

Received February 26, 2021, accepted March 15, 2021, date of publication March 18, 2021, date of current version March 29, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3067030

# Surface EMG vs. High-Density EMG: Tradeoff Between Performance and Usability for Head Orientation Prediction in VR Application

TOMMY SUGIARTO<sup>1,2,3</sup>, CHUN-LUNG HSU<sup>1</sup>, (Member, IEEE), CHI-TIEN SUN<sup>1</sup>, WEI-CHUN HSU<sup>2,3</sup>, SHU-HAO YE<sup>1</sup>, AND KUAN-TING LU<sup>1</sup>

<sup>1</sup>Information and Communication Research Laboratory, Industrial Technology Research Institute, Hsinchu 310, Taiwan

<sup>2</sup>Graduate Institute of Applied Science and Technology, National Taiwan University of Science and Technology, Taipei 106, Taiwan

<sup>3</sup>Graduate Institute of Biomedical Engineering, National Taiwan University of Science and Technology, Taipei 106, Taiwan

Corresponding author: Tommy Sugiarto (tommysugiarto@itri.org.tw)

This work was supported by the Industrial Technology Research Institute under Grant J301AR2100 and Grant L351AH1110.

**ABSTRACT** Head orientation prediction is one of the solutions to reduce end-to-end latency on Virtual Reality (VR) systems and is important since it can alleviate negative effects like motion sickness. This study compared head orientation prediction models from two different electromyography (EMG) systems: surface EMG (sEMG) and High-Density EMG (HD-EMG). The deep learning method was used to train the prediction model, and the results showed that the model with input from the pre-processed sEMG + IMU sensor outperformed the model with input from the HD-EMG + IMU sensor. However, the decreasing performance from HD-EMG was compensated by its comfort and the ease of use of its electrode. This tradeoff between performance and usability with sEMG compared to HD-EMG should be a consideration for users who want to choose between performance and ease of use for head orientation prediction purposes. Comparison with state-of-the-art head prediction methods proved that the sEMG-based model offers better performance in predictions when users change their head directions, which was quantified by calculating the dt peaks. In other words, our sEMG-based prediction model is suitable for VR applications, which require the user to perform high-intensity or abrupt movements, such as in FPS games or exercise/sports games.

**INDEX TERMS** Deep neural network, head orientation prediction, high-density electromyography, low-latency virtual reality, surface electromyography.

## I. INTRODUCTION

Virtual Reality (VR) is evolving rapidly with many applications in various fields, including medicine, navigation, entertainment, training, and education. VR systems with Head Mounted Displays (HMDs) utilize artificial sensory stimulation to induce the targeted behavior in an organism, while the organism has little to no awareness of the interference; however, this sensory stimulation sometimes fails to create a perceptual illusion because of the latency between the user's actions and the displayed image.

The time interval or time delay between a user's physical movement and the resulting update of a new frame on the display is referred as motion-to-photon (MTP) latency [1]. This MTP latency can cause several negative effects for the

user, such as motion sickness marked by symptoms like dizziness, nausea, headaches, and general discomfort [2]. For human beings, latency itself is noticeable in the range of 8–20 ms [3]. However, the end-to-end latency on a VR system can reach more than 30 ms, depending on the hardware and software configuration. A well-designed VR system consisting of a tracking sensor with a 1000 Hz sampling rate can add a 1–5 ms delay for sampling and digitization, while additional filtering and necessary calculations can add another 3–10 ms. Rendering and displaying the frame add another extra 6–15 ms. However, this last part of the delay can be reduced if the system uses a higher refresh rate display that can reach up to 120 Hz [4]. These accumulated delays can be reduced but never eliminated since they come from the hardware and software requirements. Moreover, recent literature from 2019–2020 showed that current VR-HMD still possesses an MTP latency varying between 43 and 85 ms [5]–[9].

The associate editor coordinating the review of this manuscript and approving it for publication was Donato Impedovo<sup>1</sup>.

Using an Acer device, R. Gruen *et al.* (2020) tested the latency on four different VR devices: Oculus Rift S, Valve Index, Oculus Quest, and Prim. The first of these devices was a standalone VR headset, while the others were PC-based VR headsets [8]. The authors' hardware-instrumentation-based measurements showed that the Prism device had the lowest latency, with  $54 \pm 1.9$  ms. Meanwhile, the Valve Index had the largest latency, with  $94 \pm 2.1$  ms [8]. Furthermore, a study from Stauffert *et al.* measured latency on an HTC Vive, and the result was  $54.51 \pm 8.6$  ms [6], while the older version of Oculus VR, Oculus DK2, showed latency as high as  $84 \pm 6.3$  ms, as measured by Feldstein and Ellis [9]. Thus, the MTP latency on VR system is still around 43-85 ms, even with the latest VR-HMDs like Oculus Rift S and Oculus Quest.

Furthermore, one of the solutions to reduce latency is to apply head movement predictions to the VR system. This head movement prediction system can be developed using various types of sensors and algorithms, such as using different types of predictive filters (Kalman Filter, Particle Filters) with inertial sensors [10]–[14], or using another approach utilizing biomedical signals such as Electromyography (EMG) [4], [11].

The idea behind using the EMG signal as a predictor for head movement is based on the phenomena called Electromechanical Delay (EMD). EMD is the delay between generated action potential and muscle contractions or movements. The value of EMD varies between muscle type and location, and, in some cases, can reach up to 100 ms [4], [15]. If we trace the chronological order of movements in humans, first, a signal generated from the brain propagates through nervous signaling to the target site (muscle); then, the action potential propagates through the muscles before the muscle contracts to create the desired movement. This EMD phenomena suggests that the movement from humans can be anticipated by detecting the signal with surface EMG. It was first demonstrated by Y. Barniv and S. Polak that an EMG signal can predict head movements by utilizing features extracted from the surface EMG data [4], [11]. However, this previous study still had some drawbacks, such as the utilization of handcrafted features, which added additional processing times/delays to the system.

Second, the testing data used were limited and did not cover all movement types and speeds represented in real VR applications. Third, the model was developed only for the intra-subject testing method, which means that if another new user employs the model, additional fine-tuning will be needed to adjust the pre-trained model. Another limitation when using surface EMG (sEMG) is the electrode placement, which needs to be precisely located on the muscle belly; otherwise, the data produced will not be good. The electrode placement is also susceptible to the variability between subjects or even between sessions.

This last problem can be solved by using grid electrodes (High-Density EMG). In this method, instead of placing the electrode on the exact position of the muscle, a grid electrode that contains several electrodes

in rows is placed using columns on the area of the muscle.

To address the limitations in the previous study, the present research developed and trained various deep learning models to predict head orientation from sEMG signals on the neck muscles. We also compared the results with the HD-EMG-based prediction model, which we hypothesized could solve the sensor placement problem in the sEMG system. Finally, the possibility of applying this method to predict head orientation in VR systems was also investigated for real-world applications.

