[EST Checkpoint 3] HANDIO: A Wireless Hand Gesture Recognizer Based on Muscle-Tension and Inertial Sensing

Mew
Department of Computer Science
National Tsing Hua University, Hsinchu City, Taiwan
Mew@est

Jun Luan, Ting-Chou Chien, Seungjae Lee, and Pai H. Chou
Center for Embedded Cyber-Physical Systems,
University of California, Irvine, CA 92697-2625 USA
{jluan1, tchien, leesj3, phchou}@uci.edu

***Abstract*—This paper describes a miniature, wearable wireless hand-gesture recognition (HGR) system called HANDIO. It obtains its input from not only traditional inertial sensors but also a muscle tension sensor (MTS). The addition of MTS enables recognition of a much broader range of intuitive hand gestures, particularly those involving the wrist, that would otherwise be difficult to distinguish by traditional inertial-only HGRs. Among MTSs, we choose an optical MTS over the conventional surface electromyography (sEMG) for the small size, low power consumption, wearing comfort, and good detection rate. This novel miniaturized design enables the whole system to be easily patched on the wrist area or integrated into a wearable device such as a wristband or watch without extra wiring. Experimental results show that a total of 8 hand gestures involving the wrist can be recognized with a detection rate over 93%. The average power consumption of the optical sensor is only around 258μW. This versatile system can also be used to detect other joint activities such as the elbow and knee joint.**

***Keywords*—Optical sensors, Accelerometers, Sensor fusion, Wireless sensor networks**

I. Introduction

Hand gesture recognition has drawn increasingly interest for numerous human-computer interface (HCI) applications such as gaming control, aged or disabled care, sports fitness, etc. Some early systems explored technologies such as com- puter vision and capacitive sensor [1]–[3], but their bulkiness has limited their use to controlled environments [4] such as research labs or hospitals. The emergence of low-power, miniature hardware components with sensing, computation, and communication capabilities has made possible a new generation of a wireless devices such as a wrist watch [5], wristband [6], or smartphone, making ubiquitous computing a reality.

A. Sensing Technologies

Most wearable HGRs to date perform inertial sensing (i.e., acceleration and rotation) as the primary modality for acquiring gesture data. One reason is that accelerometers (ACC) and gyroscopes are widely available in small sizes and in low power. However, they are limited to detecting the trajectory of the whole hand or forearm, such as drawing a circle or a square, but fail to differentiate the different wrist activities. Many commonly used hand gestures involve the movement of the wrist. One example is that a hand-up gesture for "stop" generates nearly the same signal as a slight forearm shaking but with smaller acceleration values in all three axes. Other gestures may include a hand swing of directions, making a fist, etc. These gestures are natural to users but can be difficult to detect by an ACC.

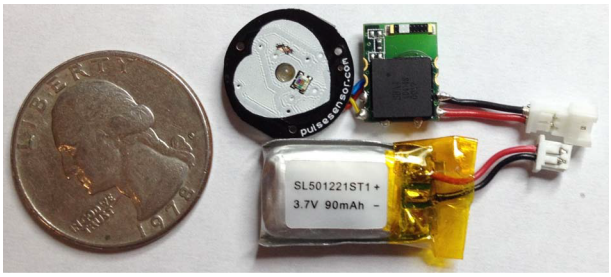


Fig. 1. Proposed hardware system. Top left: optical sensor board, top right: BLE and MPU-9250 board, bottom: battery.

Muscle tension can be sensed by surface electromyography (sEMG) or optical muscle tension sensing (OMTS). The former is most widely used but has not been amenable to wearable HGRs. It requires a minimum of two electrodes, i.e., plus and minus, to extract the differential signal generated by muscle contraction. Many applications require three with an extra ground electrode serving as a reference for the instrumentation amplifier [7]. Commercial electrodes have to be wired and are usually about 30 mm in diameter [8]. The ground electrode is critical to the performance and has to be placed far away from the target muscle to generate a stable reference. As an alternative, we explore optical sensors similar to those used in pulse oximeters for detecting oxygen concentration level (SpO2) [9] and heart rate. It uses an LED to emit light into human tissue and a PD to detect the reflected optical signal. We validate the effectiveness of OMTS in terms of size, power consumption, and wearing comfort.

B. Proposed HANDIO System

In this paper, we present HANDIO, for HAND gesture recognizer based on Inertial and Optical muscle-tension sensing. It consists of both the wearable hardware and the data-fusion algorithm.

1) Hardware: The HANDIO hardware consists of an OMTS, a triaxial accelerometer, and a Bluetooth Low Energy (BLE) system-on-chip (SoC). The optical subsystem can be made very small and lightweight by using an LED-PD combo chip [10]. The overall dimension of the whole prototype system is only 30 mm (L) $\times$ 15 mm (W) $\times$ 8 mm (H) with the largest part being the 90 mAh lithium battery, as shown in Fig. 1, and a production version can be made significantly more compact. This novel miniaturized design enables the whole system to be

easily patched on the wrist area or integrated into a wristband or watch without any extra sensor or wire.

