

FORTE: Few Samples for Recognizing Hand Gestures with a Smartphone-attached Radar

STEFANO CHIOCCARELLO, University of Padova, Italy

ARTHUR SLUYTERS, Université catholique de Louvain, Belgium

ALBERTO TESTOLIN, University of Padova, Italy

JEAN VANDERDONCKT, Université catholique de Louvain, Belgium

SÉBASTIEN LAMBOT, Université catholique de Louvain, Belgium

Radar sensing technologies offer several advantages over other gesture input modalities, such as the ability to reliably sense human movements, a reasonable deployment cost, insensitivity to ambient conditions such as light, temperature, and the ability to preserve anonymity. These advantages come at the price of high processing complexity mainly due to the spatio-temporal variations of gesture articulation performed by different people. Deep learning methods, such as CNN-LSTM and 3D CNN-LSTM, have a high potential to recognize radar-based gestures but usually require hundreds or thousands of labeled training samples and high processing power. Asking a lot of people to acquire a lot of gestures is particularly tedious and tiring to the point of being unrealistic. To overcome these challenges, we propose FORTE, a hand gesture recognition with few samples based on an optimized CNN architecture working on pre-processed raw data. Using a $k=5$ -fold cross-validation, we define and compare three alternative CNNs for recognizing hand gestures acquired in a semi-mobile context of use with a portable radar attached to a smartphone. The best CNN reaches an accuracy of 94.96% with a precision of 95.92% and a recall of 96.03% for a dataset composed of solely 5 participants producing 2 samples for 20 classes covering 1 pointing, 2 pantomimic, 3 iconic, and 14 semaphoric gestures. We suggest some implications for designing radar-based gestures and we discuss the limitations of this approach.

CCS Concepts: • **Human-centered computing** → **Gestural input**; **Graphical user interfaces**; *Interactive systems and tools*; • **Computing methodologies** → *Neural networks*; *Cross-validation*; • **Software and its engineering** → Runtime environments; • **Hardware** → Radio frequency and wireless interconnect.

Additional Key Words and Phrases: Convolutional Neural Network, Gesture-based interfaces, Gesture dataset, Hand gesture recognition, Mid-air gestures, New datasets, Radar-based interaction, Radar sensing

ACM Reference Format:

Stefano Chioccarello, Arthur Sluyters, Alberto Testolin, Jean Vanderdonckt, and Sébastien Lambot. 2023. FORTE: Few Samples for Recognizing Hand Gestures with a Smartphone-attached Radar. *Proc. ACM Hum.-Comput. Interact.* 7, EICS, Article 179 (June 2023), 25 pages. <https://doi.org/10.1145/3593231>

Authors' addresses: [Stefano Chioccarello](#), stefano.chioccarello.1@studenti.unipd.it, University of Padova, Department of Information Engineering, Via Gradenigo, 6/B, Padova, 35122, Italy; [Arthur Sluyters](#), arthur.sluyters@uclouvain.be, Université catholique de Louvain, Louvain Research Institute in Management and Organizations, Place des Doyens, 1, Louvain-la-Neuve, 1348, Belgium; [Alberto Testolin](#), alberto.testolin@unipd.it, University of Padova, Department of General Psychology and Dept. of Mathematics, via Venezia, 8, Padova, 35131, Italy; [Jean Vanderdonckt](#), jean.vanderdonckt@uclouvain.be, Université catholique de Louvain, Louvain Research Institute in Management and Organizations, Place des Doyens, 1, Louvain-la-Neuve, 1348, Belgium; [Sébastien Lambot](#), sebastien.Lambot@uclouvain.be, Université catholique de Louvain, Earth and Life Institute, Croix du Sud, 2, Louvain-la-Neuve, 1348, Belgium.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.

