

A Smart Chair Sitting Posture Recognition System Using Flex Sensors and FPGA Implemented Artificial Neural Network

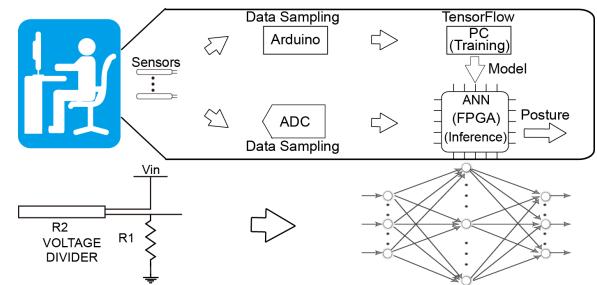
Qisong Hu, *Student Member, IEEE*, Xiaochen Tang, *Student Member, IEEE*, and Wei Tang, *Member, IEEE*

Abstract—Sitting is the most common status of modern human beings. Some sitting postures may bring health issues. To prevent the harm from bad sitting postures, a local sitting posture recognition system is desired with low power consumption and low computing overhead. The system should also provide good user experience with accuracy and privacy. This paper reports a novel posture recognition system on an office chair that can categorize seven different health-related sitting postures. The system uses six flex sensors, an Analog to Digital Converter (ADC) board and a Machine Learning algorithm of a two-layer Artificial Neural Network (ANN) implemented on a Spartan-6 Field Programmable Gate Array (FPGA). The system achieves 97.78% accuracy with a floating-point evaluation and 97.43% accuracy with the 9-bit fixed-point implementation. The ADC control logic and the ANN are constructed with a maximum propagation delay of 8.714 ns. The dynamic power consumption is 7.35 mW when the sampling rate is 5 Sample/second with the clock frequency of 5 MHz.

Index Terms—Smart Chair, Sitting Posture Recognition, Flex Sensors, Artificial Neural Network, Real-time Machine Learning.

I. INTRODUCTION

Sitting is the most common status of modern human beings and poor postures may affect head/neck posture and cervico-thoracic muscle activity [1], bring health problems, especially for young students [2]. For example, Keeping a neutral lumbar position is very important for health. However, the habitual sitting posture causes more flexed lower lumbar spine [3], which may increase health risks. According to a survey [4], people are sitting on an average of 13 hours a day. However, a long period of sitting may increase risk of obesity and metabolic diseases [5]. With bad postures, it may bring more health problems like postural pain [1]. Besides, improper sitting postures with a long period sedentary life may increase the risk of hyperflexion injury [6], and may cause musculoskeletal disorders such as back pain with deteriorating lung function [7], low back pain or injury [8], pains in muscle and connective tissues of tendons [9], increasing spine load [10], changing cervical spine position [11], neck pain [12], pressure ulcers in



some patients [13] and shoulder pain [14].

For the above-mentioned long-time sitting caused heart disease problems, wearable sensors [15], [16] were proposed to monitor the status of the heart. The warning systems were created to prevent delayed treatment, such as the wireless transceivers [17], the integrated Ultra-wideband (UWB) communication system [18], [19] using On-off Keying (OOK) and Frequency-shift Keying (FSK), and the system [20] with Delta-sigma encoder. Though, those solutions may help people avoid serious conditions, considering user experience, privacy, reliability, and hardware resource overhead, a low-power locally-implemented real-time monitoring system for directly sitting posture recognition is expected [21]. Thus, the sitting posture recognition system has become an attractive topic with help from novel sensors and machine-learning technologies.

The current sitting posture recognition systems proposed in the literature can be categorized by the sensor types. The main types of sensors include vision sensors, accelerometer sensors, pressure sensors, and textile sensors. For example, [22] reports vision-based system with the webcam and [23], [24] present recognition systems applying the Microsoft Kinect sensor. Accelerometer sensors are applied in [25], [26], and skin-mounted electromagnetic tracking sensors are used in [27]. However, the vision-based and the accelerometer-based systems suffer from poor robustness and susceptibility to the interference from the environment [24]. Moreover, the feeling of being spied results in a bad user experience. Furthermore,

Manuscript received Dec 4, 2019; This work is partially supported by United States National Science Foundation grant 1652944. Copyright (c) 2019 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending an email to pubs-permissions@ieee.org.

Qisong Hu, Xiaochen Tang, and Wei Tang are with the Klipsch School of Electrical and Computer Engineering, New Mexico State University, USA.

Correspondence should be addressed to: Wei Tang 1125 Frenger Mall, Las Cruces, New Mexico 88003 USA. E-mail: wtang@nmsu.edu

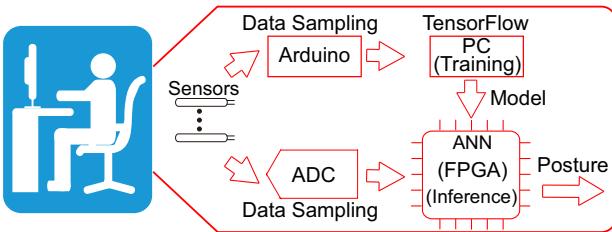


Fig. 1: System building blocks of the sitting posture recognition system.

background clutter and occlusion cause challenge to the vision-based sitting recognition system [28].

In contrast to those sensors, pressure sensors [29] and textile sensors [30] can provide an acceptable user experience with dignity and reliable results. However, these methods usually apply a large sensor array. For instance, 1176 sensors (42x28) were deployed in [29] and 96 sensors were consumed in [30]. This aggravates the processing complexity and hardware resource overhead, which is usually associated with a shortened battery life. As a result, the classifier of [29] is running on the computer which cannot be simply merged on the chair. A hardware friendly sitting detection solution was reported in [31], which applies 4x4 force-sensitive resistor (FSR) sensors and the processing algorithm is implemented on a microcontroller (MCU). However, it only detects whether someone is sitting on the chair, but cannot classify different sitting postures. Besides, the systems [29]–[31] that use personal computers (PCs) to process real-time data bring privacy issues.

To address the aforementioned problems, a novel smart chair solution in terms of sensing and processing for sitting posture recognition is proposed. The system building block diagram is illustrated in Fig. 1. The main contribution of this work is to introduce flex sensors with a machine learning algorithm to build a low-complexity hardware system for sitting posture recognition. In this work, only six passive flex sensors are attached to the chair with a sampling rate at 5 Hz. Each sensor is connected to a simple voltage divider to convert the variation of its resistance, which is caused by the deformation of the sensor, into an analog voltage. An ADC board is used to collect the analog voltage from the flex sensors and convert them into digital format. The digital signal is then processed on an FPGA with a machine learning algorithm. The algorithm is implemented as a two-layer artificial neural network (ANN) as the classifier that classifies seven different health-related sitting postures. The hyperbolic tangent (tanh) activation function is applied because of its simplicity, which made it friendly for Very-large-scale integration (VLSI) implementation [32]. For example, the hyperbolic tangent activation function was applied for optical pattern classification [33]. The overall system is implemented on an office chair.

The main research contributions of this paper are: (1) The proposed system reduces the number of sensors and computation complexity compared with the pressure and textile sensor systems, which results in less hardware overhead. This energy-efficient, private, and reliable continuous sitting posture recognition system will play a more valuable role in preventing people from getting harmed from musculoskeletal disorders

while protecting the dignity of the users; (2) The proposed method does not have a drift problem, so frequent calibrations can be avoided. Therefore, it is more reliable and power-efficient than the accelerometer-based system, which means longer battery life; (3) This proposed system provides a better user experience on privacy than the vision-based system. The rest of the paper is organized as follows: Section II presents the system design overview. Section III describes the hardware implementation. Section IV provides the experimental results. Section V concludes the paper.

