IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 7, 2020, accepted August 29, 2020, date of publication September 7, 2020, date of current version September 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3022042

Deep Learning for Intake Gesture Detection From Wrist-Worn Inertial Sensors: The Effects of Data Preprocessing, Sensor Modalities, and Sensor Positions

HAMID HEYDARIAN[1], PHILIPP V. ROUAST[1], (Member, IEEE), MARC T. P. ADAM[1,2], TRACY BURROWS[2,3], CLARE E. COLLINS[2,3], AND MEGAN E. ROLLO[2,3]

[1]School of Electrical Engineering and Computing, Faculty of Engineering and Built Environment, The University of Newcastle, Callaghan, NSW 2308, Australia
[2]Priority Research Centre for Physical Activity and Nutrition, The University of Newcastle, Callaghan, NSW 2308, Australia
[3]School of Health Sciences, Faculty of Health and Medicine, The University of Newcastle, Callaghan, NSW 2308, Australia

Corresponding author: Marc T. P. Adam (marc.adam@newcastle.edu.au)

This work was supported in part by the Bill & Melinda Gates Foundation under Grant OPP1171389. The work of Hamid Heydarian and Philipp Rouast was supported by Australian Government Research Training (RTP) Scholarships. The work of Clare Collins was supported in part by the Australian National Medical Research Council Senior Research Fellowship and in part by the University of Newcastle Faculty of Health and Medicine Gladys M Brawn Senior Research Fellowship.

ABSTRACT Wrist-worn inertial measurement units have emerged as a promising technology to passively capture dietary intake data. State-of-the-art approaches use deep neural networks to process the collected inertial data and detect characteristic hand movements associated with intake gestures. In order to clarify the effects of data preprocessing, sensor modalities, and sensor positions, we collected and labeled inertial data from wrist-worn accelerometers and gyroscopes on both hands of 100 participants in a semi-controlled setting. The method included data preprocessing and data segmentation, followed by a two-stage approach. In Stage 1, we estimated the probability of each inertial data frame being intake or non-intake, benchmarking different deep learning models and architectures. Based on the probabilities estimated in Stage 1, we detected the intake gestures in Stage 2 and calculated the F_1 score for each model. Results indicate that top model performance was achieved by a CNN-LSTM with *earliest sensor data fusion through a dedicated CNN layer* and a *target matching* technique ($F_1 = .778$). As for data preprocessing, results show that applying a consecutive combination of mirroring, removing gravity effect, and standardization was beneficial for model performance, while smoothing had adverse effects. We further investigate the effectiveness of using different combinations of sensor modalities (i.e., accelerometer and/or gyroscope) and sensor positions (i.e., dominant intake hand and/or non-dominant intake hand).

INDEX TERMS Accelerometer, deep learning, intake gesture detection, gyroscope, wrist-worn.

I. INTRODUCTION

Advances in mobile sensor technologies have enabled novel forms of dietary assessment. While dietary assessment was traditionally carried out exclusively using active methods for capturing food intake based on human effort to collect data (e.g., 24-hr recalls, food records), passive capture methods aim to reduce burden on individuals associated with collecting dietary data by using a range of different sensor technologies (e.g., inertial measurement units, microphones, and video cameras). Sensor technologies have the potential of complementing active capture methods for quantifying food intake [1] (e.g., by verifying intake activities, prompting human capture).

In recent years, the wrist-worn Inertial Measurement Unit (IMU) has emerged as a promising technology for sensor-based passive capture of food intake [2]–[4]. Mounted to the wrist, triaxial accelerometers and gyroscopes embedded in IMUs can be used to detect characteristic hand movements associated with eating and drinking (e.g., intake gestures, such as raising a fork or cup). In particular, triaxial accelerometers in IMUs measure changes in speed and

The associate editor coordinating the review of this manuscript and approving it for publication was Gang Mei.

TABLE 1. Related research on intake gesture detection using wrist-worn inertial sensors with deep learning.

Author (Year) [Ref], Subjects (Environment)	Device, Sensors (Frequency) [Hand]	Model	Data preprocessing steps in sequence of execution
Kyritsis et al. (2017) [5], 10 subjects (lab)	Microsoft Band 2, Acc/Gyro (62 Hz) [dominant hand]	SVM-LSTM	1. Smoothing (5^{th} order median filter) 2. Gravity removal (quaternion representation calculated using Madgwick's algorithm)
Papadopoulos et al. (2018) [11], 10 subjects (lab)	Microsoft Band 2, Acc/Gyro (62 Hz) [dominant hand]	Semi-supervised SVM-LSTM	1. Smoothing (5^{th} order median filter) 2. Gravity removal (high-pass FIR filter)
Kyritsis et al. (2018) [10], 10 subjects (lab)	Microsoft Band 2, Acc/Gyro (100 Hz) [dominant hand]	CNN-LSTM	1. Smoothing (5^{th} order median filter) 2. Gravity removal (high-pass FIR filter) 3. Standardization (per column)
Cho & Choi (2018) [9], 8 subjects (lab)	LG URBANE 2, Acc (50 Hz) [dominant hand]	CNN	1. Smoothing (10-point moving average filter)
Anderez et al. (2018) [8], 1 subject (lab)	Device not reported, Acc/Gyro (100 Hz) [dominant hand]	LSTM	None
Kyritsis et al. (2019) [6], 12 subjects (lab)	Microsoft Band 2, Acc/Gyro (100 Hz) [dominant hand]	CNN- LSTM	1. Mirroring [25] 2. Smoothing (5^{th} order median filter) 3. Gravity removal (high-pass FIR filter) 4. Standardization (per column)
Kyritsis et al. (2020) [22], 12 subjects (lab); 12 subjects (free-living); 12 subjects (free-living)*	Microsoft Band 2, Acc/Gyro (100 Hz) [dominant hand]; Huawei Watch 2™ and Mobvoi TicWatch™, Acc/Gyro (100 Hz) [dominant hand]; LG G-Watch, Acc/Gyro (100 Hz) [both hands]	CNN-LSTM	1. Upsampling (from 15 to 100 Hz) 2. Mirroring 3. Smoothing (moving average filter) 4. Gravity removal (high-pass FIR filter)
Current study 100 subjects (lab)	Movisens Move 3 G[1] Acc/Gyro (64 Hz) [both hands]	CNN-LSTM	1. Mirroring 2. Gravity removal (quaternion representation calculated using Madgwick's algorithm) 3. Smoothing (median, moving average, and Savitzky-Golay filters**) 4. Standardization (per column)

Note: Acc = Accelerometer, CNN = convolutional neural network, Gyro = Gyroscope, LSTM = long short-term memory, * = data was collected by different researchers in study [26], ** = comparison of different approaches

direction of the wrist, while the gyroscope measures the rotation rate of these movements. Further, wrist-worn IMUs are readily available in professional grade self-contained devices (e.g., Movisens, XSens) or smartwatches (e.g., Apple Watch, Samsung Gear).

While early approaches for detecting intake activities from wrist-worn IMUs primarily relied on traditional machine learning methods (e.g., support vector machines, random forests) [4], recent research has started to apply deep learning architectures [5], [6]. However, to the best of our knowledge, only seven studies have so far utilized deep learning for this purpose (Table 1). Hence there is a need for further research to leverage its full potential. As such, it is an open question whether data preprocessing supports deep learning models and what different sensor modalities (e.g., accelerometer and/or gyroscope, left and/or right hand) and sensor configurations (e.g., sampling rate) contribute to achieve high performance. Understanding the impact of sensor modalities and configurations is important in settings where there can be constraints on (1) the number of sensors and devices, (2) energy consumption in data collection over extended periods of time, particularly in low-income countries [7], and (3) users' acceptance towards wearing sensors on both hands. Given the different approaches in data preprocessing, it is currently not clear which data preprocessing steps achieve high model performance.

The current paper addresses this research gap by reviewing the existing deep learning models for detecting intake gestures from inertial sensors [5], [6], [8]–[11] and, based on this, proposing our own solution to this problem. In this process, we benchmarked our proposed model against existing models and clarified the impact of different data preprocessing

steps and sensor modalities on model performance. Our main contributions are as follows:

(1) Large-scale Dataset: We conducted a laboratory study and collected accelerometer and gyroscope data on both hands from 100 participants (sampling rate: 64 Hz). Data were annotated and cross-checked by two independent annotators.

