Real-Time Macro Gesture Recognition using Efficient Empirical Feature Extraction with Millimeter-Wave Technology

Alexandros Ninos, Jürgen Hasch, and Thomas Zwick

Abstract—Human Machine Interaction based on air gestures finds an increasing number of applications in consumer electronics. The availability of mmWave technology, combined with machine learning, allows the detection and classification of gestures, avoiding high-resolution LIDAR or video sensors. Nevertheless, in most of the existing studies, the processing takes place offline, takes into account only the velocity and distance of the moving arm, and can handle only gestures that are conducted very close to the sensor device, which limits the range of possible applications. Here, we use an experimental multi-channel mmWave-based system that can detect small targets, like a moving arm, up to a few meters away from the sensor. As our pipeline can estimate and take into account the angle of arrival in azimuth and elevation, it has the ability to classify a greater variety of dynamic gestures. Furthermore, the digital signal processing chain we present here, runs in real-time, incorporating an event detector. Whenever an event is detected, a novel empirical feature extraction takes place and a Multi-Layer Perceptron is deployed to infer the type of the gesture. To evaluate our setup and signal processing pipeline, a dataset with ten subjects, performing nine gestures was recorded. Our method yielded 94.3% accuracy on the test set, indicating a successful combination of our proposed sensor and signal processing pipeline for real time applications.

Index Terms—gesture recognition, empirical feature extraction, real-time, FMCW Radar

I. INTRODUCTION

INTERACTION with digital devices has been achieved with a variety of practices throughout the years, such as physical manipulation of proxy remote devices (e.g. wireless mouse) or direct physical interaction with the input device (e.g. touch screens) [1]. Despite the ease that we have earned in using these devices through direct contact, there is an increasing need for contactless interaction via gesture recognition. Varying ways have been proposed to implement this, including the attachment of sensing devices on the hand. Such an approach is sub-optimal as it adds additional peripherals to the equation [2]. Recent developments in computer vision allowed the analysis and the identification of hand motion in real time, using cameras [3]. However, in such setups the system requires a significant amount of power, which often makes it unsuitable for mobile devices or for long-term usage. Since miniaturized low-power Radar sensors became affordable [4], there was a considerable interest in the development of consumer applications. Google Soli was the first project to demonstrate that hand gesture recognition is possible with such a technology [5].

Some studies [6], [7] tackle the above problem by using continuous wave (CW) Radars, which can measure the radial-velocity between sensor and moving arm, but not the range.

Others [5], make use of Frequency Modulated Continuous Wave (FMCW) systems, which additionally take advantage of the range information.

The micro-Doppler (mD) effect is caused by signals returned from a target that incorporates vibrating or rotating structure [8]. A moving arm is such a target and in many studies a mD spectrogram, which contains the signature of the movement, is utilized. In [5], [7], [9], [10] empirical features are developed to extract as much information from the spectrogram as possible, including the length of a gesture, the ratio of negative to positive Doppler frequency, the bandwidth of mD frequency and the spectral power distribution. In [11], the authors make use of the fact that such a spectrogram is sparse and apply the Orthogonal Matching Pursuit (OMP) algorithm to extract features. Convolutional Neural Networks (CNN) are also widely used for the feature extraction of mD, reporting improved results [6], [12].

Other studies [13]–[15] use Range Doppler Maps (RDM), generated from FMCW Radars, which contain only spatial information of the movement. In these cases, many consecutive images need to be processed in order to get the temporal information. In these studies, a CNN is used for the feature extraction and Long Short Memory Cells (LSTM) are utilized for modeling the dynamics of a gesture. This comes at a cost of running a machine learning network in every frame, as well as with a high memory footprint for the LSTM. However, it combines information from the range and the radial velocity of the targets, whereas during the generation of mD images, the range information is discarded.

Using the Angle of Arrival (AoA) of the detected targets is

This paper was submitted for review on February 22, 2021

A. Ninos and J. Hasch are with Robert Bosch GmbH, Corporate Sector Research and Advance Engineering, 70465 Stuttgart, Germany (e-mail: alexandros.ninos@de.bosch.com).

T. Zwick is with Institut für Hochfrequenztechnik und Elektronik, Karlsruhe Institute of Technology (KIT), 76131 Karlsruhe, Germany.

crucial [16]. In [17] the authors developed a multi-static Radar with four receiving antennas that generated four mD images which were employed by a novel CNN architecture. Since the antennas are placed a few cm apart, each spectrogram depends on the AoA, which is exploited by the classifier to improve the accuracy. Using AoA from collocated antennas is also possible, as shown by [18]. The authors created a new heat-map with the same shape as the mD image but instead of the magnitude, they depict the AoA in each bin. Then, a custom CNN is used for feature extraction on the stacked images. Moreover, [12], [19] created three feature maps from the Radar data, Range-Time, Doppler-Time and Angle-Time by integrating range, Doppler and AoA in the time domain respectively. Then, these maps were simultaneously sent to a CNN for feature extraction. In [20] the authors also embedded AoA by creating two extra maps, one for elevation and one for azimuth. Initially, the authors calculated mD signatures by integrating the range dimension using multiple frames. Then, they calculated the phase differences for each measurement cycle, using two receiving antennas for each dimension. Moreover, they created images with the same shape as a mD map by integrating the range dimension of the phase differences. Finally yet importantly, [16] combined five attributes of the K most important bins of the Range Doppler map, in L measurement cycles to generate a feature cube $(5, K, L)$. Two of the five attributes contained information from elevation and azimuth of the detected target. Thus, their classifier in the next stage could identify 3D gestures with a high accuracy.