II. SYSTEM DESIGN

The smart chair system consists of an array of six flex sensors, an ADC board, and an FPGA implementing the Artificial Neural Network. The output of the system is the classification result of the sitting posture. We selected seven health-related sitting postures, which are shown on the top of Fig. 2. The sitting postures are: (a) sit straight; (b) left recline; (c) right recline; (d) lounge; (e) lean backward; (f) cross left leg; (g) cross right leg. Fig. 2 also illustrated the recorded sensor output waveforms, belonging to three participants, from the six sensors according to the sitting posture. The seven different postures are selected for the following reasons. Posture (a) and (d) are very common postures. Posture (b) and (c) may cause contractures and exacerbate muscle imbalances [34], while posture (e) may increase tension in the muscles, which may in turn cause pain [35]. Posture (f) and (g) may cause sagittal imbalance, coronal imbalance, pelvic obliquity, and lordosis angle [36]. The following subsections describe the sensor interface and signal processing in detail.

A. Flex Sensor

The flex sensor is made up of a polymer ink with conductive particles and plastic flake. When a flex sensor is bent into different shapes, the distance of conductive particles changes with the shape of the sensor, which results in the difference of resistance. The short flex sensor applied in the application is FS-L-0055-253-ST from Spectra Symbol. The sensor has a length of 73.7mm. The active length is 55.4 mm. The width is 6.4 mm and the thickness is 0.5 mm. Its weight is 0.27 g. The flat resistance is 25k Ohm. The bending resistance can reach 45k to 125k Ohms. The change of the resistance is converted to the change of voltages using voltage dividers with a 5V power supply. Each flex sensor, as shown in Fig. 3 (a), has two terminals. The resistance between the two terminals changes with the flexing or bending (Fig. 3 (b)) of the sensor. The individual sensor is connected to a resistor to form a voltage divider (Fig. 3 (c)). The voltage output of the voltage divider represents the distortion of the sensor.

B. Ethical approval and Participant Selection

Ethical approval for this work has been granted by The Office of Research Compliance of New Mexico State University. Each participant has been explained the Key Information and has signed the Informed Consent Form. The selected participants consist of 11 healthy young people (4 female and

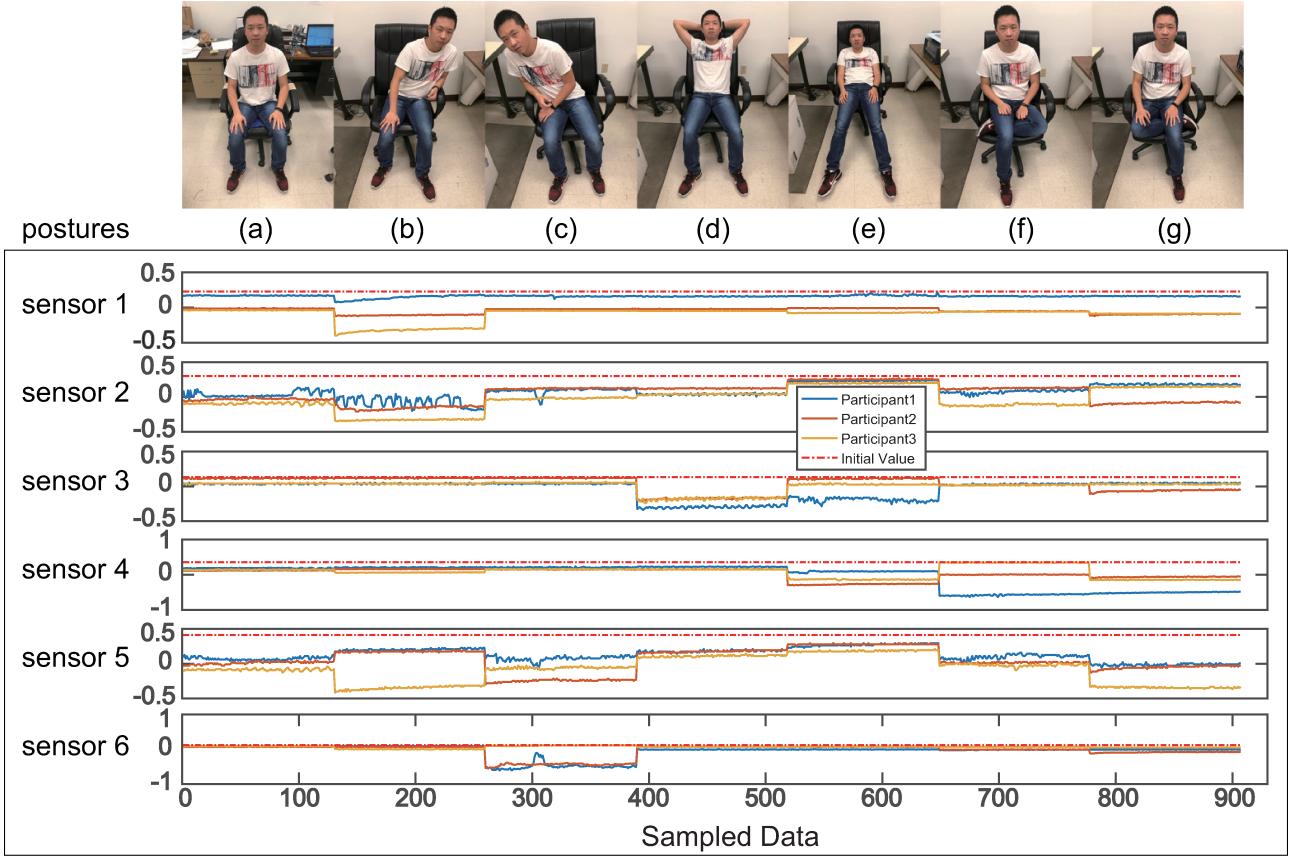


Fig. 2: Seven typical sitting postures (Top) with corresponding waveform of three participants from each flex sensor (Bottom). (a) sit straight; (b) left recline; (c) right recline; (d) lounge; (e) lean backward; (f) cross left leg; (g) cross right leg.

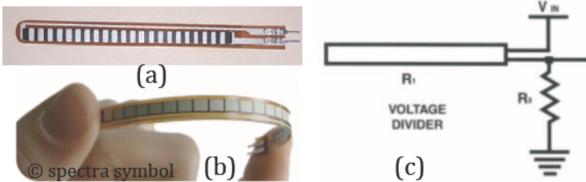


Fig. 3: Flex Sensor and its application (a) flat status (b) bending with various resistance (c) voltage divider readout circuit.

7 male) with the age ranging from 25 to 33, mean \pm standard deviation as 29 ± 2.68 ; weight ranging from 45 to 95 kg, mean \pm standard deviation as 66.63 ± 14.46 kg; height ranging from 159 to 186 cm, mean \pm standard deviation as 169 ± 9.27 cm. The participants were asked to sit on the smart chair with seven different common postures including straight, lean left, lean right, lounge, lean backward, cross left leg, and cross right leg. The participant keeps each posture for 30 seconds for data collection. The identification of each participant is anonymously coded and stored in a classified file.