(2) Proposed Model and Benchmarking: We propose a new model that achieved better performance ($F_1 = .778$) compared to current state-of-the-art deep learning models for detecting intake gestures based on inertial data, using our novel large-scale dataset. We used an effective way to fuse data (i.e., earliest sensor data fusion) from different sensor modalities (i.e., accelerometer and/ or gyroscope) and sensor positions (i.e., dominant intake hand and/ or non-dominant intake hand) and introduce a novel method to match the labels with input data (i.e., target matching technique) in the processes of training and evaluation more precisely.

(3) Data Preprocessing: Previous research has engaged various different data preprocessing steps, raising the question as to what the impact of each individual data preprocessing step is on model performance. We clarify the impact of data preprocessing approaches (i.e. mirroring, removing gravity effect, smoothing, and standardization) for deep learning models. Results demonstrated that while the combination of mirroring, removing gravity effect, and standardization improved model performance, smoothing was detrimental.

(4) Sensor Modalities and Sensor Positions: Given the multi-modal nature of the data (i.e., left and right hands, accelerometer and gyroscope), we evaluated the importance of the different modalities (e.g., only accelerometer, only gyroscope, only dominant intake hand, and only non-dominant intake hand). Results show that the proposed model using gyroscope data only ($F_1 = .771$) outperforms the same model using only accelerometer data ($F_1 = .682$). Finally, this is one of the first studies that collected inertial data from both hands to train deep learning models. Results confirm that models including data from both hands ($F_1 = .778$) yield a 19% increase in performance compared to a model using data from the dominant intake hand only ($F_1 = .654$).

The remainder of this paper is organized as follows. Section II provides a brief introduction of deep learning and its application in human activity detection and, more specifically, in the field of intake gesture detection. It then discusses the literature in the domain of automatic dietary monitoring using wrist-worn inertial sensors with deep learning and the common data preprocessing steps used. Section III introduces the implemented methods, including our data preprocessing pipeline and proposed model, along with other models for comparison purposes. In Section IV we discuss our dataset and explain the process of data collection in our study. Results of experiments are then presented, with comparisons made in Section V and finally discussion and conclusions are drawn in Section VI.

II. RELATED RESEARCH

A. FOUNDATIONS OF INTAKE GESTURE DETECTION

An *intake gesture* refers to a hand-to-mouth gesture associated with dietary intake (e.g., raising a cup to drink or a fork to eat). By contrast, an *intake activity* refers to eating and/or drinking activities that comprise a continuous sequence of individual intake gestures during an eating occasion (e.g., a meal or snack). Detecting intake gestures is typically a prerequisite for the detection of intake activities [4]. In this paper, *inertial data* refers to wrist movement data collected from tri-axial accelerometers and gyroscopes (each recorded on x, y, and z axis at a certain sample rate frequency, here: 64 Hz).We refer to accelerometers and gyroscopes as *sensor modalities* and the position of the sensor on the left and right wrists as *sensor positions*. Further, we refer to a sensor data point as a *frame* and a frame of an intake gesture as an *intake frame*.

When using machine learning for intake gesture detection, the collected data is commonly segmented into windows of a particular length (e.g., 2 seconds) in order to create temporal input data for the model [4]. One of the widely-used data segmentation approaches applied in the current work is the *sliding window* technique [12]. In this approach a window of a certain length moves over frames, where the frames within the window create a unit of sequential data (referred to as a *temporal element*). The last frame within a window is referred to as the *target frame*. Temporal elements are used to train, validate, and test a machine learning model.

B. DEEP LEARNING

Deep learning, also known as deep neural networks (DNNs), refers to artificial neural networks with multiple hidden layers of non-linear information processing, where each layer uses the output of previous layer as the input [13]. Convolutional Neural Networks (CNNs) are a specific type of DNN designed to automatically learn features from data with a coherent spatial structure. They are often used to avoid hand-crafted or heuristic features [14]. Recurrent Neural Networks (RNNs) are DNNs with additional self-connections suitable for processing sequential data [15]. Long Short-Term Memory (LSTM) is a type of RNN that provides an additional gating mechanism to remember information selectively [16]. Intake gesture and activity detection can be categorized as a specific type of Human Activity Recognition. Deep learning approaches have widely been utilized in the field of human activity recognition using wearable sensors (e.g., [17]–[21]). While deep learning may be able to uncover features tied to complex body motions, the combination of CNN and LSTM in particular has shown advantages in this field [18].

C. MACHINE LEARNING FOR DETECTING INTAKE GESTURES

A recent systematic review identified that up to January 2019, the majority of studies using inertial sensor data for intake gesture and activity detection employed traditional machine

learning approaches [4]. The majority of existing studies used Support Vector Machine (SVM, 21 studies), Random Forest (19 studies), Decision Tree (16 studies), rule-based algorithms (11 studies), Hidden Markov Model (HMM, ten studies) [4], and K-nearest neighbors (KNN, nine studies). Naive Bayes was used mostly for benchmarking purposes (11 studies). Deep learning has only been used in seven studies [5], [6], [8]–[11], [22] to date. Six of these seven models employed LSTM.

The existing approaches for intake gesture detection from inertial sensor data can be divided into two groups based on the utilization of temporal context in sequential data. Approaches such as KNN and SVM do not take into account the temporal aspect of data. In contrast, approaches such as HMM and LSTM consider previous data frames to predict the state of the current data frame. The latter group have recently been more successful and gained more attention [23]. In the following, we provide an overview of the deep learning approaches that have been applied in this context, including an overview of the sensor modalities, and data preprocessing that they considered.

D. DEEP LEARNING FOR DETECTING INTAKE GESTURES

Recent studies show that deep learning approaches, and specially the combination of CNN and LSTM, are promising in detecting intake activities. Table 1 provides an overview of existing studies that have employed deep learning for intake gesture detection from inertial sensor data. As can be seen from the table, the current state-of-the-art approaches for this context divide the model into two consecutive networks: first a CNN to extract temporal features, then a LSTM to learn the temporal patterns, where the LSTM uses the CNN's output as input. In the CNN, the number of layers varies based on computational power capacity, size, and complexity of input data. Whereas, in the LSTM, the existing studies commonly used one [10], [22] or two [5], [6] layers based on the complexity of temporal patterns and the size of dataset.

Kyritsis *et al.* [5] employed an SVM for modeling sub-gestures, whose output was fed to a LSTM network to model the temporal context of inertial data. The LSTM served as a replacement for a HMM used in a previous study [24]. In a later study [10], the authors replaced the SVM with a CNN as part of an end-to-end network to detect intake gestures without using sub-gesture labels. This approach was later enhanced in their later study [6] by taking advantage of their more detailed labelling system at the sub-gesture level. Papadopoulos *et al.* [11] trained a deep network using standard learning techniques (supervised learning) and then fine-tuned the pre-trained model to a new person. The fine-tuning step was done using unlabeled samples of the new person (unsupervised learning). While six of the deep learning studies used data collected from lab settings, Kyritsis and colleagues [22] recently investigated detecting intake events from data collected in different free-living settings using a combination of CNN and LSTM.

E. DATA PREPROCESSING

Existing studies have applied a range of different data preprocessing steps before the data was fed into the deep learning model. The most common steps include (1) *smoothing* (median filter [5], [6], moving average filter [9], [22]), (2) removing the earth's *gravitational effect* on accelerometer data (quaternion representation calculated using Madgwick's algorithm [5], high-pass FIR filter [6], [10], [22]), and (3) *standardizing* the values [6], [10]. However, as shown in Table 1, there is currently no unified approach to data preprocessing and a range of different methods is applied in different studies.

Further, in addition to the three steps discussed above (smoothing, removing gravity effect, standardizing), *mirroring* is an additional data preprocessing step that some recent research applied before the other steps [22], [25]. Mirroring enables researchers to transform data by flipping left to right and vice versa [25]. This may be helpful to achieve data uniformity, for instance, to uniform inertial data into dominant vs non-dominant intake hand and account for situations where some subjects are left-handed while other subjects are right-handed. Importantly, there has been no research on the effectiveness of the data preprocessing steps of (1) mirroring, (2) smoothing, (3) removing gravity, and (4) standardization in increasing the performance of deep learning models. In the current paper, we address these gaps. [1]

III. METHODS

In order to detect intake gestures, we adopted a two-stage approach as shown in Fig. 1 (see [6], [27] for a similar approach). In *Stage 1*, we estimated the state probability of each frame being an intake frame. In *Stage 2*, we detected the intake gestures by finding the peaks in the probabilities that were higher than a certain threshold, and at least 2 seconds apart. In the following section, we provide detailed descriptions of the data preprocessing steps that we applied, the data segmentation approach that we implemented, our proposed deep learning model along with a baseline model and a benchmark model that we used for the frame-level intake detection in *Stage 1*. We also introduce our *earliest sensor data fusion* method through a dedicated CNN layer and *target matching* technique as a part of the proposed model. This section ends with a detailed description of the gesture-level intake detection in *Stage 2*.

