WAGENINGEN
UNIVERSITY & RESEARCH

Eat-Radar: Continuous Fine-Grained Intake Gesture Detection Using FMCW Radar and 3D Temporal Convolutional Network With Attention

IEEE Journal of Biomedical and Health Informatics

Wang, Chunzhuo; Sunil Kumar, T.; De Raedt, Walter; Camps, Guido; Hallez, Hans et al

https://doi.org/10.1109/JBHI.2023.3339703

This publication is made publicly available in the institutional repository of Wageningen University and Research, under the terms of article 25fa of the Dutch Copyright Act, also known as the Amendment Taverne.

Article 25fa states that the author of a short scientific work funded either wholly or partially by Dutch public funds is entitled to make that work publicly available for no consideration following a reasonable period of time after the work was first published, provided that clear reference is made to the source of the first publication of the work.

This publication is distributed using the principles as determined in the Association of Universities in the Netherlands (VSNU) 'Article 25fa implementation' project. According to these principles research outputs of researchers employed by Dutch Universities that comply with the legal requirements of Article 25fa of the Dutch Copyright Act are distributed online and free of cost or other barriers in institutional repositories. Research outputs are distributed six months after their first online publication in the original published version and with proper attribution to the source of the original publication.

You are permitted to download and use the publication for personal purposes. All rights remain with the author(s) and / or copyright owner(s) of this work. Any use of the publication or parts of it other than authorised under article 25fa of the Dutch Copyright act is prohibited. Wageningen University & Research and the author(s) of this publication shall not be held responsible or liable for any damages resulting from your (re)use of this publication.

For questions regarding the public availability of this publication please contact openaccess.library@wur.nl

Eat-Radar: Continuous Fine-Grained Intake Gesture Detection Using FMCW Radar and 3D Temporal Convolutional Network With Attention

Chunzhuo Wang, *Graduate Student Member, IEEE*, T. Sunil Kumar, *Member, IEEE*, Walter De Raedt, Guido Camps, Hans Hallez, *Member, IEEE*, and Bart Vanrumste, *Senior Member, IEEE*

***Abstract*—Unhealthy dietary habits are considered as the primary cause of various chronic diseases, including obesity and diabetes. The automatic food intake monitoring system has the potential to improve the quality of life (QoL) of people with diet-related diseases through dietary assessment. In this work, we propose a novel contactless radar-based approach for food intake monitoring. Specifically, a Frequency Modulated Continuous Wave (FMCW) radar sensor is employed to recognize fine-grained eating and drinking gestures. The fine-grained eating/drinking gesture contains a series of movements from raising the hand to the mouth until putting away the hand from the mouth. A 3D temporal convolutional network with self-attention (3D-TCN-Att) is developed to detect and segment eating and drinking gestures in meal sessions by processing the Range-Doppler Cube (RD Cube). Unlike previous radar-based research, this work collects data in continuous meal sessions (more realistic scenarios). We create a public dataset comprising 70 meal sessions (4,132 eating gestures and 893 drinking gestures) from 70 participants with a total duration of 1,155 minutes. Four eating styles (fork & knife, chopsticks, spoon, hand) are included in this dataset. To validate the performance of the proposed approach, seven-fold cross-validation method is applied. The 3D-TCN-Att model achieves a segmental F1-score of 0.896 and 0.868 for eating and drinking gestures, respectively. The results of the proposed approach indicate the feasibility of using radar for fine-grained eating and drinking gesture detection and segmentation in meal sessions.**

***Index Terms*—FMCW radar, human activity recognition, food intake monitoring, eating gesture detection, deep learning.**

Manuscript received 18 November 2022; revised 28 June 2023 and 25 October 2023; accepted 29 November 2023. Date of publication 5 December 2023; date of current version 6 February 2024. This work was supported in part by KU Leuven under Grant C3/20/016, in part by Good Brother Project (Network on Privacy-Aware Audio- and Video-Based Applications for Active and Assisted Living with COST Action 19121) and in part by China Scholarship Council (CSC) under Grant 202007650018. *(Corresponding author: Chunzhuo Wang.)*

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Institutional Review Board (IRB) of KU Leuven under Application No. G-2021-4025-R4.

Chunzhuo Wang is with e-Media Research Lab, 3000 Leuven, Belgium, also with ESAT-STADIUS Division, KU Leuven, 3000 Leuven, Belgium, and also with Life Science Department, IMEC, 3001 Heverlee, Belgium (e-mail: chunzhuo.wang@kuleuven.be).

T. Sunil Kumar is with the Vellore Institute of Technology, Chennai 600127, India (e-mail: telagamsetti.sunilkumar@vit.ac.in).

Walter De Raedt is with Life Science Department, IMEC, 3001 Heverlee, Belgium (e-mail: walter.deraedt@imec.be).

Guido Camps is with the Division of Human Nutrition and Health, Department of Agrotechnology and Food Sciences, Wageningen University and Research, 6700EA Wageningen, Netherlands, and also with OnePlanet Research Center, 6708WE Wageningen, The Netherlands (e-mail: guido.camps@wur.nl).

Hans Hallez is with M-Group, DistriNet, Department of Computer Science, KU Leuven, 8200 Sint-Michiels, Belgium (e-mail: hans.hallez@kuleuven.be).

Bart Vanrumste is with e-Media Research Lab, 3000 Leuven, Belgium, and also with ESAT-STADIUS Division, KU Leuven, 3000 Leuven, Belgium (e-mail: bart.vanrumste@kuleuven.be).

This article has supplementary downloadable material available at https://doi.org/10.1109/JBHI.2023.3339703, provided by the authors.

Digital Object Identifier 10.1109/JBHI.2023.3339703

I. Introduction

UNHEALTHY dietary habits are closely related to multiple chronic health problems such as overweight, diabetes, malnutrition, eating disorder, and cardiovascular diseases [1]. These diet-related diseases can not only affect individuals' daily life but also result in substantial financial and societal costs [2]. According to WHO reports, the overweight and obesity have been considered as a global public health problem affecting people's quality of life (QoL) [3], and imbalanced diet is the primary cause of this issue. In order to keep a healthy diet, it is necessary to track dietary intake.

Recently, different automatic food intake monitoring systems have been proposed for this purpose. These solutions can be categorized into 3 types: video-based, wearable sensor-based, and portable sensor-based approaches. The video-based method [4], [5], [6] uses cameras to detect eating gestures. The wearable sensor-based approach mainly utilizes an inertial sensor [7], [8], [9], a miniature camera [10], an acoustic sensor [11] or integrates multiple sensors [12] to monitor hand-to-mouth movements or chewing movements. The smart plate [13], the dining tray [14], and the smart table surface [15] are categorized into the portable sensor-based approach. Existing methods have been proven feasible for food intake monitoring; however, some disadvantages can emerge. The wearable sensors require precise placement, which can be uncomfortable for long-term duration. Battery concerns are also associated with such sensor types. As for video-based methods, issues related to lighting conditions and privacy may arise. In order to account for these drawbacks, we propose a contactless, privacy-preserving radar-based approach for in-home food intake monitoring.

2168-2194 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Wageningen UR. Downloaded on March 01,2024 at 10:23:46 UTC from IEEE Xplore. Restrictions apply.

Recently, radar, as an emerging technology, combined with machine learning, has been gradually applied to a variety of scenarios, such as health monitoring [16], [17], and human activity recognition (HAR) [18]. Much radar-based HAR research in indoor/kitchen scenarios has been conducted [19], [20], [21]. Luo et al. [19] used the Doppler-Time map (DT map) from two radars and a convolutional neural network (CNN) to recognize 15 activities (eating activity included) from 3 participants in a kitchen environment, with an overall classification rate of 92.8%. Gorij et al. [20] deployed two radars to classify 8 activities from 12 participants. They manually extracted features and used a Random Forest (RF) classifier, with a weighted F1-score of 93.63%. Maitre et al. [21] utilized Range-Time maps (RT map) from three ultra-wideband (UWB) radars and applied a CNN and long-short-term-memory network combined model (CNN-LSTM) to recognize 15 different activities from 10 participants, yielding an F1-score of 93% for eating activity detection. It should be noted that the above research mainly focused on multi-activities recognition. For each activity, they can classify the session/period coarsely. Specifically, they recognized the eating session from the start to the end, but did not recognize fine-grained eating gestures.

The fine-grained eating gesture recognition includes details such as how many eating gestures happened in that period (counting the number of eating/drinking gestures) and the duration of each detected gesture. In this study, the eating/drinking gesture is defined as the movement from raising the left/right hand to the mouth with the fork/knife/spoon/chopsticks/containers until putting away the hand from the mouth.

Xie et al. [22] were the first to explore the potential of using mmWave Frequency Modulated Continuous Wave (FMCW) radar sensors for fine-grained eating behavior monitoring. They collected eating gestures with five classes (fork, fork & knife, spoon, chopstick, and bare hand) from 6 participants. They detected eating gestures by searching the repetitive pattern of hand and arm movements that bring food from the table to mouth. Once an eating gesture was detected, an unsupervised clustering method was utilized to segment the detected eating gesture. After segmentation, a CNN-based classifier was used to classify the segmented data. In a recent work [23], the authors collected a total of 324 eating and drinking gestures from 6 participants in a standing scenario. Using CNN-Bidirectional LSTM (CNN-BiLSTM) model, they achieved accuracies of 84% and 76% for the detection of eating and drinking activities, respectively, under Leave-One-Subject-Out (LOSO) validation.