Automatic detection of a gesture is a pivotal part of a real-time recognition system. [13], [14] achieved that by using a CNN combined with an LSTM. In [20] a Faster R-CNN object detection framework was used to identify the Region of Interest in the feature map that could contain a gesture and send it to the classifier. On the other hand, in [16], [21] the power content of the returning signal in each frame was used to identify if a significant event took place.

The aforementioned work focused on the detection and identification of micro-gestures, which are small hand or finger movements performed only a few centimeters away from the sensing device. On the contrary, we developed a system that is capable of recognizing macro-gestures, so that a user could interact with the device using his/her arm at a distance of a few meters. Figure 1 exhibits such a typical interaction setup. Furthermore, our proposed system does not use a CNN or LSTM, for inferring the gesture type, which usually require hardware accelerators, but relies on empirical feature extraction which can be implemented in an embedded system. Finally yet importantly, we evaluate the effect of certain system parameters in the average accuracy of the classifier.

In this paper, we present a real time gesture recognition system that uses an experimental Radar sensor which utilizes four receiving and two transmitting antennas. The Digital Signal Processing (DSP) chain is depicted in Fig. 7. For each measurement frame, we apply 2D-FFT and Constant False Alarm Rate (CFAR) calculation [22], then mD is calculated using the bin with the highest amplitude in range dimension.

Fig. 1. Macro-gesture example, “swipe-up” in front of the sensing device.

The next step contains a significant-event detection algorithm, which finds a region of the mD stream that contains a motion. Since a mD map alone is not sufficient for classifying complex 3D gestures, we also add the AoA as input to the pipeline. That is why, in the selected mD region we calculate the AoA in elevation and azimuth and generate two new feature maps. For the last step we extract features as described in detail in Section II-D and use a small neural network for the classification.

The remainder of this paper is structured as follows: Section II introduces the experimental hardware setup that we used, the signal model, and the novel signal processing pipeline for gesture recognition. Section III explains the macro-gestures and more details on the recorded dataset. Finally, Section IV presents the results of our method and compares it against the state of the art.

II. RADAR SYSTEM

This section describes the experimental hardware, the modulation and the signal processing chain that was used for the data collection.

A. Hardware Setup

In this work, we use a two transmitter (TX) and four receiver (RX) FMCW Radar sensor. Fig. 2 shows a simplified block diagram consisting of a high frequency and a baseband part. The high frequency part is located on a separate printed circuit board (PCB) and is connected to the baseband with a connector. It includes the two-channel transmitter and four-channel receiver, a Voltage Controlled Oscillator (VCO) and a Phase-Locked Loop (PLL) capable of fast linear frequency ramps of up to 2 GHz bandwidth, as well as the transmit and receive antennas. The setup allows multiple input multiple output (MIMO), operating multiple transmitters in Time Domain Multiplexing (TDM). The system parameters of the Radar are given in table II.

The baseband board contains the analog interface electronics, analog to digital converters (ADC), digital logic, and power supply. The digital logic itself consists of an FPGA part, which controls the real-time operation of the Radar

Fig. 2. Simplified block diagram of the Radar sensor setup

Fig. 3. Radar hardware with frontend and baseband

sensor and an ARM micro-controller, which acquires the data and communicates with a host PC, which is connected over Ethernet. Fig. 3 shows an image of frontend and baseband.

The Radar sensor can operate in standalone mode, processing the Radar data on its own, or in slave mode, transmitting the measured Radar data to the PC without further processing. In our setup we choose slave mode and transmit the measured data directly to the PC for signal processing and classification.

The center frequency is at 77 GHz, which is suitable for automotive applications but not for commercial and industrial applications. However, the novel signal processing pipeline that we developed can be used by a sensor that operates at 60 GHz without modifications. Since the frequency differs by only 20%, no noticeable difference in reflectivity, velocity unambiguity or resolution is expected.

Last but not least, the antenna array consists of elements with three series fed patches and an additional matching structure at the feedline. The antennas are optimally positioned at a distance of approximately half lambda horizontally and the transmitters are placed accordingly at a distance of lambda to generate a MIMO virtual array. Thus, it is possible to calculate the AoA in azimuth and elevation.