A. DATA PREPROCESSING

In order to investigate the influence of data preprocessing on model performance, the current method contains differing implementations for the four different data preprocessing steps discussed above and as shown in Fig. 1 (i.e., mirroring, removing gravity effect, smoothing, and standardization). The details of each of these four steps are introduced in the following section.

[1] The Move 3 (G) version of the Movisens Move 3 additionally contains gyroscope (https://www.movisens.com/en/products/activity-sensor-move-3).

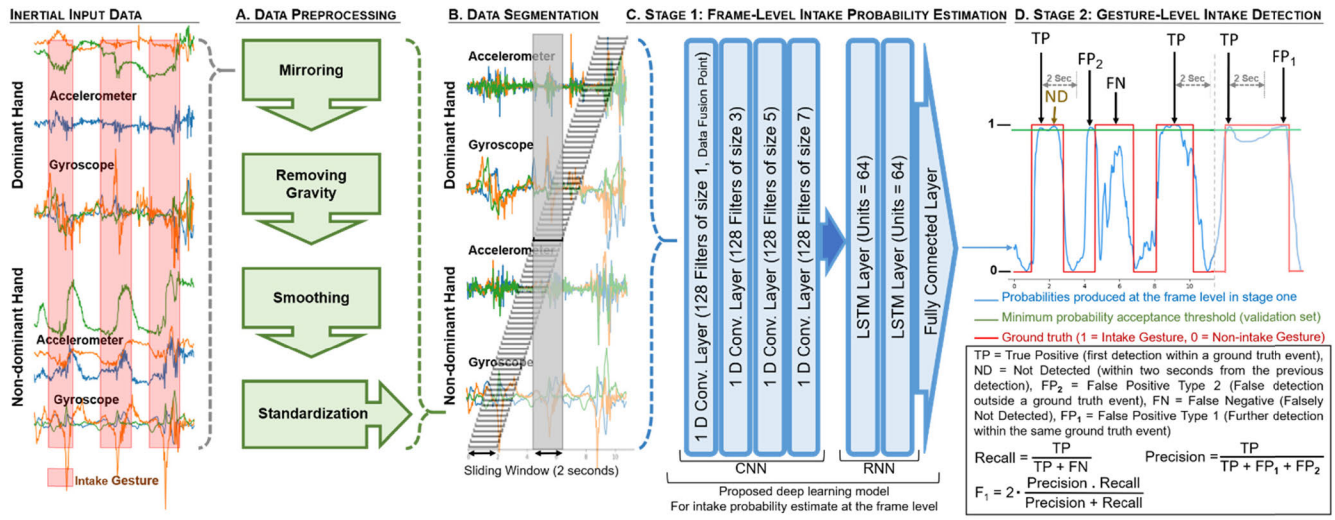


FIGURE 1. Inertial data composition, data preprocessing, data segmentation, and the two-stage approach for intake gesture detection.

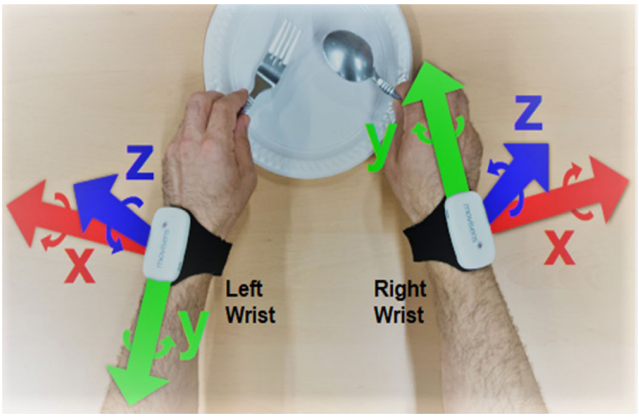


FIGURE 2. Axes and rotations of accelerometer and gyroscope sensors on the left and right wrists.

1) MIRRORING

Sensor data corresponds to the sensor's internal coordinate system. To mirror acceleration data horizontally, we flipped the sign for the x axis, which corresponds to the horizontal direction (see Fig. 2). We also flipped the signs for x and y axis to compensate for the difference in sensor orientation between left and right wrist in our experiments[2] (see [22] for a similar approach). Combined, this yields the transformation

$$\left[a'_x, a'_y, a'_z\right] = \left[-(-a_x), -a_y, a_z\right] = \left[a_x, -a_y, a_z\right]$$

For the gyroscope data, we flipped the signs of the y and z axis to mirror rotations horizontally; as before, we also flipped the signs for x and y axis to compensate for different sensor orientations. This yields

$$\left[g'_x, g'_y, g'_z\right] = \left[-g_x, -(-g)_y, -g_z\right] = \left[-g_x, g_y, -g_z\right]$$

Mirroring the sensor data horizontally (i.e., transforming data from left wrist as if it had been recorded on right wrist and vice versa) can be useful in several ways. For example, it allows achievement of data uniformity by transforming all dominant hands to be right hands, and all non-dominant hands to be left hands. It can also be used for data augmentation, similar to horizontal flipping when working with 2D images. In this study, we use mirroring to uniform input data into dominant vs non-dominant intake hand. To achieve this, we mirror the data of the left-handed participants to match them to the right-handed participants.

2) REMOVING THE GRAVITY EFFECT

Because of the Earth's gravitational force, the acceleration signal reflects (1) the acceleration due to the wrist movements of interest, and (2) the acceleration caused by earth's gravity. Removing the effect of gravity could potentially improve model performance, because the model does not need to learn this additional complexity by itself.

In order to remove the effect of the earth's gravitational field on the acceleration, we estimate a quaternion that represents the sensor's orientation relative to the earth by using sensor fusion of accelerometer and gyroscope via Madgwick's algorithm [28]. We use this quaternion to rotate the acceleration vector and then subtract the gravity vector. Since the chosen approach accounts for small errors in the sensor data, this step is operationalized before smoothing to avoid information loss. In the Supplemental Material, we provide a pseudo-code listing of the used algorithm along with a reference to the original article with the full derivation of the underlying formulas.

3) SMOOTHING

We compared a range of different smoothing methods. This includes the median (used in [5], [6]), and moving average

[2]We deliberately decided for the sensor orientation shown in Fig. 2 to ensure that all participants wear the sensors uniformly. Specifically, participants were instructed to wear the sensor such that they were able to read the label on the sensor. Another approach would have been to wear the sensors in the same direction which changes the mirroring formula for accelerometer to $\left[a'_x, a'_y, a'_z\right] = \left[-a_x, a_y, a_z\right]$ and for gyroscope to $\left[g'_x, g'_y, g'_z\right] = \left[g_x, -g_y, -g_z\right]$.

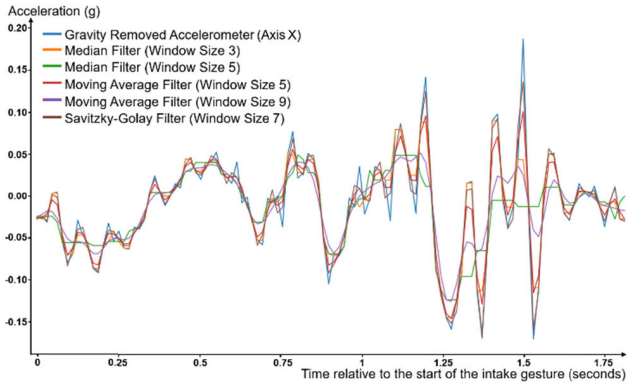


FIGURE 3. Effect of different smoothing approaches of gravity-removed accelerometer data from an intake gesture.

(used in [9]) filters that have been applied in prior works as well as a 5th order Savitzky-Golay filter [29] that has not been applied in this context so far. Based on our experiments, median filter with a window size of five frames outperformed other smoothing methods on our 64 Hz data. The general purpose of smoothing is to remove noise associated with short-term fluctuations in the sensor data (e.g. slight wrist tremor, technical sensor limitations) [30]. Fig. 3 illustrates the effect of different smoothing approaches on gravity-removed accelerometer data.