To reduce the system power, the LED is controlled by fast switching pulse width modulation (PWM) signal. Our experimental result also shows that the optical MTS is free from baseline wondering and motion artifact. This feature avoids the heavyweight filtering process and complicated pattern recognition algorithms, thus further saving the system power.

2) Fusion Algorithm: We also propose a fusion algorithm, which includes rapid optical signal induced segmentation, Dynamic Time Warp (DTW) [11], and decision fusion. The fusion of optical and acceleration data increases the recognizable gestures. We tested a total of 8 wrist-involved hand gestures with a detection rate over 93%. The average power consumption of the optical sensor is 258μW, which is only 0.4% of the overall system power.

The contribution of this work lies in the novel hardware sensing platform design and the fusion algorithm. Even though this work focuses only on the HGR, this system is also versatile enough for detecting other joint or muscle activities such as elbows and knees movement.

C. Paper Outline

This paper first provides a background on the OMTS sensor and accelerometer-based hand gesture recognition systems. We then describe the proposed sensing system and sensor-fusion algorithm. Experimental results are analyzed with a discussion of the implications. We also demonstrate the versatility of the system by applying it to other joint movement detection. Finally, we conclude with a summary and future works.

II. BACKGROUND AND RELATED WORKS

Single ACC-based hand gesture recognition has been well studied. One advantage of ACC is its availability on many off-the-shelf portable devices. Examples include TI Chronos Watch [5], Nintendo Wii remote controller, and virtually all smartphones. A digital ACC consumes less than 1 mW in active mode. Acceleration data can be easily processed locally or transferred through the wireless interface. On the software side, simple filters such as low-pass filter (LPF) or median filter [12] are usually used to smooth the signal before using either feature-based or template-based detection algorithms to detect a gesture. Feature-based methods use machine-learning techniques such as Hidden Markov Model (HMM) or Support Vector Machine (SVM) [3] to classify the signal according to the extracted features. High accuracy is reported [13] but a large training set is required to ensure high detection rate. In template recognition, a widely used technique is dynamic time warping (DTW). It can get started with only one template for each gesture [14]. The signal is usually divided into fixedlength segments, and dynamic programming is used to find

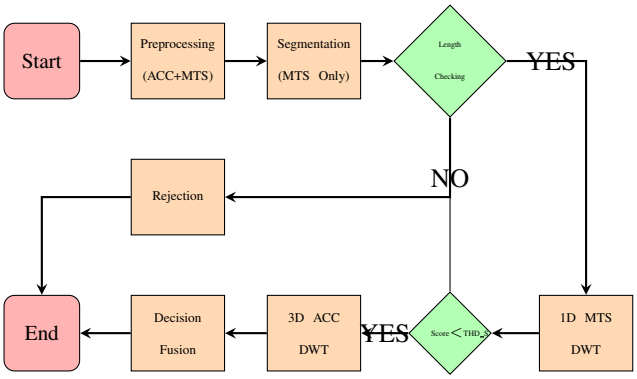


Fig. 2. Detection Algorithm

the best matching subsequence with the preset templates. The complexity of DTW is O(pST) with S being the segment size, T the template size, and p the number of preset templates. It can be easily implemented on a mobile platform such as a smartphone and many modern microcontroller units (MCU).

To overcome the limitations of single ACC-based systems inability to detect wrist movements, multiple previous works suggest to couple sEMG with ACC. In [15], [16], HMM, Kmeans clustering, and decision fusion are used to fuse the multi-channel sEMG signal with ACC. A set of more than 18 gestures can be detected with a detection rate of over 90%. However, a multi-channel sEMG needs to be wired, and electrodes are big in size and can be uncomfortable to wear. Most applications require an extra troublesome step of skin preparation [8], which generally includes hair removal, cleaning the skin, and applying conductive gel. The gel may cause skin irritation and also introduces instability as the gel dries over the time, thereby causing degradation of the signal quality and baseline wondering. The closest one to our system is [17], which uses a single-channel sEMG with three electrodes 30 mm in diameter. The muscle contraction signal from sEMG is used as on-off control and only five gestures can be detected. It is made wearable with a much smaller size than the other two but still an order of magnitude larger than our proposed system and cannot solve the electrode problem. Therefore, ACC-sEMG-based systems are only suitable for a controlled environment.

New sensing technologies, especially low-power wearable sensors, provide potential solutions to this problem. Optical sensor-based MTS is very promising in replacing sEMG in this application due its size and wearing comfort. A wellknown application of optical sensor is pulse oximetry, which is acquired from an LED-PD pair. The LED emits light into human tissue while the PD measures the transmittance or reflectance of light. When used for muscle tension, only the reflectance can be measured. It is a fairly complicated process since the muscle movement has a triple effect: the muscle fiber contraction changes both the light absorbance and the reflected light path, while the blood volume in the muscle also affects the reflected light. As a result, muscle movements manifest as changes to the signal frequencies and amplitude in the PD output. It is shown in [18] that optical sensors can differentiate between isometric and isotonic contraction and in work [19], [20], similar optical sensors are used to monitor the upper limb movement for controlling prosthetic limbs. In all these works, the optical signal is only used for on-off control. Our study shows that far more details are provided even in a singlechannel MTS. We will show how to utilize the MTS signal in template matching.