B. FMCW Signal Model

Our Radar system creates a sequence of L chirps, using linear frequency modulation [23], with each of them lasting

Fig. 4. Linear FMCW with four chirps and return signal from one target

	Range	Radial Velocity
Resolution	$\frac{c}{2B_{\text{chirp}}}$	$\frac{\lambda}{2 \cdot L \cdot T_{\text{chirp}}}$
Maximum	$\frac{c \cdot K}{2B_{\text{chirp}}}$	$\frac{1}{2 \cdot T_{\text{chirp}}}$

TABLE I

RESOLUTION AND MAXIMUM UNAMBIGUOUS VALUE FOR RANGE AND RADIAL VELOCITY

T_{chirp} . The received and down-converted signal contains the beat f_B and Doppler frequency f_D together with the phase information ϕ . Fig. 4 shows a sequence with four chirps and the echo signal from a target near the Radar.

The following equation describes the baseband signal:

$$s(t, l) = e^{j2\pi(f_B t - f_D l T_{\text{chirp}} + \phi)} \quad (1)$$

where l is the chirp index. The above continuous time signal is sampled; for each chirp, FFT processing is applied, as shown in equation 2.

$$S(m, l) = \sum_{k=0}^{K-1} s(k, l) \cdot e^{-j2\pi \frac{k \cdot m}{K}} \quad (2)$$

Sample index is denoted with k , number of samples with K and beat frequency index with m . When this procedure is completed for all chirps, a second FFT is applied for each range gate, in order to estimate f_D . Equation 3 shows the result:

$$Q(m, n) = \sum_{l=0}^{L-1} S(m, l) \cdot e^{-j2\pi \frac{l \cdot n}{L}} \quad (3)$$

where n is the Doppler frequency index. As mentioned in [23] due to the two measurements for beat and Doppler frequency, the range R and radial velocity v_r for each detected target can be calculated as follows:

$$R = -(f_B + f_D) \frac{T_{\text{chirp}} \cdot c}{2 \cdot B_{\text{chirp}}} \quad (4)$$

$$v_r = -f_D \frac{\lambda}{2} \quad (5)$$

where B_{chirp} is equal to the chirp bandwidth and c is the speed of light. A more rigorous signal model can be found in [23]. Table I provides the formulas for calculating the maximum unambiguous range and radial velocity of a target, as well as their resolution.

Fig. 5. Ambiguity function in azimuth direction for experimental sensor with two transmitting and four receiving elements

C. Estimating the Angle of Arrival

The principle of Digital Beamforming (DBF) is a well established technique for estimating the AoA of the detected targets [24]. In short, let N be the number of receiving antennas placed in a one-dimensional axis. The direction of arrival for the incoming signal is the same for all of them, if plane waves impinge from the target. With that assumption, it is possible to calculate the steering vector for the angular direction θ as in equation 6:

$$w(\theta) = \begin{bmatrix} e^{j2\pi \frac{x_1 \sin \theta}{\lambda}} \\ \vdots \\ e^{j2\pi \frac{x_N \sin \theta}{\lambda}} \end{bmatrix} \quad (6)$$

λ being the wavelength, $k = 1, \dots, N$ the antenna index and x_k the position in the axis [25].

Hardware imperfections like differences in transmission line lengths, manufacturing uncertainties and variances in radio-frequency components, create the necessity for antenna calibration [26]. Following the tutorial in [27], we conducted two calibration measurements using a corner reflector and a rotating Radar in azimuth and elevation respectively. Thus, we were able to extract the steering vector for each direction. Fig. 5 shows the ambiguity function for azimuth; the highest value of the autocorrelation has been achieved along the main diagonal only when $\theta = [-60, 60]$. Therefore, there are no ambiguities for a field of view (FoV) of 120° . In a similar fashion, we found that the unambiguous FoV in elevation is 60° .

FMCW MIMO Radar is widely used in target detection due to the high angular resolution that it offers. A MIMO system that is consisted of M transmitting and N receiving elements could be equivalent, as far as the angular resolution is concerned, to an array with $M \cdot N$ receiving elements. For a MIMO Radar to easily separate the signals transmitted by different antennas, the most intuitive and simple way is TDM. In other words, each transmitter transmits its own waveform alternatively, and there is no overlap between any two transmissions [24], [28].

A typical example is shown in Fig. 6, in which the transmitting duration and pause between chirps is selected so that the total duration of two consecutive chirps is equal to that of

Fig. 6. Time Division Multiplexing using FMCW

Name	Value
Antenna Gain	10 dBi
Transmit power	12 dBm
FoV (az/el)	120/60°
Antennas (receiving/transmitting)	4/2
Center Frequency	76.75 GHz
Chirp duration	66 μs
Chirps per Tx Antenna	64
Break between chirps	50 μs
Bandwidth	1.5 GHz
Measurement duration	15 ms
Break between measurements	5 ms
Total duration	20 ms

TABLE II
SYSTEM PARAMETERS

Fig. 4. This way, the resolution and maximum unambiguous radial velocity will be the same in both cases.