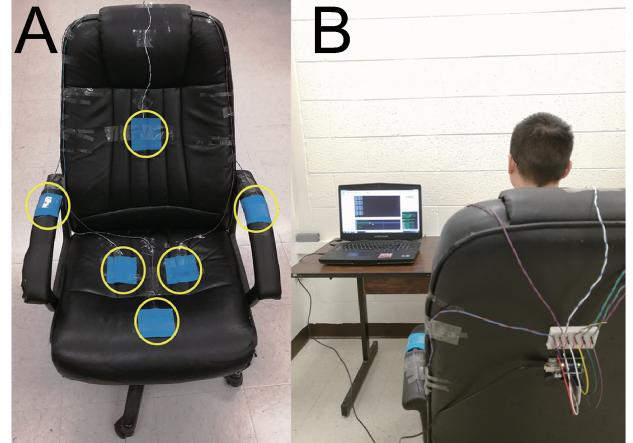


Fig. 4: Flex sensor based sitting posture recognition system (left) and the circuit deployment (right). Wires and circuits are implemented on the chair.

C. Sensor Interface

The flex sensors are attached to the chair and connected to the input pins of the Arduino board or the ADC board through individual voltage dividers. Each value of the voltage is sampled by the ADC in the Arduino board for training or the ADC board for inference. The flow of data, from sensors

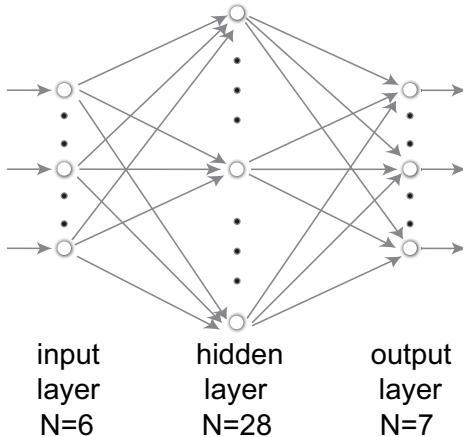


Fig. 5: Two layer Artificial Neural Network for sitting posture classification.

to Arduino board or ADC board, is shown in Fig.1 as well. The ADC in the Arduino board has a 10-bit resolution with a sampling rate of up to 9.6 kS/s while the ADC board equipped by the device with a 16-bit resolution and sampling rate up to 200 kS/s. The Tensorflow 1.12 is installed on a PC with a Core i5 CPU, a GTX 1060 GPU and an Ubuntu 16.04 Operating System. The Arduino device was used to collect the training data. The training process is performed off-line on a PC with TensorFlow. In the training process, the voltage at one end of each sensor was sampled by the built-in ADCs of the Arduino. The training data were then transmitted to the PC using Universal Asynchronous Receiver/Transmitter (UART) and processed by the python program on the PC. In the inference stage, the system uses an ADC board to obtain real-time data from the sensors. Then the ADC sends the data to the FPGA for inference. The sensors are deployed on the top of the seat, the backrest, and the armrests on the chair as shown in Fig. 4. The hardware board and power supply are attached on the back of the chair.

D. Artificial Neural Network

A two-layer Artificial Neural Network is applied in the system as shown in Fig. 5. The ANN has six inputs from the six flex sensors. The input layer is not considered as an ANN layer since it does not perform computing. The inputs are connected to the internal nodes in the hidden layer. Then the output layer generates a classification result, which is one of the seven different sitting postures.

The ANN design procedure includes four main steps. The first step is to find the number of layers (k) and the number of the nodes (l) in the hidden layer. This is done using TensorFlow simulation on the PC with the stored training data to perform cross-validations to evaluate the classification accuracy of each combination of (k, l) , and find the optimized numbers. We used 10-fold cross-validation to find out the best combination of k and l . In the experiment, we set the number of layers k to 1 or 2. This is because when k is larger than 2, the computation overhead will increase more than double.

Then the number of nodes l at the hidden layer is set between 13 and 30. This is because our number of the input node is 6, and the number of the hidden layer nodes should be 3-5 times of the number of the input nodes, which is a rule-of-thumb of choosing the number of nodes in the hidden layers. The results of each combination of k and l for the 10-fold cross-validation are in Table I.

In Table I, the first column is the number of layers; the second column is the number of nodes of the hidden layer; from the third to the twelfth columns are the accuracy of the first round to the tenth round of the 10-fold cross-validation; the last column is the accuracy with each combination where m stands for mean and s stands for standard deviation. The experiment shows that the best combination is $k = 2$, $l = 28$.

The second step is to obtain the floating-point weight and bias by training the ANN with back propagation (BP) algorithm with iterative and adaptive process, in which we set the feed batch size as 100 and the epoch as 990. In the third step, the fixed point weight and bias are finalized using Matlab fixed point converter by evaluating the system performance at the different numbers of bits and fractions. This step also estimates the fixed-point model with the activation function. In this experiment, the word length is finalized as 9 bits with 5 bits as the fraction. The activation function is finalized as a 9-bit Look-up Table. The last step before hardware implementation is to simulate the overall fixed-point model in Matlab with the raw input data and evaluate the system performance.

E. ANN vs. SVM

In order to compare the performance between different machine-learning methods, we also studied the support vector machine (SVM) algorithm as an alternative method in our sitting posture recognition system. It is well known that compared to neural network methods, SVM with linear kernel has lower computational overhead, which makes it easier to be implemented in wearable sensors. However, neural network methods may achieve higher accuracy in scenarios with looser power consumption requirements. To compare their performance, We built a linear SVM based classifier. The sensor inputs are applied as features. We used the same 10-fold cross-validation to find out the best penalty parameter C for the SVM. The optimized C is found as 128. Then we trained and evaluated the SVM model with the same dataset as the ANN. The SVM achieved an accuracy of 88.37%, which is lower than the ANN (97.43%). The results of the 10-fold cross-validation are shown in Table II.

In Table II, the first column is the penalty parameters, and the parameters range from 2^{-5} to 2^9 ; from the second to the eleventh columns are the accuracy of the first round to the tenth round of the 10-fold cross-validation with different penalty parameters; the twelfth column is the average accuracy and the standard deviation. C stands for penalty parameter, $Rnd1$ to $Rnd10$ stand for the first to the tenth fold cross-validation, m and s stands for mean and standard deviation, respectively. The results show that in our smart chair setup, ANN achieves a higher accuracy than SVM. It is possible

TABLE I: 10-fold cross-validation for different combination of number of layers and number of nodes in the hidden layer.