## II. RELATED STUDIES

Previous literature on head tracking and prediction for VR was dominated by the usage of various types of filters, including Kalman Filters (KFs), Extended Kalman Filters (EKFs), and Particle Filters (PFs) [10], [12], [13], [16]–[18]. The head tracking sensors used in the VR system were also different from magnetic trackers, which utilize electromagnetic transmitter–receivers to track head orientation with an Inertial Measurement Unit (IMU) sensor that combines three different sensors (an accelerometer, gyroscope, and magnetometer). In 1997, A Kiruluta *et al.* used magnetic trackers with Kalman Filters to predict head movement under smooth and abrupt conditions. This Kalman Filter method was based on a Constant Acceleration (CA) assumption, which assumes that there is no change in angular velocity between two sampling points. The results showed that predictions under abrupt movements are not good, with a mean error up to  $14.84^\circ$ . Meanwhile, the smooth movement condition showed a mean error of  $4.52^\circ$  [12].

Testing different types of movement is also considered an important parameter when developing head movement prediction since some models might only work on specific types of movement like slow movement but perform worse on faster movement. Therefore, most studies used different speeds or movement intensities to prove that their prediction models have better generalization. In 2009, H. Himberg *et al.* tested three different types of movement, benign, moderate, and aggressive, for predicting head movement with a magnetic tracker. The authors proposed using the delta quaternions method on EKF and predicted head movement 50 ms in the future. Their results for moderate movement were promising, with an average error of  $0.31^\circ$ ; with aggressive movement, the average error was  $1.11^\circ$ . However, the aggressive movement defined in this study was not very intense since the error for no prediction under aggressive movement was only  $3.79^\circ$  compared to other studies that used aggressive movements with no-prediction errors up to  $10^\circ$ , which indicates that an abrupt movement [17].

In 2017, A.G Agundez *et al.* also utilized extrapolation combined with various filtering methods to predict head movement. This study used an Oculus Rift DK2 with the tracker sampling rate equal to 75 Hz. The data were collected from 10 users playing a First Person Shooter game on a VR system. The authors also compared several extrapolation

methods (linear and polynomial), but their results showed that linear extrapolation was the best, with yaw and pitch errors of  $0.125^\circ$  and  $0.200^\circ$ , respectively. These results were even better when combined with a smoothing filter (Savitsky–Golay Filter), with the lowest error being only  $0.04^\circ$  for 13 ms in future predictions. This study showed promising results using a simple extrapolation method for predicting a user's future head orientation. However, these studies did not include predictions based on roll movements, since this movement was too small and happened in the roll direction; thus, the extrapolation was not able to handle the movement. This also proved that the extrapolation method was not a good choice for predicting very low intensity or almost no-movement conditions. Another limitation in this study was the prediction time, which was only 13 ms in the future. Thus, for VR systems that have latency greater than 13 ms, this result will not be useful [19].

Moreover, a recent study from X. Hou *et al.* in 2020 utilized Long Short-Term Memory (LSTM)- and Multi-Layer Perceptron (MLP)-based models to predict the 6DoF of a VR user's movement [20]. In this study, the authors studied 20 participants using an HTC Vive with the 6DoF VR apps of Virtual Museum and Virtual Rome. Then, with the multi-axis model from LSTM and MLP, the authors produced position and orientation predictions of the user based on 840,000 sample points. Here, the LSTM model achieved better result for a relatively slower speed and more regular motion, while the MLP model achieved better results for sessions with quicker variations and abrupt changes [20]. Even though this study predicted 6DoF motion among VR users, the data collected from the sample did not explore the worst-case scenarios for when the user performs abrupt movements while playing VR games, focusing instead on scenery-based VR apps. Another limitation in this study was their use of 60 window time data (666 ms) to predict only 11 ms in the future, which is not practical since recent VR-HMDs still have latency between 43 and 85 ms [5]–[9].

Modern VR devices cannot be separated by the development of mobile VR systems that rely on cloud-based volumetric streaming system. This kind of VR device also needs a motion prediction system to overcome latency. Thus, in 2020, S. Gul *et al.* used a Microsoft Holo Lens with an autoregressive model to predict the 6DoF position and orientation of a VR user [21]. This study collected data from five users while freely interacting with static virtual objects. Then, the authors used 32 window data (160 ms) to predict the user's next 20–100 ms of movement in the future. The results showed that if the user's pose linearly changed and there was no changing direction, the prediction would be accurate. Otherwise, the prediction was worse than that under the baseline method [21].

In 2005, Y. Barniv *et al.* were the first to study head movement predictions based on EMG data [11]. The authors used features extracted from 32 EMG electrodes on the neck muscles to predict head movement. The authors continued their study in 2006 with some improvements [4]. However,

as previously described in the introduction section, this study still has some drawbacks, such as the utilization of EMG features and the intra-subject testing method.

To sum up, state-of-the-art VR head orientation prediction still has limitations in predicting high-intensity and abrupt movements when the user moves at a high speed and changes his or her movement direction abruptly. Moreover, how far in advance the system can predict future head orientations is another limitation that needs to be improved since the best result with an error of less than  $1^\circ$  achieved only a 13 ms prediction time [19]. Moreover, head movement prediction with EMG data still has problems with model generalization and electrode placement, as electrodes need to be placed precisely on the muscle belly when using an sEMG sensor.

Based on above literature review and state-of-the-art head orientation prediction studies on VR systems, this research was developed to address the limitations of predictions for high-intensity or abrupt movements by using an sEMG placed on the neck muscle. The second purpose of this study was to compare the results of an HD-EMG-based prediction model with those of the intra- and inter-subject testing method, which we hypothesized could solve the sensor placement problem in the sEMG system. Finally, the possibility of applying this method to predict head orientation in a VR system was also investigated for real world applications.

### III. METHOD AND EQUIPMENT

#### A. SUBJECT AND EXPERIMENTAL PROTOCOL

In total, 31 subjects participated in this study. All subjects were divided into one of two groups: those using a surface EMG (sEMG) and those using HD-EMG. The sEMG group consisted of 20 subjects (Age =  $29.9 \pm 7.7$  years old), while the HD-EMG group consisted of 11 subjects (Age =  $27.5 \pm 3.9$  years old). Since the main purpose of this study was not to measure motion sickness itself, the current subject recruitment plan to recruit only from the young adult population was considered sufficient for this study. Moreover, based on the results of D. Saredakis *et al.*, age was not considered to be a main contributor to VR motion sickness, while other factors such as VR content, visual stimulation, and exposure time were considered noteworthy [22]. All subjects were informed about this study's experimental procedure and signed their informed consent before the experiment.