The modulation that we used has a chirp duration for each transmitter of $66 \mu s$, a bandwidth of 1.5 GHz and 64 transmitted chirps per transmitting antenna. We also added a pause of $50 \mu s$ after each chirp in order to satisfy maximum unambiguous velocity and velocity resolution that we believe are suitable for gesture recognition application. Therefore, one complete measurement frame takes almost 15 ms. In order for the signal processing chain to run smoothly we added an extra 5 ms and the total sampling period reached 20 ms. The system parameters are summarized in Table II

D. Processing Pipeline for Gesture Recognition

After we get the recorded signal from the ADC of receiving channels, we apply a 2D-FFT for calculating the RDM and then we estimate the noise floor using a CFAR algorithm [22]. In the next step, we replace the bins of RDM that do not exceed the noise floor, with zeros. Then the mD vector is calculated by selecting at each range vector, the bin with the highest amplitude. We chose not to integrate over the range vector, as is usually the case in FMCW Radar, in order to be able to apply DBF in the selected bins. This way we significantly reduce the processing time for each frame, since DBF is applied only to a few bins of the mD vector. The above procedure is repeated for each measurement frame and the mD vectors are concatenated so as to create the mD image, which is sparse due to the CFAR thresholding. This image, constitutes the first out of the three feature maps that will be used in the next stages of the pipeline.

After that, we developed an algorithm which identifies if a significant event took place in order to initiate the classification process. According to this algorithm, if the maximum radial velocity in the mD image is higher than a given threshold for a certain amount of time, then an event is taking place. These two hyper-parameters depend on the event type that needs to be detected. For dynamic hand gestures we selected a velocity threshold equal to 0.8 m/s and a minimum duration of 200 ms . The machine learning model expects input with a certain shape, that is why we set the total event duration as a third hyper-parameter equal to 50 frames. However, this raises a problem in case the user completes a gesture faster than the pre-fixed value, because she/he would have to wait until the remaining frames are recorded and sent to the classifier [16]. Hence our event detector also searches for the ending point of the gesture, by checking if the maximum velocity is below 0.8 m/s for 200 ms . If the end of the gesture is detected, then the remaining frames of the event window are filled with zeros and sent to the next stage of the pipeline.

After event detection, we create two more feature-maps with the same shape as the mD image, whose amplitude refers to the AoA in azimuth and elevation respectively. Thus, in total three images transform the arm movement, as it was captured by the sensor, in a compact form that contains valuable information about the radial velocity and the AoA. We believe that the information from the range of the target is not as valuable as the radial velocity, that is why we did not use it in our work. In case there are multiple users in the FoV, then range information would become very important. A tracking algorithm needs to cluster the detections from the users into objects [29] and separately recognize the performed gesture for each tracked object. However, this is not within the scope of this paper.

Next, we developed an empirical method that generates an one-dimensional signal from each feature-map. For the mD image we use the highest velocity of each frame and for the two AoA maps, we use the median angle of each frame. At this stage, we have three one-dimensional signals which contain information on the radial velocity and the AoA in azimuth and elevation. For simplifying the classification even further, we decided to extract certain features out of the three signals. These are the following: (1) Number of zero-crossings in radial velocity, (2, 3) arguments of maximum and minimum radial velocity, (4, 5) maximum and minimum radial velocity, (6, 7, 8, 9) maximum and minimum angle in azimuth and elevation, (10, 11, 12, 13) angle in azimuth and elevation when radial velocity reached its maximum and minimum value. We use the above 13 values for training a Multi Layer Perceptron (MLP) with 32 neurons in the hidden layer [30]. Fig. 7 shows a diagram with the proposed signal processing pipeline.

III. GESTURES

Previous studies have focused on the detection of hand gestures conducted on top of a mmWave sensor at a distance of a few centimeters. This could be very useful for sensors integrated in a mobile phone, like the Google Pixel 4 [31]. However, our goal is to simulate a smart home application in which a device is attached on a wall or on a TV. That is why

Fig. 7. Signal processing chain for gesture recognition with mmWave sensor.

during data collection, the Radar was facing the room and each subject was located at a distance approximately 2 meters from it. We found that a range higher than two meters would deteriorate the results due to the relatively low transmitting power of the device and large FoV in azimuth and elevation.

We selected ten gestures for the subjects to perform: (0) Random movement/walk, (1) Pull, (2) Push, (3) Swipe up, (4) Swipe down, (5) Swipe right, (6) Swipe left, (7) Rotate, (8) Wave, (9) Push-Pull. We recruited 10 subjects with various heights and ages to perform the above gestures in an intuitive manner. Each subject had to repeat each gesture 15 times; we were able to collect $10 \times 10 \times 15 = 1500$ samples. In order to simulate a real scenario as good as possible, each subject conducted the gestures at three different aspect angles in relation to the device. The first location was in front of the sensor, the second around 30° to the left and the third around 30° to the right.