# of layers	# of nodes	Accuracy										Analysis (m±s)
		Rnd1	Rnd2	Rnd3	Rnd4	Rnd5	Rnd6	Rnd7	Rnd8	Rnd9	Rnd10	
1	N/A	0.758	0.705	0.710	0.727	0.712	0.735	0.712	0.745	0.657	0.757	0.7216±0.0301
2	13	0.945	0.905	0.962	0.934	0.929	0.940	0.932	0.917	0.949	0.929	0.9341±0.0161
2	14	0.960	0.948	0.957	0.943	0.903	0.937	0.952	0.942	0.952	0.934	0.9427±0.0163
2	15	0.943	0.939	0.952	0.979	0.932	0.980	0.940	0.919	0.934	0.951	0.9467±0.0196
2	16	0.952	0.949	0.966	0.960	0.971	0.955	0.917	0.949	0.972	0.951	0.9543±0.0157
2	17	0.959	0.959	0.940	0.948	0.946	0.949	0.974	0.925	0.954	0.955	0.95077±0.0129
2	18	0.935	0.957	0.969	0.942	0.931	0.965	0.946	0.949	0.952	0.968	0.95138±0.0133
2	19	0.942	0.957	0.975	0.974	0.928	0.972	0.955	0.929	0.939	0.979	0.95492±0.0196
2	20	0.962	0.965	0.971	0.955	0.955	0.982	0.951	0.937	0.975	0.962	0.96138±0.0128
2	21	0.975	0.931	0.982	0.982	0.940	0.992	0.963	0.939	0.955	0.959	0.9617±0.0208
2	22	0.951	0.968	0.959	0.963	0.955	0.968	0.951	0.955	0.969	0.968	0.96063±0.0073
2	23	0.959	0.942	0.969	0.963	0.949	0.963	0.969	0.943	0.991	0.946	0.95939±0.0151
2	24	0.966	0.960	0.975	0.955	0.975	0.952	0.945	0.948	0.971	0.971	0.96186±0.0114
2	25	0.963	0.957	0.975	0.969	0.937	0.979	0.974	0.955	0.974	0.935	0.96184±0.0155
2	26	0.975	0.945	0.974	0.943	0.928	0.982	0.955	0.962	0.955	0.966	0.95846±0.0167
2	27	0.963	0.960	0.966	0.948	0.968	0.975	0.963	0.949	0.971	0.940	0.96032±0.0112
2	28	0.957	0.946	0.982	0.963	0.959	0.991	0.966	0.962	0.969	0.982	0.96754±0.0135
2	29	0.988	0.949	0.985	0.963	0.929	0.968	0.982	0.952	0.963	0.959	0.96369±0.0179
2	30	0.977	0.942	0.972	0.963	0.939	0.965	0.968	0.954	0.983	0.952	0.96138±0.0146

TABLE II: 10-fold cross-validation for different penalty parameters with the linear SVM.

C	Accuracy										Analysis (m±s)
	Rnd1	Rnd2	Rnd3	Rnd4	Rnd5	Rnd6	Rnd7	Rnd8	Rnd9	Rnd10	
0.0312	0.6863	0.6906	0.7151	0.7151	0.6806	0.6763	0.6734	0.7036	0.6791	0.7094	0.69295±0.0164
0.0625	0.7022	0.7165	0.7266	0.7511	0.7079	0.7165	0.7108	0.7094	0.6763	0.7453	0.71626±0.0213
0.125	0.777	0.7525	0.7899	0.7986	0.7727	0.7914	0.777	0.7799	0.741	0.7971	0.77771±0.0187
0.25	0.7626	0.7914	0.7914	0.8014	0.7842	0.7871	0.7799	0.7856	0.7511	0.8058	0.78405±0.0165
0.5	0.8158	0.8014	0.823	0.836	0.7971	0.8245	0.8273	0.8216	0.7957	0.8432	0.81856±0.0161
1	0.8432	0.823	0.8432	0.8475	0.8043	0.8475	0.8489	0.8345	0.8259	0.8518	0.83698±0.0151
2	0.8561	0.8317	0.8561	0.859	0.8288	0.8604	0.8576	0.8388	0.8388	0.8719	0.84992±0.0143
4	0.8604	0.8345	0.859	0.859	0.8331	0.859	0.859	0.8489	0.8504	0.8748	0.85381±0.0126
8	0.8691	0.8374	0.8619	0.8647	0.8518	0.8647	0.859	0.8518	0.8633	0.8791	0.86028±0.0113
16	0.8791	0.8547	0.8734	0.8705	0.8662	0.8719	0.8791	0.8662	0.8705	0.895	0.87266±0.0105
32	0.8777	0.8619	0.8863	0.8806	0.8705	0.8777	0.8892	0.8676	0.8691	0.9065	0.87871±0.0129
64	0.8806	0.8647	0.8921	0.882	0.8863	0.8748	0.8892	0.8691	0.8719	0.9094	0.88201±0.0131
128	0.882	0.8719	0.8921	0.8806	0.8892	0.8777	0.8863	0.8705	0.8734	0.9137	0.88374±0.0128
256	0.882	0.8691	0.8921	0.882	0.8878	0.8748	0.8849	0.8719	0.8734	0.9137	0.88317±0.0130
512	0.8791	0.8647	0.8935	0.882	0.8878	0.8719	0.8835	0.8719	0.8719	0.9137	0.882±0.0141

that other machine learning algorithms or other sensor systems may achieve a higher accuracy, which is an interesting topic for future studies.

III. HARDWARE IMPLEMENTATION

Several efforts are made to implement the ANN on the FPGA. This section presents the specific steps and considerations of the hardware-algorithm interface.

A. Data Normalization

The first step in hardware implementation is to normalize the input signal. With a 5V power supply, the input signal from the flex sensor is between 2.5 V and 3.3 V, which is normalized to a data range from -1 to +1 for the input of the ANN. Normalization of the input signal is achieved by a Look-up Table (LUT) in order to avoid a hardware divider. Then the ANN performs multiply-and-accumulate (MAC) operations

using the input data, as well as the weights and biases data. The data range of the weight and bias is between -9.7 and 7.949 which is truncated to the range from -8 to 7.949. The output ranges of the two-layer ANN are (-6.254, 7.239) at the hidden layer and (-25, 29) at the output layer. Since the floating-point operation is power-hungry in hardware, we implement the fixed-point operation for MAC in the ANN. For example, in this design, all data, weights, and biases used as input operands for the MAC operation in the ANN are quantized to signed 2's complement binary numbers with a word length of 9-bit, including 5 bits for the fraction. The outputs of MAC have a word length of 18-bit with 10 bits for the fraction. These parameters for fixed-point binary number guarantee enough range to cover and present data, weights, and biases with reasonable accuracy and the performance in classification.

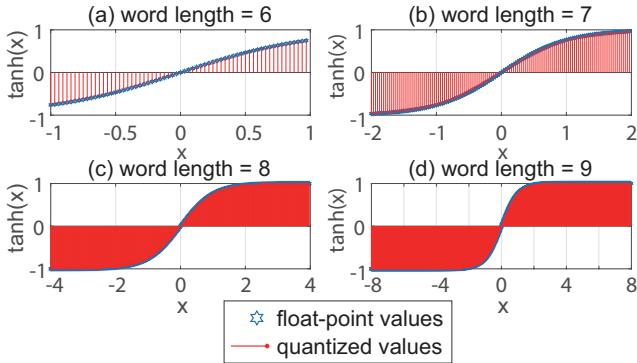


Fig. 6: Quantization of Hyperbolic tangent with different word lengths.

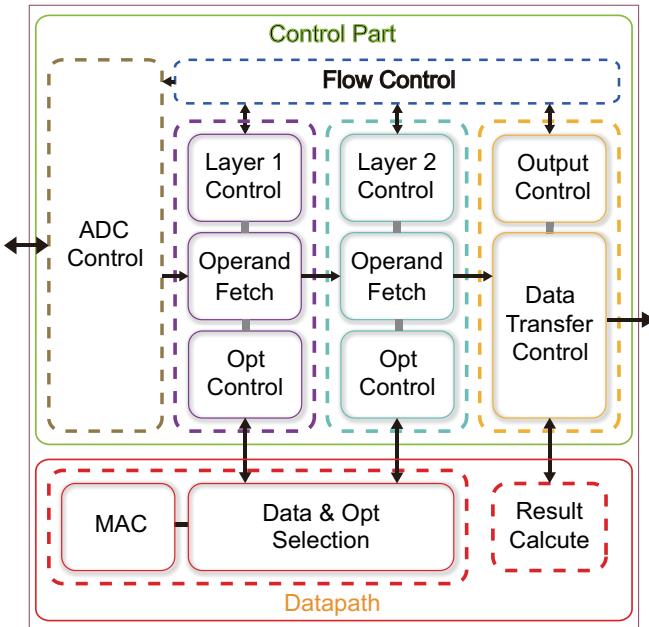


Fig. 7: Hardware architecture of the ANN implemented on the FPGA.