Through running multiple experiments with smoothing filters of different sizes, it was noted that choosing bigger smoothing filters distorts the data and reduces model performance. Therefore, we chose window sizes of three and five frames for median filter, five and nine frames for moving average[3] filter and seven frames for Savitzky-Golay filter to minimize the distortion effect of these filters, while they still retain the smoothing effect.

4) STANDARDIZATION

Following the common standardization process, the mean of the signal was deducted and divided by its standard deviation (see [6], [10] for a similar approach). This step is done separately for each participant and each of the 12 possible channels, that is, for each axis (x, y, and z) for each modality (accelerometer, gyroscope) and hand (left, right). This makes sure that all sensor data is unitless, using the same scale. Further, it may mitigate potential between-subjects variance due to interpersonal differences in wrist movements.

B. DATA SEGMENTATION

A temporal element is a sequence of frames to be fed to the model. Similar to [27], [31], [32], we employed a fixed overlapping sliding window [12] with a two second window size and a one frame step size, which allows to include the maximum number of temporal elements in the training data. Considering each sensor modality produces three values per reading (x, y, and z axis) and the 64 Hz sampling rate, each temporal element comprised a two-dimensional matrix consisting of 128 frames. Thereby, each frame contained three, six, or twelve values depending on what sensor modalities (accelerometer and/or gyroscope) and sensor positions (one or both hands) were considered in the model.

C. STAGE 1: DEEP LEARNING MODELS FOR FRAME-LEVEL INTAKE PROBABILITY ESTIMATION

Based on the current state-of-the-art of deep learning for intake gesture recognition, we implemented and compared the following models: (1) A CNN model as a baseline, (2) an adaptation of Kyritsis's model [6] as a benchmark, and (3) our proposed CNN-LSTM model. Table 2 provides an overview of the specifications of the three models. These models to classify frames according to our binary classification (i.e., intake vs non-intake), yielding probabilities of each frame being an intake or non-intake frame.

Training configuration: We used cross-entropy for loss calculation and the Adam optimizer for training. The dataset is naturally imbalanced as it contains more non-intake frames than intake frames. To correct this, we scaled the minibatch loss (see [27] for a similar approach). Based on experiments, we found that the proposed and baseline models performed best using an exponentially decaying, rather than a constant, learning rate. In particular, we used a learning rate starting at 3e-4 and decaying at a rate of 0.93 per epoch until it remains constant at 2e-7.[4] We also ran experiments to compare different batch sizes for input data (32, 64, 128, 256 and 512), which showed that a batch size of 256 performed best. All the above decisions were based on model performance on the validation set. To measure performance in *Stage 1*, we used unweighted average recall (UAR) of the classification categories. We evaluated the performance of each model based on UAR of intake and non-intake classification categories *at the frame level* and kept the ten best instances of each model.

1) BASELINE: CNN MODEL

As a baseline, we implement a CNN model (see e.g. [9]). The baseline model contains seven one-dimensional CNN layers with 64 filters in the first and second layers, 128 filters in the third and fourth layers, 256 filters in the fifth and sixth layers and 512 in the last CNN layer. There was a max pooling layer after each CNN layer. The model ends with a flatten and a fully connected layer with two units for the binary classification. The size of filters is kept to 6 in CNN layers. Therefore, the model considers the temporal context of data by extracting features from sequences of frames.

2) BENCHMARK: KYRITSIS'S MODEL

As a benchmark, we implemented an adaptation of the model proposed by Kyritsis *et al.* [6]. Thereby, there are two important differences between the dataset in the present work and

[3]The moving average filter introduces a delay between the smoothed data and the activity labels. This delay is equal to half the filter size, rounded down (i.e., for a window size of nine frames the delay is four frames). Hence, we moved forward the filter's output by half the window size.

[4]The original work by Kyritsis *et al.* [6] used a constant training learning rate at 1e-3. Therefore, we implemented two variants of this model, one with the original constant learning rate and one with the described exponentially decaying learning rate.

TABLE 2. Overview of the parameters and specifications of the models.

Model / Layer	Baseline: CNN		Benchmark: Kyritsis et al.		Proposed Model	
	S, U	Shape	S, U	Shape	S, U	Shape
Input Data	-, -	128×12	-, -	128×12	-, -	128×12*
Conv. 1	6, 64	128×64	6, 64	128×64	1, 128	128×128
M. Pool 1	2, -	64×64	2, -	64×64		
Conv. 2	6, 128	64×64	6, 128	64×128	3, 128	126×128
M. Pool 2	2, -	32×64	2, -	32×128		
Conv. 3	6, 128	32×128			5, 128	122×128
M. Pool 2	2, -	16×128				
Conv. 4	6, 128	16×128			7, 128	116×128
M. Pool 4	2, -	8×128				
Conv. 5	6, 256	8×256				
M. Pool 5	2, -	4×256				
Conv. 6	6, 256	4×256				
M. Pool 6	2, -	2×256				
Conv. 7	6, 512	2×512				
M. Pool 7	2, -	1×512				
Fully C. 1			-, 5	32×5		
LSTM 1			-, 64	32×64	-, 64	116×64
LSTM 2			-, 64	32×64	-, 64	116×64
Flatten 1	-, -	512				
Fully C. 2	-, 2	2	-, 2	32×2	-, 2	116×2

Note: Stride in all convolutional layers is set to one, Conv. = Convolutional, F = Number of filters, Fully C. = Fully Connected, M. Pool = Max Pooling, S = Filter's kernel size | Pooling size, Shape = Output shape, U = Number of units | Number of filters, * = The model is adapted to the shape of input data, which is (1) 128x12 when the model uses data from both hands (i.e., dominant / non-dominant hand) and both modalities (i.e., accelerometer, gyroscope), (2) 128x6 when the model uses data from one hand and both modalities, or one modality and both hands, and (3) 128x3 when the model uses data from one hand (e.g., dominant hand) and one modality (e.g., accelerometer);

the dataset used in the original work. First, the sampling frequency is 64 Hz in the current work, while it was 100 Hz in the original work. This was addressed by setting the size of convolutional filters in the CNN to 6 instead of 10 so it still corresponded to approximately 0.1 of a second of sequential input data. Second, the current dataset does not include labels of sub-gestures. In the original work, the CNN was separately trained using sub-gesture labels to produce a sub-gesture probability distribution that is inputted to the LSTM [6]. Hence, because our dataset does not include labelling for sub-gestures, we trained the entire CNN-LSTM in one step. Adding sub-gesture labels may improve the model performance.

3) PROPOSED MODEL

The proposed model contains a four-layer CNN for feature extraction and a two-layer LSTM to find the temporal patterns (see Fig. 1, Stage 1). The activation function in all CNN layers is ReLU. Each CNN layer contains 128 filters, while the filter shifts one frame at a time. Filter sizes in the first to last layers are one, three, five, and seven, respectively. The features learned by the CNN layers are used as input by the LSTM layers. The proposed model contains two LSTM layers.[5] Each LSTM layer contains 64 units, uses the hyperbolic tangent activation function, applies the sigmoid function for the recurrent step and returns the full sequence to the output. What distinguishes our model from existing ones are the proposed (1) *earliest sensor data fusion through a dedicated CNN layer* and (2) *target matching technique*, as described below.

Earliest sensor data fusion through dedicated CNN layer: We used a CNN layer with filter size one as the first CNN layer. Setting the filter size to one means that this layer considers only one frame at a time, which consists of the sensor input for that frame (i.e., twelve values from tri-axial accelerometers and gyroscopes on both wrists). Therefore, this layer is intended to specialize in fusing the features from different channels and sensors, without considering the temporal context. In contrast, the following CNN layers have filter sizes greater than one and hence specialize in learning from the temporal context.

Target matching technique: When convolving a sequence with a filter of size greater than one (without padding), the length of the resulting sequence will be shortened. Our target matching technique adjusted the label sequence accordingly. In the current study (64 Hz sample rate, 2 seconds window), the length of the temporal element is 128. If we count indices starting from one, the index of the target frame is 128 initially. It remains 128 after the first layer. The next three CNN layers apply filters with filter size three, five, and seven in ascending order. Therefore, the size of the temporal element shrinks to 126, 122, and at last 116. Since the filters shrink the temporal element from both sides equally, the index of the target frame changes to 127, 125, and at last 122. As Fig. 4 illustrates, our target matching algorithm calculates the index of the target frame and adjusts the index of *target label* accordingly. Target label is the last element of the corresponding label sequence that is relevant for model prediction.