For the collection of the dataset, we developed a data logging application using the Bokeh library [32]. When the subject conducts a gesture, our DSP chain will detect it and plot the three feature maps. The administrator should either discard the gesture (in case that something went wrong) or select the gesture type that took place and save the data. In that case, the application saves the raw Radar data and the processed feature maps, then the detection process is resumed.

IV. RESULTS

A. Automatic Gesture Detection

Fig. 8 shows a mD image of five gestures and the output of the event detection algorithm. The first and second are a “push” and “pull” respectively, for which the algorithm correctly finds the start and end point. The third gesture is a “push-pull”; the algorithm manages to identify that it is a single event. On the other hand, the fourth and fifth gestures

Fig. 8. Micro Doppler stream with five different gestures, the red bounding box shows the start and end of the gesture as was estimated by our algorithm.

have a small break between them and the algorithm managed to correctly separate them.

It is important to point out that we fine-tuned the gesture detection part, so that small movements would be identified and would trigger the classifier. The latter is responsible for discarding random motions by classifying them as "Random movement".

B. Feature Maps

As already mentioned, feature maps are generated after an event is detected. Fig. 9 shows a typical example of the maps from a "pull" gesture, whereas Fig. 10 shows the ones from a "push". Even for a naked eye it is easy to extract important features, like the fact that target velocity is always positive during the former and always negative during the latter. It should be noted that in both cases the AoA in azimuth and elevation does not show a significant change. However, things are quite different in Figures 11 and 12 which show the results from a "swipe left" and a "swipe right" gesture respectively. In these cases, radial velocity changes sign and AoA in azimuth dimension has a significant change over time. However, when the subject performs a "swipe right" from a different position, for example 30° from bore-sight, then the radial velocity will not change sign and will look like a "pull". In that case the azimuth feature map will be important for the classification since it will be able to capture positive or negative slope of the AoA, as shown in Fig. 13. In a similar fashion, the elevation feature map is important for the "swipe up" and "swipe down" gestures. Therefore, the above heat-maps contain the needed information to correctly classify hand gestures.

Fig 14 shows the one-dimensional signal of the same "swipe right" as in Fig 13, after applying the method explained above. The first subplot contains the envelope of the mD image and the second subplot the median angle of each measurement frame. The increase in azimuth from -20 to 5 is an important feature that can be used at the next stage for classification.

C. Supervised Learning

During training we split the dataset in two parts, train and test. The train set contains 80% of the samples and the test set 20%. In order to prevent the model from over-fitting, we used weight regularization and dropout [30], [33]. Fig. 15 shows

Fig. 9. Feature maps generated from the DSP chain during a "pull" gesture. The first one refers to Micro-Doppler, the second one to AoA in azimuth and the last one in elevation.

Fig. 10. Feature maps generated from the DSP chain during a "push" gesture. The first one refers to Micro-Doppler, the second one to AoA in azimuth and the last one in elevation.

Fig. 11. Feature maps generated from the DSP chain during a "swipe left" gesture. The first one refers to Micro-Doppler, the second one to AoA in azimuth and the last one in elevation.

Fig. 12. Feature maps generated from the DSP chain during a "swipe right" gesture. The first one refers to Micro-Doppler, the second one to AoA in azimuth and the last one in elevation.

Fig. 13. Feature maps generated from the DSP chain during a “swipe right” gesture, subject was located 30° from bore-sight. The first one refers to Micro-Doppler, the second one to AoA in azimuth and the last one in elevation.

Fig. 14. Time-series generated from the DSP chain during a “swipe right” gesture. The first subplot refers to Micro-Doppler and the second one to AoA in azimuth and elevation.

the confusion matrix for the test set. The accuracy is higher than 90% for all classes and 94.3% overall, which shows that the signal processing chain transforms the data coming from the mmWave sensor, in a representation which is well-suited for training a MLP. We also used a CNN architecture for extracting features from the feature-maps but we could not achieve better results in comparison to the empirical feature extraction approach.

In addition, we investigated the effect of certain system parameters in the classification accuracy. In the first case, we evaluated the significance of high frame rate by using a 2x and 3x down-sampled dataset. Moreover, we assessed the importance of having two transmitting elements in comparison to one, also called Single Input Multiple Output (SIMO), which would reduce the amount of virtual antennas used for angle estimation by half. Last but not least, we combined the above cases and created a dataset that used one transmitting antenna and the frame rate was 2x down-sampled. Fig. 16 contains the confusion matrices for the above cases. It is worth pointing out that reducing the number of transmitters or decreasing the frame rate, reduces the power consumption and lowers the hardware cost. Results showed that decreasing the frame rate 3x, significantly decreased the average accuracy, which is a clear sign that a low frame rate cannot capture the dynamics of the arm movement. However, a system with frame rate of 25 Hz or one transmitting antenna achieves average accuracy above 90%. For certain consumer applications one could compromise with lower accuracy in order to achieve

Fig. 15. Confusion matrix with the original modulation. Accuracy in all classes is higher than 90% and 94.3% overall.