A more complex challenge arises when dealing with real-life continuous meal sessions. Previous works on radar-based eating activity recognition collected data in limited conditions with short durations. In [19], [20], [21], each participant's eating and drinking duration was less than 5 min. However, the real meal session has a longer duration (10–20 min per meal in our case, Table II). Additionally, meal sessions naturally involve lots of non-feeding activities seamlessly intertwined with intake gestures [24], resulting in a more realistic and challenging scenario for radar-based intake gesture detection. In this paper, we build a new end-to-end approach for fine-grained intake gesture detection using a contactless mmWave FMCW radar and a sequence-to-sequence (seq2seq) 3D temporal convolutional network with self-attention (3D-TCN-Att). The key highlights of this work are as follows:

1) A radar-based system is applied in the food intake monitoring domain to detect eating gestures in meal sessions. A seq2seq 3D-TCN-Att model is applied to process the Range-Doppler Cube (RD Cube) and generate frame-wise predictions. Combined with IoU-based segment-wise evaluation, this method is able to not only count the number of intake gestures, but also segment the time intervals of intake gestures.
2) We complete an intensive comparison between the proposed approach to the state-of-the-art techniques reported in the literature. Firstly, different radar data representations, namely RD Cube and DT map, are selected to find the most suitable one based on machine learning performance. Secondly, different models: CNN-(Bi)LSTM, pretrained 3D-CNNs, and transformers, are used to compare the performance. Results demonstrate the feasibility of radar-based intake gesture detection.
3) We produce a publicly available dataset[1] consisting 70 meal sessions from 70 participants. In total, 4,132 eating gestures and 893 drinking gestures are recorded. To our knowledge, this is the first public radar dataset targeting eating gesture detection in real meal sessions.

II. Methods

A. FMCW Radar Principle

For an FMCW radar with single transmitter (TX) and single receiver (RX), as called single-input-single-output (SISO) radar, its TX periodically transmits a frequency modulated sinusoidal signal with bandwidth B, start frequency f_c, frequency slope S, period T_c, as shown in Fig. 1. The signal in one periodic time is called a chirp signal. A sequence of chirps represents a frame (Fig. 1(c) and (d)). The row dimension of the frame is called the fast time dimension, and the column dimension is known as the slow time dimension, as the time scale between two chirps is much larger than the time scale between two sample points within a chirp. The RX antenna receives the signals reflected from objects in the radar's field of view (FOV). By mixing the received signal and transmitted signal, the mixer generates the Intermediate Frequency (IF) signal, also called the beat signal. The instantaneous frequency (f_b) of IF signal is equal to the difference of the instantaneous frequency between the transmitted and received signal. Similarly, the phase of IF signal (ϕ) is the difference of phase between the transmitted and received signal.

1) Range Measurement: For a static object with a distance d to the radar (Fig. 1(b)), the time delay of the received signal is $\tau = \frac{f_b}{S} = \frac{2d}{c}$, where c is the speed of light. Hence, the relation between the range d and beat frequency f_b is $d = \frac{f_b c}{2S}$. To calculate the range, a Fast Fourier Transform (Range-FFT) is applied to convert the IF signal from the time domain to the frequency domain for each chirp. The initial phase (ϕ) of the IF signal is $\phi = 2\pi f_c \tau = \frac{4\pi d}{\lambda}$, where λ is the wavelength.

2) Velocity Measurement: For a moving object with range d and velocity v, there is a tiny displacement Δd between two consecutive chirps, as shown Fig. 1(c). This causes an IF frequency

[1][Online]. Available: https://rdr.kuleuven.be/dataset.xhtml?persistentId=doi:10.48804/NQSH4F

Authorized licensed use limited to: Wageningen UR. Downloaded on March 01,2024 at 10:23:46 UTC from IEEE Xplore. Restrictions apply.

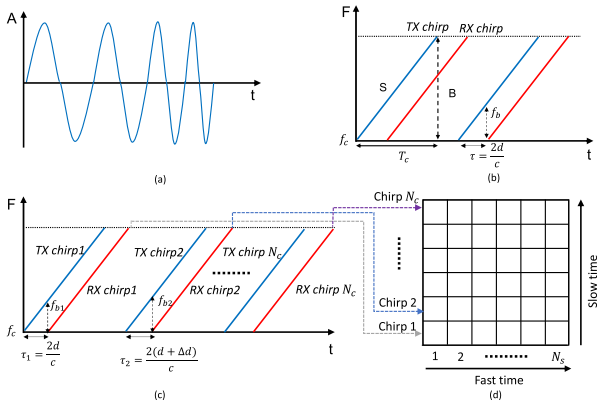


Fig. 1. (a) Example of the time-amplitude response from radar TX signal. The x-axis represents the time (t), and the y-axis represents the amplitude (A) of the signal. (b) The time-frequency response of TX signal (blue line) and RX signal (red line). The x-axis represents the time (t), and the y-axis represents the frequency (F). B, f_c, S, T_c are the bandwidth, start frequency, slope, and period of TX signal. f_b, d, c, τ are the beat frequency, the distance between the radar and the object, the speed of light, and the time delay between TX and RX signal. (c) The consecutive TX and RX signals for a moving object. τ_1, τ_2 represent the time delay for the first and second chirp, Δd is the displacement of the moving target in one TX period. (d) The form of the initial data frame after mixing TX and RX signals. Each row represents the N_s sample points for a chirp after mixing and analog to digital conversion (ADC), N_c chirps are stacked together to form the initial data frame. The row dimension of the frame is called the fast time dimension, and the column dimension is known as the slow time dimension, as the time scale between two chirps is much larger than the time scale between two sample points in a chirp.

change $\Delta f_b = f_{b2} - f_{b1} = S(\tau_2 - \tau_1) = S(\frac{2(d+\Delta d)}{c} - \frac{2d}{c}) = \frac{2S\Delta d}{c}$ and a phase change $\Delta\phi = \frac{4\pi\Delta d}{\lambda} = \frac{4\pi v T_c}{\lambda}$. Normally, the Δf_b caused by Δd is negligible compared to the scale of the sweep bandwidth (B), whereas the phase change is more sensitive to the object's movement between chirps in one frame. Hence, to estimate the velocity, an FFT across the chirps (along slow time axis, see Fig. 1(d)) for each range bin in one frame is applied after Range-FFT processing, as called Doppler-FFT. More details of the FMCW radar principle are in the website of Texas Instruments (TI).[2]

It should be noted that an analog to digital conversion (ADC) is used to digitize the beat signal before any processing. Suppose the number of samples in a chirp is N_s after the ADC and the number of chirps in a frame is N_c, the dimension of a radar data frame R is $N_c \times N_s$ (Fig. 1(d)).

B. Radar System Information

The experiment utilizes a 76 GHz TI IWR1843 radar board and a DCA1000EVM data capture card to build the data collection system, as shown in Fig. 2. The configurations of the radar system are specified in Table I. The radar has a 4 GHz bandwidth, with 3 TX and 4 RX antennas, powered with 5 V. The power consumption under described configuration is 1.8 W. The system generates 25 frames per second (fps). Two USB interfaces (one for IWR1843 radar board, one for DCA1000EVM) are connected to the laptop for system configuration. One Ethernet cable with high-speed data transmission capability (1 Gbps) is used to transfer raw data from the radar to laptop via the software TI mmWave Studio.[3]

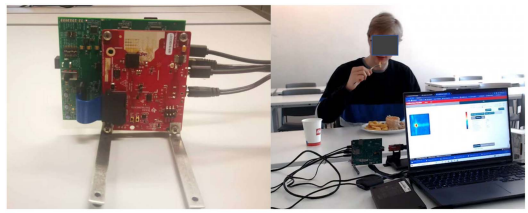


Fig. 2. (a) TI IWR1843 radar board (Red) and DCA1000EVM board (Green). (b) Scene of data collection. The participant was eating with a pair of fork and knife. The radar was placed on the table.

TABLE I
CONFIGURATIONS OF FMCW RADAR

Parameter	Configuration
Start frequency	77 GHz
Sweep bandwidth	3.8 GHz
Sweep slope	50 MHz/μs
Frame rate	25 fps
Sample rate	2,000 ksps
Number of chirps in one frame	128
Number of samples in one chirp	128
Number of used transmitter, receivers	1, 4
Range resolution	4 cm
Velocity resolution	4 cm/s

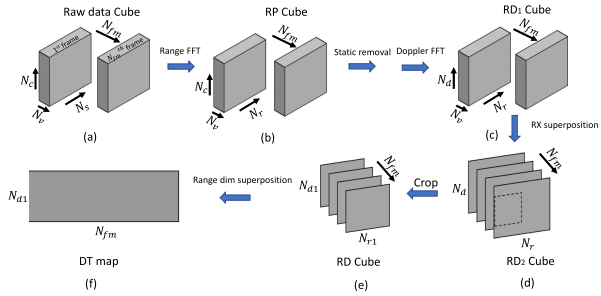


Fig. 3. Radar signal pre-processing pipeline. (a) The raw data cube. (b) The range profile cube (RP Cube). (c) The RD_1 Cube. (d) The RD_2 Cube after RX superposition. (e) The cropped RD Cube (to reduce the computation cost). (f) The Doppler-Time map (DT map).

C. Radar Signal Processing

The raw data exported from the radar are defined as a 5D data cube of size $N_{fm} \times N_{tx} \times N_{rx} \times N_c \times N_s$ where N_{fm}, N_{tx}, N_{rx}, N_c and N_s are the numbers of the radar frames in one meal session, transmitters (1), receivers (4), chirps in one frame (128), and sample points (128) in one chirp, respectively. Considering that we only use 1 transmitter and 4 receivers, the term $N_{tx} \times N_{rx}$ is replaced with N_v which denotes the number of virtual antennas ($N_v = 4$). Hence, a 4D data cube $N_{fm} \times N_v \times N_c \times N_s$ represents the raw data of a meal session. As mentioned in Section II-A, a frame of a SISO radar is represented by R (with size $N_c \times N_s$, Fig. 1(d)), the 4D data cube can also be written in $N_{fm} \times N_v \times R$. Subsequently, several steps are applied to extract the meaningful features, as shown in Fig. 3.

[2][Online]. Available: https://training.ti.com/node/1139153

[3][Online]. Available: https://www.ti.com/tool/MMWAVE-STUDIO

Authorized licensed use limited to: Wageningen UR. Downloaded on March 01,2024 at 10:23:46 UTC from IEEE Xplore. Restrictions apply.