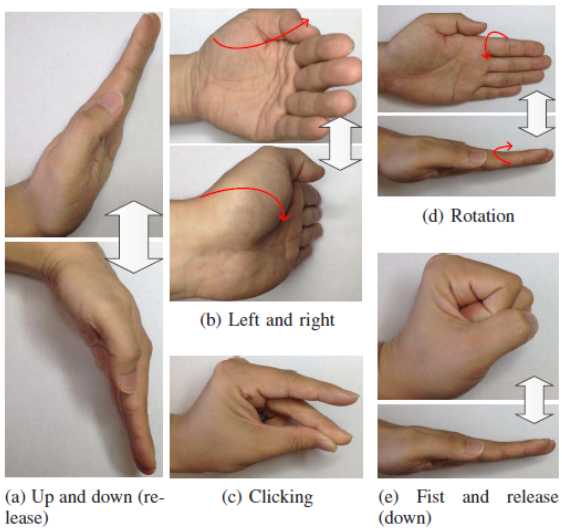


Fig. 3. Gestures set. (No distinction between the down gesture from the up position and the release gesture from the fist position.)

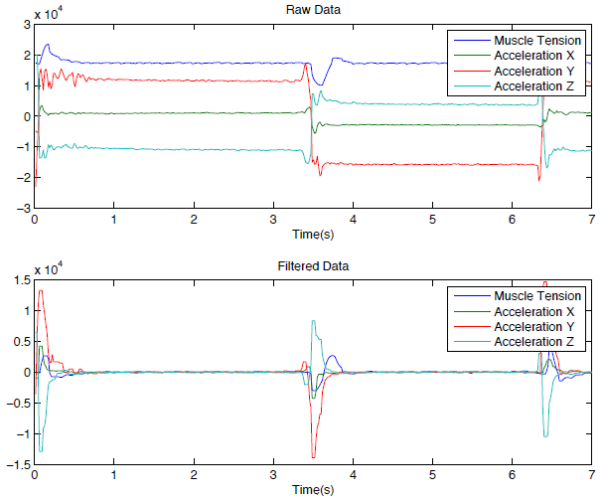


Fig. 4. Filtered Signal

III. TECHNICAL APPROACH

We propose a recognition algorithm based on DTW and decision fusion. The flowchart of the algorithm is shown in Fig. 2.

A. Gesture Set

Fig. 3 shows the tested gestures. What distinguishes them from those that can be handled by previous ACC-based systems [4], [13], [14] is their use of significant wrist movements, such as hand up, clicking, and making a fist.

B. Preprocessing and Segmentation

In the preprocessing stage, a median filter is used for both MTS and ACC to remove the high-frequency noise. A high-pass filter (HPF) is then used to remove the gravity factor in all three axes of ACC. After the filtering, only the MTS signal is normalized to reduce the influence of signal amplitude variation due to different speeds of performing the same gesture. The raw and the filtered signals are shown in Fig. 4.

Segmentation is based solely on the amplitude of the MTS signal. A double-thresholding algorithm, as shown in Algorithm 1, runs point by point to divide the signal into segments. Previous single ACC-based systems [13], [14] usually use a fixed-size sliding window with overlap to segment the signal. The signal content within the window has to be checked constantly to see if any valid gestures exist. Longer delay will be caused by the fixed window size. By setting the starting point and ending point in the signal of interest, our MTS-based segmentation improves the system response time while naturally rejecting some segments based on their length. Any signal that is too short or too long will be automatically discarded and will not go to the computationally complex dynamic programming.

The algorithm keeps counting the samples across or below the corresponding amplitude threshold. Based on a series of counting conditions, the algorithm resets the counter or outputs the starting and ending points of the given segment. Fig. 5 shows a segmented MTS signal with its starting and ending points. This algorithm requires that the gestures be done separately with enough resting interval. Otherwise, several segments may be counted as one and may be discarded due to oversize.

Input: current signal sample: $current\_sample$, current sample position: $current\_position$
Output: segment starting and ending points: idstart_point, end_point or NONE
if $current\_sample$ <*LO_THD* **and** $start\_flag == FALSE$ **then**
  if $jump\_flag == TRUE$ **then**
    $start\_point \leftarrow current\_position$
  else
    $reset\_cont \leftarrow reset\_cont + 1$
    if $reset\_cont$ >*RESET_COUNT* **then**
      RESET()
    end
  end
else if $current\_sample$ ≥*HI_THD* **and** $start\_flag == FALSE$ **then**
  $reset\_count \leftarrow 0$
  $jump\_flag \leftarrow$ TRUE
  $signal\_cont \leftarrow signal\_cont + 1$
  if $signal\_cont$ >*SIGNAL_COUNT* **then**
    $start\_flag \leftarrow$ TRUE
  end
else if $current\_sample$ <*LO_THD* **and** $start\_flag == TRUE$ **then**
  if $jump\_flag == TRUE$ **then**
    $end\_point \leftarrow current\_position$
    $jump\_flag \leftarrow$ FALSE
  end
  $ending\_cont \leftarrow ending\_cont + 1$
  if $ending\_cont$ >*ENDING_COUNT* **then**
    return $start\_point, end\_point$
  end
else if $current\_sample$ ≥*HI_THD* **and** $start\_flag == TRUE$ **then**
  if $jump\_flag == FALSE$ **then**
    $ending\_cont \leftarrow 0$
    $end\_point \leftarrow current\_position$
  end
end
return NONE;