Frame Rate (Hz)	Transmitters	Average Accuracy (%)
50	2	94.3
25	2	92.0
17	2	87.6
50	1	90.3
25	1	88.3

TABLE III
AVERAGE ACCURACY OF THE TEST SET

lower cost or lower power consumption. Table III summarizes the average accuracy that was achieved in the test set under different approaches.

D. Deployment

For the real-time implementation, we used an Intel i7-8650U @1.9GHz. The median processing runtime is 10.1 ms when a gesture is not detected (2D-FFT, CFAR, Angle Estimation, micro-Doppler, event detection) and 40.3 ms it is detected. Therefore, we discard the next two frames that are waiting to be processed so that the system achieves real-time performance.

We selected a Raspberry Pi 4 as target hardware to deploy our application, since it has low cost and capacity to run the Python package that we developed. Results showed that median processing runtime in case that an event is not detected is 31.5 ms. According to Table III, if a frame rate equal to 25 Hz is selected, the average accuracy will drop by 2.3 %. On the same time, the allowed processing runtime can be up to 40 ms, more than enough for the Raspberry Pi to handle it. In case that an event is detected and the gesture recognition pipeline is triggered, the median runtime reaches 67.4 ms and two frames need to be discarded for achieving real-time performance.

In conclusion, if the designer is willing decrease the average accuracy by 2.3 %, our suggested method can be deployed in a low cost system. In case that a hardware accelerator is to be used, the low-level signal processing steps, which require more than 30 ms, could be transferred to it. Then a Raspberry Pi, or any similar micro-controller, would be responsible only

Fig. 16. Confusion matrices for different modulation parameters and overall accuracy of test set: A) frame rate decreased 2x, 92%, B) frame rate decreased 3x, 87.6% C) one transmitting antenna, 90.3% D) frame rate decreased 2x and one transmitting antenna, 88.3%.

Fig. 17. Confusion matrix for the feature encoder combined with CNN. Accuracy in all classes is higher than 84% and 90.5% overall.

	Our method	Feature Encoder
Average Accuracy	94.3	90.5
Memory Footprint (MB)	0.015	35.7
Floating point operations	1602	18.3M

TABLE IV
COMPARISON WITH STATE OF THE ART

the bins, like in our DSP chain. This way we made sure that only bins above the noise floor will be used for training the machine learning model. Also we used the last 50 frames like in our case and we used the same train and test set as with our DSP chain. Last but not least, we experimented with the number of bins used in each frame and found that 50 instead of 25 achieved the highest average accuracy in the test set, equal to 90.5%. Fig. 17 shows the confusion matrix for the different classes. Table IV summarizes the comparison between our method and the one from [16]. It is important to mention, that the number of floating point operations refer to the Machine Learning model. In our case, more calculations are needed for extracting the empirical features.

V. CONCLUSION

We developed a real-time mmWave based macro-gesture recognition system using an experimental setup with four receiving and two transmitting antennas. We recorded a dataset with nine gestures from ten different subjects, which were positioned approximately two meters from the Radar in three different locations. The proposed signal processing chain and feature extraction method transforms ADC data from the sensor into feature maps, extracts empirical features in a two-step approach, and feeds the result to an MLP for classification. Furthermore, a custom made event detection algorithm detects if a gesture took place and only then activates the machine learning part. Our method achieved 94.3% average accuracy. Furthermore, we evaluated the effect of two system parameters, the frame rate and the antenna number, in the classification accuracy. We found that decreasing the frame rate two times or using one transmitting antenna, does not significantly decrease the average accuracy, even though it

for the gesture recognition part; thus a frame rate of 50 Hz could be achieved.

E. Comparison with state of the art approach

In [16] a thorough comparison can be found between several machine learning approaches for gesture recognition with mmWave sensors. The “Multi-Feature encoder + CNN” yielded high accuracy and on the same time has as significantly smaller memory footprint in comparison to the rest. That is why we consider this method as state of the art and decided to implement it and train a model with the dataset that we collected.

In a nutshell, the algorithm sorts the bins of each Range Doppler Map (RDM) according to their amplitude, selects the first 25 and for each of them saves five attributes, amplitude, range, radial velocity, AoA in azimuth and elevation. If an event is detected, then it creates a tensor with the last 40 frames. The three-dimensional tensor is called feature cube and is used to train a CNN and a MLP. In order to make a fair comparison, we applied CFAR thresholding before selecting

can lower the cost of the sensor. Moreover, we deployed our method in a low-cost embedded system and found that a real-time performance is possible. Last but not least, we compared our method with the state of the art approach and found out that our model achieved higher accuracy in the test set, even though our model is significantly smaller.