1) Range FFT: For each frame R, we apply a FFT transform along fast-time axis (in each chirp, see Fig. 1(d)), obtained range profile cube (RP Cube) with size $N_{fm} \times N_v \times N_c \times N_r = N_{fm} \times N_v \times R_1$, N_r represents the number of range bins after range-FFT, R_1 represents the new frame in range profile cube (RP Cube). In this study, $N_r = N_s = 128$.

2) Static Objects Removal: To remove the signal of static objects, for each frame R_1 in the RP Cube, we calculate the average value through all chirps (along the slow-time axis, see Fig. 1(d)), and every original chirp is subtracted by the mean chirp as follows [25]:

$$R_2(i,j) = R_1(i,j) - \frac{1}{N_c}\sum_{k=1}^{N_c} R_1(k,j) \tag{1}$$

where $R_1(i,j)$ represents the value at i-th chirp ($i \in [1, N_c]$) and j-th range bin ($j \in [1, N_r]$) in frame R_1, R_2 represents the new frame after removing static objects.

3) Doppler FFT: For each frame R_2, we apply a FFT along slow-time across chirps to generate a Range-Doppler cube (RD_1 Cube) with size $N_{fm} \times N_v \times N_d \times N_r = N_{fm} \times N_v \times R_3$, N_d represents the number of doppler bins after doppler-FFT, R_3 represents the new frame in RD_1 Cube. In this study, $N_d = N_c = 128$.

4) RX Superposition: Because of the narrow width (less than 1 m) of the moving target (human), the angular resolution obtained by 4 receiving antennas is insufficient to differentiate objects in the cross-range dimension. Therefore, we simply superpose the RD_1 Cubes from 4 RX channels into one with an improved signal-to-noise ratio (SNR) [26]. The new RD_2 Cube has size $N_{fm} \times N_d \times N_r = N_{fm} \times R_4$, R_4 represents the new 4-in-1 RD frame:

$$RD_2(t,i,j) = \frac{1}{N_v}\sum_{a=1}^{N_v} |RD_1(t,a,i,j)| \tag{2}$$

where $RD_2(t,i,j)$ represents the value at t-th frame ($t \in [1, N_{fm}]$) i-th doppler bin ($i \in [1, N_d]$) and j-th range bin ($j \in [1, N_r]$), $a \in [1, N_v]$.

5) Crop ROI: The obtained RD_2 Cube contains frames with size 128×128. Because this study focuses on short-range HAR, and intake behavior has low speed, it is observed that the long-range area and high-velocity area are never reached. To reduce the computation cost, we crop the frames from RD_2 Cube to only keep the region of interest (ROI). The cropped frame size is $N_{d1} \times N_{r1}$, where $N_{d1} = 64$ (max velocity: ± 1.28 m/s) and $N_{r1} = 32$ (max range: 1.28 m). After this step, the final RD Cube is prepared as 3D input data with size $N_{fm} \times N_{d1} \times N_{r1}$ ($N_{fm} \times 64 \times 32$). Fig. 4 shows the example of an RD frame.

6) Generate the Doppler-Time Map for Comparison: This extra step is used to generate the DT map for data representation comparison, which will be discussed in Section III-C. The DT map is generated by summing the absolute value of the signal over the range dimension [27].

$$DT(t,n) = \frac{1}{N_{r1}}\sum_{j=1}^{N_{r1}} RD(t,n,j) \tag{3}$$

where $DT(t,n)$ represents the DT map value at t-th frame ($t \in [1, N_{fm}]$) and n-th doppler bin ($n \in [1, N_{d1}]$), $RD(t,n,j)$ represents the RD Cube value at t-th frame, n-th doppler bin, and j-th range bin ($j \in [1, N_{r1}]$). The dimension of the DT map is $N_{fm} \times N_{d1}$.

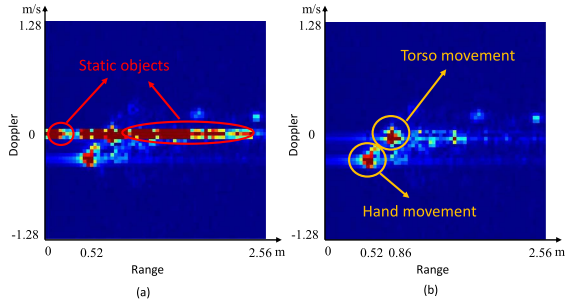


Fig. 4. Range-doppler (RD) frame represents the transient movement taking a glass of water from table to mouth. (a) the RD frame without static object removal; (b) the RD frame after static object removal. This is a movement from plate to mouth (the hand is moving away from radar), so it has negative velocity. The torso (body) is moving towards the radar with lower velocity compared to the hand.

TABLE II
MEAL DATA STATISTICS

Parameter	Values
Participants	70
Fork & knife: chopsticks:spoon:hand	27:16:15:12
Female : male	25 : 45
Duration ratio of other : eating : drinking	11.21:2.59:1
Mean meal duration	(16.50±5.24) min
Total duration	1155 min

D. Data Collection and Annotation

The ethical committee of KU Leuven has approved this research (Reference number: G-2021-4025-R4), and written informed consent from each participant was collected. 70 participants (25 females and 45 males) were recruited in the experiment. Each of them had one meal session. The data were collected in seven different rooms with different layouts at the KU Leuven University. The participants sat directly in front of the radar and ate their meals. The distance between the food plate and the radar was around 0.3 m to 0.6 m, as shown in Fig. 2(b). A camera was used to record video for annotation. It should be noted that some of the participants also wore IMU wristbands on two hands for another study, which will not be discussed here. Table II shows detailed information of meals.

The consumed food was served by Alma (the student restaurant), a Chinese restaurant, a sandwich shop, and a convenience store. The participants could use forks & knives, chopsticks, spoons, and hands to eat the meal. They were allowed to eat at their own pace. They could stop, talk, watch smartphones, move for a while, and prepare/stir food. A glass of water or a can of soft drink was served together with the meal. During the meal, the participants could drink at their own pace. However, to increase the number of drinking gestures, some participants were required to drink several times extra before and after the meal. This requirement is due to the fact that people have few drinking gestures compared to the amount of eating and other gestures in a meal. Detailed information on eating styles is shown in Table III.

All participants were encouraged to eat their meals in their own way. Thus some non-feeding movements were naturally

Authorized licensed use limited to: Wageningen UR. Downloaded on March 01,2024 at 10:23:46 UTC from IEEE Xplore. Restrictions apply.

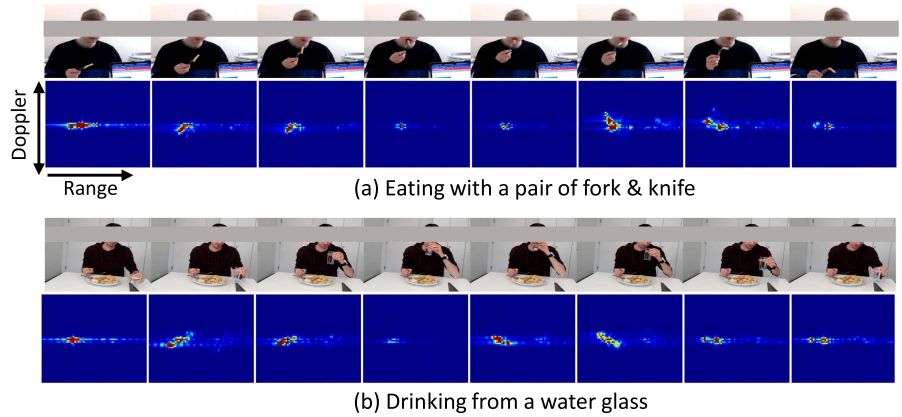


Fig. 5. Video frames and the corresponding RD frames of eating and drinking gestures. For the RD frames, the x-axis represents the range dimension; the y-axis represents the doppler dimension. It should be noted that the actual amounts of frames for each eating/drinking gesture are different, and we manually selected the fixed number of frames for each gesture in this figure.

TABLE III
DETAILS OF DIFFERENT EATING STYLES

Style	#Eating gesture	Food
Fork & knife	1,632	salad, fries, chicken, beef burger, cake
Chopsticks	1,479	rice, fried noodles, chicken, vegetables
Spoon	671	rice, chicken, vegetables
Hand	350	chicken sandwich, egg sandwich

TABLE IV
EATING AND DRINKING GESTURE DETAILS

Parameter	Values	
	Eating	Drinking
Number of gestures	4,132	893
Duration range of gestures	1.00–17.08 s	1.88–24.56 s
Median duration of gestures	2.56 s	4.64 s
Mean ± std of gestures	2.93 ± 1.42 s	5.24 ± 2.34 s

included during meal sessions. Additionally, to further mimic free-living environments, 22 of 70 meal sessions were collected by asking the participants to perform more non-feeding gestures and confounding gestures [9], such as touching nose, combing hair with hand, adjusting eyeglasses, using smartphones, and reading magazines.

We used ELAN [28] to annotate the video. The data are annotated into 3 classses, namely eating, drinking, and other. The definitions of eating and drinking gesture are mentioned in Section I. The annotation work was performed by two annotators who followed the same instruction. One is the first author, the other one is independent of this research. The task was divided by sequence. The first author reviewed the annotation from the independent annotator and adjusted it if needed to minimize ambiguity. There are 4,132 eating gestures and 893 drinking gestures in total. The dataset statistics are indicated in Table IV. Fig. 5 shows the RD frames of eating and drinking gestures.

E. Sequence-to-Sequence Neural Network Architecture

Before feeding the data into the model, prior works [19], [20], [21] tend to use a sliding window to generate data segments with fixed temporal length. Each window is assigned a label through a majority voting mechanism. Alternatively, they label the window by checking if it contains an entire target activity. We found both approaches have limitations for real-world continuous eating gesture detection. The challenge lies in the variability of eating gesture durations and the fixed length of the sliding window. Firstly, if the duration of the eating gesture is longer than the window length, this gesture may be divided into two or more segments. The segment that contains the minor part of the eating gesture may be labeled as other. Secondly, if a single window encompasses multiple eating gestures with a short duration, it becomes challenging to accurately quantify the total number of eating gestures. To mitigate this issue, we choose the seq2seq architecture instead of the many-to-one architecture. For the seq2seq model, the prediction is generated for each frame rather than for an entire window, referred to as frame-wise prediction. Afterward, a sequence of frame-wise predictions sharing the same value is considered as a segmental interval.