D. STAGE 2: GESTURE-LEVEL INTAKE DETECTION

For each model, the algorithm in *Stage 2* finds local maxima based on the frame-level probabilities estimated in *Stage 1* (see [5], [6] for a similar approach). The algorithm performs a maximum search on the probabilities above a minimum probability acceptance threshold. This threshold is estimated separately for each model by finding the value that optimizes

[5]We ran several experiments with LSTM, bidirectional LSTM, as well as Gated Recurrent Units (GRU), and different numbers of layers. We chose the present model based on its performance.

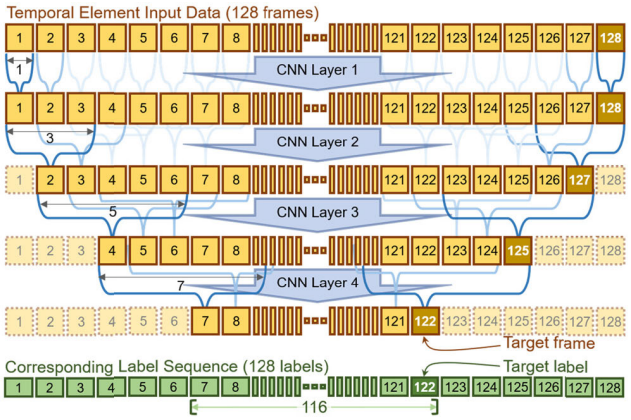


FIGURE 4. Illustration of the proposed target matching technique with the temporal element passing through CNN layers (*Stage 1*).

model performance on validation set. Local maxima that are at least two seconds apart from the previous local maximum are detected as intake gestures. Thereby, we utilized the evaluation scheme of Kyritsis *et al.* [6] (see Fig. 1, Stage 2). According to this scheme a true positive (TP) is the first correct intake detection in a ground truth event. Further detections within the same ground truth event count as false positive type 1 (FP_1). An intake detection that is not within a ground truth event is a false positive type 2 (FP_2). A ground truth event that is not detected counts as false negative (FN). Based on this, we calculated precision as the number of true positives divided by number of all detections (i.e., TP, FP_1 and FP_2), and recall as the number of true positives divided by the number of all ground truth events (i.e., TP and FN). Using these calculations, We then calculated the F_1 score *at the gesture level* as the harmonic average of precision and recall [27] (see Fig. 1). We first calculated the F_1 score on the validation set to identify the best instance for each model as the representative instance of that model. Using these representative instances, we then calculated the F_1 score on the test set to report the results.

IV. DATASET

We recruited 102 individuals through social media posts and noticeboards at the University of Newcastle. We excluded one participant due to a data collection error and another participant because they did not provide consent to their data being used by other researchers in subsequent studies. Hence, the final dataset contained 100 individuals (69 male, 31 female). 24 participants did not report their dominant intake hand. For these participants, the dominant intake hand was identified by inspecting the video recordings. The study was approved by the University of Newcastle Human Research Ethics Committee (approval number H-2017-0208).

A. DATA COLLECTION SETUP

Data was collected from both hands using wrist-worn tri-axial accelerometers and tri-axial gyroscopes at a sampling rate frequency of 64 Hz (Movisens Move 3 G). Fig. 2 shows the

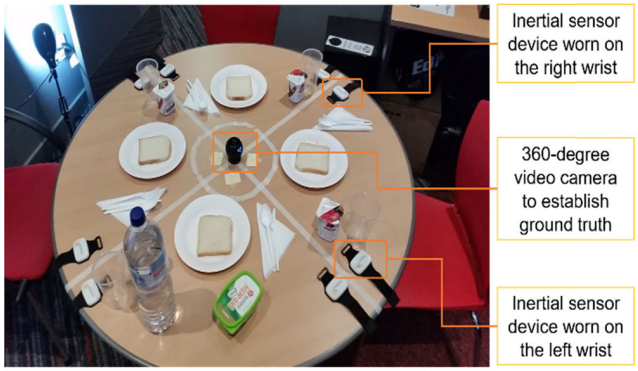


FIGURE 5. Data collection setup including wrist-worn sensors for four participants and video camera in the center of the table.

axes and rotation direction of inertial sensors used on the left and right hands. The data collection setup included a group setting of four participants who each individually consumed a standardized meal of lasagna, bread, yogurt, and water (no shared dishes). However, some sessions were conducted with two or three participants due to participant availability. Fig. 5 shows the data collection setup.

B. GROUND TRUTH AND DATA LABELING

Ground truth was established by video recording the experiments using a 360-degree camera (see Fig. 5) with a clapping method [5] used to synchronize inertial data from different sensor positions (right and left hands) and ground truth. Two research assistants annotated the collected data and cross-checked each other's work as a quality check.

C. DATASET SPLITS

We randomly split the dataset of 100 participants into a training set of 61 participants, validation set of 20 participants, and test set of 19 participants. The training set was used to train the models (*Stage 1*). The validation set was used to evaluate the trained models (*Stage 1*). It was also used to calculate the minimum probability acceptance threshold, and to select the best instance of each model (*Stage 2*). To rule out that comparisons are biased towards a particular model, the test set was only used to report the results on unseen data (*Stage 2*).

V. EXPERIMENTS AND RESULTS

In order to compare the effects of data preprocessing, sensors modalities, and sensor positions, we calculate the F_1 of the corresponding model implementations as they perform on the test set. Further, in order to statistically evaluate how different models directly compare to each other, we use pairwise comparisons based on 500 bootstrapped samples. In other words, we use bootstrapping to randomly create 500 samples of the original test set. For each model implementation, we then (1) calculate F_1 scores for each of the 500 bootstrapped samples and (2) run pairwise t-tests to directly compare individual models. Because we use the exact same 500 random samples on each model implementation, we can directly compare their performance. The results of the pairwise comparisons are shown in Tables 3–6.

TABLE 3. Effect of different data preprocessing combinations on the performance of the proposed model.

	Data Preprocessing					Pairwise comparisons (p values)							
#	Mirrored	Gravity Removed	Smoothed	Std	F_1	#1	#2	#3	#4	#5	#6	#7	#8
1	yes	yes	-	yes	.778	-							
2	yes	yes	yes	yes	.776	<.001	-						
3	yes	yes	yes	-	.772	<.001	<.001	-					
4	yes	yes	-	-	.763	<.001	<.001	<.001	-				
5	yes	-	-	yes	.752	<.001	<.001	<.001	<.001	-			
6	yes	-	-	-	.752	<.001	<.001	<.001	<.001	.870	-		
7	yes	-	yes	yes	.745	<.001	<.001	<.001	<.001	<.001	<.001	-	
8	-	-	-	-	.743	<.001	<.001	<.001	<.001	<.001	<.001	.340	-

Note: Ordered by model performance as measured by F_1, the smoothing filter is median filter (window size = 3), std = standardized, p-values refer to the significance of pairwise comparisons based on 500 randomly bootstrapped samples from the test set.

TABLE 4. Effect of different smoothing filters on the performance of the proposed model.

	Smoothing					Pairwise comparisons (p values)				
#	Mirrored	Gravity Removed	Smoothing Filter (Window Size)	Std	F_1	#1	#2	#3	#4	#5
1	yes	yes	Median (3)	yes	.776	-				
2	yes	yes	Moving (9)	yes	.773	<.001	-			
3	yes	yes	Sav_gol (7)	yes	.766	<.001	<.001	-		
4	yes	yes	Median (5)	yes	.757	<.001	<.001	<.001	-	
5	yes	yes	Moving (5)	yes	.757	<.001	<.001	<.001	.700	-

Note: Ordered by model performance as measured by F_1, moving = moving average filter, sav_gol = 5th order Savitzky-Golay filter, std = standardized, p-values refer to the significance of pairwise comparisons using paired t-tests based on 500 randomly bootstrapped samples from the test set.

A. DATA PREPROCESSING

As can be seen in Table 3, the experiments indicate that the best data preprocessing results can be achieved by combining mirroring, removing the gravity effect, and standardization. Of the different smoothing methods used in the experiments (see Table 4), median filter (window size = 3, $F_1 = .776$) outperformed moving average filter (window size = 9, $F_1 = .773$) and Savitzky-Golay (window size = 7, $F_1 = .766$), while no use of smoothing achieved the best result ($F_1 = .778$).