2573-0142/2023/6-ART179 \$15.00

<https://doi.org/10.1145/3593231>

1 INTRODUCTION

The acquisition and the recognition of gestures for gesture-based interaction [74] usually fall into four categories: *touch* input captured by surface technologies [86], *motion* input captured by computer vision [54], *sensor* input captured by wearable devices [32], and *radar* input captured by radar sensing technologies [90]. Hand gestures recognition spans the last three categories [20] with their intrinsic limitations: computer vision, which acquires gestures from image-based devices [69], is sensitive to ambient conditions [92], particularly lighting, limited field of view [29], transient or permanent vision occlusion [13], and privacy concerns raised by a visible device [8, 90]. Wearable computing, which acquires gestures from smart devices, such as smart rings [25], smartwatches [38], or smartphones, streams raw data in real-time to track gestures, but is sensitive to noise, articulation variation, obstrusiveness [8], and ecological validity [36].

In contrast, radar sensing [90] can be reliably operated under any light condition (e.g., under dark, cloudy, foggy conditions), in any direction (e.g., in a reflective setup) and is privacy friendly (e.g., a hidden radar preserves anonymity). Furthermore, radars could detect gestures below or behind a surface [8, 9], through fabrics [42], thus raising the need to study radar interaction in new conditions [63]. These technologies are suitable for some applications, such as people monitoring, activity recognition [9], gesture interaction through fabrics [42] and material [78], material recognition [24], tangible interaction [91], and virtual reality [34].

Template-based recognizers [15, 76, 77] working with pattern matching [55] could be considered complementary to deep learning [16, 51] depending on the number and quality of samples available for training and recognition [76]. Template-based recognizers are often praised for their ability to become accurate with a few samples [73, 86], to eliminate retraining or remodeling when samples are modified (e.g., for customization [68]), to benefit from efficient implementation and to allow geometric interpretation [74]. For example, some template-based recognizers have been developed in that spirit for 2D [52, 71, 76, 77] and 3D trajectory gestures [17, 18].

While these methods typically recognize simple gestures in particular during the prototyping phase, they give way to deep learning recognizers as soon as the gestures become more complicated to process and more challenging to recognize. However, deep learning recognizers usually require a high number of samples to become accurate, need to be retrained or remodeled each time samples are modified, imply a sophisticated implementation that requires development expertise, and no longer allow geometric interpretation [72].

When different people are asked to produce the same gesture to train such methods, they are quite likely to respond positively in that sense but inevitably produce different radar signals or different articulations [57, 77] due to their human morphological and physiological variations [81]. For example, a person with a large palm area will not produce the same radar echo as a person with a smaller, leaner hand. Since human individual differences in physiological parameters and the expression of gesture intent can lead to individual differences [88] in radar signals when performing the same gesture, most of the template-based recognizers employed in this context are not generic enough to work sufficiently in user-independent scenarios. Reciprocally, two apparently distinct gestures can lead to two very similar radar signals simply because their radar signatures, essentially conditioned by distance and permittivity, are very similar.

While template-based recognizers turn out to be accurate in user-dependent scenarios since they take into account only variations that are pre-recorded, they are not accurate enough in user-independent scenarios. For example, through the improvement of algorithms including signal preprocessing, feature extraction, and classification optimization, existing works on radar-based gesture recognition can efficiently recognize a limited number of gesture classes in user-dependent scenarios [67]: 65% in 4-gesture task [80], 96% in a 3-gesture task [61], 98% in a 4-hand gesture

manipulation [93]. In particular, a recognition rate of up to 84% was obtained for 16 gesture classes in a user-dependent scenario with 5 templates, but this performance largely deteriorates to 20% in a user-independent scenario [66].

These limitations become less restrictive as deep learning progresses. For example, *few-shot learning* [49] (i.e., when the classifier becomes operational already with few samples) promises to reach high accuracy. Real-time retraining becomes available under certain circumstances, and its implementation becomes easier to integrate into real applications. Furthermore, when relying on geometric modeling [14], a graphical representation and interpretation of the recognition process are still possible, although not as easy as in the case of template-based recognizers. In this paper, we are pursuing the same goal as in few-shot learning, but not necessarily using this method. Therefore, we prefer to call it “few samples” [67].