For the sEMG experiment, 3 pairs of wireless sEMG sensors from Delsys Trigno (Delsys, MA, USA) were placed on the subject's neck muscles. A total of 6 sEMG sensors with a sampling rate of 2000 Hz were placed on the left and right Sternocleidomastoid, Splenius Capitis, and Trapezius muscles. We decided to include only 3 pairs of muscles since the muscles on the neck are relatively small, and with the current wireless sensor size, it was only possible to place 3 pairs of sensors without sacrificing the signal quality due to poor sensor placement. All the sensors were then secured by medical-grade tape to prevent any displacement during the experiment.

Meanwhile, for the HD-EMG experiment, 2 arrays of sensors were placed on the left and right back-sides of the neck muscles. Each of the array sensors consisted of  $4 \times 8$  electrodes. The HD-EMG is an EMG sensor that uses 2D grid electrodes featuring many closely spaced electrodes (3-6 mm center to center). This device generally measures activity distributed over an area under the grid electrode [23]. In this study, a wireless 64 channel HD-EMG system from Sessantaquattro (OT Bioelettronica, Torino, Italy) was used with a two  $4 \times 8$  electrodes configuration. The HD-EMG system was also equipped with 2 auxiliary input channels that were used as input for the analog trigger to enable synchronization with the IMU sensor.

To monitor head movement, 1 IMU sensor from Delsys Trigno IM was used and placed on the subject's forehead. The IMU sensor consisted of a triaxial accelerometer, a gyroscope, and a magnetometer. The IMU sensor data were resampled to 1000 Hz, and the range for each sensor was  $\pm 6$  g for acceleration,  $\pm 2000^\circ/\text{s}$  for angular velocity, and  $\pm 4900 \mu\text{T}$  for the magnetometer. For the sEMG experiment, since the IMU and sEMG came from the same system, both of them were directly synchronized. However, for the HD-EMG, since the IMU sensor was different and separated with HD-EMG, the synchronization process was done with a separate trigger box that connected the HD-EMG system and IMU sensor system.

After the sensor was placed on the designated muscle, both of the experimental groups followed same tasks for each subject. Every subject was instructed to perform 5 types of head movements:

- 1) Continuous head rotation: the subjects rotated their heads left and right continuously at their preferred speed
- 2) Continuous head rotation (faster): like the previous task, but each subject was asked to perform a faster movement than before
- 3) Continuous head flexion/extension: The subjects were asked to move their heads up and down (flexion and extension) continuously at their preferred speed
- 4) Continuous head roll: the subjects were asked to perform a continuous head roll left and right at their preferred speed
- 5) Continuous free head movement: in this task, the subjects were asked to move their heads in any direction at their preferred speed

All these trials were repeated 3 times, and during each repetition, the data were recorded for 10 seconds. The purpose of the first 4 tasks was to isolate each movement direction for the EMG signal, while the last task was used to mimic real human head motion when using a VR system (a free movement task).

## B. DATA PROCESSING AND MODEL TRAINING

The raw data from the IMU sensor were resampled to 1000 Hz and combined using sensor fusion and a complementary filter algorithm to generate head orientation in terms

of pitch, roll, and yaw (in degrees). The pitch and roll angle can be measured using (1) and (2), where  $g_x$ ,  $g_y$ , and  $g_z$  are the normalized measured acceleration from the accelerometer on each axis, and  $B_x$ ,  $B_y$ , and  $B_z$  are components of the magnetometer sensor [24], [25]:

$$\text{pitch}(\theta) = \tan^{-1}\left(\frac{-g_x}{\sqrt{g_y^2 + g_z^2}}\right) \quad (1)$$

$$\text{roll}(\phi) = \tan^{-1}\left(\frac{g_y}{g_z}\right) \quad (2)$$

## C. MEANWHILE, THE YAW ANGLE CAN BE FOUND WITH (3)

$$\text{yaw}(\psi) = \tan^{-1}\frac{B_z \sin \phi - B_y \cos \phi}{B_x \cos \theta + B_y \sin \theta \sin \phi + B_z \sin \theta \cos \phi} \quad (3)$$

We used a simple complementary filter to combine the outputs from the accelerometer and gyroscope to remove the drift in the angular estimation for each sensor in both devices. The angle calculated from accelerometer data was fed into a low pass filter, while the angle from the integrated gyroscope data was fed into a high pass filter. The final estimation of the angle was obtained from the sum of both measurements (4):

$$\alpha_0 = k\alpha_{gyro} + (1 - k)\alpha_{acc} \quad (4)$$

where  $\alpha_0$  is the final estimation of the angle (pitch, roll, and yaw),  $\alpha_{gyro}$  is the angle estimated from integration of the gyroscope data, and  $\alpha_{acc}$  is the angle estimated from accelerometer data (1) and (2). Meanwhile,  $k$  is the weighting factor, for which we used  $k = 0.98$  [26].

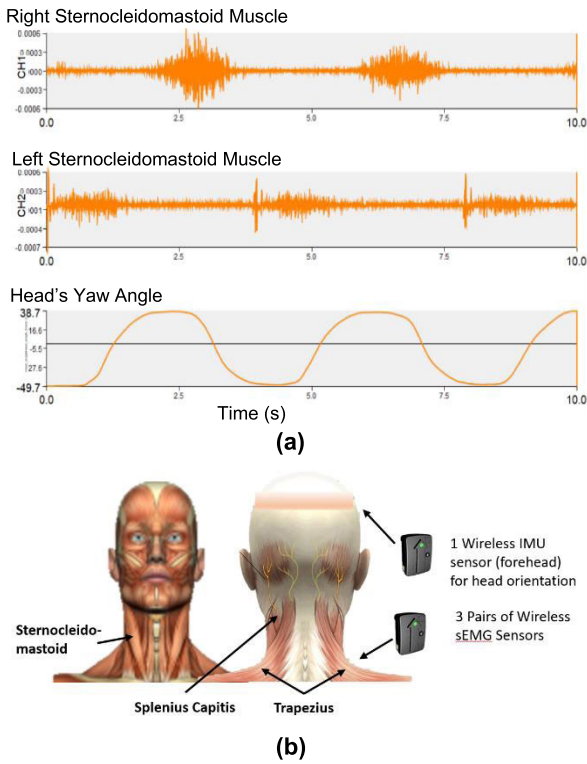
The head orientation data were then shifted 50 ms ahead of the sEMG and HD-EMG signals to become the ground truth for model training. The 50 ms shifting time was chosen based on the EMD value from the EMG signal. However, since the actual EMD value is difficult to measure due to the variation between subjects and muscles, we needed to define a specific dt as the part of the pattern recognition solution. This dt value should be large enough to compensate for the end-to-end latency from the VR system, which can range up to 20 ms.

## 1) SEMG DATA PROCESSING AND FEATURE EXTRACTION

The raw data of the sEMG signal were resampled to 1000 Hz and filtered with a 2nd order Butterworth bandpass filter with a band frequency of 50-500 Hz. The filtered signal was then fully rectified and smoothed with the moving average. The sample of the raw sEMG signal from the sternocleidomastoid muscle and head's yaw angle is shown in Figure 1.