Table 4 reveals more details on the effect of using different smoothing filters (i.e., median, moving average, and Savitzky-Golay) combined with other data preprocessing steps on the performance of the proposed model.

B. SENSOR MODALITIES AND SENSOR POSITIONS

The proposed model was adapted for three-channel and six-channel input data. Therefore, we were able to train and test it with the best preprocessed data (i.e., mirrored, gravity effect removed and standardized) from different sensor modalities and sensor positions combinations listed in Table 5.

C. TARGET MATCHING

To evaluate the impact of the target matching technique, we also ran an implementation of the model without target matching. The results show that the model without target matching yields lower model performance ($F_1 = .733$) than the model with target matching ($F_1 = .733$). Based on pairwise comparisons of these two model implementations using paired t-tests on 500 randomly generated samples from the test set, we can confirm that the difference in model performance is significant ($p < .001$).

D. MODEL BENCHMARKING

We implemented two variations of the benchmark model by Kyritsis [6], namely one with the original constant learning rate and one with the exponentially decaying learning rate technique. Table 6 shows results of testing these two models along with the baseline and proposed models. However, in this comparison it is important to note that in the original work by Kyritsis [6] the CNN was trained separately using sub-gesture annotations which are not available for our data.

E. WHERE DO THE MODELS STRUGGLE?

To identify limitations of the model to detect eating gestures, we investigated (1) types of intake gestures the model struggled to detect (false negatives, see Fig. 6) and (2) non-intake hand gestures the model tended to detect as an intake event (false positives, see Fig. 7).

In terms of *false negatives*, some types of intake events were more difficult than others for the model to detect. This could be because these intake events occur only occasionally (e.g., licking finger, licking food from knife, or eating with knife; see a-c in Fig. 6). Therefore, the model sees less examples of these intake events through training.

TABLE 5. Results of using different sensor modalities and sensor positions.

	Sensor Configuration			Pairwise comparisons (p values)								
#	Sensor Modality	Sensor Position	F_1	#1	#2	#3	#4	#5	#6	#7	#8	#9
1	Accelerometer and Gyroscope	Both	.778	-								
2	Accelerometer and Gyroscope	Dominant	.654	<.001	-							
3	Accelerometer and Gyroscope	Non-dominant	.497	<.001	<.001	-						
4	Accelerometer	Both	.682	<.001	<.001	<.001	-					
5	Accelerometer	Dominant	.583	<.001	<.001	<.001	<.001	-				
6	Accelerometer	Non-dominant	.351	<.001	<.001	<.001	<.001	<.001	-			
7	Gyroscope	Both	.771	<.001	<.001	<.001	<.001	<.001	<.001	-		
8	Gyroscope	Dominant	.620	<.001	<.001	<.001	<.001	<.001	<.001	<.001	-	
9	Gyroscope	Non-dominant	.454	<.001	<.001	<.001	<.001	<.001	<.001	<.001	<.001	-

Note: Ordered by model performance as measured by F_1 in each sensor modality category, p-values refer to the significance of pairwise comparisons using paired t-tests based on 500 randomly bootstrapped samples from the test set.

TABLE 6. Results of benchmarking against other models.

	Model Benchmarking		Pairwise comparisons (p values)			
#	Model (learning rate)	F_1	#1	#2	#3	#4
1	Proposed model (exponentially decaying)	.778	-			
2	Benchmark: Kyritsis et al. [6] (constant)	.762	<.001	-		
3	Baseline CNN (exponentially decaying)	.758	<.001	<.001	-	
4	Benchmark: Kyritsis et al. [6] (exponentially decaying)	.757	<.001	<.001	.080	-

Note: Ordered by model performance as measured by F_1, p-values refer to the significance of pairwise comparisons using paired t-tests based on 500 randomly bootstrapped samples from the test set.

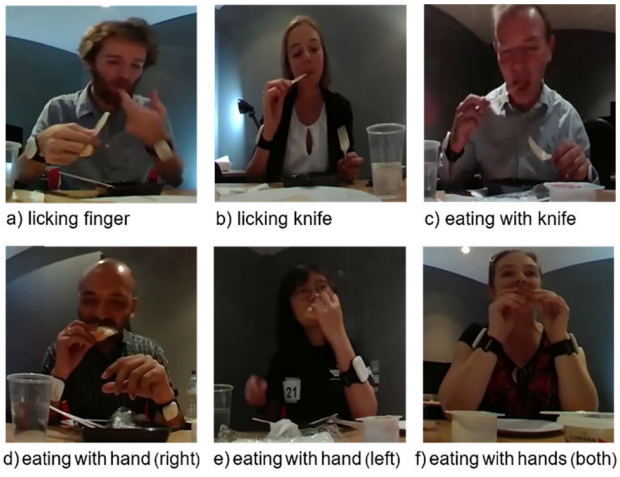


FIGURE 6. Examples of intake events causing false negatives in test set.

FIGURE 7. Examples of hand gestures causing false positives in test set.

Another reason may be that some intake events can be performed with shorter hand to mouth movements and therefore may involve less hand gestures (e.g., moving head towards food, or having multiple bites from a piece of bread; see d-f in Fig. 6).

In terms of *false positives*, the hand gestures misclassified as eating mainly pertain to two categories. The first category refers to hand movements that occur when a participant touches their face (e.g., nose, mouth, glasses, or forehead; see a-f in Fig. 7). The second category contains hand movements that happen when a participant delays an intake gesture (e.g. due to a conversation or blowing the bite) or initiates but does not complete the intake gesture (e.g., because of food being too hot or food falling off the cutlery; see g-i in Fig. 7).

Table 7 provides an overview of the recall levels achieved for different intake categories (i.e., eat, or drink), hand

TABLE 7. False negatives and true positives for different intake events.

Label	Sub label	TP	FN	Recall
Intake	Drink	69	9	.885
	Eat	674	185	.785
Hand involved	Dominant	551	140	.797
	Non-dominant	181	46	.797
	Both	11	8	.579
Eating utensil	Cup	69	9	.885
	Spoon	306	57	.843
	Fork	279	81	.775
	Hand	86	36	.705
	Knife	2	4	.333
	Finger	1	7	.125
Overall	-	743	194	.793

Note: Based on the test set; ordered by recall in each label category.

involved (i.e., dominant, non-dominant, or both), and eating utensil (i.e., spoon, fork, cup, hand, knife, or finger). Results indicate that the least detected eating utensil were fingers (Recall =.125) and knife (Recall =.333). In total, the number of true positives, false negatives, false positives type 1, and false positives type 2 were 743, 194, 41, and 188, respectively. Therefore, precision was.764 while recall was.793.

VI. DISCUSSION AND CONCLUSIONS

Using deep learning to detect intake gestures from inertial sensor data holds great potential for a wide range of application areas (e.g., life-logging, patient monitoring; [3], [4]). However, at this stage, only few studies have applied deep learning to this task, with a lack of research on the effects of data preprocessing, sensor modalities, and sensor positions on the performance of deep learning models. In the current study, we set out to address this gap by clarifying the role of these factors with a dataset of 100 participants.

In terms of *data preprocessing*, a combination of mirroring, removing gravity effects, and standardization improved model performance ($F_1 = .778$), while even the best-performing smoothing (median filter, window size = 3) had adverse effects ($F_1 = .776$). Even though the difference in F_1 is relatively small ($\Delta F_1 = .002$), it is notable that smoothing was detrimental to model performance, particularly because smoothing was frequently applied to this task in machine learning approaches before the application of deep learning (see [4] for a review). A possible explanation for the detrimental impact of smoothing is that deep learning models are better able to utilize the rich information provided in the inertial data than previous architectures. In general, the purpose of smoothing is to remove noise associated with short-term fluctuations in the signal data (e.g. slight wrist tremor, technical limitations of the sensor). However, applying smoothing inevitably also removes information related to the activity. Given that hand-to-mouth movements are a natural daily activity that is critical for human survival, individuals without movement impairments are able to perform this task effortlessly, leading to little noise in the data. At the same time, the advances in sensor technology have improved sensor accuracy. Against this backdrop, smoothing may do more harm than good, and deep architectures are capable to utilize the rich information. Following this line of thought, it would be interesting to further explore the impact of smoothing for populations that exhibit higher degrees of noise in intake gesture movements (e.g. elderly users, small children). Also, it is important to note that our results are based on a sampling rate of 64 Hz and hence the results with regard to smoothing may need to be re-evaluated in datasets with different sampling rates.