Algorithm 1: MTS segmentation algorithm

C. Dynamic Time Warping Algorithm

As shown in Fig. 2, DTW is the core algorithm of detection. It is performed to both MTS and ACC signals. After preprocessing and segmentation, MTS signal is first checked with all stored templates using one-dimensional DTW. Only those templates with sufficiently high scores can be sent to the next step. If the segmented signal does not match any pattern over the threshold, then it will be discarded. We find this rejection step to be able to effectively reduce the number of false-positive detections due to casual, unintended movement. The basic DTW algorithm is shown in Algorithm 2.

DTW finds a match between two time sequences by dynamic programming. The idea is to find the shortest path in the cost matrix as shown in Fig. 6, where each cell represents the similarity score between the two corresponding subsequences. The distance between two MTS samples is just the absolute value,

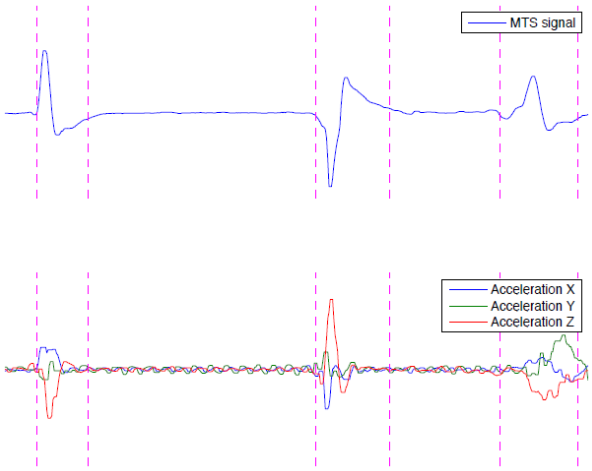


Fig. 5. Signal Segmentation

Input: signal sequence A: seq_a of length M, signal sequence B:seq_b of length N
Output: similarity score: $matching\_score$
initialization
  cost matrix $cost\_matrix$
  slope matrices $slope\_x$ and $slope\_y$
end
for $i = 2$ *to* $M + 1$ **do**
  for $j = 2$ *to* $N + 1$ **do**
    if $ABS(i - j) \leq$ *WINDOW_SIZE* **then**

$$neighbors \leftarrow \begin{pmatrix} cost\_matrix(i-1,j) \\ cost\_matrix(i,j-1) \\ cost\_matrix(i-1,j-1) \end{pmatrix};$$

      if $slope\_x(i-1,j)$ == *SLOPE_THRESHOLD* **then**
        $neighbors(1) \leftarrow$ INF
      end
      if $slope\_y(i,j-1)$ == *SLOPE_THRESHOLD* **then**
        $neighbors(2) \leftarrow$ INF
      end
      $cost\_matrix(i,j) \leftarrow$
      MIN$(neighbors)$ + DISTANCE$(seq_a(i-1), seq_b(j-1))$
      SLOPE_UPDATE()
    end
  end
end
return $cost\_matrix(M+1, N+1)$

Algorithm 2: Dynamic Time Warping

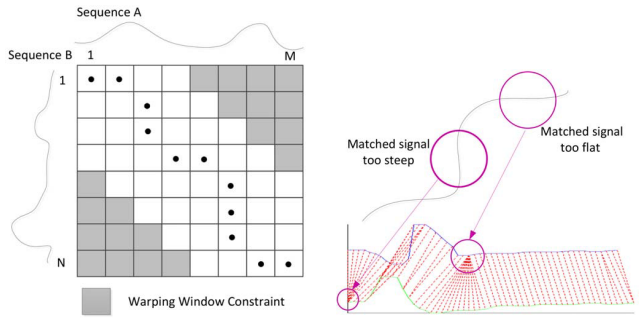
while for 3D acceleration samples, the distance is calculated based on Euclidean distance as in shown in Eq. (1). The matched signals in all four dimensions are shown in Fig. 7.

$$\sqrt{(ACCX_1 - ACCX_2)^2 + (ACCY_1 - ACCY_2)^2 + (ACCZ_1 - ACCZ_2)^2} \quad (1)$$

Two constraints are implemented here: warping window constraint and slope constraint [11]. Warping window constraint (line 1) prevents one point from matching any point too far away by setting global forbidden area, shown as the gray cells in Fig. 6, to eliminate the path far off the diagonal. The local slope constraint (line 2) avoids the alignment paths that are too steep or too shallow as shown in Fig. 6. The slope value along the optimal path from the starting point to each cell is stored in separate slope matrices. When exceeding the threshold, the path will be discarded along the way. The slope matrix is updated on line 3. If the two sequences are too different from each other, then the template sequence will be automatically rejected before DTW.