B. Look-up Table

Each output of the MAC should be processed by the nonlinear activation function. Thanks to the nature of differentiability, the Hyperbolic tangent activation function is fit for the back propagation algorithm. For example, the Hyperbolic tangent is also applied in the ANN [32], [33] which is used as a classifier. The activation function is also implemented using a LUT to save the hardware resource. In order to fit the fixed point system with the word length of 9-bit and 5 bits as the fraction, the LUT is designed to cover the input range from -8 to +8 with a standard deviation of residual as 0.0056. The floating-point Hyperbolic tangent sequences and quantized Hyperbolic tangent sequences with a word length of input data as 6 bits, 7 bits, 8 bits, and 9 bits are shown in Fig. 6. The LUT for Hyperbolic tangent has a size of 512x9

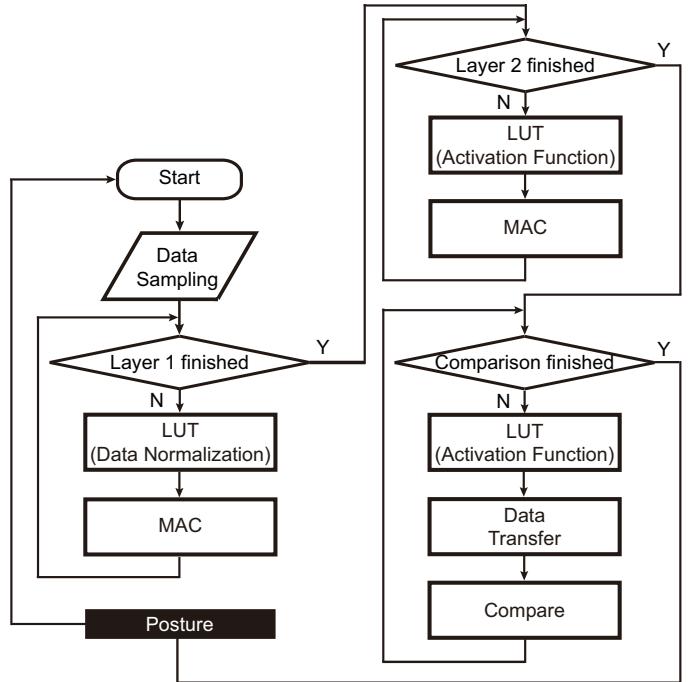


Fig. 8: Flowchart of the ANN on the FPGA.

bits.

C. System Implementation

The system architecture implemented on the FPGA is illustrated in Fig. 7. The flowchart of the ANN on the FPGA is shown in Fig.8.

The system consists of a Control Part module and a Data-path module.

1) Control Logic: The Control Part contains an ADC logic module, a flow control module and three control models for computation at the input layer, the hidden layer, and the output of the ANN. The system operates as follows: First, the ADC Control module sends control signals to the ADC and receives data from the ADC with a sampling rate of 5 samples/second. Then, the Flow Control module manages the ANN processing flow. Each layer of the ANN starts running after receiving the beginning signals from the Flow Control module. At each layer, the Layer Control module sends the address of the operands to the Operand Fetch module. Then the Operand Fetch module fetches the corresponding operands from the LUT. Next, the Operation Control module sends the operand from the Operand Fetch module, along with control signals, to the Datapath module. The difference between Layer 1 control logic and Layer 2 control logic is the address generation operations due to the structure of the LUT. The control logic for the output layer consists of the Output Control module and the Data Transfer Control module. The Output Control module starts work after receiving the beginning signal from the Flow Control module. Then the Output Control module sends the control signals to the Data Transfer Control module. After that, the Data Transfer Control module converts data from signed

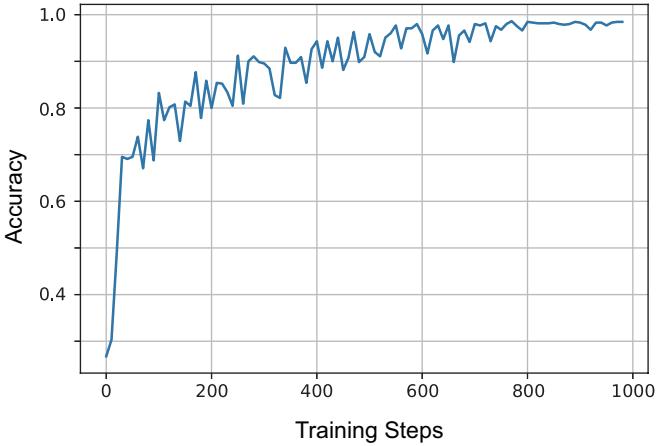


Fig. 9: Accuracy curve during the training stage.

2's complement binary code to the true binary code, which facilitates the comparison operations.

2) Datapath: The Datapath module is composed of a Data & Operation Selection module, a Multiply-Accumulate (MAC) module and a Result Calculate module. The Data & Operation Select module contains the ports connected to Control Part and the MAC and multiplexers. The data and control signals are sent to this module from the Control Part, then the specific data and operation are selected by controlling the multiplexers. Then the MAC module performs the MAC calculation. After calculation, the result is sent back to the Data & Operation Section unit. Finally, The result is sent to a specific layer. The Result Calculate unit is responsible for finding the maximum value from the seven different values. Finally, the output data are transferred to the Datapath Result Calculate module. The output of the system is the classified sitting posture.

IV. EXPERIMENT RESULTS

In order to evaluate the performance of the smart chair sitting recognition system, we obtained data from a total of 11 people (4 females and 7 males). Each person participated in the experiment is required to sit on the chair to exercise all seven postures. The participant keeps one posture for 30 seconds and then change to another posture. The sensing system records data from sensors during the experiment. The total data are separated into two parts. The first part contains data from 9 people (4 females and 5 males) for training and evaluation, including the cross-validation of the ANN model. The second part of the data (2 people) is only used to validate the generalization of the ANN model to avoid over-fitting.