The temporal convolutional network (TCN) using dilated convolution and skip connection was originally introduced by Lea et al. [29] to recognize long-range temporal sequences. To date, it has been utilized to process time-series signals in the healthcare and disease diagnosis domains [8], [30], [31] due to its outstanding performance [32]. Meanwhile, the self-attention exhibits superior performance in natural language processing (NLP) [33] and video-based HAR [34]. Inspired by its success in [35], [36], we integrated the multi-head self-attention into the TCN model, composing the 3D-TCN-Att model to further enhance the temporal sequence processing ability. Noted that existing TCNs [29], [37] are not suitable in our case, as we employ 3D data. Therefore, the 3D-TCN using Conv3D is developed to process time-series RD Cube.

1) 3D-TCN Module: As illustrated in Fig. 6, the 3D-TCN module contains a series of L dilated residual layers, where L is the number of layers. Each layer consists of dilated convolutions with ReLU activation, and a residual connection that combines the input of the current layer and the convolution results, as depicted in Fig. 6(b). The Conv3D kernel with size $3 \times 3 \times 3$ convolves input data in range (R), doppler (D), and time (T) dimensions. The number of Conv3D kernels in each layer is 32. The 3D-TCN contains 2 types of dilated layers, the shape-invariant layer and shape-variant layer. The shape-variant layer (Fig. 6(b)) is used to reduce data shape. Specifically, the size of the range and doppler dimensions are reduced whereas the time

Authorized licensed use limited to: Wageningen UR. Downloaded on March 01,2024 at 10:23:46 UTC from IEEE Xplore. Restrictions apply.

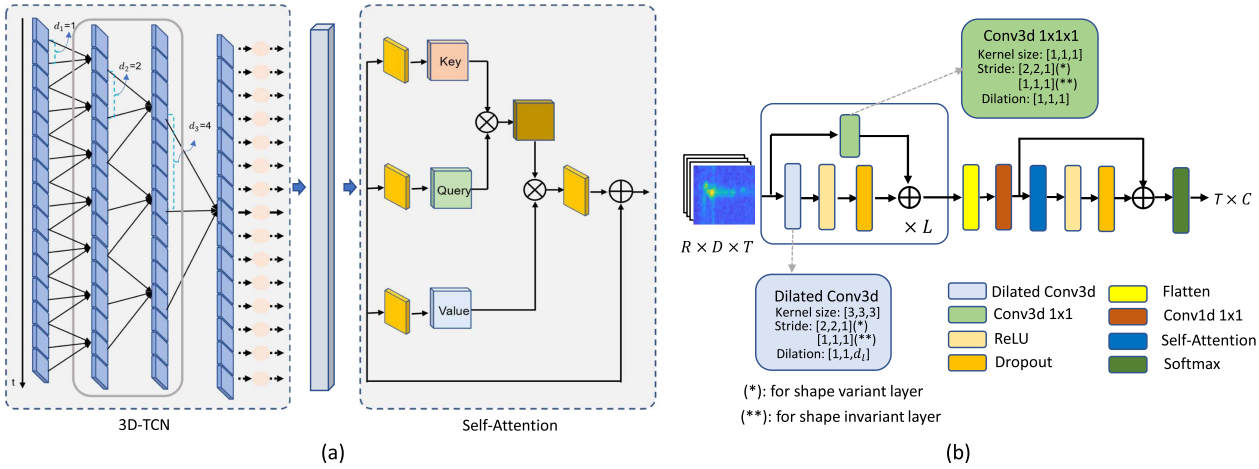


Fig. 6. (a) Framework of the 3D-TCN-Att model. From left to right, the RD cube is sent into the input layer (the first layer) of the 3D-TCN module. The extracted features from 3D-TCN are fed into the attention module for prediction refinement. (b) Detailed architecture of the shape-variant (*) and shape-invariant (**) dilated residual layer in 3D-TCN module. The main difference between the two layers is the stride setting. d_l represents the dilated factor, $d_l = 2^{l-1}$ $(1 \leq l \leq L)$, where l denotes the order of layers, L is the amount of layers. The R, D, T from the input dimension represent the range (R), doppler (D), and time (T). The C in output is the number of classes.

dimension remains by setting the stride as (2,2,1). The residual connection has the same stride to keep the same reduction. In contrast, the input and output dimensions of shape-invariant layer remain the same by setting the stride as (1,1,1). The convolutions in both layers are in SAME mode. Finally, the dilation only applies in the time dimension; for the range and doppler dimension, the dilation rate is 1. The dilation factor of the time dimension in Conv3D at each layer is doubled such that $d_l = 2^{l-1}$ $(1 \leq l \leq L)$. After each dilated layer, a 0.3 dropout is applied.

2) Attention Module: The attention function can be described as a mapping between the query $(Q = X_1 W_q)$, key $(K = X_1 W_k)$, and value $(V = X_1 W_v)$, where $X_1 \in \mathbb{R}^{T \times d_{in}}$ is the input of attention module (also the output of 3D-TCN module), W_q, W_k and $W_v \in \mathbb{R}^{d_{in} \times d_{model}}$, d_{in} is the dimension of output sequence from the 3D-TCN model ($d_{in} = 3$), and d_{model} represents the dimension of the attention module. The results obtained from the attention can be described as follows:

$$\text{Attention}\ (Q, K, V) = \text{Softmax}\left(\frac{QK^\top}{\sqrt{d_{model}}}\right) V \qquad (4)$$

The aforementioned calculation only involves one single attention graph, whereas multi-head attention utilizes multiple attention graphs to learn attention weights across multiple aspects. The number of heads (h) denotes the number of attention graphs.

$$MultiHead(Q, K, V) = \text{Concat}(head_1, \ldots, head_h) W^O$$
$$\text{where}\ head_i = \text{Attention}\ (QW_i^Q, KW_i^K, VW_i^V) \qquad (5)$$

where $W^O \in \mathbb{R}^{h \cdot d_h \times d_{in}}$, W_i^Q, W_i^K and $W_i^V \in \mathbb{R}^{d_{model} \times d_h}$, and d_h represents the dimension of each single head attention, with $d_h = d_{model}/h$. After applying a residual connection between the input and output of the multi-head attention and a Softmax activation, the final prediction $Y \in \mathbb{R}^{T \times C}$ is obtained, where C is the number of classes ($C = d_{in}$).

TABLE V
ARCHITECTURE OF 3D-TCN-ATT MODEL AND THE OUTPUT SIZE OF EACH LAYER

Layer	Type[a]	Output
Input[b]	-	$1 \times 32 \times 64 \times 1000$
Layer 1	Shape variant[c]	$32 \times 32 \times 32 \times 1000$
Layer 2	Shape variant	$32 \times 16 \times 16 \times 1000$
Layer 3	Shape variant	$32 \times 8 \times 8 \times 1000$
Layer 4	Shape variant	$32 \times 4 \times 4 \times 1000$
Layer 5	Shape invariant	$32 \times 4 \times 4 \times 1000$
Layer 6	Shape invariant	$32 \times 4 \times 4 \times 1000$
Layer 7	Shape invariant	$32 \times 4 \times 4 \times 1000$
Layer 8	Shape invariant	$32 \times 4 \times 4 \times 1000$
Layer 9	Shape invariant	$32 \times 4 \times 4 \times 1000$
Flatten	-	512×1000
Conv1D & Permute	-	1000×3
Self-Attention	-	1000×3
ReLU & Dropout	-	1000×3
Softmax	-	1000×3

[a] There are two types of dilated residual layer, namely the shape variant layer and the shape invariant layer, as shown in Fig. 6(b).
[b] Here we take a 40 s data segment for example, hence the input data size is $1 \times 32 \times 64 \times 1000$.
[c] The first shape variant layer is different from others; its stride is [1,2,1], the size of range dimension remains the same to achieve the same size between range and doppler dimension.

The loss function of the 3D-TCN-Att model is a combination of the cross-entropy loss as the classification loss, and a truncated mean squared error (T-MSE) as the smoothing loss. Details of the combined loss function can be found in [38]. According to experiments, we built the 3D-TCN module with 4 shape variant layers, and 5 shape invariant layers. The self-attention module has 4 heads, each with a dimension of 8 ($d_{model} = 32$). To train the model, an RMSprop optimizer with a learning rate of 0.0005 is employed. Notably, the positional encoding (PE) is not incorporated in attention module since we did not observe performance improvement after its inclusion. The architecture information is summarized in Table V. All the training, and testing experiments were deployed on an Intel 8-core Xeon Gold 6240 CPU@2.6 GHz (Cascadelake) with 20 GB RAM per core, and two pieces of NVIDIA Tesla V100-SXM2-32 GB GPU from Vlaams Supercomputer Centrum (VSC) .[4]

[4][Online]. Available: https://www.vscentrum.be/

Authorized licensed use limited to: Wageningen UR. Downloaded on March 01,2024 at 10:23:46 UTC from IEEE Xplore. Restrictions apply.

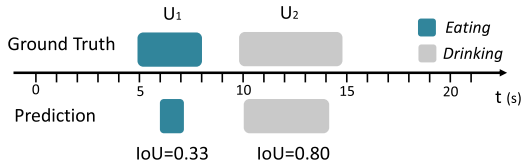


Fig. 7. Example of IoU calculation.

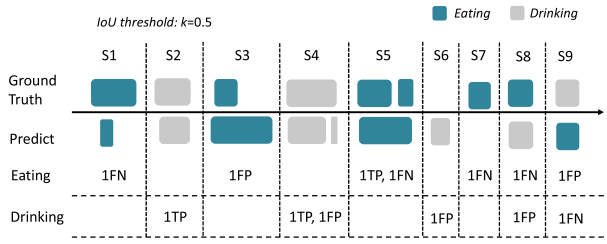


Fig. 8. Segment-wise evaluation examples when the threshold k is 0.5.