D. Decision Fusion

The final matching score of the segmented signal to each template is given by Formula (2). It is a weighted average of



(a) Cost matrix and warping window constraint (b) Slope constraint

Fig. 6. Dynamic Time Warping with Constraints

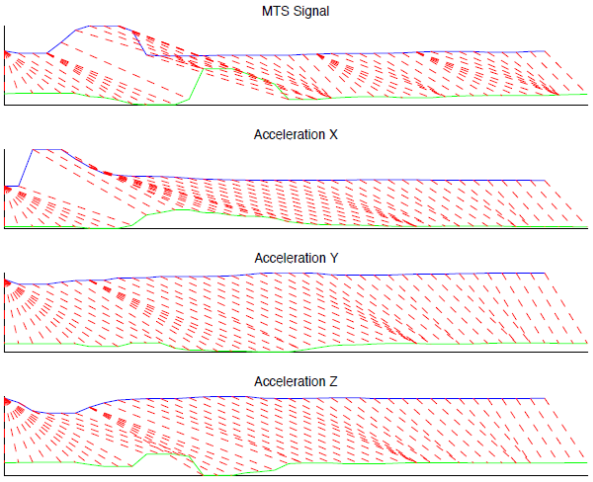


Fig. 7. Signal matched with Dynamic Time Warping

both MTS and ACC scores with the weight γ = 0.66. The template with the best score is selected as the output gesture.

$$score\_final = \gamma \cdot score\_ACC + (1 - \gamma) \cdot score\_MTS \quad (2)$$

IV. IMPLEMENTATION

Fig. 8 shows the block diagram of the proposed HGR system. It consists of a light emitter with a current source, a photo diode (PD) with an OpAMP, an inertial sensor, and a Bluetooth Low Energy (BLE) system-on-chip (SoC). To quickly validate our

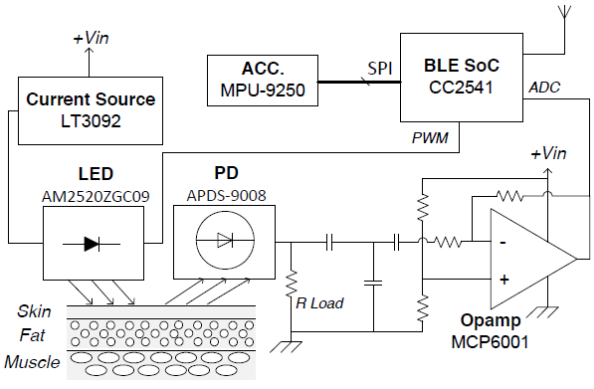


Fig. 8. Proposed hybrid system

Fig. 9. Sensing location on the wrist area.

design, we implement our sensor board using a commercially available pulse-sensor design [21] but customize the board to our design. For the accelerometer and BLE board, we designed a prototype as shown in Fig. 1. Both boards are connected using jumpers. The overall dimension of the system including the battery is 30 mm (L) $\times$ 15 mm (W) $\times$ 8 mm (H). The optical sensor board for prototyping purpose may appear bulky but can be shrunk easily by fabricating a flexible PCB, as the optical sensors themselves are small. Even without optimization, the whole system is lightweight and can be easily adhered onto the wrist area or further integrated into a wearable sensing platform such as a wrist band or a watch. The sensing location of the optical sensor is shown in Fig. 9.

A. Optical Subsystem

In the optical subsystem, the LED is driven by a PWMcontrolled current source [22] for brightness control. The LED current is limited to 500μA to prevent the on-surge current during fast switching. The color of the LED is green with a peak wavelength of 515 nm, which matches the peak wavelength of the PD response curve. This way, the PD response is maximized, thus further reducing the system power consumption. The green LED color is also proved to be more robust to motion artifact than other colors [23]. The LED on-time is set to approximately 200μs in every 16 ms by PWM control as shown in Fig. 11. In response to the LED, the PD outputs a current that flows through the load resistor to generates the voltage signal. The low-pass filter removes noise and motion artifact in the voltage signal, and the OpAMP amplifies it for sampling by the on-chip ADC on the CC2541.

B. Accelerometer

We use MPU-9250 [24] as the ACC controlled by the SPI interface of CC2541. MPU-9250 is actually a 9 degree-off-reedom (9-DoF) inertial sensor consisting of a triaxial ACC, gyroscope, and compass. We use only the ACC for this test while the gyro and compass are reserved for future use. The typical operating current of the gyroscope is 3.2 mA and only around 450μA for the ACC. The compass takes around 280μA and can be helpful to gesture recognition but it is susceptible to the disturbance of other magnetic materials in the vicinity [25].