The sEMG signals were treated in two different ways for the two different types of model training. The first one used only the pre-processed data as explained above, while the second one applied an additional feature extraction method after pre-processing the raw signal. Several features used in this study are explained below:





**FIGURE 1.** Sample of right and left sternocleidomastoid muscle raw data with head’s yaw angle (a) and illustration of sensor placement location on neck muscle (b).

Moving Average [27]: This feature was used to calculate the absolute amplitude averaged over a small window of time to obtain smoother data (5):

$$Mov\ Avg = \frac{1}{W} \sum_{i=t-W}^t S(i). \quad (5)$$

Integrated EMG [27] is summation of the absolute values of the sEMG signal amplitude (area under the curve), expressed as (6)

$$iEMG = \sum_{i=t-W}^t S(i). \quad (6)$$

EMG Variance [27] is defined as the average square values of the deviation. The variance of EMG is also another power index and defined as (7)

$$VAR = \frac{1}{W-1} \sum_{i=t-W}^t S_i^2. \quad (7)$$

The EMG Average Slope [11] is computed as the average gradient of the rectified sEMG signal (8):

$$AVG\ SL = \frac{\sum_{i=t-W}^t |S(i) - S(i-1)|}{W}. \quad (8)$$

The curve complexity of EMG [11] is the sum of the gradient of the rectified sEMG signal. This feature measures the curve “shape” according to Hou and Dey [20] (9):

$$CurveComp = \sum_{i=t-W}^t |S(i) - S(i-1)| \quad (9)$$

where  $W$  is the window size, and  $S$  is the preprocessed sEMG signal.

## 2) HD-EMG SIGNAL PROCESSING

Raw data of the HD-EMG signal were filtered with a 2nd order Butterworth bandpass filter featuring a band frequency of 50-500 Hz. The filtered signal was fully rectified and the Root Mean Square (RMS) values for every 10 ms window data were calculated. A 10 ms window was chosen experimentally based on the additional processing/delay time for the system and the features/information stored in the windowed data.

After that, the pre-processed HD-EMG signals were reshaped by row x column to become the input for the convolutional layers. Several configurations of row x column for the HD-EMG signal were examined to obtain the model with the best performance.

## 3) MODEL TRAINING

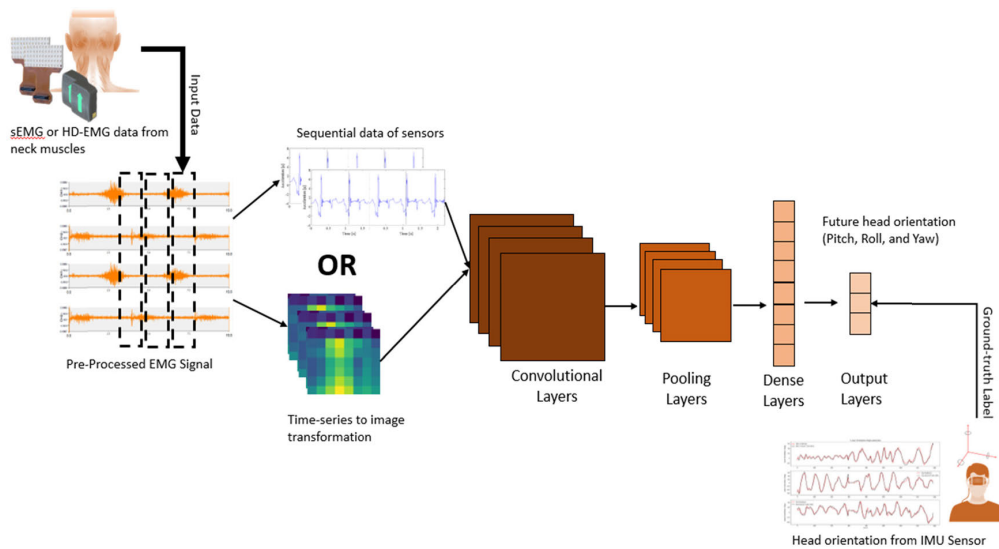
To determine which model offers the best performance in predicting future head orientation, various deep learning model architectures were examined in this study. The model architectures tested in this study included 1D- and 2D- Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), Artificial Neural Network (ANN), and a hybrid model of 1D-CNN and LSTM.

The CNN architecture performs convolutional operations on the input data, followed by pooling and fully connecting the layers. The CNN layers enable automatic feature extraction followed by pooling and fully connecting the layers to construct a deep network. CNN has extensive applications in object detection and classification problems, including for self-driving cars and healthcare applications [28], [29]. This architecture features more (deep) layers that are capable of performing automatic feature selection from raw input data, which makes deep learning superior to machine learning algorithms since the former does not require input from hand-crafted features for learning.

Specifically, for the model with a 2D-CNN architecture, additional time-series to image transformation were needed. All the features input into the 2D-CNN model were reshaped as row x column x step, where step is the moving window (35 ms in our case). For example, the pre-processed data from HD-EMG were  $N \times 64 \text{ channel} \times 35 \text{ ms (step)}$ , where  $N$  is the number of samples. Then, the data were reshaped to  $(N \times 8 \times 8 \times 35)$ .

Furthermore, we tried two methods to input our data into the model. First, we directly input the 3D arrays  $(N \times 8 \times 8 \times 35)$  into the custom 2D-CNN model without transforming the arrays into an image form. Secondly, we used the original shape of the HD-EMG data without time-windowing  $(N \times 8 \times 8)$  and then created a normalized heatmap image for each sample from the  $8 \times 8$  electrode. This heatmap of the HD-EMG data provides a graphical representation that uses a color-code to represent different values. Finally, each heatmap image of HD-EMG was used as the input for the 2D-CNN model.

Also used in this study was LSTM model architecture, which was developed to take advantage of sequential data



**FIGURE 2.** The illustration for input and output data for the model training process on the CNN Model.

such as the time series data of sensors like sEMG. LSTM utilizes memory cells to store contextual information; then, with the inclusion of some logic gates, the model can learn the temporal features from the sequential data [28]. In this study, we developed a custom network that consisted of each architecture (2D-CNN, 1D-CNN, and LSTM), and the number of layers, kernels, and filters was found by experimental trial-and-error. The final model architecture is shown in Figure 3 (1D-CNN). All models in this study used the Mean Absolute Error (MAE) as the loss function (10):

$$MAE = \frac{1}{n} \sum_{j=1}^n |y_j - \hat{y}_j| \quad (10)$$

where  $\hat{y}$  is the head orientation prediction, and  $y$  is the ground truth.

The mean absolute error of the testing data in the pitch, roll, and yaw directions was measured and compared between the different models. The total parameters, model sizes, and total Floating Point Operations per Second (FLOPS) were also calculated for each of the models to compare the complexity between the models.