A. Validation results

The first part of the dataset collected from 9 participants has a total of 9791 samples from each sensor. The data are then organized in rows. Each row has six data samples corresponding to six sensors. The setup of the system when collecting the data are shown in Fig.4. Due to the different initial resistance of each sensor, the voltages at the six sensors have different ranges. Thus, data of each column (sensor) are normalized with its specific parameters into the range between -1 and 1 for training the ANN model. The curves for each sensor with three participants are also shown at the bottom of Fig. 2, corresponding to seven typical sitting postures. To recognize the different sitting postures, each row of the data is sent to the training stage or evaluation stage as an independent feature set. Then, 6527 (67%) samples are selected randomly to be the data for training a model, while other 3264 (33%) samples are selected as data to evaluate both of the trained floating-point model and the quantized fixed-point model. The 2-layer model with 28 nodes in the hidden layer achieves the accuracy at 97.86% at the training stage. The accuracy curve during training is shown in Fig. 9. At the evaluation stage, the data for evaluation is used to evaluate both of

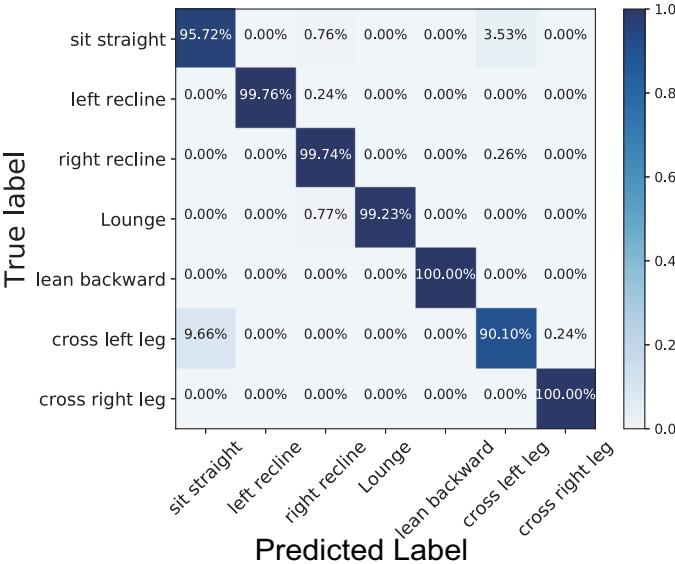


Fig. 10: Confusion matrix with the floating-point model and the data from the first part of the dataset.

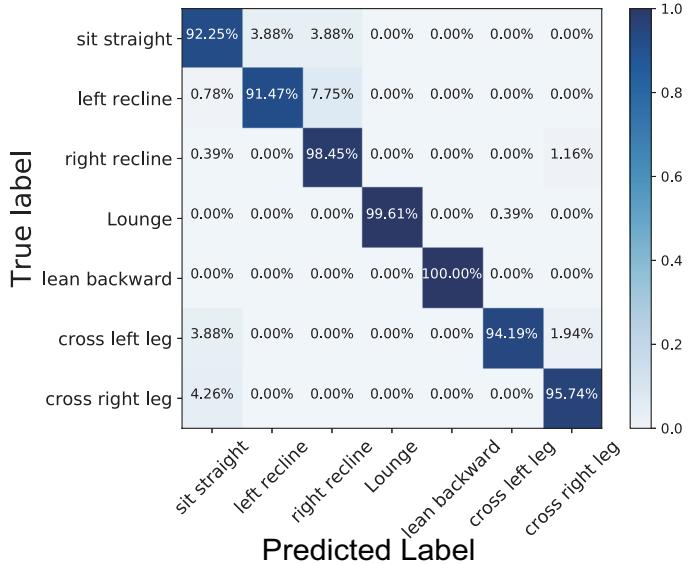
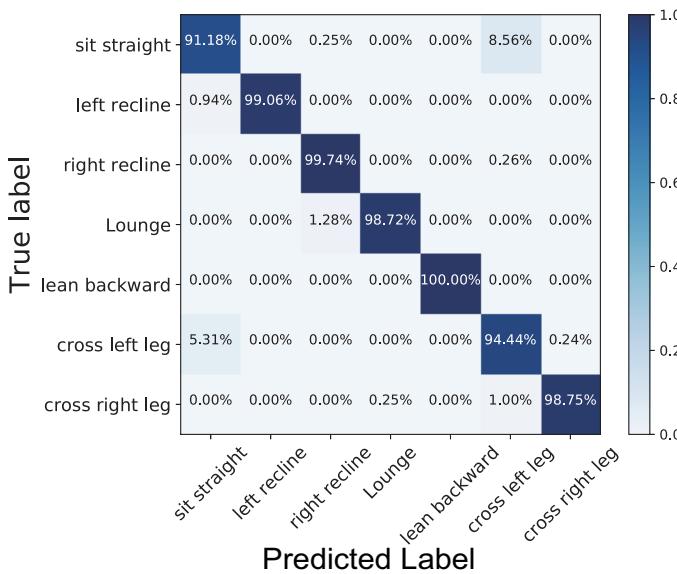


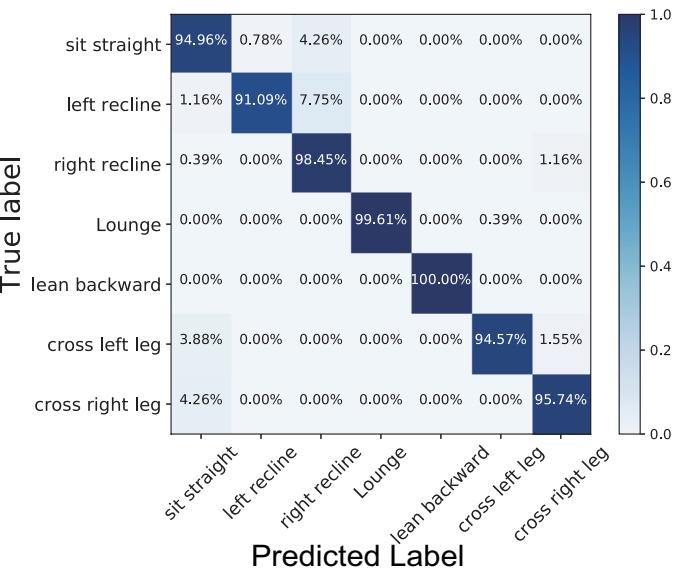
Fig. 11: Confusion matrix with the floating-point model and the data from the second part of the dataset.

TABLE III: Comparison between this work and state-of-the-art works

Works	Sensor Type	Number of Sensors (Power of Sensors)	Power of Sampling Device	Algorithm	Platforms (Hardware Resource)	Power of Processor	Processing Time	Accuracy
[22]	Camera	2 (N/A)	N/A	SVM	MATLAB (N/A)	N/A (PC)	N/A	61.9%
[23]	Kinect	2 (5W)	N/A	KNN	N/A (N/A)	35W avg (i7 CPI)	N/A	94%
[25]	Accelerometer	1 (1.65mW)	232.5mW (Arduino)	SVM	MCU PC (N/A)	N/A (PC)	N/A	95.33%
[29]	Pressure Distribution Sensors	42x28 (N/A)	N/A	PCA	PC (N/A)	35W avg (Pentium CPU)	N/A	96%
[30]	Textile Sensors	96 (N/A)	1.89mW (ADC)	Bayes	MCU PC (N/A)	N/A (PC)	N/A	82%
[31]	Force Sensitive Resistor	4x4 ((N/A))	198mW max (MCU)	Vector Measurement Algorithm	MCU PC (N/A)	N/A (PC)	N/A	N/A
[37]	Strain Gauge Sensor	8 (N/A)	N/A	Bayes	Mobility Monitor (N/A)	2.7W max (Mobility Monitor)	N/A	72.1% \pm 12%
[38]	Accelerometer	1 (N/A)	N/A (Smartphone)	ANN	Smartphone FPGA (81 DSP 569 FF 3466 LUT)	120mW (FPGA)	270ns	94.6%
this work	Flex Sensor	6 (2.34mW)	100mW max (ADC)	ANN	ADC FPGA (755 Slice Register 911 FF 659 LUT)	7.33mW (FPGA)	267.4μs	97.43%

**Fig. 12:** Confusion matrix with the fixed-point model and the data from the first part of the dataset.

the floating-point model and the fixed-point model, obtaining the accuracy at 97.78% and 97.43% respectively. The posture lean backward can be recognized with accuracy at 100% for both the floating-point model and the fixed-point model. The confusion matrix which shows the classification result with the floating-point model and data from the first part of the dataset is shown in Fig.10. The fixed-point system results

**Fig. 13:** Confusion matrix with the fixed-point model and the data from the second part of the dataset.

are shown in Fig. 12. According to the confusion matrices, the lowest accuracy, 90.1%, is recorded on the posture of cross-left-leg when applying the floating-point model, which is caused by the confusion between the posture of sit-straight and the posture of cross-left-leg. To find out the reason of notable confusion between these two sitting postures, the data from different participants has been analyzed. Some data about

posture cross left leg is close to the value about the posture sit straight, as shown in the Fig.2. The confusion in the data which was caused by personal sitting habits results in the confusion between posture sit straight and posture cross left leg.