Our results show that using the proposed *target matching* technique increased model performance by 4.18% (i.e., F_1 of.733 vs F_1 of.778) in the proposed model. This can be explained by the notion that with target matching, the model learns to use the temporal context of data to predict the state of the very target frame instead of a frame in the neighborhood of the target frame. Further, in CNNs, a convolutional layer is generally followed by a pooling layer (e.g., in the benchmark model [5]). Widely used in the context of image processing, pooling layers assist in (1) making the network invariant to local translation, (2) reducing the computational complexity by downsampling the output of the previous layer to reduce the statistical burden on the next layer, and (3) handling inputs of varying size [13]. However, replacing pooling with a convolutional layer has shown no loss in accuracy on image recognition tasks [33]. During network design our experiments indicated that pooling layers were not beneficial to model performance, which may be due to lower dimensionality and higher density of inertial data compared to image data. Similarly, there are examples of CNN-LSTM models for wearable sensor-based activity recognition that do not include pooling layers [18] or where a pooling layer only comes after the first of two convolutional layers [34].

Interestingly, despite not including pooling layers which are known to reduce the computational complexity, the number of floating point operations (FLOPs) required to run inference in real-time at the sensor's sampling rate of 64 Hz with the proposed model is within the capabilities of current smartphone devices. Specifically, we found that our implementation of the proposed model requires 3.8 GFLOP/s, which is higher than the benchmark model's 0.5 GFLOP/s but lower than the capabilities of GPUs in mobile devices (e.g. 727 GFLOP/s for Adreno 630 [35] used in Google Pixel 3, Nokia 9 PureView, and Sony Xperia XZ2). By processing the inertial data on the user's mobile phone, one could design real-time interventions that support and encourage individuals to maintain a healthier diet [36].

As for *sensor modalities and sensor positions*, this is one of the first deep learning studies to consider inertial intake data from both hands. Based on multiple experiments, the best model performance was achieved by using earliest fusion (i.e., dedicating the first CNN layer to data fusion at the frame level). This was achieved by configuring this layer to only

convolve data from one frame at a time. Further, the results show that models using both hands ($F_1 = .778$) are essential for top model performance compared to models using the dominant intake hand only ($F_1 = .654$). It is important to note that collecting data from both hands might not be feasible in everyday environments, particularly because users tend to only wear one smart device on their wrist. Our results show that if data can only be collected from one hand then it is critical to use the dominant eating hand as model performance substantially drops if only the non-dominant intake hand is available ($F_1 = .497$). However, in more controlled settings such as aged care, hospital, and field studies, it might be feasible to collect data from both hands. For models with data from both hands, using both gyroscope and accelerometer data ($F_1 = .778$) outperforms using only gyroscope ($F_1 = .771$) or only accelerometer data ($F_1 = .682$). However, using both modalities achieves only a 0.9% increase in performance compared to a model using only gyroscope. Therefore, in a limited resource environment (e.g. energy constraints in multi-day recording settings), using only a gyroscope may still achieve acceptable performance. One application area of these results could be in settings with resource constraints (e.g., extended periods of data collection and limited energy supply in low-income countries [7]). However, the energy saving effect of removing accelerometer may be marginal.[6]

In terms of future work, it is noteworthy that deep learning has only recently been used for food intake gesture detection from wrist-mounted inertial sensors [4]. As a result, there is a lack of pre-trained models in this area which limits the possibility of warm-starting (i.e., initializing the deep network using the weights from an already trained model). To the best of our knowledge, pre-trained deep learning models also do not exist in the field of human activity recognition based on inertial data from wearable sensors. This eliminates the possibility of fine-tuning a pre-trained model. Research using other modalities has shown that warm-starting can be effective in improving model performance (e.g. video data [27], [37]). Hence, the creation of pre-trained models appears an interesting avenue for future research in this area. Another interesting aspect of temporal data is the sampling frequency, which varies across different inertial measurement devices. Understanding the optimal sampling rate is important to further improve model performance [38]. Thereby, it is important to note though that changing the sampling frequency inherently changes the temporal structure of the network, which is essential to consider to allow for an adequate model comparison. Finally, another area for future work may be the extension of the sliding window beyond 2 seconds. Longer sequential input data may help the model to identify eating gestures without overfitting the model. In gesture-level intake detection (*Stage 2*), we followed the evaluation scheme introduced in [6] to ensure that our approach is comparable to the current state-of-the-are approaches. This could be enhanced in future research to improve the model performance. For instance, introducing a maximum probability acceptance threshold that the probabilities must drop below between two detections may reduce false positive type 1 (FP_1).

[6]According to the manufacturer, the energy consumption of running the employed sensor device with only the gyroscope activated is approximately 450uA, compared to 85uA when only the accelerometer is activated. However, when comparing the gyroscope used in this study with earlier sensor generations one can observe an overall trend towards higher energy efficiency. For instance, the data sheet for the Bosch BMI055 from 2014 reports a consumption of about 5000uA (https://bosch-sensortec.com).

REFERENCES

[1] A. F. Subar, L. S. Freedman, J. A. Tooze, S. I. Kirkpatrick, C. Boushey, M. L. Neuhouser, F. E. Thompson, N. Potischman, P. M. Guenther, V. Tarasuk, J. Reedy, and S. M. Krebs-Smith, "Addressing current criticism regarding the value of self-report dietary data," *J. Nutrition*, vol. 145, no. 12, pp. 2639–2645, 2015, doi: 10.3945/jn.115.219634.

[2] O. Amft and G. Tröster, "Recognition of dietary activity events using on-body sensors," *Artif. Intell. Med.*, vol. 42, no. 2, pp. 121–136, 2008, doi: 10.1016/j.artmed.2007.11.007.

[3] Y. Dong, A. Hoover, and E. Muth, "A device for detecting and counting bites of food taken by a person during eating," in *Proc. IEEE Int. Conf. Bioinf. Biomed.*, 2009, pp. 265–268, doi: 10.1109/BIBM.2009.29.

[4] H. Heydarian, M. Adam, T. Burrows, C. Collins, and M. E. Rollo, "Assessing eating behaviour using upper limb mounted motion sensors: A systematic review," *Nutrients*, vol. 11, no. 5, pp. 1–25, 2019, doi: 10.3390/nu11051168.

[5] K. Kyritsis, C. Diou, and A. Delopoulos, "Food intake detection from inertial sensors using LSTM networks," in *Proc. Int. Conf. Image Anal. Process.*, 2017, pp. 411–418, doi: 10.1007/978-3-319-70742-6_39.

[6] K. Kyritsis, C. Diou, and A. Delopoulos, "Modeling wrist micromovements to measure in-meal eating behavior from inertial sensor data," *IEEE J. Biomed. Health Informat.*, vol. 23, no. 6, pp. 2325–2334, 2019, doi: 10.1109/JBHI.2019.2892011.

[7] T. Burrows, C. Collins, M. Adam, K. Duncanson, and M. Rollo, "Dietary assessment of shared plate eating: A missing link," *Nutrients*, vol. 11, no. 4, pp. 1–14, 2019, doi: 10.3390/nu11040789.

[8] D. O. Anderez, A. Lotfi, and C. Langensiepen, "A hierarchical approach in food and drink intake recognition using wearable inertial sensors," in *Proc. 11th Pervasive Technol. Rel. Assistive Environ. Conf.*, 2018, pp. 552–557, doi: 10.1145/3197768.3201542.

[9] J. Cho and A. Choi, "Asian-style food intake pattern estimation based on convolutional neural network," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, 2018, pp. 1–2, doi: 10.1109/ICCE.2018.8326311.

[10] K. Kyritsis, C. Diou, and A. Delopoulos, "End-to-end learning for measuring in-meal eating behavior from a smartwatch," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, 2018, pp. 5511–5514, doi: 10.1109/EMBC.2018.8513627.

[11] A. Papadopoulos, K. Kyritsis, I. Sarafis, and A. Delopoulos, "Personalised meal eating behaviour analysis via semi-supervised learning," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, 2018, pp. 4768–4771, doi: 10.1109/EMBC.2018.8513174.

[12] A. Dehghani, O. Sarbishei, T. Glatard, and E. Shihab, "A quantitative comparison of overlapping and non-overlapping sliding windows for human activity recognition using inertial sensors," *Sensors*, vol. 19, no. 22, pp. 1–19, 2019, doi: 10.3390/s19225026.