C. Bluetooth Low Energy SoC

CC2541 from TI is an integrated MCU and radio frequency transceiver SoC that runs TIs BLE protocol stack [26] and the application code on a single chip. Industrystandard profiles such as GATT (for attribute access) and GAP (for connection and discovery) are already implemented. The HGR system works as a slave server while the mobile device (smartphone) works as a master client. The BLE stack runs on a low-overhead, non-emptive, event-driven task executive called OSAL, which implements tasks as call-back functions.

The CC2541 MCU performs MTS sampling, ACC reading, and data transmission over BLE. The MCU remains in low-power mode normally. On each timer rollover, the MCU wakes up, reads its ADC for the MTS signal, and reads the ACC data from SPI. After all the data is acquired and stored into a local buffer, the MCU queues a sending event in the scheduler and go to low-power mode 3 (LPM3) again, from which the MCU can be waken by interrupts only. The sending event will cause the queued data to be sent the mobile client. This process repeats at a rate of 62.5 Hz.

V. EVALUATION

A. Detection Accuracy

We tested our HGR system on three subjects, two male and one female. One important aspect of HGR is the noise rejection. In real situations, gestures coexist with random activities. These random activity should be rejected instead of being recognized as a valid gesture. Otherwise, the system will become hard to use due to many false positives. We collected 40 non-gesture noise patterns, including random hand or wrist shake with both short and long wrist activities from each subject. Another 40 samples of each gesture from each person are also record. The total collection time is over 1 hour each. The system is implemented on a laptop computer to repeatedly study the data. The detection rate is defined in Eq. (3).

$$Detection\_Rate = 1 - \frac{True\_Negatives + False\_Positives}{Total\_Gestures} \quad (3)$$

As shown in Table I, the detection rate is over 93% with the highest over 95%. We also implemented a single ACC-based HGR with the sliding window algorithm. A threshold is set to reject the noise based on the DTW score. The result clearly indicates that our system is superior to single ACC-based system in terms of noise rejection and detection accuracy.

The detailed result for each gesture is shown in Table II. The detection rate is just the true positive rate, since no noise gesture is involved. Even though for some gestures, ACConly results in slightly higher detection rate than the fusion method does, the fusion method helps improve the accuracy of the majority at a very small cost. The overall result shows the advantage of decision fusion over the single ACC-based method. The same gesture is usually done multiple times with different stopping points or angles to record multiple templates for each gesture. This multi-template setting helps eliminate ambiguity. Fig. 10 shows the influence of the weight γ in Eq. (2). The detection rates for all three people go beyond 93% when γ = 66%.

B. Power Consumption

The power consumption of the system is measured using a current probe. The current values are integrated over a period of 5 minutes to calculate the average power consumption. Fig. 11 shows the periodic current signal of the optical sensor board. Even though the LED takes around 500μA when turned on, the duty of PWM is around 1.25%. The optical sensor consumes around 258μW on average. The majority of the power is consumed by BLE communication sending and receiving data. The average power consumption of the CC2541 and MPU-9250 is around 57.5 mW. Thus, the power consumption increased by the MTS is only around 0.4% of the system power. The current

TABLE I
TOTAL DETECTION RATE

Det. Alg.	Sub.	TN	FP	Dec. Rate
Acc + MTS	Male 1	18	4	93.89%
	Male 2	11	5	95.56%
	Female	18	3	94.17%
Acc	Male	25	17	88.33%
	Male 2	14	9	93.61%
	Female	31	7	89.44%

TABLE II
DETECTION RATE OF EACH GESTURE

Sub.	Gesture	Fusion Rate	Acc. Rate	Temp.	Length
Male 1 γ = 0.66	Up	92.5%	87.5%	4	240ms-928ms
	Dn/Rel.	90.0%	60.0%	6	304ms-1312ms
	Rot.←	92.5%	95.0%	2	304ms-1120ms
	Rot.→	90.0%	90.0%	2	288ms-1200ms
	Fist	97.5%	67.5%	2	448ms-1120ms
	Clicking	97.5%	32.5%	4	336ms-1040ms
	Left	95.0%	92.5%	1	480ms-1200ms
	Right	97.5%	100.0%	1	288ms-768ms
	Tol.	94.4%	78.1%	22	240ms-1312ms
Male 2 γ = 0.82	Up	97.5%	100.0%	5	512ms-1408ms
	Dn/Rel.	92.5%	95.0%	5	480ms-1264ms
	Rot.←	97.5%	97.5%	4	592ms-2880ms
	Rot.→	100.0%	100.0%	4	560ms-2384ms
	Fist	97.5%	67.5%	1	656ms-1376ms
	Clicking	92.5%	100.0%	3	1072ms-1488ms
	Left	100.0%	50.0%	2	784ms-1040ms
	Right	95.0%	45.0%	2	672ms-1296ms
	Tol.	96.56%	81.9%	26	480ms-2880ms
Female γ = 0.75	Up	97.5%	100.0%	4	784ms-1392ms
	Dn/Rel.	97.5%	100.0%	5	624ms-1424ms
	Rot.←	92.5%	82.5%	2	1024ms-1840ms
	Rot.→	92.5%	90.0%	2	1264ms-2544ms
	Fist	92.5%	80.0%	2	448ms-1408ms
	Clicking	90.0%	80.0%	3	688ms-2480ms
	Left	95.0%	77.5%	2	912ms-1488ms
	Right	90.0%	80.0%	1	912ms-1264ms
	Tol.	94.4%	86.3%	21	448ms-2544ms

system can last for more than 6 hours without recharging. Further optimization for future work includes lowering the SoC's Tx power, adjusting the sampling rate, and enabling threshold detection in the accelerometer.