Most of the previous work on gesture recognition using radar sensing, with a few exceptions, is characterized by the following context of use: (R_1) the radar used is homemade or assembled from electronic components [29], (R_2) the radar is stationary in a fixed position in the environment [7, 57], (R_3) the number of recognized gesture classes is limited (e.g., 5 classes in Amin et al. [6], 6 classes in Zhang et al. [94]) while (R_4) the number of samples required to recognize them is very high (e.g., 2,000 samples in Lan et al. [39]), and (R_5) the gestures recognized are mostly (simple) directional gestures (e.g., the 8 directional swipes in Patra et al. [59]).

To overcome these five limitations, this paper presents FORTE (Few samples fOR Recognizing hand gesTurEs on a smartphone-attached radar), a method for hand gesture recognition with a Commercially available Off-The-Shelf (COTS), smartphone-attached radar, that accurately recognizes 20 motion gesture classes, each class being populated with only 10 samples from 5 users.

To this end, the remainder of this paper is organized as follows to present its contributions. Section 2 reviews existing works on deep learning methods used for recognizing hand gestures with radar sensing, then examines radar-based interaction in general and in particular with a smartphone-attached radar. Section 3 motivates the selection of the COTS, a smartphone-attached radar, and describes a dataset of 20 motion gesture classes with only 10 templates per class (2 samples $\times$ 5 users) acquired with this radar. Section 4 defines three optimized convolutional neural networks (CNNs) that support short-cut learning to recognize gestures from the acquired dataset. Section 5 compares their respective cross-validation to identify the best configuration. From these results and the experience gained with the comparison, Section 6 suggests some implications for the design of radar-based gestures. Section 7 reflects on the limitations of this work. Finally, Section 8 concludes this paper and discusses future avenues for research in radar-based gesture recognition.

Overall, this paper follows a similar approach to [66] (see Table 1 for a comparison) but with three complementary goals: (1) to become efficient in both user-independent and user-dependent scenarios ([66] is only efficient in the user-dependent scenario), (2) to prove that the pipeline is flexible enough to accommodate other algorithms than template-based recognizers when needed, and (3) to preserve quality properties such as few samples [67]. This is why we instead consider CNNs.

2 RELATED WORK

This section reviews some work related to deep learning hand gesture recognition, which is traditionally used for this purpose, then discusses radar-based interaction in general and in particular for a smartphone-attached radar.

2.1 Hand Gesture Recognition and Deep Learning

Until now, hand gesture recognition has been a favorite topic of interest through popular COTS devices (e.g., Intel RealSense [11], Microsoft Kinect Azure, UltraLeap Leap Motion Controller, 3D

	This paper	Sluÿters <i>et al.</i> [66]
Gesture body scale (Vatavu [75])	Hand, arm, and body-level	
Number of gesture types (Aigner et al. [3])	6	5
Dataset size	4400 samples in total	80 samples per sensor
Number of gesture classes	20	16
Number of users	22	1
Number of samples/class/user	10	5 per sensor
Size of the training set	5 users $\times$ 2 samples = 10 samples	1 user $\times$ 4 samples = 4 samples
Gesture recognition approach	CNN	Template-matching
Scenario	User-independent	User-dependent
Accuracy	94.96%	84.5%
Sensor(s)	Walabot Developer (EU/CE)	Walabot Developer (EU/CE), Horn antenna, Leap Motion Controller
Context	Mobile (standing) & stationary (sitting)	Stationary (sitting)
Sensor position	Smartphone-attached	Pedestal-attached

Table 1. Comparison between this paper and the work by Sluÿters et al. [66]: this paper built a CNN obtaining an accuracy of 94.96% in a user-independent scenario for 20 gesture classes, each class with 5 users with 2 samples each. We also provide a rich complete radar-based dataset of 20 gesture classes of 220 samples each (22 users $\times$ 10 samples). This entire dataset was used for testing.