Other metrics calculated for the turning point of the head (maximal or minimal degree point of head orientation) were also investigated. The delta  $\theta$  peaks ( $d\theta$  peaks) were calculated using the difference of the degree between the peak of the ground truth and the closest peak of prediction. This metric was used to measure how good the model predicted the head orientation when the head was turning or changing direction. Delta time peaks ( $dt$  peaks) were calculated from the difference of the time between the peak of the ground truth and the closest peak of prediction. This metric was used to measure how accurately the model was able to detect whether the head would stop or change direction (turning around).

The inter-subject and intra-subject testing methods were used in this study. For the intra-subject testing method, the testing data came from part of the subject data that included in

the training data. Meanwhile, the inter-subject testing method utilized testing data from a subject never exposed to the training process. Therefore, in the inter-subject testing method, the testing subject was a completely new subject (one of the subjects). Figure 2 illustrates the input and output data for the model training process.

To compare our results with state-of-the-art head orientation prediction, we used the linear extrapolation method to predict the pitch, roll, and yaw of the head orientation. This linear extrapolation is the baseline for the head orientation predictions used on VR devices such as the Oculus Rift [30] and tested in other studies [19]. The formula for linear extrapolation is provided in (11):

$$y(x_k)^* = y(x_{k-2}) + \frac{(x_k - x_{k-1})}{x_{k-1} - x_{k-2}} (y(x_{k-2}) - y(x_{k-1})) \quad (11)$$

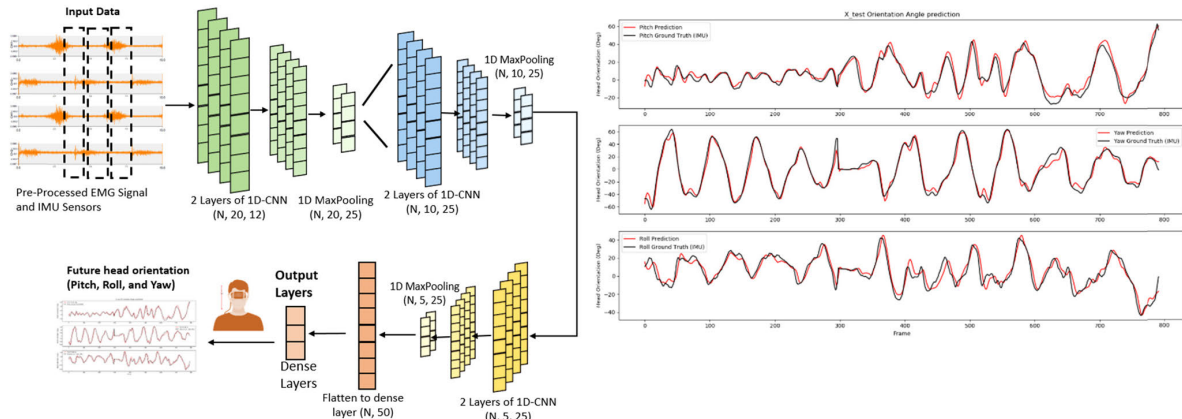
where  $y(x_k)^*$  is the predicted value, and  $y(x_k)$  is the measured value.

#### IV. RESULTS

The results for the sEMG-based and HD-EMG-based predictions are shown in Table 1. All the results, except for the models with only sEMG and HD-EMG features, were trained with three axis output from the same model. We also trained another model that separates each axis output for each model, and the result is shown in Table 2. The results from Table 1, Table 2, and Table 3 came from the intra-subject testing method.

The results for the  $d\theta$  peaks and  $dt$  peaks are expressed in degrees and milliseconds, respectively. A summary of all trained model  $d\theta$  peaks and  $dt$  peaks is shown in Table 3.

All of the above results were testing using the intra-subject testing method. However, for the inter-subject testing method, the model was only trained on the best input combination result: sEMG feature extraction + IMU and preprocessed



**FIGURE 3.** The testing result from the best performing model (1D-CNN) on the pre-processed EMG + IMU input combination. The red line is the prediction using the model, while the black line is the ground truth (right figure). Detailed model architecture of the best performing model for pre-processed EMG + IMU (1D-CNN) is shown (left figure).

**TABLE 1.** Mean absolute error of the models from sEMG- and HD-EMG-based head orientation predictions (3-axis output model). This result came from the intra-subject testing method.

Input Combination	Best Model	Mean Absolute Error (deg)		
		Pitch	Roll	Yaw
sEMG Feature Extraction	2D-CNN-based Model	37.42	23.42	16.00
sEMG Features + IMU	2D-CNN-based Model	4.90	10.93	5.43
Preprocessed sEMG + IMU	1D-CNN-based Model	3.08	4.36	4.50
HD-EMG	2D-CNN-based Model	26.33	28.90	18.40
HD-EMG + IMU	1D-CNN-based Model	7.17	10.41	8.73
IMU Only	1D-CNN-based Model	3.57	4.79	5.47
Baseline (IMU with extrapolation)	Linear Extrapolation	2.66	1.68	1.46

sEMG + IMU. The summary for this result is shown in Table 4.

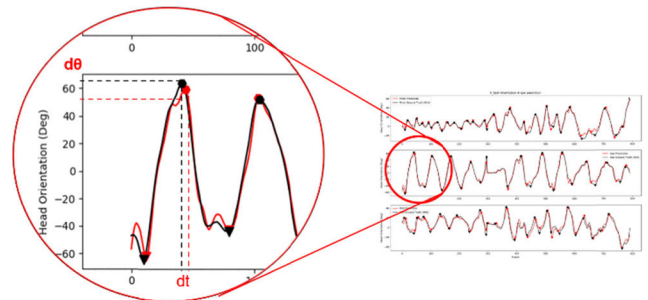
The result for testing data to determine the best input combination (pre-processed sEMG + IMU) on the 1D-CNN model is shown in Figure 3. Meanwhile for prediction result on testing data for HD-EMG + IMU input combination can be seen in Figure 4.

**V. DISCUSSION**

The results for the sEMG group showed that the combination of preprocessed sEMG + IMU outperformed the other input combinations, including the one using sEMG features. This occurred because the chosen features were not optimized for this application, as the features chosen were based on a previous study that also used sEMG signals for modeling purposes [27]. Another reason why the model with pre-processed data performed better than that with extracted features was because this study trained the model via the deep learning method, which is specialized for learning automatic feature representations in large sets of raw data. This results agrees with a study from A. Phinyomark *et al.* reviewed

**TABLE 2.** Mean absolute error of the model from the sEMG- and HD-EMG-based head orientation prediction (1-axis output model). This result came from the intra-subject testing method.