The second part of the dataset has 1806 rows of the data collected from the other 2 participants (2 male). This part of data are used to validate the generalization of the ANN model, since the data is alien to the ANN model. The accuracy of the floating-point model and the fixed-point model reaches 96% and 96.3%, respectively, which validates the generalization of the ANN model. The confusion matrix of the classification result with the floating-point model and the data from the second part of the dataset is shown in Fig.11. The fixed-point system results are shown in Fig. 13.

As shown in the confusion matrices, when the fixed-point model is applied, over 7% of data from posture left recline is recognized as posture right recline, resulting in the low accuracy, 91%, of posture left recline.

B. Hardware and power results

Based on the verification of the fixed-point model and hardware structure design, a processing system is realized on an FPGA board of Spartan 6 XC6SLX9, using 755 slice registers, 659 slice LUTs, and 911 Flip-Flop pairs. The processing system has a maximum propagation delay as 8.714 ns with dynamic power as 7.35 mW, when the sampling rate is 5 Sample/second and the clock frequency is 5 MHz.

The sensor power consumption is very low in our system. Each voltage divider consists of two parts, one is the flex sensor while another part is a resistor with resistance as 42k ohm. According to the data we collected, the highest voltage is 3.3 V on the sensor, which means the current is 0.078 mA for each voltage divider with the power supply of 5 V. The total power consumption is 2.34 mW for all 6 voltage dividers and sensors.

V. DISCUSSION AND FUTURE WORK

The comparison between our system and state-of-the-art works, [22], [23], [25], [29]–[31], [37], [38], is presented in Table III. To the best of our knowledge, this work has achieved the lowest power consumption, the lowest hardware simplicity and the highest accuracy among the related works. The proposed system can be more energy-efficient and powerful. For example, the ADC board consumes the power up to 100mW, which is over 91% of the power budget of the entire system. In order to reduce power consumption, an integrated ADC will be designed to replace the ADC board. Besides, this recognition system will be a part of a smart health monitoring system which brings challenges of processing large amounts of data with higher dimension and achieving high performance with complicated tasks. The algorithms with complicated architecture like Convolutional Neural Network (CNN) will be considered as a potential choice to meet the challenges. Besides, the system will be optimized with respect to the real-time processing, the user usability and acceptability as well as mobility and comfortability.

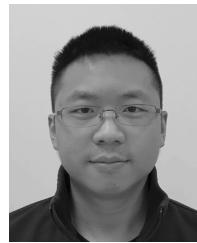
VI. CONCLUSION

A low-power private smart sitting posture recognition system was realized. The system achieved an accuracy of 97.78% with floating-point model and 97.43% with 9-bit fixed-point model. The dynamic power consumption is 7.35 mW with sampling rate as 5 Sample/second and maximum propagation delay as 8.714 ns. The primary novelty of the paper is the new type of sensor combined with fixed-point two-layer ANN model to achieve high accuracy, low computing overhead, and power consumption. The proposed system brings longer battery life, better user experience, and robustness compared to other types of sensing systems.

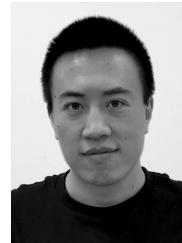
REFERENCES

- [1] J. P. Caneiro, P. O’Sullivan, A. Burnett, A. Barach, D. O’Neil, O. Tveit, and K. Olafsdottir, “The influence of different sitting postures on head/neck posture and muscle activity,” *Manual therapy*, vol. 15, no. 1, pp. 54–60, 2010.
- [2] J. Kratěnová, K. ŽEjglicová, M. Malý, and V. Filipová, “Prevalence and risk factors of poor posture in school children in the czech republic,” *Journal of School Health*, vol. 77, no. 3, pp. 131–137, 2007.
- [3] K. O’Sullivan, P. O’Dea, W. Dankaerts, P. O’Sullivan, A. Clifford, and L. O’Sullivan, “Neutral lumbar spine sitting posture in pain-free subjects,” *Manual Therapy*, vol. 15, no. 6, pp. 557–561, 2010.
- [4] Ruben and Castaneda, “10 ways poor posture can harm your health,” <https://health.usnews.com/wellness/slideshows/10-ways-poor-posture-can-harm-your-health>, accessed February 1, 2018.
- [5] B. A. Franklin, “Health implications of low cardiorespiratory fitness, too little exercise, and too much sitting time: changing paradigms and perceptions,” *American Journal of Health Promotion*, vol. 25, no. 4, pp. exi–exv, 2011.
- [6] S. McGill and S. Brown, “Creep response of the lumbar spine to prolonged full flexion,” *Clinical Biomechanics*, vol. 7, no. 1, pp. 43–46, 1992.
- [7] R. Tattersall and M. Walshaw, “Posture and cystic fibrosis.” *Journal of the royal society of medicine*, vol. 96, no. Suppl 43, p. 18, 2003.
- [8] T. A. Beach, R. J. Parkinson, J. P. Stothart, and J. P. Callaghan, “Effects of prolonged sitting on the passive flexion stiffness of the in vivo lumbar spine,” *The Spine Journal*, vol. 5, no. 2, pp. 145–154, 2005.
- [9] E. Grandjean and W. Hunting, “Ergonomics of posture—review of various problems of standing and sitting posture,” *Applied ergonomics*, vol. 8, no. 3, pp. 135–140, 1977.
- [10] M. De Wall, M. Van Riel, C. Snijders, and J. Van Wingerden, “The effect on sitting posture of a desk with a 10 inclination for reading and writing,” *Ergonomics*, vol. 34, no. 5, pp. 575–584, 1991.
- [11] K. M. Black, P. McClure, and M. Polansky, “The influence of different sitting positions on cervical and lumbar posture,” *Spine*, vol. 21, no. 1, pp. 65–70, 1996.
- [12] L. J. Haugie, I. M. Fiebert, and K. E. Roach, “Relationship of forward head posture and cervical backward bending to neck pain,” *Journal of Manual & Manipulative Therapy*, vol. 3, no. 3, pp. 91–97, 1995.
- [13] T. Defloor and M. H. Grypdonck, “Sitting posture and prevention of pressure ulcers,” *Applied Nursing Research*, vol. 12, no. 3, pp. 136–142, 1999.
- [14] M. P. Bullock, N. E. Foster, and C. C. Wright, “Shoulder impingement: the effect of sitting posture on shoulder pain and range of motion,” *Manual therapy*, vol. 10, no. 1, pp. 28–37, 2005.
- [15] P. Li, X. Zhang, M. Liu, X. Hu, B. Pang, Z. Yao, H. Jiang, and H. Chen, “A 410-nw efficient qrs processor for mobile ecg monitoring in 0.18- μ m cmos,” *IEEE transactions on biomedical circuits and systems*, vol. 11, no. 6, pp. 1356–1365, 2017.
- [16] X. Tang, Q. Hu, and W. Tang, “A real-time qrs detection system with pr/rt interval and st segment measurements for wearable ecg sensors using parallel delta modulators,” *IEEE transactions on biomedical circuits and systems*, vol. 12, no. 4, pp. 751–761, 2018.
- [17] W. Tang and E. Culurciello, “A low-power high-speed ultra-wideband pulse radio transmission system,” *IEEE transactions on biomedical circuits and systems*, vol. 3, no. 5, pp. 286–292, 2009.
- [18] Q. Hu, X. Tang, and W. Tang, “Integrated asynchronous ultrawideband impulse radio with intrinsic clock and data recovery,” *IEEE Microwave and Wireless Components Letters*, vol. 27, no. 4, pp. 416–418, 2017.