III. Evaluation and Experiments

A. Evaluation Scheme

1) Frame-Wise Evaluation: To evaluate a seq2seq model, firstly, frame-wise f1-scores for eating and drinking frames are calculated, as our dataset is unbalanced (Table II). Secondly, the Cohen Kappa is applied as an overall frame-wise metric. The equations are given below.

$$f1\text{-}score_{c=\{E,D\}} = \frac{2tp_c}{2tp_c + fp_c + fn_c} \tag{6}$$

$$Kappa = \frac{P_o - P_e}{1 - P_e} \tag{7}$$

$$P_o = \sum_{c=N,E,D} \frac{tp_c}{Total} \tag{8}$$

$$P_e = \sum_{c=N,E,D} \frac{tp_c + fp_c}{Total} \times \frac{tp_c + fn_c}{Total} \tag{9}$$

where $Total$ is the number of data frames, c represents the class, N, E, D represent the class of other, eating, and drinking gesture, respectively. The tp, fp, and fn are frame-wise true positive, false positive, and false negative, respectively.

However, in the perspective of HAR, frame-wise evaluation is unable to give us the results such as how many eating/drinking gestures are detected. To address this issue, we also utilized segment-wise evaluation.

2) Segment-Wise Evaluation: The segmental F1-score proposed by [8], [29] is applied for segment-wise evaluation. The intersection over union (IoU) of each predicted intake gesture is first computed. The IoU is defined as $\frac{A \cap B}{A \cup B}$, the ratio of overlap between temporal intervals of the ground truth segment (A) and the predicted segment (B). Fig. 7 shows the example of IoU calculation. Once the IoU of each intake gesture is calculated, a threshold k is applied to determine segment-wise True Positive (TP), False Negative (FN), and False Positive (FP). Fig. 8 shows evaluation examples. If the IoU of a segment pair is greater than the threshold k, it is a TP (Fig. 8 S2), otherwise it is an FN segment (Fig. 8 S1) or FP segment (Fig. 8 S3). The decision tree is as follows:

$$\text{Segment} = \begin{cases} TP, & IoU \geq k \\ FP, & IoU < k, length_{gt} < length_p \\ FN, & IoU < k, length_{gt} > length_p \end{cases} \tag{10}$$

where $length_{gt}$ and $length_p$ are the time lengths of the ground truth and predicted eating/drinking gesture, respectively. Additionally, if multiple predicted segments exist inside the range of a given ground truth gesture, only one is classified as a TP, all others are FPs (Fig. 8 S4). Conversely, if a predicted segment spans multiple ground truth eating/drinking gestures, only one is recognized as TP and the remainder are FNs (Fig. 8 S5). We also defined the corresponding punishment for the misclassification between the eating and drinking gesture (see Fig. 8 S8–S9). After the generation of segmental TP, FP, and FN, the segmental F1-score is calculated as:

$$F1\text{-}score_{c=\{E,D\}} = \frac{2TP_c}{2TP_c + FP_c + FN_c} \tag{11}$$

where c represents the class, E, D represents eating and drinking class, respectively.

The segment-wise method provides three main advantages. Firstly, it effectively addressed both over-segmentation errors (Fig. 8 S4) and under-segmentation errors (Fig. 8 S5); secondly, it accommodates minor temporal discrepancies between the ground truth and prediction, which can occur due to annotation variability. Thirdly, it provides clear information including how many gestures are detected. Three thresholds k are selected as 0.1, 0.25 and 0.5 according to [29].

B. Models for Benchmark

Different deep learning models are selected as benchmarks. Specifically, the CNN-(Bi)LSTM [17], [39], pretrained 3D-CNNs, and transformers. It should be noted that the output modes of all models are configured as sequence output.

1) CNN-(Bi)LSTM: The CNN-(Bi)LSTM model consists of three subnetworks: three CNN layers, one LSTM layer, and two fully connected layers (FCN). The CNN part contains three layers with 16, 32, and 32 kernels with size 3×3, respectively. The activation ReLU is used for each layer. Each CNN layer is followed by a maxpooling layer with 2×2 pooling size. The output of the CNN part is flattened and fed into the LSTM part. The LSTM layer contains 128 hidden units followed by a 0.3 dropout. The FCN contains two dense layers. The first layer has 128 neurons with ReLU activation, and the second layer has 3 neurons with Softmax activation. The Timedistributed layer is used to wrap the model. Based on experiments, the window length of CNN-LSTM and CNN-BiLSTM is 10 s and 20 s, respectively.

2) Pretrained 3D-CNNs: In video-based HAR domain, numerous 3D-CNN-based models have been proposed and pretrained on large scale RGB-based datasets [18]. We conducted experiments using four widely used pretrained 3D-CNN models: 3DResNet-18 [40], 3DResNet-50 [41], X3D [42], R(2+1)D [43]. These models were modified to fit our data processing pipeline. The 3DResNet-18 and R(2+1)D models were built based on Torchvision package, while the 3DResNet-50 and X3D models were from FAIR PyTorchVideo package. All models were pretrained on the Kinetics dataset [44]. In our experiments, we modified the output layers for frame-wise

Authorized licensed use limited to: Wageningen UR. Downloaded on March 01,2024 at 10:23:46 UTC from IEEE Xplore. Restrictions apply.

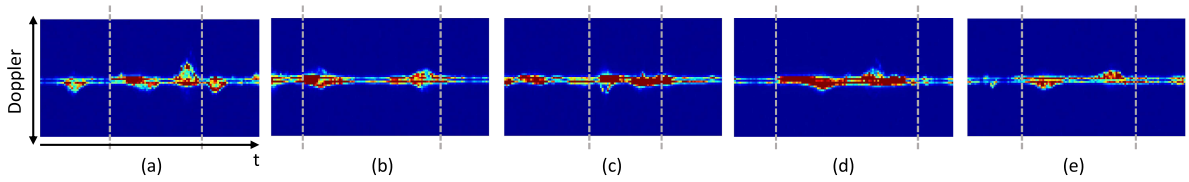


Fig. 9. Examples of DT map for different eating and drinking gestures. The duration of each example is 5 s. Signals between grey dashed line represent different eating and drinking gestures: (a) Eating with a pair of fork&knife; (b) Drinking from a water glass; (c) Eating with a pair of chopsticks; (d) Eating with a spoon; (e) Eating with a hand.

prediction and adjusted the number of predicted classes to 3. The window length of these models is 10 s. All models were trained for 100 epochs with the RMSprop optimizer, a learning rate of 0.0005, and a batch size of 64.

3) Transformers: Two transformer models were employed for our experiments: the vanilla Transformer [33], and the ViViT model [45]. Based on experiments, both were trained from scratch. The vanilla Transformer has the same encoder architecture as introduced in [33]; however, as it can only process 1D data, a 3-layer CNN module is used to extract spatial features from 3D radar data and flatten it into 1D intermediate data. The added 3-layer CNN is identical to the CNN component used in the CNN-LSTM model. Hence, the CNN-vanilla Transformer is called CNN-VT in this study. Meanwhile, based on our experiments, the number of encoder layer in CNN-VT model is 1, with 8 heads, each with the dimension of 64. For the ViViT model, the patch size for each input frame is 16, the number of transformer layer is 6, and each layer has 8 heads with a head dimension of 64. The window length, number of epochs, and learning rate are consistent with those of pretrained 3D-CNNs.

C. Radar Data Representations

For radar-based HAR systems, another common radar data representation is the DT map. Compared to RD Cube that contains range, velocity, and time dimensions, the DT map only includes information from time and velocity dimensions. For the eating activity, apart from velocity, the range information of the arm and the head (or torso) is also an essential feature. Thus, this research focuses on using RD Cube for eating gesture detection. To validate our hypothesis, we also test the DT map as input, with the dimension $N_f \times N_{d1}$. Fig. 9 shows the examples of DT map. Due to the dimension difference between the DT map (2D) and the RD Cube (3D), proposed models used for the RD Cube (3D-TCN, CNN-(Bi)LSTM) are modified as 2D types to fit the DT map.

D. Validation Strategy

To evaluate the model performance on unseen data, the dataset is divided into 7 folds. For each fold, the number of meals in the train/valid/test set is 50/10/10, respectively. It should be noted that the split of dataset in this validation method is meal level, there is no information overlap between the train, valid, and test set. (The meal_id lists for 7-fold experiments are available along with the dataset.)

IV. Results

Table VI presents the frame-wise results on DT map and RD Cube. Both the f1-scores and kappa scores from three models

TABLE VI
Overall Frame-Wise Results for 7-Fold Cross-Validation

Data	Model	f1-score E	f1-score D	Kappa
DT map (2D)	CNN-LSTM	0.519	0.523	0.450
	CNN-BiLSTM	0.570	0.636	0.523
	2D-TCN	0.742	0.805	0.715
RD Cube (3D)	CNN-LSTM	0.654	0.697	0.605
	CNN-BiLSTM	0.713	0.763	0.672
	3DResNet-18	0.699	0.720	0.653
	3DResNet-50	0.588	0.575	0.520
	X3D	0.716	0.732	0.667
	R(2+1)D	0.696	0.682	0.636
	CNN-VT	0.654	0.666	0.589
	ViViT	0.577	0.474	0.449
	3D-TCN	0.772	**0.819**	0.736
	3D-TCN-Att	**0.776**	0.817	**0.745**

The bold entities represent the best results in each column.

(CNN-LSTM, CNN-BiLSTM, and 2D/3D-TCN) in RD Cube were higher than those of in DT map. Using DT map data, the 2D-TCN obtained the highest performance with a kappa of 0.715, an eating f1-score of 0.742, and a drinking f1-score of 0.805. For RD Cube, the 3D-TCN-Att obtained the highest kappa of 0.745 and eating f1-score of 0.776. The 3D-TCN yielded the best drinking f1-score of 0.819.