Input Combination	Best Model	Mean Absolute Error (deg)		
		Pitch	Roll	Yaw
sEMG Features + IMU	2D-CNN-based Model	4.58	11.55	6.09
Preprocessed sEMG + IMU	1D-CNN-based Model	3.27	3.96	3.78
HD-EMG + IMU	1D-CNN-based Model	6.40	10.23	8.76
IMU Only	1D-CNN-based Model	4.87	4.01	3.93



**FIGURE 4.**  $d\theta$  peaks and  $dt$  peaks calculated from the sEMG-based model.

the recognition of several EMG patterns using the feature learning and deep learning methods and showed that the EMG pattern recognition system based on the deep learning method offers better classification accuracy than its counterparts such as Support Vector Machine, K-Nearest Neighbors, Multi Layers Perceptron, and Random Forest [31].

In terms of model architecture, the pre-processed sEMG and sEMG features utilized different model architectures in their best performing models. Pre-processed sEMG used 1D-CNN as its best model, while the sEMG features utilized the model with 2D-CNN architecture. This could be related to the nature of 2D-CNN architecture, which learns from spatial

**TABLE 3. Summary of  $d\theta$  peaks and  $dt$  peaks for all trained models.**

Input Combination		$d\theta$ Peaks (deg)			$dt$ Peaks (ms)		
		<i>Pitch</i>	<i>Roll</i>	<i>Yaw</i>	<i>Pitch</i>	<i>Roll</i>	<i>Yaw</i>
sEMG Features + IMU	+	5.19	12.75	7.63	15.52	17.05	15.12
Preprocessed sEMG + IMU	+	2.69	4.91	5.85	2.12	3.45	3.03
HD-EMG + IMU	+	8.03	22.11	12.87	3.18	2.27	1.55
IMU Only		4.08	6.68	6.12	2.00	4.80	3.50
Baseline (IMU with extrapolation)		1.87	2.35	1.85	26.9	29.26	36.33

features instead of sequences, the latter of which can only be found in sEMG features because of the large dimension of its features (6 channels \* 5 features = 30 total features). Meanwhile, the data from pre-processed sEMG carried information on its sequence of signals. Thus, in this case, 1D-CNN was more suitable and performed better for this input data. We also found that the model using input data directly from the reshaped array without transformation to an image form achieved better performance in terms of MAE and processing speed compared to the model with image transformation input. This is likely because of the size of the image, which, after transformation, is far larger than the size of an array. Therefore, if the dimension of an array is sufficient to become the input for the 2D-CNN model, it is better to retain the array form instead of first transforming it into an image.

Compared to the result from HD-EMG, both the pre-processed EMG and sEMG features provided better predictions, except in the roll direction for sEMG features + IMU input, for which the HD-EMG + IMU combination provided better results with a  $0.52^\circ$  error difference. This performance could be related to the number of data collected for both groups since the sEMG group features more data compared to the HD-EMG group (20 subjects compared to 11 subjects in the HD-EMG group). However, the result from HD-EMG proved that input from the HD-EMG data could also be used to predict future head orientation. The lower accuracy of the HD-EMG model could be compensated for by the ease of its use and comfort for the subject when using grid electrodes from the HD-EMG system. When using the HD-EMG system, the subject does not necessarily know exactly where the muscle belly is located, which is required when using sEMG electrodes to obtain a high-quality sEMG signal. This performance vs. usability tradeoff creates another opportunity for HD-EMG to replace sEMG for predicting future head orientations if the model accuracy can be improved in the future by collecting more data and fine-tuning the model.

Studies from Barniv *et al.* [11] and Polak *et al.* [4] also utilized features extracted from sEMG to predict head motion. However, the target prediction was different since this study predicted head angular velocity (rad/sec), while our study predicted head orientation (pitch, roll, and yaw in degrees).

**TABLE 4. Performance comparison for the inter-subject testing method.**

Input Combination	Best Model	Mean Absolute Error (deg)		
		<i>Pitch</i>	<i>Roll</i>	<i>Yaw</i>
sEMG Features + IMU	2D-CNN-based Model	5.26	11.72	5.62
Preprocessed sEMG + IMU	1D-CNN-based Model	4.85	6.50	4.47

Another difference is that the previous study used one output/axis model, providing three separate models for three different movement axes. In Table 2, we can see that the model with one axis output offers slightly better performance compared to the three-axis-output model. However, this factor was compensated for by the size of the model, which was three times larger due to containing three separate models. The accuracy of each axis model can be increased by tuning each model with separate parameters because, in this study, each separated model used the same parameters.

Using EMG signals with the deep learning method showed positive trends marked by an increasing number of published papers related to this topic every year from 2014 to 2018 [32]. Most of these papers investigated applications in hand gesture classification, speech, and emotion applications, as well as sleep stage classification [33], [34]. However, the high-variability of the sEMG signal remains a problem, and most of these studies was still utilized intra-subject or intra-session methods for testing their models. Besides classification models that still use the intra-session testing method, problems like the regression model with deep learning on sEMG still utilize intra-session testing methods, as shown in a study by J. Chen, who performed a continuous estimation of human lower limb joints with sEMG signals [35]. Likewise, in 2019, Y. Chen developed a continuous estimation model of the upper limb joint angle with the sEMG and deep learning models [36]. These previous studies proved that sEMG signals are better for use in intra-session testing, and the results in Table 4 agree with these previous claims. However, this study also proved that even with the risk of decreasing performance, inter-session and inter-subject testing methods can be used in sEMG-related studies. This result could be achieved if the training data were large enough to obtain better generalization performance. Our results for the inter-subject testing method in this study open the opportunity for real-world applications under conditions where the user data were never exposed to any training data before. Thus, there is no need to develop another transfer learning/ fine-tuning model for convenience of time-saving purposes. However, fine tuning or transfer learning will always be a good option if we do not want to sacrifice model performance.

We also compared our results with state-of-the-art head orientation predictions in VR applications using the extrapolation method. This extrapolation method has been used for recent commercialized VR headsets, such as Oculus Rift [30]. Another study proved that linear extrapolation outperforms the polynomial extrapolation method [19]. In terms of MAE,



extrapolation that only uses IMU sensor data outperformed the other model with sEMG and HD-EMG data. However, the result for dt peaks using the extrapolation method was superior to all EMG-based predictions. This indicates that the extrapolation method tends to inaccurately predict the time when one's head changes direction, which is indicated by the peak on the graph. These  $d\theta$  peaks and dt peaks are important when a VR user engages in intense or abrupt movements, such as when playing First Person Shooter games or exercise/sports-based games. Therefore, even though our EMG-based prediction model did not offer better overall performance in terms of MAE, our prediction model outperformed the extrapolation method in predicting when the user would change their head direction (dt peaks). In other words, our sEMG-based prediction model is suitable for VR applications that require the user to perform high-intensity or abrupt movements, such as in FPS games or exercise/sports-based games. An illustration of  $d\theta$  peaks and dt peaks is provided in Figure 4.

Future work based on this study could implement this system in a real VR system as the default head motion prediction model. The predicted head orientation can be used to pre-render the frame, which can then be stored in cache memory. After that, if the predicted head orientation is found to be correct in the future, the cached predicted frame will be rendered to the VR headset. Otherwise, the actual view from the real head orientation will be rendered. Furthermore, this pre-rendering from predicted head orientation could be integrated with game engines like Unity or included in some VR games that require the user to move abruptly, such as First Person Shooter (FPS) games.

