learning for large-scale wearable sensor data in Parkinson's disease: Concepts, promises, pitfalls, and futures," *Movement Disorders*, vol. 31, pp. 1314–1326, Sep. 2016.

[20] M. Kaboli, D. Feng, and G. Cheng, "Active tactile transfer learning for object discrimination in an unstructured environment using multimodal robotic skin," *Int. J. Humanoid Robot.*, vol. 15, no. 1, p. 27, 2017.

[21] M. Kaboli and G. Cheng, "Robust tactile descriptors for discriminating objects from textural properties via artificial robotic skin," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 985–1003, Aug. 2018.

[22] H. Ghasemzadeh, R. Jafari, and B. Prabhakaran, "A body sensor network with electromyogram and inertial sensors: Multimodal interpretation of muscular activities," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 2, pp. 198–206, Mar. 2010.

[23] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "Improvements to Platt's SMO algorithm for SVM classifier design," *Neural Comput.*, vol. 13, no. 3, pp. 637–649, 2001.

[24] S. Nashat, A. Abdullah, S. Aramvith, and M. Z. Abdullah, "Support vector machine approach to real-time inspection of biscuits on moving conveyor belt," *Comput. Electron. Agricult.*, vol. 75, no. 1, pp. 147–158, 2011.

[25] P. Chen and S. Liu, "An improved DAG-SVM for multi-class classification," in *Proc. 5th Int. Conf. Natural Comput.*, Aug. 2009, pp. 460–462.

[26] A. Vilouras, H. Heidari, W. T. Navaraj, and R. Dahiya, *At-Home Computer-Aided Myoelectric Training System for Wrist Prosthesis*. Cham, Switzerland: Springer, 2016.

[27] H. Heidari, E. Bonizzoni, U. Gatti, and F. Maloberti, "A 0.18-μm CMOS current-mode Hall magnetic sensor with very low bias current and high sensitive front-end," in *Proc. Sensors*, Nov. 2014, pp. 1467–1470.

[28] H. Heidari, E. Bonizzoni, U. Gatti, and F. Maloberti, "A current-mode CMOS integrated microsystem for current spinning magnetic Hall sensors," in *Proc. IEEE Int. Symp. Circuits Syst.*, Jun. 2014, pp. 678–681.

[29] J. K. Perng, B. Fisher, S. Hollar, and K. S. J. Pister, "Acceleration sensing glove (ASG)," in *Int. Symp. Wearable Comput., Dig. Papers*, Oct. 2002, pp. 178–180.

[30] W. Klimesch, P. Sauseng, and S. Hanslmayr, "EEG alpha oscillations: The inhibition–timing hypothesis," *Brain Res. Rev.*, vol. 53, pp. 63–88, Jan. 2007.

[31] J. Allen, "Photoplethysmography and its application in clinical physiological measurement," *Physiol. Meas.*, vol. 28, no. 3, p. R1, 2007.

[32] O. Amft, H. Junker, P. Lukowicz, and G. Troster, C. Schuster, "Sensing muscle activities with body-worn sensors," in *Proc. Int. Workshop Wearable Implant. Body Sensor Netw.*, Apr. 2006, pp. 138–141.

[33] M. B. I. Reaz, M. S. Hussain, and F. Mohd-Yasin, "Techniques of EMG signal analysis: Detection, processing, classification and applications," *Biol. Procedures*, vol. 8, no. 1, pp. 11–35, Dec. 2006.

[34] R. J. K. Jacob, *The Use of Eye Movements in Human-Computer Interaction Techniques: What You Look at is What You Get*. San Mateo, CA, USA: Morgan Kaufmann, 1990.

[35] A. Vilouras, H. Heidari, S. Gupta, and R. Dahiya, "Modeling of CMOS devices and circuits on flexible ultrathin chips," *IEEE Trans. Electron Devices*, vol. 64, no. 5, pp. 2038–2046, May 2017.

[36] F. Miao, Y. Cheng, H. Yi, Q. He, and L. Ye, "A wearable context-aware ECG monitoring system integrated with built-in kinematic sensors of the smartphone," *Sensors*, vol. 15, pp. 11465–11484, May 2015.

[37] R. Jafari, H. Noshadi, S. Ghiasi, and M. Sarrafzadeh, "Adaptive electrocardiogram feature extraction on distributed embedded systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 17, no. 8, pp. 797–807, Aug. 2006.

[38] K. K. Biswas and S. K. Basu, "Gesture recognition using Microsoft Kinect," in *Proc. Int. Conf. Automat., Robot. Appl.*, Dec. 2012, pp. 100–103.