Table VII shows the overall segment-wise results. In segment-wise evaluation, the RD Cube still obtained better performance on all three models (CNN-LSTM, CNN-BiLSTM, and 2D/3D-TCN). For $k = 0.1$, the 3D-TCN-Att model yielded the highest F1-score of 0.924 for eating and 0.893 for drinking, compared to the X3D (E: 0.890; D: 0.825) and the CNN-BiLSTM (E: 0.884; D: 0.866). When the threshold was raised to 0.5, the performance of all models decreased. However, 3D-TCN-Att still obtained the best results (E: 0.896; D: 0.868) with the least reduction (E: 2.8%, D: 2.5%), compared to X3D (E: 0.890→0.834; D: 0.825→0.785).

In addition to segmental F1-scores, the number of parameters for each model and their inference time (the latency) were also indicated in Table VII. Regarding DT map data, the amount of parameters in 2D-TCN was the least (0.115 M). For RD Cube data, the number of parameters in all pretrained 3D-CNN models was significantly larger than CNN-LSTM (0.294 M) and 3D-TCN-Att models (0.302 M). To assess the model's real-time ability after deployment, we conducted tests to measure the inference time for processing 40 s data. Models were deployed on a laptop with Intel Core i7 10750 CPU @2.6 GB, 6 cores (without GPU). Results in Table VII represent the average inference time across 100 test runs. The CNN-LSTM had the lowest latency compared to other models (0.217 s). The 3D-TCN-Att required 1.062 s to generate predictions.

Table VIII shows the average segmental F1-score ($k = 0.5$) across 7-fold cross-validation experiment. To further compare the performance difference between 3D-TCN-Att and other

Authorized licensed use limited to: Wageningen UR. Downloaded on March 01,2024 at 10:23:46 UTC from IEEE Xplore. Restrictions apply.

TABLE VII
OVERALL SEGMENT-WISE RESULTS FOR 7-FOLD CROSS-VALIDATION

Data	Model	Eating Segmental F1-score			Drinking Segmental F1-score			#Params	Latency
		$k=0.1$	$k=0.25$	$k=0.5$	$k=0.1$	$k=0.25$	$k=0.5$		
DT map (2D)	CNN-LSTM	0.735	0.715	0.609	0.745	0.696	0.578	0.228 M[a]	**0.021 s**
	CNN-BiLSTM	0.763	0.749	0.669	0.813	0.782	0.694	0.442 M	0.031 s
	2D-TCN	0.890	0.885	0.855	0.887	0.880	0.858	**0.115 M**	0.085 s
RD Cube (3D)	CNN-LSTM	0.843	0.837	0.766	0.816	0.795	0.740	0.294 M	0.217 s
	CNN-BiLSTM	0.884	0.877	0.828	0.866	0.852	0.812	0.573 M	0.233 s
	3DResNet-18	0.873	0.865	0.807	0.832	0.805	0.732	33.353 M	20.582 s
	3DResNet-50	0.798	0.789	0.729	0.642	0.614	0.554	31.634 M	5.871 s
	X3D	0.890	0.883	0.834	0.825	0.819	0.785	2.980 M	1.870 s
	R(2+1)D	0.844	0.840	0.792	0.748	0.725	0.655	31.297 M	15.398 s
	CNN-VT	0.846	0.833	0.768	0.760	0.749	0.699	1.667 M	0.485 s
	ViViT	0.749	0.728	0.645	0.603	0.570	0.501	7.235 M	0.855 s
	3D-TCN	0.921	0.918	0.890	0.888	0.883	0.864	0.300 M	0.991 s
	3D-TCN-Att	0.924	0.922	**0.896**	0.893	0.889	**0.868**	0.302 M	1.062 s

[a] The M represents million. So 0.228 M represents 0.228 million parameters.
The bold entities represent the best results in each column.

TABLE VIII
AVERAGE RESULTS ACROSS 7-FOLD CROSS-VALIDATION ($k=0.5$)

Data	Model	F1-score (mean±std)		p[a]	
		E	D	E	D
DT map (2D)	CNN-LSTM	0.594±0.094	0.575±0.083	***	***
	CNN-BiLSTM	0.670±0.081	0.693±0.037	***	***
	2D-TCN	0.853±0.031	0.856±0.078	**	0.23
RD Cube (3D)	CNN-LSTM	0.768±0.031	0.731±0.080	***	***
	CNN-BiLSTM	0.826±0.024	0.806±0.089	**	**
	3DResNet-18	0.807±0.024	0.727±0.070	***	***
	3DResNet-50	0.722±0.081	0.583±0.111	***	***
	X3D	0.830±0.041	0.777±0.099	**	***
	R(2+1)D	0.779±0.077	0.663±0.156	***	***
	CNN-VT	0.768±0.033	0.699±0.060	***	***
	ViViT	0.645±0.040	0.498±0.051	***	***
	3D-TCN	0.888±0.037	0.857±0.061	*	0.08
	3D-TCN-Att	**0.895±0.035**	**0.861±0.054**	-	-

[a] The p values are obtained by applying paired t-test (7-fold experiment) between 3D-TCN-Att and other models for eating and drinking detection performance. $p<0.05$: *, $p<0.01$: **, $p<0.001$: ***.
The bold entities represent the best results in each column.

TABLE IX
OVERALL MISCLASSIFICATION RESULTS

Data	Model	#(E→D)[a]	#(D→E)
DT map (2D)	CNN-LSTM	70 (1.69%)	32 (3.58%)
	CNN-BiLSTM	39 (0.94%)	24 (2.69%)
	2D-TCN	23 (0.56%)	**10 (1.12%)**
RD Cube (3D)	CNN-LSTM	25 (0.61%)	25 (2.80%)
	CNN-BiLSTM	22 (0.53%)	20 (2.24%)
	3DResNet-18	66 (1.60%)	28 (3.14%)
	3DResNet-50	227 (5.49%)	104 (11.65%)
	X3D	61 (1.48%)	26 (2.91%)
	R(2+1)D	90 (2.18%)	63 (7.05%)
	CNN-VT	61 (1.48%)	26 (2.91%)
	ViViT	94 (2.27%)	107 (11.98%)
	3D-TCN	**19 (0.46%)**	24 (2.69%)
	3D-TCN-Att	23 (0.56%)	24 (2.69%)

[a] #(E→D) denotes the number of ground truth eating gesture predicted as drinking gesture (Fig. 8 S8). #(D→E) represents the reverse case (Fig. 8 S9).
The bold entities represent the best results in each column.

models, paired t-test was applied. For eating gesture detection, the 3D-TCN-Att obtained the highest mean F1-score (0.895±0.035, $p<0.05$). For drinking gesture detection, the model still yielded the highest mean F1-score with statistical significance (0.861±0.054, $p<0.05$) for all models except 2D-TCN ($p=0.23$) and 3D-TCN ($p=0.08$).

Table IX presents the number of misclassifications between eating and drinking gestures. The 3D-TCN-Att model wrongly predicted 23 ground truth eating gestures (0.56% of total ground truth eating gestures) as drinking gestures (E → D, see Fig. 8 S8), which was slightly higher than that of the 3D-TCN model (0.46%). For the case of misclassifying drinking as eating (D → E, see Fig. 8 S9), 2D-TCN had the least number.

TABLE X
PERFORMANCE OF EATING GESTURE DETECTION FOR FOUR EATING STYLES USING 3D-TCN-ATT ($k=0.5$)

	Fork & knife	Chopsticks	Spoon	Hand
F1-score	0.897	0.920	0.901	0.813

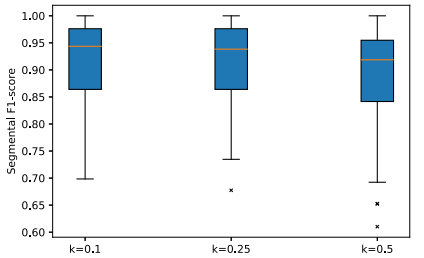


Fig. 10. Box plot of the results from 70 meals. The outlier with the lowest F1-score for $k=0.5$ is the meal eaten with hand.

Four different eating styles were included in our dataset (Table III). To investigate if the eating style has any impact on the performance, the segment-wise F1-score for each eating style on 3D-TCN-Att with RD Cube data was calculated, as shown in Table X. The 3D-TCN-Att obtained the highest F1-score for eating with chopsticks (0.920). The detection of eating with hand had the lowest performance (0.813) compared to the other three eating gestures. It should be clarified that this analysis did not involve eating style classification. The results were obtained from the 7-fold validation. The entire dataset (70 meals) was categorized into four groups based on eating styles to facilitate this analysis.

Fig. 10 denotes the segmental F1-score distribution for 70 meal sessions. All the meals had F1-scores higher than 0.7 for $k=0.1$. When k was increased to 0.5, one meal session's F1-score was equal to or lower than 0.6 (the lowest outlier), which was eaten by hand.

V. DISCUSSION

The proposed radar-based food intake monitoring system obtained promising results. There are several elements that distinguish our work from existing approaches [22], [23]. Firstly, we developed an end-to-end framework for eating behavior

Authorized licensed use limited to: Wageningen UR. Downloaded on March 01,2024 at 10:23:46 UTC from IEEE Xplore. Restrictions apply.