## VI. CONCLUSION

The result from this study showed that the head orientation prediction model with input from a combination of pre-processed sEMG + IMU outperforms the model using HD-EMG + IMU. However, the decreased performance on HD-EMG was compensated for by the comfort and ease of use of the HD-EMG electrode compared with sEMG. This tradeoff should be considered if the user wants to choose between performance and usability. In terms of the best model performance, the 1D-CNN-based model with input from pre-processed sEMG + IMU data achieved the best results with the MAE of the pitch, roll, and yaw equal to  $3.08^\circ$ ,  $4.36^\circ$ , and  $4.50^\circ$ , respectively.

The results from the inter-subject testing method used in this study also open up an opportunity for real-world applications in which the user does not have to perform additional transfer learning or fine-tuning processes, as these processes can take a longer time. However, transfer learning and fine-tuning the model will always be a good option if the user does not want to sacrifice the performance of the prediction model. Moreover, our results also proved that the sEMG-based model offers better performance in predicting when the user will change his or her head direction, which was quantified by calculating the dt peaks. In other words, our sEMG-based

prediction model is suitable for VR applications that require the user to perform high-intensity or abrupt movements, such as in FPS games or exercise/sports-based games.

## REFERENCES

- [1] J. Zhao, R. S. Allison, M. Vinnikov, and S. Jennings, "Estimating the motion-to-photon latency in head mounted displays," in *Proc. IEEE Virtual Reality (VR)*, Mar. 2017, pp. 313–314.
- [2] J. Jerald, *The VR Book: Human-Centered Design for Virtual Reality*. Morgan & Claypool: San Rafael, CA, USA, 2015.
- [3] B. D. Adelstein, T. G. Lee, and S. R. Ellis, "Head tracking latency in virtual environments: Psychophysics and a model," in *Proc. Hum. Factors Ergonom. Soc. Annu. Meeting*, vol. 47, no. 20. Los Angeles, CA, USA: Sage, 2003, pp. 2083–2087.
- [4] S. Polak, Y. Barniv, and Y. Baram, "Head motion anticipation for virtual-environment applications using kinematics and EMG energy," *IEEE Trans. Syst., Man, A, Syst. Humans*, vol. 36, no. 3, pp. 569–576, May 2006.
- [5] H. Zhang, M. Chen, X. Li, K. Li, X. Chen, J. Wang, H. Liang, J. Zhou, W. Liang, H. Fan, R. Ding, S. Wang, and D. Deng, "Overcoming latency with motion prediction in directional autostereoscopic displays," *J. Soc. Inf. Display*, vol. 28, no. 3, pp. 252–261, Mar. 2020.
- [6] J.-P. Stauffert, F. Niebling, and M. E. Latoschik, "Simultaneous runtime measurement of motion-to-photon latency and latency jitter," in *Proc. IEEE Conf. Virtual Reality 3D User Interfaces (VR)*, Mar. 2020, pp. 636–644.
- [7] S. Pape, M. Kruger, J. Muller, and T. W. Kuhlen, "Calibratio: A small, low-cost, fully automated Motion-to-Photon measurement device," in *Proc. IEEE Conf. Virtual Reality 3D User Interfaces Abstr. Workshops (VRW)*, Mar. 2020, pp. 234–237.
- [8] R. Gruen, E. Ofek, A. Steed, R. Gal, M. Sinclair, and M. Gonzalez-Franco, "Measuring system visual latency through cognitive latency on video see-through AR devices," in *Proc. IEEE Conf. Virtual Reality 3D User Interfaces (VR)*, Mar. 2020, pp. 791–799.
- [9] I. T. Feldstein and S. R. Ellis, "A simple video-based technique for measuring latency in virtual reality or teleoperation," *IEEE Trans. Vis. Comput. Graphics*, early access, Mar. 13, 2020, doi: 10.1109/TVCG.2020.2980527.
- [10] F. Ababsa, M. Malle, and D. Roussel, "Comparison between particle filter approach and Kalman filter-based technique for head tracking in augmented reality systems," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, vol. 1, Apr. 2004, pp. 1021–1026.
- [11] Y. Barniv, M. Aguilar, and E. Hasanbelliu, "Using EMG to anticipate head motion for virtual-environment applications," *IEEE Trans. Biomed. Eng.*, vol. 52, no. 6, pp. 1078–1093, Jun. 2005.
- [12] A. Kiruluta, M. Eizenman, and S. Pasupathy, "Predictive head movement tracking using a Kalman filter," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 27, no. 2, pp. 326–331, Apr. 1997.
- [13] A. van Rhijn, R. van Liere, and J. D. Mulder, "An analysis of orientation prediction and filtering methods for VR/AR," in *Proc. IEEE Virtual Reality (VR)*, Mar. 2005, pp. 67–74.
- [14] F.-E. Ababsa and M. Malle, "Inertial and vision head tracker sensor fusion using a particle filter for augmented reality systems," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2004, p. III-861.
- [15] W. H. Zangemeister, S. Lehman, and L. Stark, "Simulation of head movement trajectories: Model and fit to main sequence," *Biol. Cybern.*, vol. 41, no. 1, pp. 19–32, Jun. 1981.
- [16] N. Enayati, E. De Momi, and G. Ferrigno, "A quaternion-based unscented Kalman filter for robust Optical/Inertial motion tracking in computer-assisted surgery," *IEEE Trans. Instrum. Meas.*, vol. 64, no. 8, pp. 2291–2301, Aug. 2015.
- [17] H. Himberg and Y. Motai, "Head orientation prediction: Delta quaternions versus quaternions," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 39, no. 6, pp. 1382–1392, Dec. 2009.
- [18] C. H. Kang, C. G. Park, and J. W. Song, "An adaptive complementary Kalman filter using fuzzy logic for a hybrid head tracker system," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 9, pp. 2163–2173, Sep. 2016.
- [19] A. Garcia-Agundez, A. Westmeier, P. Caserman, R. Konrad, and S. Göbel, "An evaluation of extrapolation and filtering techniques in head tracking for virtual environments to reduce cybersickness," in *Proc. Joint Int. Conf. Serious Games*. Cham, Switzerland: Springer, 2017, pp. 203–211.
- [20] X. Hou and S. Dey, "Motion prediction and pre-rendering at the edge to enable ultra-low latency mobile 6DoF experiences," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 1674–1690, 2020.

[21] S. Gül, D. Podborski, T. Buchholz, T. Schierl, and C. Hellige, "Low-latency cloud-based volumetric video streaming using head motion prediction," presented at the 30th ACM Workshop Netw. Oper. Syst. Support Digit. Audio Video, Istanbul, Turkey, 2020, doi: 10.1145/3386290.3396933.