PMDTEC Depth-sensing camera, smartwatches [40]) and sensors such as those using Electromyography (EMG) or Inertial Measurement Units (IMUs) [32, 47]. These devices are suitable for 3D gesture interaction in general [20] and for many purposes [89]. Non-COTS devices and sensors are also widely developed and considered for hand gesture recognition, such as AILI [44], a custom hidden device that provides hand skeleton data without privacy concerns.

There are many robust deep learning algorithms for gesture segmentation, recognition, and interaction [50, 71], which are regularly subjected to online and offline recognition competition [16–18]. For example, Akl and Valaee [4] exploited Dynamic Time Warping (DTW) [54] and affinity properties to recognize gestures based on accelerometers. Li *et al.* [43] recognized finger gestures with high precision using WE-kNN algorithm. DEEPGRU (Deep Gesture Recognition Utility) [50] is a CNN composed of a set of stacked gated recurrent units followed by two fully connected layers and a novel global attention model to recognize human gestures and actions based on a skeleton vector. Although it has been proven to be accurate on several datasets, it has not yet been applied to radar-based gestures. Furthermore, DeepGAN [51] consists of a Generative Adversarial Network (GAN) model synthesizing new gesture samples, which is particularly useful when asking a lot of people to acquire a lot of gestures is too tedious. To train DeepGAN’s generator without requiring a discriminator, DeepNAG [51] relies on a DTW-based differentiable loss function with the average Hausdorff distance. The majority of algorithms used to recognize hand gestures belong to Machine Learning (e.g., k -Nearest Neighbors, SVMs, Ensemble Learning, Decision Trees, Hidden Markov Models, and Bayesian networks) and to Deep Learning, CNNs being the most frequent, often combined with Long Short-Term Memory (LSTM) or SVMs.

Therefore, we will also investigate CNNs. Although deep learning algorithms demonstrated superior accuracy compared to conventional methods when large gesture datasets are available for training, their performance substantially decreases when data are limited, such as when the number of samples per gesture class is reduced [41]. Instead of relying on gesture synthesis, we will investigate CNNs to preserve the “few samples” property.

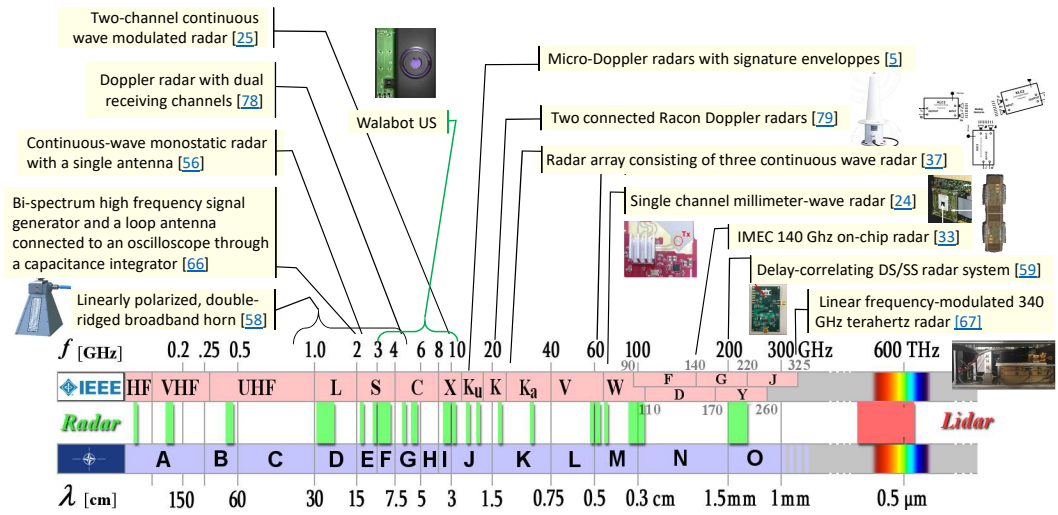


Fig. 1. Classification of radar-based interaction according to the frequency band [56] of the radar (Bottom image of waves and frequency ranges used by radar used with permission from Wolff [87]).