TABLE XI
COMPARING THE LITERATURE ON IN-MEAL EATING GESTURE DETECTION

Modality	Study/Dataset	#Subjects	#Meal	Model	Validation	Performance
IMU	FIC [7]	12	21	CNN-LSTM	LOSO	Eating F1: 0.923
	OREBA-IMU [4]	100	100	ResNet-10 CNN-LSTM	train:valid:test=61:20:19	Eating F1: 0.837 Drinking F1: 0.770
	Clemson [47]	271	271	Non-machine learning	-	Eating F1: 0.814
Video	Hossain *et al.* [48]	28	84	Faster R-CNN+AlexNet	4-fold	Eating accuracy: 0.854
	OREBA-Video [4]	100	100	ResNet-50 CNN-LSTM	train:valid:test=61:20:19	Eating F1: 0.869 Drinking F1: 0.761
Acoustic	Sazonov *et al.* [49]	20	70	SVM	3-fold	Average weighted accuracy: 0.847
	K. Lee [50]	10	40	ANN	train:test=7:3	Chewing accuracy: 0.914 Swallowing accuracy: 0.784
Radar	Proposed	70	70	3D-TCN-Att	7-fold	Eating F1: 0.896 Drinking F1: 0.868 ($k = 0.5$)

detection in continuous meal sessions. In contrast to previous radar-based works, continuous meal sessions involve lots of other non-feeding activities, which are seamlessly performed together with intake gestures during meal sessions. These activities are continuous, with varying durations and natural transitions between activities [24], making the eating gesture detection in meal session more challenging, but more meaningful in terms of real-life application [46]. Secondly, we evaluated our model on a significantly larger dataset with more participants (70 vs. 6). Thirdly, we detected intake gestures from a dataset that includes various other classes of activities, whereas the work in [22] focused on the classification between 5 eating styles (drinking and other classes were not reported in their data.). For the study in [23], the duration ratio of eating and non-eating activity was 1.2 : 1 (eating : non-eating). However, in real meal sessions, intake gestures occur sparsely, resulting in an imbalanced dataset (ratio of duration for drinking : eating : others = 1 : 2.59 : 11.21 in our case).

When comparing our study to existing food intake monitoring studies using different sensors but with a similar experiment setup, as presented in Table XI, the size of our dataset is comparable to others. Furthermore, the radar performance aligns with existing approaches (IMU-, video-, and acoustic-based sensor) in similar settings, which provides evidence that the radar is a promising modality for further investigation.

Currently, the conventional method in HAR is using a sliding window and there is only one label per window. In real-life scenarios, the duration of intake gestures and the time distance between two intake gestures vary greatly (Table IV). It is difficult to annotate the window as eating, drinking or other, because it is possible that one window can contain multiple short eating gestures, or only a part of one eating gesture. Hence we applied the more desirable seq2seq architecture to obtain frame-wise predictions. Frame-wise labels are used in the training process, the seq2seq architecture can also enable the ability of segmentation for detected gestures.

Several models were implemented to evaluate the performance. Results show that the performance of pretrained 3D-CNNs is lower than that of 3D-TCN-Att. One potential reason is the substantial dissimilarity between the pretrained video dataset and our radar dataset. Although both are 3D data, the information provided by video and RD cube is fundamentally different. As for transformers, they require large-scale datasets and significant computational resources due to their lack of inductive biases. The success of transformers in video domain relies on extensive training on large-scale datasets. However, radar-based HAR is still at an early stage compared to video-based study, resulting in a scarcity of large-scale radar datasets. Due to the limited data size, we did not take full advantage of pretrained 3D-CNNs and transformers, which are worth for further investigation in radar domain.

The CNN-(Bi)LSTM models exhibited significantly lower latency compared to others. This is primarily because CNN-RNNs utilized Conv2D, while others relied on Conv3D, which required more computational resources. Although 3D-TCN-Att had a higher latency (1.062 s) compared to CNN-RNNs, it was still much shorter than the waiting time for integrating received data (40 s), which implies that mainstream computers can support the proposed method in real-time scenarios. The Conv3D enabled 3D-TCN-Att to extract and process spatial-temporal features simultaneously, resulting in better performance at the cost of increased computation. Using specially designed 2D CNN to learn spatial-temporal representations can be a potential solution to mitigate the cost [51].

The performance of using RD Cube data was higher than that of using DT map data. This result is consistent with our hypothesis in Section III-C, which suggests that the range information between head and hand is an important feature for radar-based food intake detection.

The performance of drinking gesture detection was lower than that of eating gesture detection. One reason is that the amount of drinking gestures was much less than that of eating (893 vs. 4132). Meanwhile, the detection performance for the gesture of eating with hand was the lowest. One potential reason is that there is more variety in eating with hands. Moreover, by investigating misclassification cases, we found that 3.43% of the gestures belonging to eating with hand were predicted as drinking, which was higher than the other three eating gestures (0.37% for fork & knife, 0.14% for chopsticks, 0.30% for spoon). This observation suggests that the gesture of eating with hand is similar to the drinking gesture.

We investigated the meal that obtains the lowest F1-score (the lowest outlier in Fig. 10) by checking the video. Two typical wrong cases were selected, as shown in Fig. 11. For the FP example, the participant used one hand to touch his nose, while using the other hand to hold the sandwich close to his mouth, which was similar to eating sandwich with hand. For the FN example, the participant mainly used his wrists to rotate the sandwich to his mouth, no obvious hand/arm movement.

Several sensors have been proposed for food intake monitoring, each with its own advantages and limitations. Despite the benefits of radar discussed in Section I, there are still several technical challenges inherent to radar sensor. Firstly, radar data requires complex signal processing procedures compared to IMU and video data. Secondly, there is less context information in radar compared to video. Thirdly, for radar sensor, recognizing eating gestures from multiple people is more challenging if people sit closely and eat at same time. Using an FMCW radar

Authorized licensed use limited to: Wageningen UR. Downloaded on March 01,2024 at 10:23:46 UTC from IEEE Xplore. Restrictions apply.

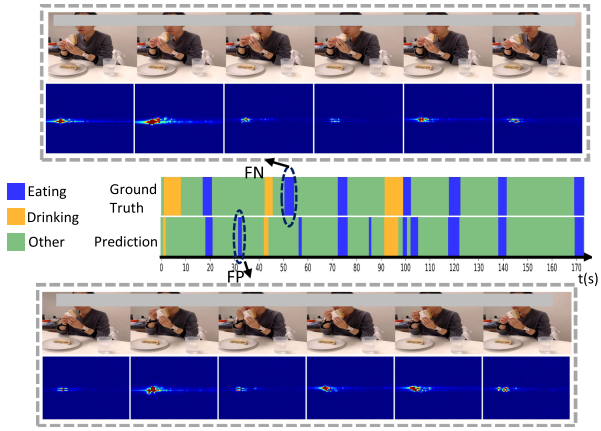


Fig. 11. Failure examples for the lowest outlier meal in Fig. 10. In the FP example, the participant used his right hand to touch nose while using the left hand to hold the sandwich, which was similar to delivering food to mouth by hand. In the FN example, the participant rotated his wrists to eat food without obvious arm and body movement.

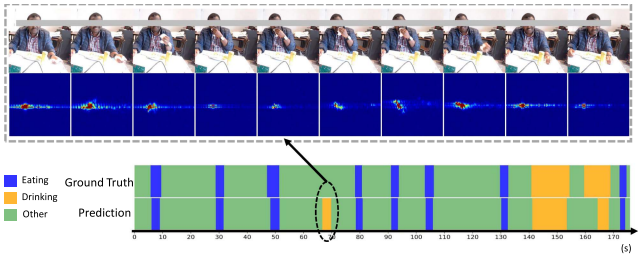


Fig. 12. Example of failure case. The participant was wiping mouth with tissues. The model predicted it as a drinking gesture.

for food intake monitoring has yet to be explored broadly. We propose this method as a good supplement to smartwatch-based and video-based approaches. One application scenario is in-door activity recognition, since video monitoring at home has higher privacy concern and people could not wear a smartwatch all the time. Apart from food intake monitoring solely, this method can also be explored to assess upper limb tremors of the elderly continuously and unobtrusively in real-life meal sessions [52].

We collected the dataset at the university, not in a home-like environment. However, since we applied static object removal, the stationary layout information was removed in this step. The results on the data collected in seven different rooms with different layouts indicated that the proposed method can detect eating gestures in different environments.

However, there are still few limitations in this research. Firstly, the data were collected in static environments, which means there were no other moving objects/subjects in the ROI of radar during data collection. The performance in environments containing other moving objects and persons still needs further exploration. A typical radar noise for indoor scenarios caused by multipath problem [53] should also be considered in further study. Secondly, we observed that the model occasionally misclassifies certain deceptive movements as intake gestures. For instance, actions like wiping the mouth with a napkin, which was similar to drinking gesture from radar signal, as illustrated in Fig. 12. Thirdly, this study mainly concentrated on the single-subject scenario. The multiple-subject eating scenario should be explored as future work. One potential solution is to incorporate an additional feature dimension, namely the angle information provided by the multiple-input-multiple-output (MIMO) radar. By integrating this information, the Range-Doppler-Angle-Time 4D data can be used to recognize activities from multiple subjects.

VI. Conclusion

In this research, we proposed Eat-Radar, a continuous food intake detection system, using an FMCW radar sensor and a seq2seq 3D-TCN-Att model. The detection and segmentation of eating and drinking gestures were realized by processing RD Cube from the radar. The effectiveness of this approach was validated on a collected dataset containing 70 meal sessions. The results from the proposed approach and other existing methods explored in our work demonstrated the feasibility of using FMCW radar for food intake monitoring. The proposed end-to-end framework including the seq2seq deep learning model and the segment-wise evaluation method has the potential to be applied in other radar-based HAR research.

Acknowledgment

The authors would like to thank the participants who participated in the experiments for their efforts and time.

References

[1] "Obesity and overweight," Accessed: Jun. 5, 2022. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/obesity-and-overweight

[2] L. M. Donini et al., "Malnutrition in elderly: Social and economic determinants," *J. Nutr. Heal. Aging*, vol. 17, no. 1, pp. 9–15, 2013.

[3] W. H. Organization, "Obesity: Preventing and managing the global epidemic," in *Report of a WHO Consultation.* Geneva, Switzerland: World Health Org., 2000.

[4] P. V. Rouast and M. T. P. Adam, "Single-stage intake gesture detection using CTC loss and extended prefix beam search," *IEEE J. Biomed. Heal. Informat.*, vol. 25, no. 7, pp. 2733–2743, Jul. 2021.

[5] P. V. Rouast and M. T. P. Adam, "Learning deep representations for video-based intake gesture detection," *IEEE J. Biomed. Heal. Informat.*, vol. 24, no. 6, pp. 1727–1737, Jun. 2020.