C. Time Delay

The recognition system has also been ported to an iPhone4s. The measured computation delay is around 50 ms in the worst case with an average of 14 ms. Even with an expansion of around 30 gestures, the delay is under 100 ms.

D. Other Applications

We also tested the system on elbow and knee joints. The sensing locations are shown in Fig. 12. Signal samples are shown in Fig. 13. The degree of freedom of movement is generally less than the wrist joint. The MTS signal can be used to detect the muscle movement around the joint while the ACC data provides the detailed movement, such as moving direction, timing, etc. The fusion of both MTS and ACC can be synergistic in classifying the movements.

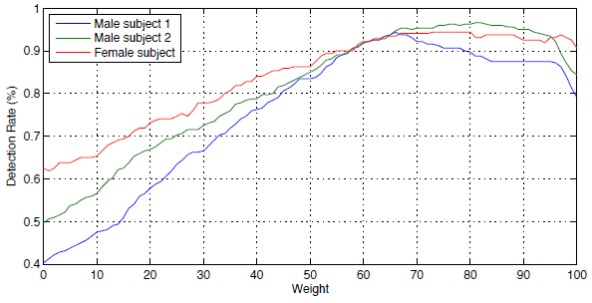
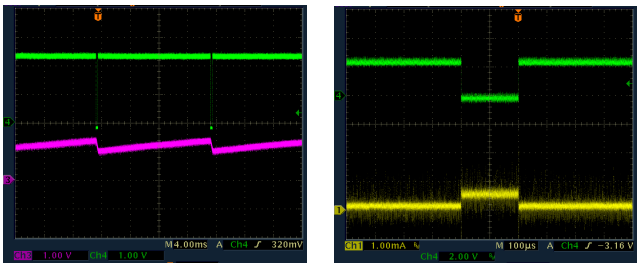
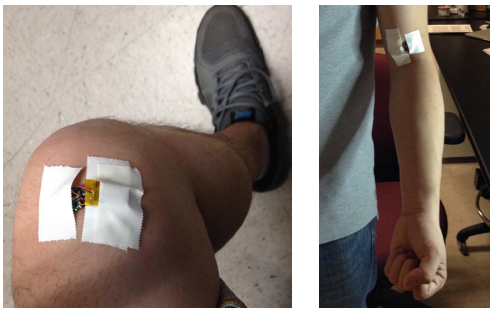


Fig. 10. Decision Weight



(a) PWM and PD output. Green:PWM, pink: PD output (b) Current consumption. Green:PWM, yellow: optical sensor board current signal

Fig. 11. LED control and current consumption.



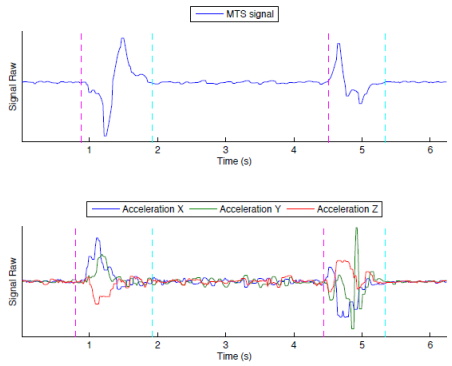
(a) Knee (b) Elbow

Fig. 12. Sensing locations on the knee and elbow.

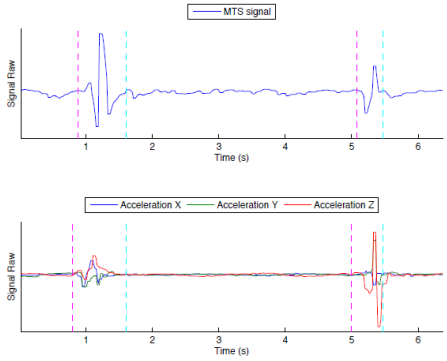
VI. CONCLUSIONS

In this paper, we have presented HANDIO, a wireless hybrid hand gesture recognition system of OMTS and ACC. Compared with previous works, the proposed system strikes the most balance among gesture variety, wearability, and power consumption. The OMTS enables the detection of wrist movements, which greatly increases the total number of detectable gestures of the current single ACC-based HSG.

In addition to the hardware, we also describe a DTW and decision fusion-based detection algorithm, which shows a high detection rate. By fusing the MTS and ACC signals, the system response time is also significantly improved. We also show that the same system can be used in many other body-area sensor applications. Altogether we are building the next-generation, highly wearable sensor networks for accurate human movement monitoring and tracking.



(a) Knee bending and recovering while sitting



(b) Elbow bending and recovering

Fig. 13. Signals from the knee and elbow joint.