REFERENCES

- [1] C. Gu, J. Wang, and J. Lien, "Motion Sensing Using Radar: Gesture Interaction and Beyond," *IEEE Microwave Magazine*, vol. 20, pp. 44–57, Aug. 2019.
- [2] B.-S. Lin, I.-J. Lee, P.-Y. Chiang, S.-Y. Huang, and C.-W. Peng, "A Modular Data Glove System for Finger and Hand Motion Capture Based on Inertial Sensors," *Journal of Medical and Biological Engineering*, vol. 39, pp. 532–540, Aug. 2019.
- [3] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz, "Online Detection and Classification of Dynamic Hand Gestures with Recurrent 3d Convolutional Neural Networks," pp. 4207–4215, IEEE, June 2016.
- [4] J. Hasch, E. Topak, R. Schnabel, T. Zwick, R. Weigel, and C. Waldschmidt, "Millimeter-Wave Technology for Automotive Radar Sensors in the 77 GHz Frequency Band," *IEEE Transactions on Microwave Theory and Techniques*, vol. 60, pp. 845–860, Mar. 2012.
- [5] J. Lien, N. Gillian, M. E. Karagozler, P. Amihoud, C. Schwesig, E. Olson, H. Raja, and I. Poupyrev, "Soli: Ubiquitous gesture sensing with millimeter wave radar," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, p. 142, 2016.
- [6] Y. Kim and B. Toomajian, "Hand Gesture Recognition Using Micro-Doppler Signatures With Convolutional Neural Network," *IEEE Access*, vol. 4, pp. 7125–7130, 2016.
- [7] M. G. Amin, Z. Zeng, and T. Shan, "Hand Gesture Recognition based on Radar Micro-Doppler Signature Envelopes," *arXiv preprint arXiv:1811.12467*, 2018.
- [8] V. C. Chen, F. Li, S.-S. Ho, and H. Wechsler, "Micro-Doppler effect in radar: phenomenon, model, and simulation study," *IEEE Transactions on Aerospace and electronic systems*, vol. 42, no. 1, pp. 2–21, 2006.
- [9] S. Lan, Z. He, K. Yao, and W. Chen, "Hand Gesture Recognition using a Three-dimensional 24 GHz Radar Array," in *IEEE/MTT-S International Microwave Symposium-IMS*, pp. 138–140, IEEE, 2018.
- [10] M. Ritchie and A. M. Jones, "Micro-Doppler Gesture Recognition using Doppler, Time and Range Based Features," pp. 1–6, IEEE, Apr. 2019.
- [11] G. Li, R. Zhang, M. Ritchie, and H. Griffiths, "Sparsity-based dynamic hand gesture recognition using micro-Doppler signatures," in *IEEE Radar Conference (RadarConf)*, pp. 0928–0931, IEEE, 2017.
- [12] X. Cai, J. Ma, W. Liu, H. Han, and L. Ma, "Efficient convolutional neural network for FMCW radar based hand gesture recognition," pp. 17–20, ACM Press, 2019.
- [13] S. Wang, J. Song, J. Lien, I. Poupyrev, and O. Hilliges, "Interacting with Soli: Exploring Fine-Grained Dynamic Gesture Recognition in the Radio-Frequency Spectrum," pp. 851–860, ACM Press, 2016.
- [14] Z. Zhang, Z. Tian, and M. Zhou, "Latern: Dynamic Continuous Hand Gesture Recognition Using FMCW Radar Sensor," *IEEE Sensors Journal*, vol. 18, pp. 3278–3289, Apr. 2018.
- [15] Y. Wang, S. Wang, M. Zhou, Q. Jiang, and Z. Tian, "TS-I3d Based Hand Gesture Recognition Method With Radar Sensor," *IEEE Access*, vol. 7, pp. 22902–22913, 2019.
- [16] Y. Sun, T. Fei, X. Li, A. Warnecke, E. Warsitz, and N. Pohl, "Real-Time Radar-Based Gesture Detection and Recognition Built in an Edge-Computing Platform," *IEEE Sensors Journal*, pp. 1–1, 2020.
- [17] Z. Chen, G. Li, F. Fioranelli, and H. Griffiths, "Dynamic Hand Gesture Classification Based on Multistatic Radar Micro-Doppler Signatures Using Convolutional Neural Network," pp. 1–5, IEEE, Apr. 2019.
- [18] S. Skaria, A. Al-Hourani, M. Lech, and R. J. Evans, "Hand-Gesture Recognition Using Two-Antenna Doppler Radar With Deep Convolutional Neural Networks," *IEEE Sensors Journal*, vol. 19, pp. 3041–3048, Apr. 2019.
- [19] Y. Wang, X. Jia, M. Zhou, X. Yang, and Z. Tian, "Rammar: RAM Assisted Mask R-CNN for FMCW Sensor Based HGD System," pp. 1–6, IEEE, May 2019.
- [20] Y. Sun, T. Fei, S. Gao, and N. Pohl, "Automatic Radar-based Gesture Detection and Classification via a Region-based Deep Convolutional Neural Network," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4300–4304, IEEE, 2019.
- [21] L. Liang, M. Popescu, M. Skubic, M. Rantz, T. Yardibi, and P. Cuddihy, "Automatic Fall Detection Based on Doppler Radar Motion Signature," IEEE, 2011.
- [22] M. A. Richards, *Fundamentals of radar signal processing*. New York: McGraw-Hill Education, second edition ed., 2014.
- [23] M. Kronauge and H. Rohling, "New chirp sequence radar waveform," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, pp. 2870–2877, Oct. 2014.
- [24] G. Hakobyan and B. Yang, "High-Performance Automotive Radar: A Review of Signal Processing Algorithms and Modulation Schemes," *IEEE Signal Processing Magazine*, vol. 36, pp. 32–44, Sept. 2019.
- [25] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE signal processing magazine*, vol. 13, no. 4, pp. 67–94, 1996.
- [26] M. Harter, J. Hildebrandt, A. Ziroff, and T. Zwick, "Self-Calibration of a 3-D-Digital Beamforming Radar System for Automotive Applications With Installation Behind Automotive Covers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 64, pp. 2994–3000, Sept. 2016.
- [27] C. Vasanelli, F. Roos, A. Durr, J. Schlichenmaier, P. Hugler, B. Meinel, M. Steiner, and C. Waldschmidt, "Calibration and Direction-of-Arrival Estimation of Millimeter-Wave Radars: A Practical Introduction," *IEEE Antennas and Propagation Magazine*, pp. 0–0, 2020.
- [28] A. Zwanetski and H. Rohling, "Continuous wave MIMO radar based on time division multiplexing," in *13th International Radar Symposium*, pp. 119–121, IEEE, 2012.
- [29] T. Wagner, R. Feger, and A. Stelzer, "Radar Signal Processing for Jointly Estimating Tracks and Micro-Doppler Signatures," *IEEE Access*, vol. 5, pp. 1220–1238, 2017.
- [30] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. Adaptive computation and machine learning, Cambridge, Massachusetts: The MIT Press, 2016.
- [31] B. Rakowski, "Pixel 4 is here to help," 2019.
- [32] B. D. Team, "Python library for interactive visualization," 2020.
- [33] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