[13] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

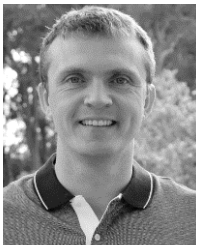
[14] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013, doi: 10.1109/TPAMI.2013.50.

[15] F. Moya Rueda, R. Grzeszick, G. Fink, S. Feldhorst, and M. ten Hompel, "Convolutional neural networks for human activity recognition using body-worn sensors," *Informatics*, vol. 5, no. 2, pp. 1–17, 2018, doi: 10.3390/informatics5020026.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
[17] Y. Chen and Y. Xue, "A deep learning approach to human activity recognition based on single accelerometer," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2015, pp. 1488–1492, doi: 10.1109/SMC.2015.263.
[18] F. Ordóñez and D. Roggen, "Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, pp. 1–25, 2016, doi: 10.3390/s16010115.
[19] N. Y. Hammerla, S. Halloran, and T. Ploetz, "Deep, convolutional, and recurrent models for human activity recognition using wearables," in *Proc. Int. Jt. Conf. Artif. Intell. (IJCAI)*, 2016, pp. 1533–1540. [Online]. Available: http://arxiv.org/abs/1604.08880
[20] M. M. Hassan, M. Z. Uddin, A. Mohamed, and A. Almogren, "A robust human activity recognition system using smartphone sensors and deep learning," *Future Gener. Comput. Syst.*, vol. 81, pp. 307–313, Apr. 2018, doi: 10.1016/j.future.2017.11.029.
[21] C. A. Ronao and S.-B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert Syst. Appl.*, vol. 59, pp. 235–244, Oct. 2016, doi: 10.1016/j.eswa.2016.04.032.
[22] K. Kyritsis, C. Diou, and A. Delopoulos, "A data driven end-to-end approach for in-the-wild monitoring of eating behavior using smartwatches," *IEEE J. Biomed. Health Informat.*, early access, Apr. 3, 2020, doi: 10.1109/JBHI.2020.2984907.
[23] R. I. Ramos-Garcia and A. W. Hoover, "A study of temporal action sequencing during consumption of a meal," in *Proc. Int. Conf. Bioinf., Comput. Biol. Biomed. Informat. (BCB)*, 2007, pp. 68–75, doi: 10.1145/2506583.2506596.
[24] K. Kyritsis, C. L. Tatli, C. Diou, and A. Delopoulos, "Automated analysis of in meal eating behavior using a commercial wristband IMU sensor," in *Proc. 39th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2017, pp. 2843–2846, doi: 10.1109/EMBC.2017.8037449.
[25] *The Food Intake Cycle (FIC) Dataset | Multimedia Understanding Group*. Accessed: Feb. 25, 2020. [Online]. Available: https://mug.ee.auth.gr/intake-cycle-detection/
[26] M. Mirtchouk, D. Lustig, A. Smith, I. Ching, M. Zheng, and S. Kleinberg, "Recognizing eating from body-worn sensors: Combining free-living and laboratory data," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 1, no. 3, pp. 1–20, Sep. 2017, doi: 10.1145/3131894.
[27] P. V. Rouast and M. T. P. Adam, "Learning deep representations for video-based intake gesture detection," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 6, pp. 1727–1737, Jun. 2020, doi: 10.1109/JBHI.2019.2942845.
[28] S. O. H. Madgwick, "An efficient orientation filter for inertial and inertial/magnetic sensor arrays," Univ. Bristol, Bristol, U.K., Tech. Rep. 25, 2010.
[29] *Scipy.Signal.Savgol_Filter–SciPy v0.16.1 Reference Guide*. Accessed: Feb. 13, 2020. [Online]. Available: https://docs.scipy.org/doc/scipy-0.16.1/reference/generated/scipy.signal.savgol_filter.html
[30] A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares Procedures.," *Anal. Chem.*, vol. 36, no. 8, pp. 1627–1639, Jul. 1964, doi: 10.1021/ac60214a047.
[31] P. Rivera, E. Valarezo, M.-T. Choi, and T.-S. Kim, "Recognition of human hand activities based on a single wrist IMU using recurrent neural networks," *Int. J. Pharma Med. Biol. Sci.*, vol. 6, no. 4, pp. 114–118, 2017, doi: 10.18178/ijpmbs.6.4.114-118.
[32] S. L. Lau and K. David, "Movement recognition using the accelerometer in smartphones," in *Proc. IEEE Future Netw. Mobile Summit*, Florence, Italy, Jun. 2010, pp. 1–9. [Online]. Available: https://ieeexplore.ieee.org/document/5722356.
[33] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," in *Proc. 3rd Int. Conf. Learn. Represent.*, 2015, pp. 1–14. [Online]. Available: http://arxiv.org/abs/1412.6806
[34] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 4580–4584, doi: 10.1109/ICASSP.2015.7178838.
[35] A. Ignatov, R. Timofte, W. Chou, K. Wang, M. Wu, T. Hartley, and L. Van Gool, "AI benchmark: Running deep neural networks on Android smartphones," in *Proc. Eur. Conf. Comput. Vis.*, vol. 11133, 2018, pp. 288–314. [Online]. Available: http://arxiv.org/abs/1810.01109
[36] T. J. Noorbergen, M. T. P. Adam, J. R. Attia, D. J. Cornforth, and M. Minichiello, "Exploring the design of mHealth systems for health behavior change using mobile biosensors," *Commun. Assoc. Inf. Syst.*, vol. 44, no. 1, pp. 1–37, 2019.
[37] P. Rouast, M. Adam, T. Burrows, and R. Chiong, "Using deep learning and 360 video to detect eating behavior for user assistance systems," in *Proc. Eur. Conf. Inf. Syst.*, 2018, pp. 1–11. [Online]. Available: https://aisel.aisnet.org/ecis2018_rp/101
[38] A. Khan, N. Hammerla, S. Mellor, and T. Plötz, "Optimising sampling rates for accelerometer-based human activity recognition," *Pattern Recognit. Lett.*, vol. 73, pp. 33–40, 2016, doi: 10.1016/j.patrec.2016.01.001.

HAMID HEYDARIAN received the B.Sc. degree in computer engineering (software) from Kharazmi University, Iran, in 2002. He is currently pursuing the Ph.D. degree in information technology at The University of Newcastle (UON), Australia. He is also a Senior Software Developer and a casual academic at UON. His research interests include inertial signal processing using deep learning and its related applications in dietary intake assessment and passive dietary monitoring.

PHILIPP V. ROUAST (Member, IEEE) received the B.Sc. and M.Sc. degrees in industrial engineering from the Karlsruhe Institute of Technology, Germany, in 2013 and 2016, respectively. He is currently pursuing the Ph.D. degree in information systems with The University of Newcastle (UON), Australia. He is also a Graduate Research Assistant at UON. His research interests include deep learning, affective computing, HCI, and related applications of computer vision.



MARC T. P. ADAM received the undergraduate degree in computer science from the University of Applied Sciences Würzburg, Germany, and the Ph.D. degree in information systems from the Karlsruhe Institute of Technology, Germany. He is currently an Associate Professor in computing and information technology with The University of Newcastle, Australia. His researches into the interplay of users' cognition and affects in human–computer interaction. He is a founding member of the Society for NeuroIS.

TRACY BURROWS is currently an Associate Professor in nutrition and dietetics at The University of Newcastle, and also a Researcher at the Hunter Medical Research Institute. She is currently a National Health and Medical Research Fellow. Her research areas focus on dietary assessment, eating behaviors, in addition to the management of overweight, obesity, and addictive eating.

CLARE E. COLLINS is currently a Professor in nutrition and dietetics with the School of Health Sciences and Priority Research Centre for Physical Activity and Nutrition, The University of Newcastle. She holds a National Health and Medical Research Council of Australia and also a Faculty of Health and Medicine Gladys M Brawn Senior Research Fellowships. She is a Fellow of the Australian Academy of Health and Medical Sciences, the Nutrition Society of Australia, the Dietitians Association of Australia, and the Royal Society of NSW. Her research focuses on using technology for personalized dietary assessment and nutrition management based on lifestage and chronic disease risk.

MEGAN E. ROLLO received the BAppSci, BHlthSci(Nutr&Diet), and Ph.D. degrees from the Queensland University of Technology, Australia. She is currently a Research Fellow in nutrition and dietetics with the School of Health Sciences and Priority Research Centre for Physical Activity and Nutrition, The University of Newcastle, Australia. She has research interests in technology-assisted dietary assessment and personalized behavioral nutrition interventions.