[22] D. Saredakis, A. Szpak, B. Birchead, H. A. D. Keage, A. Rizzo, and T. Loetscher, "Factors associated with virtual reality sickness in head-mounted displays: A systematic review and meta-analysis," *Frontiers Human Neurosci.*, vol. 14, p. 96, Mar. 2020.

[23] D. F. Stegeman, B. U. Kleine, B. G. Lapatki, and J. P. Van Dijk, "High-density surface EMG: Techniques and applications at a motor unit level," *Biocybern. Biomed. Eng.*, vol. 32, no. 3, pp. 3–27, Jan. 2012.

[24] M. J. Pedley, "Tilt sensing using a three-axis accelerometer: Freescale semiconductor application note," *Appl. Note AN3461 Rev. 6*, 2013, vol. 1, pp. 2012–2013.

[25] T. J. F. S. Ozyagcilar, "Implementing a tilt-compensated eCompass using accelerometer and magnetometer sensors: J Freescale semiconductor, AN," *Appl. Note AN4248 Rev. 4.0*, 2012, vol. 4248.

[26] M. Nowicki, J. Wietrzykowski, and P. Skrzypczynski, "Simplicity or flexibility? Complementary filter vs. EKF for orientation estimation on mobile devices," in *Proc. IEEE 2nd Int. Conf. Cybern. (CYBCONF)*, Jun. 2015, pp. 166–171.

[27] A. Phinyomark, P. Phukpattaranont, and C. Limsakul, "Feature reduction and selection for EMG signal classification," *Expert Syst. Appl.*, vol. 39, no. 8, pp. 7420–7431, Jun. 2012.

[28] H. F. Nweke, Y. W. Teh, M. A. Al-garadi, and U. R. Alo, "Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges," *Expert Syst. Appl.*, vol. 105, pp. 233–261, Sep. 2018.

[29] T. Sugiarto, C.-L. Hsu, C.-T. Sun, S.-H. Ye, K.-T. Lu, and C.-H. Wang, "An automatic COPD diagnosis with deep learning on topology-preserving multi spectral image of EEG data," in *Basic & Clinical Pharmacology & Toxicology*, vol. 124. Hoboken, NJ, USA: Wiley, 2019, pp. 13–14.

[30] S. M. LaValle, A. Yershova, M. Katsev, and M. Antonov, "Head tracking for the oculus rift," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 187–194.

[31] A. Phinyomark and E. Scheme, "EMG pattern recognition in the era of big data and deep learning," *Big Data Cognit. Comput.*, vol. 2, no. 3, p. 21, Aug. 2018.

[32] D. Buongiorno, G. D. Cascarano, A. Brunetti, I. De Feudis, and V. Bevilacqua, "A survey on deep learning in electromyographic signal analysis," in *Intelligent Computing Methodologies*. Cham, Switzerland: Springer, 2019, pp. 751–761.

[33] A. H. Al-Timemy, G. Bugmann, J. Escudero, and N. Outram, "Classification of finger movements for the dexterous hand prosthesis control with surface electromyography," *IEEE J. Biomed. Health Informat.*, vol. 17, no. 3, pp. 608–618, May 2013.

[34] U. Cote-Allard, C. L. Fall, A. Drouin, A. Campeau-Lecours, C. Gosselin, K. Glette, F. Laviolette, and B. Gosselin, "Deep learning for electromyographic hand gesture signal classification using transfer learning," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 4, pp. 760–771, Apr. 2019.

[35] J. Chen, X. Zhang, Y. Cheng, and N. Xi, "Surface EMG based continuous estimation of human lower limb joint angles by using deep belief networks," *Biomed. Signal Process. Control*, vol. 40, pp. 335–342, Feb. 2018.

[36] Y. Chen, S. Yu, K. Ma, S. Huang, G. Li, S. Cai, and L. Xie, "A continuous estimation model of upper limb joint angles by using surface electromyography and deep learning method," *IEEE Access*, vol. 7, pp. 174940–174950, 2019.



**TOMMY SUGIARTO** was born in Indonesia, in 1992. He received the B.S. degree in instrumentation physics from the Department of Physics, University of Indonesia, in 2015, and the M.S. degree in biomedical engineering from the National Taiwan University of Science and Technology, in 2018, where he is currently pursuing the Ph.D. degree in biomedical engineering.

From 2015 to 2016, he worked with the Indonesian National Institute of Aeronautics and Space, as a Research Assistant. Since 2018, he has been working as an Associate Engineer with the Industrial Technology Research Institute, Taiwan. His research interests include artificial intelligence in biomedical applications, gait analysis, human motion analysis, and wearable sensors.



**CHUN-LUNG HSU** (Member, IEEE) received the Ph.D. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, in 2000.

From 2000 to 2011, he was an Assistant/Associate Professor with the Department of Electrical Engineering, National Dong Hwa University (NDHU), Hualien, Taiwan. He currently works with the Information and Communication Research Laboratories (ICLs), Industrial Technology Research Institute (ITRI), Hsinchu, Taiwan. He is mainly responsible for ultra-low-power SoC design, heterogeneous stacking IC design, high-level synthesis design for embedded systems, and artificial intelligence (AI) deep learning HW/SW system design-related projects. His major research interests include low-power circuit design, VLSI design/testing, and tolerance scheme development for image systems. He is a member of the Institute of Electrical, Information and Communication Engineers (IEICE), Japan, and the IET Society.



**CHI-TIEN SUN** received the B.S. degree from Feng Chia University, Taichung, Taiwan, in 1992, and the M.S. degree from Yuan Ze University, Taoyuan, Taiwan, in 1994, both in electrical engineering. He is currently pursuing the Ph.D. degree with the Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan, with a focus on the efficient architecture design of neural networks.

He is also a Manager with the System Integration and Application Department, Information and Communication Research Laboratory (ICL), Industrial Technology Research Institute (ITRI), Hsinchu. His current research interests include analog and digital signal processing, embedded systems and VLSI design, artificial intelligence, and machine learning.



**WEI-CHUN HSU** received the Ph.D. degree from National Taiwan University.

She is currently a Professor in biomedical engineering with the National Taiwan University of Science and Technology (NTUST), Taiwan. She also leads the Laboratory of Movement Diagnosis and Rehabilitation Engineering at the Graduate Institute of Biomedical Engineering, NTUST. Her research interests include human motion analysis and biomechanics.



**SHU-HAO YE** received the Ph.D. degree from the Department of Computer Science, National Chiao Tung University (NCTU), Taiwan. He is currently working as an Engineer with the Information and Communication Laboratory (ICL), Industrial Technology Research Institute (ITRI), Taiwan.



**KUAN-TING LU** received the M.S. degree in computer science from National Chiao-Tung University (NCTU), Taiwan. He is currently an Associate Engineer with the Information and Communication Laboratory (ICL), Industrial Technology Research Institute (ITRI), Taiwan.