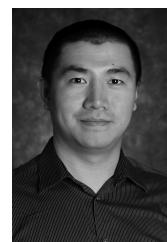
- [19] Q. Hu, C. Yi, J. Kliewer, and W. Tang, "Asynchronous communication for wireless sensors using ultra wideband impulse radio," in *2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS)*. IEEE, 2015, pp. 1–4.
- [20] X. Tang, Q. Hu, and W. Tang, "Delta-sigma encoder for low-power wireless bio-sensors using ultrawideband impulse radio," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 64, no. 7, pp. 747–751, 2016.
- [21] S. C. Mukhopadhyay, "Wearable sensors for human activity monitoring: A review," *IEEE Sensors Journal*, vol. 15, no. 3, pp. 1321–1330, March 2015.
- [22] J. C. T. Mallare, D. F. G. Pineda, G. M. Trinidad, R. D. Serafica, J. B. K. Villanueva, A. R. Dela Cruz, R. R. P. Vicerra, K. K. D. Serrano, and E. A. Roxas, "Sitting posture assessment using computer vision," in *2017 IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, Dec 2017, pp. 1–5.
- [23] S. Bei, Z. Xing, L. Taocheng, and L. Qin, "Sitting posture detection using adaptively fused 3d features," in *2017 IEEE 2nd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, Dec 2017, pp. 1073–1077.
- [24] W. Min, H. Cui, Q. Han, and F. Zou, "A scene recognition and semantic analysis approach to unhealthy sitting posture detection during screen-reading," *Sensors*, vol. 18, no. 9, p. 3119, 2018.
- [25] S. Ma, W. Cho, C. Quan, and S. Lee, "A sitting posture recognition system based on 3 axis accelerometer," in *2016 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, Oct 2016, pp. 1–3.
- [26] W. Y. Wong and M. S. Wong, "Trunk posture monitoring with inertial sensors," *European Spine Journal*, vol. 17, no. 5, pp. 743–753, 2008.
- [27] M. A. Finley and R. Y. Lee, "Effect of sitting posture on 3-dimensional scapular kinematics measured by skin-mounted electromagnetic tracking sensors," *Archives of physical medicine and rehabilitation*, vol. 84, no. 4, pp. 563–568, 2003.
- [28] M. Ramanathan, W. Yau, and E. K. Teoh, "Human action recognition with video data: Research and evaluation challenges," *IEEE Transactions on Human-Machine Systems*, vol. 44, no. 5, pp. 650–663, Oct 2014.
- [29] H. Z. Tan, L. A. Slivovsky, and A. Pentland, "A sensing chair using pressure distribution sensors," *IEEE/ASME Transactions on Mechatronics*, vol. 6, no. 3, pp. 261–268, Sep. 2001.
- [30] J. Meyer, B. Arnrich, J. Schumm, and G. Troster, "Design and modeling of a textile pressure sensor for sitting posture classification," *IEEE Sensors Journal*, vol. 10, no. 8, pp. 1391–1398, Aug 2010.
- [31] S. Ahn, Y. Jeong, D. Kim, and H. Kim, "Development of the non-wearable system with FSR sensors for correction of sitting position," in *2015 Second International Conference on Computing Technology and Information Management (ICCTIM)*, April 2015, pp. 140–143.
- [32] A. H. Namin, K. Leboeuf, R. Muscedere, H. Wu, and M. Ahmadi, "Efficient hardware implementation of the hyperbolic tangent sigmoid function," in *2009 IEEE International Symposium on Circuits and Systems*, May 2009, pp. 2117–2120.
- [33] B. Zamanlooy and M. Mirhassani, "Efficient vlsi implementation of neural networks with hyperbolic tangent activation function," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 1, pp. 39–48, Jan 2014.
- [34] R. Holicky, "Healing the crooked spine: Solving posture problems," Online, February 2014. [Online]. Available: <http://www.newmobility.com/2014/02/scoliosis-solving-posture-problems/>
- [35] "Common posture mistakes and fixes," Online, 2014. [Online]. Available: [Commonposturemistakesandfixeshttps://www.nhs.uk/live-well/exercise/common-posture-mistakes-and-fixes/](https://www.nhs.uk/live-well/exercise/common-posture-mistakes-and-fixes/)
- [36] W. HS, O. JC, and W. SY, "Effects of asymmetric sitting on spinal balance," *J Phys Ther Sci.*, vol. 28, no. 2, pp. 355–359, 2016.
- [37] D. Waltisberg, O. Amft, D. P. Brunner, and G. Tröster, "Detecting disordered breathing and limb movement using in-bed force sensors," *IEEE Journal of Biomedical and Health Informatics*, vol. 21, no. 4, pp. 930–938, July 2017.
- [38] N. B. Gaikwad, V. Tiwari, A. Keskar, and N. Shivaprakash, "Efficient fpga implementation of multilayer perceptron for real-time human activity classification," *IEEE Access*, vol. 7, pp. 26 696–26 706, 2019.



Qisong Hu (S'17) received the Bachelor of Engineering degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2013. He is currently working toward the Ph.D. degree in Klipsch School of Electrical and Computer Engineering, New Mexico State University, Las Cruces, NM, USA. His research interests include low-power RF circuits and system, low-power mixed-signal circuits, and sensing system.



Xiaochen Tang (S'17) received the B.E. degree in electronics science and technology and the M.E. degree in microelectronics and solid-state electronics from the Harbin Institute of Technology, Harbin, China, in 2010 and 2012, respectively. He is currently pursuing the Ph.D. degree in electrical engineering with New Mexico State University, Las Cruces, NM, USA. His research interests include low-power biomedical integrated circuits and wearable computing.



Wei Tang (S'06–M'12) received the B.Sc. degree in microelectronics from Peking University, Beijing, China, in 2006, and the Ph.D. degree in electrical engineering from Yale University, New Haven, CT, USA, in 2012. He joined the Klipsch School of Electrical and Computer Engineering, New Mexico State University, Las Cruces, NM, USA, as an Assistant Professor in 2012 and was promoted to an Associate Professor in 2018. He is currently an Associate Professor with the Klipsch School of Electrical and Computer Engineering, New Mexico State University. His research interests include low-power analog/mixed-signal/digital/RF integrated circuits design and implementation, hardware-friendly digital signal processing and image processing algorithms, and biomedical sensors and wearable devices, circuits and systems. He was a recipient of the National Science Foundation Faculty Early Career Award in 2017 and currently holds the Paul W. and Valerie Klipsch Distinguished Professorship of New Mexico State University.