ACKNOWLEDGMENTS

This work was sponsored by the National Institute of Health (NIH) STTR Phase II Grant 2R42HL112435-03. The content is solely the responsibility of the authors and does not necessarily represent the official views of the sponsors.

REFERENCES

[1] D. Ashbrook T. Starner, J. Auxier and M. Gandy. The gesture pendant: a self-illuminating, wearable, infrared computer vision system for home automation control and medical monitoring. *Wearable Computers, The Fourth International Symposium*, pages 87–94, Oct 2000.

[2] J. Rekimoto. Gesturewrist and gesturepad: unobtrusive wearable interaction devices. In *Wearable Computers, 2001. Proceedings. Fifth International Symposium*, pages 21–27, 2001.

[3] S. Mitra and T. Acharya. Gesture recognition: A survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions*, 37(3):311–324, May 2007.

[4] J. Mantyjarvi S. Kallio G. Savino L. Jozzo J. Kela, P. Korpipaa and D. Marca. accelerometer-based gesture control for a design environment. *Personal Ubiquitous Comput.*, 10(5):285–299, Jul. 2006. [Online]. Available: http://dx.doi.org/10.1007/s00779-005-0033-8.

[5] Ti chronos watch. http://www.ti.com/tool/ez430-chronos.

[6] Wrist based hrm reference design. http://www.ti.com/lit/ug/tidu125/tidu125.pdf.

[7] L. Couts. A designers guide to instrumentation amplifiers. chap. 1, 2006.

[8] P. Konrad. The abc of emg. *A practical introduction to kinesiological electromyography*, 1, 2005.

[9] R. Sankar T. Rusch and J. Scharf. Signal processing methods for pulse oximetry. *Computers in Biology and Medicine*, 26(2):143–159, 1996. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0010482595000496.

[10] Cobp photo reflector with red and ir led. http://www.njr.com/semicon/products/NJL5501R.html.

[11] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions*, 26(1):43–49, Feb 1978.

[12] M. Mathie N. Lovell D. Karantonis, M. Narayanan and B. Celler. Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring. *Information Technology in Biomedicine, IEEE Transactions*, 10(1):156–167, Jan 2006.

[13] W. Gao D. Mace and A. K. Coskun. Improving accuracy and practicality of accelerometer-based hand gesture recognition. In *2nd Workshop on Interacting with Smart Objects, in conjunction with the ACM International Conference on Intelligent User Interfaces (IUI). Citeseer*, 2013.

[14] J. Wickramasuriya J. Liu, L. Zhong and V. Vasudevan. uwave: Accelerometer-based personalized gesture recognition and its applications. *Pervasive and Mobile Computing*, 5(6):657–675, 2009, perCom 2009. [Online]. Available:http://www.sciencedirect.com/science/article/pii/S1574119209000674.

[15] Y. Li V. Lantz K. Wang X. Zhang, X. Chen and J. Yang. A framework for hand gesture recognition based on accelerometer and emg sensors. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions*, 41(6):1064–1076, Nov 2011.

[16] W.-h. Wang J.-h. Yang V. Lantz X. Zhang, X. Chen and K. q. Wang. Hand gesture recognition and virtual game control based on 3d accelerometer and emg sensors. *Proceedings of the 14th International Conference on Intelligent User Interfaces, ser. IUI 09. New York, NY, USA: ACM*, pages 401–406, 2009. [Online]. Available:http://doi.acm.org/10.1145/1502650.1502708.

[17] J. Cannan and H. Hu. A multi-sensor armband based on muscle and motion measurements. In *Robotics and Biomimetics (ROBIO), 2012 IEEE International Conference*, pages 1098–1103, Dec 2012.

[18] A. Chianura and M. E. Giardini. An electrooptical muscle contraction sensor. *Medical & biological engineering & computing*, 48(7):731–734, 2010.

[19] H. Han and J. Kim. Novel muscle activation sensors for estimating of upper limb motion intention. In *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, page 37673770, Sept 2009.

[20] O. Kulyk E. M. Bowman A. K. Bansal, S. Hou and I. D. W. Samuel. Wearable organic optoelectronic sensors for medicine. *Advanced Materials*, 2014. [Online]. Available: http://dx.doi.org/10.1002/adma.201403560.

[21] Pulse sensor. http://pulsesensor.com.

[22] Programmable current source lt3092. http://www.linear.com/product/LT3092.

[23] K.-I. Yamakoshi P. Rolfe S. Tanaka J. Lee, K. Matsumura and T. Yamakoshi. Comparison between red, green and blue light reflection photoplethysmography for heart rate monitoring during motion. In *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE*, pages 1724–1727, July 2013.

[24] Mpu9250 motion sensor. http://www.invensense.com/mems/gyro/mpu9250.html.

[25] C. T. Baten D. Roetenberg, H. J. Luinge and P. H. Veltink. Compensation of magnetic disturbances improves inertial and magnetic sensing of human body segment orientation. *Neural Systems and Rehabilitation Engineering, IEEE Transactions*, 13(3):395–405, 2005.

[26] Bluetooth low energy software stack. http://www.ti.com/tool/ble-stack.