[39] Z. Ren, J. Yuan, J. Meng, and Z. Zhang, "Robust part-based hand gesture recognition using Kinect sensor," *IEEE Trans. Multimedia*, vol. 15, no. 5, pp. 1110–1120, Aug. 2013.

[40] D. Kim *et al.*, *Digits: Freehand 3D Interactions Anywhere Using a Wrist-Worn Gloveless Sensor*, Office for Official Publications of the European Communities, 2013.

[41] Y. Xue, Z. Ju, K. Xiang, J. Chen, and H. Liu, "Multiple sensors based hand motion recognition using adaptive directed acyclic graph," *Appl. Sci.*, vol. 7, no. 4, p. 358, 2017.

[42] H. Wen, J. R. Rojas, and A. K. Dey, "Serendipity: Finger gesture recognition using an off-the-shelf smartwatch," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2016, pp. 3847–3851.

[43] H. K. Chu, J. K. Mills, and W. L. Cleghorn, "Design of a high sensitivity capacitive force sensor," in *Proc. IEEE Conf. Nanotechnol. (IEEE NANO)*, Aug. 2008, pp. 29–33.

[44] *Measuring Small Changes in Capacitive Sensors*, M. T. Inc., 2005.

[45] SingleTact. *SingleTact Datasheet*. Accessed: Jun. 2018. [Online]. Available: https://www.singletact.com/SingleTact_Datasheet.pdf

[46] J. Travis and J. Kring, *LabVIEW for Everyone: Graphical Programming Made Easy and Fun*. Upper Saddle River, NJ, USA: Prentice-Hall, 2006.

[47] G. Cauwenberghs and T. Poggio, "Incremental and decremental support vector machine learning," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2000, pp. 388–394.

[48] R. Sitaram, S. Lee, S. Ruiz, M. Rana, R. Veit, and N. Birbaumer, "Real-time support vector classification and feedback of multiple emotional brain states," *NeuroImage*, vol. 56, pp. 753–765, May 2011.

[49] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *J. Mach. Learn. Res.*, vol. 2, pp. 45–66, Nov. 2001.
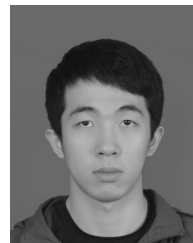
Xiangpeng Liang (S'18) received the B.Eng. degree from the Guangzhou College, South China University of Technology, China, and the M.Sc. (Hons.) degree from the University of Glasgow in 2017. He is currently pursuing the Ph.D. degree in electronic and electrical engineering with the Microelectronics Lab with research interests in the field of neuromorphic chip. He has spent a year as a placement with Zengyi Technology, Beijing.

Rami Ghannam (SM'18) received the B.Eng. (Hons.) degree from King's College, the DIC and M.Sc. degrees from Imperial College London, and the Ph.D. degree. He is currently a Lecturer in Electronic and Electrical Engineering with research interests in the broad field of photonics. Following his Ph.D. from Cambridge University in 2007, he has spent the past 10 years in the field of photovoltaics. He has held previous industrial appointments at Nortel Networks, IBM Research GmbH, and Starkon S.A.E. He is also a member of the Microelectronics Lab, where he is also investigating the use of PV cells for energy harvesting applications in wearable and implantable electronic devices. He is a member of IET.

Hadi Heidari (S'11–M'15–SM'17) received the Ph.D. degree in microelectronics from the University of Pavia, Italy, in 2015, where he worked on Integrated CMOS Sensory Microsystems. He is currently a Lecturer (Assistant Professor) with the School of Engineering and the Lead of the Microelectronics Lab, University of Glasgow, U.K. He has authored over 80 articles in peer reviewed journals (e.g., the IEEE SOLID-STATE CIRCUITS JOURNAL, TRANSACTIONS CIRCUITS AND SYSTEMS I, and the IEEE TRANSACTIONS ELECTRON DEVICES) and in international conferences. He has organized several conferences, workshops and special sessions, e.g., he is also the Founder Chairs of the U.K.-China Emerging Technology Workshop, the U.K. Circuits and Systems Workshop, and a member of organizing committee of SENSORS from 2017 to 2018, BioCAS in 2018, and PRIME in 2019. He is a member of the IEEE Circuits and Systems Society Board of Governors, the IEEE Sensors Council, and the IEEE Solid-State Circuits Society Administrative Committee. He has received several best paper awards from the IEEE international conferences, including ISCAS'14, PRIME'14, ISSCC'16, and travel scholarship from IEEE NGCAS'17. He has a grant portfolio of £1 million funded by major research councils and funding organizations, including European Commission, U.K.'s EPSRC, Royal Society, and Scottish Funding Council. He is an Editor for the Elsevier *Microelectronics Journal* and a lead Guest Editor for four journal special issues.