Alexandros Ninou received his Diploma in Electrical and Computer Engineering from the National Technical University of Athens in 2014. He is currently pursuing the Dr.-Ing degree in electrical engineering with the Institut für Hochfrequenztechnik und Elektronik (IHE) at the KIT, Germany and the corporate research of the Robert Bosch GmbH, Germany. His research interests are Radar signal processing, mm-wave based human motion recognition and machine learning.

Jürgen Hasch (Senior Member, IEEE) received the Dipl.-Ing. and Dr.-Ing. degrees from the University of Stuttgart, Germany, in 1996 and 2007, respectively. He is a Senior Expert on RF Technology at the Corporate Research of Robert Bosch GmbH in Renningen, Germany. He has authored more than 40 scientific papers and holds more than 20 patents. His main research interests include RF-based sensing technologies, integrated millimeter wave sensors in the 60–240 GHz range. Dr. Hasch currently serves

as Chair of the IEEE MTT-27 Technical Committee of Connected and Autonomous Systems and a member of ITG-FA Mikrowellentechnik. He serves as a reviewer for several journals and conferences.

Thomas Zwick (S'95–M'00–SM'06) received the Dipl.-Ing. (M.S.E.E.) and the Dr.-Ing. (Ph.D.E.E.) degrees from the Universität Karlsruhe (TH), Germany, in 1994 and 1999, respectively. From 1994 to 2001 he was a research assistant at the Institut für Höchstfrequenztechnik und Elektronik (IHE) at the Universität Karlsruhe (TH), Germany. In February 2001 he joined IBM as research staff member at the IBM T. J. Watson Research Center, Yorktown Heights, NY, USA. From October 2004 to September 2007,

Thomas Zwick was with Siemens AG, Lindau, Germany. During this period he managed the RF development team for automotive Radars. In October 2007, he became a full professor at the Karlsruhe Institute of Technology (KIT), Germany. He is the director of the Institut für Hochfrequenztechnik und Elektronik (IHE) at the KIT.

Thomas Zwick is author or co-author of over 400 technical papers and 20 patents. Together with his team he received over 20 best paper awards on international conferences. He served on the technical program committees (TPC) of several scientific conferences. In 2013 Dr. Zwick was general chair of the international Workshop on Antenna Technology (iWAT 2013) in Karlsruhe and in 2015 of the IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM) in Heidelberg. He also was TPC chair of the European Microwave Conference (EuMC) 2013. In 2017 he was General TPC Chair of European Microwave Week. From 2008 until 2015 he has been president of the Institute for Microwaves and Antennas (IMA). T. Zwick became selected as a distinguished microwave lecturer for the 2013 – 2015 period with his lecture on "QFN Based Packaging Concepts for Millimeter-Wave Transceivers". Since 2017 he is member of the Heidelberg Academy of Sciences and Humanities. In 2018 Thomas Zwick became appointed IEEE Fellow, since 2019 he is member of acatech.