[6] J. Qiu, F. P. W. Lo, S. Jiang, Y. Y. Tsai, Y. Sun, and B. Lo, "Counting bites and recognizing consumed food from videos for passive dietary monitoring," *IEEE J. Biomed. Heal. Informat.*, vol. 25, no. 5, pp. 1471–1482, May 2021.

[7] 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. Heal. Informat.*, vol. 25, no. 1, pp. 22–34, Jan. 2021.

[8] C. Wang, T. S. Kumar, W. De Raedt, G. Camps, H. Hallez, and B. Vanrumste, "Drinking gesture detection using wrist-worn IMU sensors with multi-stage temporal convolutional network in free-living environments," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2022, pp. 1778–1782.

[9] S. Zhang, R. Alharbi, W. Stogin, M. Pourhomayun, B. Spring, and N. Alshurafa, "Food watch: Detecting and characterizing eating episodes through feeding gestures," in *Proc. BodyNets Int. Conf. Body Area Netw.*, 2017, pp. 91–96.

[10] A. Doulah, T. Ghosh, D. Hossain, M. H. Imtiaz, and E. Sazonov, ""Automatic ingestion monitor version 2" – A novel wearable device for automatic food intake detection and passive capture of food images," *IEEE J. Biomed. Heal. Informat.*, vol. 25, no. 2, pp. 568–576, Feb. 2021.

[11] K. S. Lee, "Joint audio-ultrasound food recognition for noisy environments," *IEEE J. Biomed. Heal. Informat.*, vol. 24, no. 5, pp. 1477–1489, May 2020.

[12] S. Zhang et al., "NeckSense: A multi-sensor necklace for detecting eating activities in free-living conditions," *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 4, no. 2, pp. 1–26, 2020.

[13] G. Mertes, L. Ding, W. Chen, H. Hallez, J. Jia, and B. Vanrumste, "Measuring and localizing individual bites using a sensor augmented plate during unrestricted eating for the aging population," *IEEE J. Biomed. Heal. Informat.*, vol. 24, no. 5, pp. 1509–1518, May 2020.

Authorized licensed use limited to: Wageningen UR. Downloaded on March 01,2024 at 10:23:46 UTC from IEEE Xplore. Restrictions apply.

[14] M. P. Lasschuijt et al., "Concept development and use of an automated food intake and eating behavior assessment method," *J. Vis. Exp.*, vol. 168, 2021, Art. no. e62144.
[15] B. Zhou et al., "Smart table surface: A novel approach to pervasive dining monitoring," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Smart*, 2015, pp. 155–162.
[16] H. Zhao et al., "A noncontact breathing disorder recognition system using 2.4-GHz digital-IF doppler radar," *IEEE J. Biomed. Heal. Informat.*, vol. 23, no. 1, pp. 208–217, Jan. 2019.
[17] J. Maitre, K. Bouchard, and S. Gaboury, "Fall detection with UWB radars and CNN-LSTM architecture," *IEEE J. Biomed. Heal. Informat.*, vol. 25, no. 4, pp. 1273–1283, Apr. 2021.
[18] Z. Sun, Q. Ke, H. Rahmani, M. Bennamoun, G. Wang, and J. Liu, "Human action recognition from various data modalities: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3200–3225, Mar. 2023.
[19] F. Luo, S. Poslad, and E. Bodanese, "Kitchen activity detection for healthcare using a low-power radar-enabled sensor network," in *Proc. IEEE Int. Conf. Commun.*, 2019, pp. 1–7.
[20] A. Gorji, T. Gielen, M. Bauduin, H. Sahli, and A. Bourdoux, "A multi-radar architecture for human activity recognition in indoor kitchen environments," in *Proc. IEEE Radar Conf.*, 2021, pp. 1–6.
[21] J. Maitre, K. Bouchard, C. Bertuglia, and S. Gaboury, "Recognizing activities of daily living from UWB radars and deep learning," *Expert Syst. Appl.*, vol. 164, 2021, Art. no. 113994.
[22] Y. Xie, R. Jiang, X. Guo, Y. Wang, J. Cheng, and Y. Chen, "mmEat: Millimeter wave-enabled environment-invariant eating behavior monitoring," *Smart Heal.*, vol. 23, 2022, Art. no. 100236.
[23] Y. Wu, Y. Chen, S. Shirmohammadi, and C. Hsu, "AI-assisted food intake activity recognition using 3D mmWave radars," in *Proc. 7th Int. Workshop Multimedia Assist. Dietary Manage.*, 2022, pp. 81–89.
[24] S. Zhu, R. G. Guendel, A. Yarovoy, and F. Fioranelli, "Continuous human activity recognition with distributed radar sensor networks and CNN–RNN architectures," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art no. 5115215.
[25] M. Piriyajitakonkij et al., "SleepPoseNet: Multi-view learning for sleep postural transition recognition using UWB," *IEEE J. Biomed. Heal. Informat.*, vol. 25, no. 4, pp. 1305–1314, Apr. 2021.
[26] Z. Ni and B. Huang, "Gait-based person identification and intruder detection using mm-wave sensing in multi-person scenario," *IEEE Sens. J.*, vol. 22, no. 10, pp. 9713–9723, May 2022.
[27] B. Vandersmissen et al., "Indoor person identification using a low-power FMCW radar," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 7, pp. 3941–3952, Jul. 2018.
[28] H. Sloetjes and P. Wittenburg, "Annotation by category - ELAN and ISO DCR," in *Proc. 6th Int. Conf. Lang Resour. Eval. Lr.*, 2008, pp. 816–820.
[29] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," in *Proc. 30th IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1003–1012.
[30] B. Filtjens, P. Ginis, A. Nieuwboer, P. Slaets, and B. Vanrumste, "Automated freezing of gait assessment with marker-based motion capture and multi-stage spatial-temporal graph convolutional neural networks," *J. Neuroeng. Rehabil.*, vol. 19, no. 1, pp. 1–14, 2022.
[31] D. Jarrett, J. Yoon, and M. Van Der Schaar, "Dynamic prediction in clinical survival analysis using temporal convolutional networks," *IEEE J. Biomed. Heal. Informat.*, vol. 24, no. 2, pp. 424–436, Feb. 2020.
[32] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, *arXiv:1803.01271*.
[33] A. Vaswani, N. Shazeer, N. Parmar, and J. Uszkoreit, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[34] J. Selva, A. S. Johansen, S. Escalera, K. Nasrollahi, T. B. Moeslund, and A. Clapes, "Video transformers: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 12922–12943, Nov. 2023.
[35] X. Zhang, L. Han, W. Zhu, L. Sun, and D. Zhang, "An explainable 3D residual self-attention deep neural network for joint atrophy localization and Alzheimer's disease diagnosis using structural MRI," *IEEE J. Biomed. Heal. Informat.*, vol. 26, no. 11, pp. 5289–5297, Nov. 2022.
[36] S. P. Singh, M. K. Sharma, A. Lay-Ekuakille, D. Gangwar, and S. Gupta, "Deep convLSTM with self-attention for human activity decoding using wearable sensors," *IEEE Sens. J.*, vol. 21, no. 6, pp. 8575–8582, Mar. 2021.
[37] R. Xue and X. Bai, "2D-Temporal convolution for target recognition of SAR sequence image," in *Proc. IEEE 6th Asia-Pacific Conf. Synth. Aperture Radar*, 2019, pp. 1–4.
[38] Y. A. Farha and J. Gall, "MS-TCN: Multi-stage temporal convolutional network for action segmentation," in *Proc. 32th IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3570–3579.
[39] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. Baik, "Action recognition in video sequences using deep bi-directional LSTM with CNN features," *IEEE Access*, vol. 6, pp. 1155–1166, 2018.
[40] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and ImageNet ?," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6546–6555.
[41] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 6201–6210.
[42] C. Feichtenhofer, "X3D: Expanding architectures for efficient video recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 200–210.
[43] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6450–6459.
[44] J. Carreira and A. Zisserman, "Quo Vadis, action recognition? A new model and the Kinetics dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6299–6308.
[45] M. Arnab, G. Dehghani, C. Heigold, M. S. Lučić, and C. Schmid, "ViViT: A video vision transformer," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 6816–6826.
[46] X. Li, Y. He, and X. Jing, "A survey of deep learning-based human activity recognition in radar," *Remote Sens.*, vol. 11, no. 9, 2019, Art. no. 1068.
[47] Y. Shen, J. Salley, E. Muth, and A. Hoover, "Assessing the accuracy of a wrist motion tracking method for counting bites across demographic and food variables," *IEEE J. Biomed. Heal. Informat.*, vol. 21, no. 3, pp. 599–606, May 2017.
[48] D. Hossain, T. Ghosh, and E. Sazonov, "Automatic count of bites and chews from videos of eating episodes," *IEEE Access*, vol. 8, pp. 101934–101945, 2020.
[49] E. S. Sazonov, O. Makeyev, S. Schuckers, P. Lopez-Meyer, E. L. Melanson, and M. R. Neuman, "Automatic detection of swallowing events by acoustical means for applications of monitoring of ingestive behavior," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 3, pp. 626–633, Mar. 2010.
[50] K. Lee, "Food intake detection using ultrasonic doppler sonar," *IEEE Sens. J.*, vol. 17, no. 18, pp. 6056–6068, Sep. 2017.
[51] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3D residual networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5534–5542.
[52] N. Gillani and T. Arslan, "Unobtrusive detection and monitoring of tremors using non-contact radar sensor," in *Proc. 4th Int. Conf. Bio- Eng. Smart Technol.(BioSMART)*, 2021, pp. 1–4.
[53] R. Feng, E. De Greef, M. Rykunov, H. Sahli, S. Pollin, and A. Bourdoux, "Multipath ghost recognition for indoor MIMO radar," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5104610.

Authorized licensed use limited to: Wageningen UR. Downloaded on March 01,2024 at 10:23:46 UTC from IEEE Xplore. Restrictions apply.