2.2 Radar-based Interaction

As many different methods recognize gestures depending on the type of radar, we classify and explore them according to the frequency band, as defined in the IEEE 521-2019 standard [56], along the electromagnetic continuum (Fig. 1):

- **L=1–2 GHz:** based on a 2 GHz radar, Viunytskyi and Totsky [80] apply bispectrum-based processing of the signal envelope to recognize four gestures: top-down (60%), bottom-up (60%), left-to-right (80%), and right-to-left (60%). The accuracy is low given the high frequency and depends on indoor interference.
- **S=2–4 GHz:** based on a 2.4 GHz radar, Sakamoto et al. [61] exploit a CNN trained on 60-time domain I-Q plots for three gestures: palm back and forth, palm rotating, and making a fist (96%).
- **C=4–8 GHz:** based on a 5.8 GHz radar, Zhang et al. [94] rely on a CNN to classify four hand gestures with an accuracy of 98%, a rate that can be affected by the distance and the gesture scale.
- **X=8–12 GHz:** based on a 10 GHz, Ehrnsperger et al. [23] compared various recognizers, such as a Support Vector Machine (SVM), to conclude that ML methods are more accurate but require more computational power.
- **Ku=12–18 GHz:** based on a 12.8 GHz radar, Amin et al. [6] recognized five gestures (*i.e.*, swipe, rotate, flip, call, and snap) from the envelopes of their micro-Doppler signatures, which capture the distinctions among different hand movements and their corresponding positive and negative Doppler frequencies. The best precision of 95.23% was obtained with a kNN and the Manhattan distance.
- **K=18–24 GHz:** based on two 24 GHz radars, RACon [95] recognized six gestures with different angles between people and the radar by relying on an improved Dynamic Time Warping (DTW) for an accuracy of 96%.
- **Ka=24–40 GHz:** based on three 25 GHz radars, Lan et al. [39] recognized ten gesture classes of 2k samples each (92%: swipe left, right, up, down, front, back, lift, tap, open, clench) based

on a decision tree with three features: temporal and frequency signatures with magnitude difference, phase difference, and spectra power integral.

- **V=40–70 GHz:** based on a 64 GHz, Patra et al. [59] tested two low-complexity classifiers, *i.e.*, the unsupervised Self Organized Map (SOM: 60%) and the supervised Learning Vector Quantization (LVQ: 75%), to recognize eight swipe gestures: forward, backward, left, right, back to front, front to back, left to right and right to left.
- **W=70–110 GHz:** based on a 77 GHz radar, Du et al. [22] combined the micro-Doppler features, the instantaneous azimuths and elevations in their 3D-CNN after removing noise by channel and spatial attention-based feature refinement. Ten gestures are recognized with an average accuracy of 96%: left to right, right to left, lower left to upper right, upper left to lower right, backward, forward, left and right, double press, clench, and snap.
- **F=90–140 GHz:** based on a 140 GHz radar, IMEC [35] can recognize a wide range of micro gestures by combining a CNN with a Long Short-Term Memory (LSTM).
- **G=140–220 GHz:** based on a 183–205 GHz delay-correlating direct sequence spread spectrum radar, Tang et al. [70] used its de/modulation architecture to detect the position and motion of cardboard tubes.
- **~1 THz:** based on a 340 GHz terahertz radar, Wang et al. [82] wanted to achieve a good trade-off between accuracy and latency by using a CNN to coarsely classify the six gestures of the parent class, *i.e.*, horizontal swipe, vertical swipe, press, zoom, slide, and circle, and then an intention model to refine them according to their direction, *e.g.*, left vs. right swipe. Twelve gestures are recognized with an accuracy of 94% in 0.033 s.

The Magic Carpet [58], a Doppler radar that recognizes body gestures by signal processing, and **RADARCAT** [90], a radar that recognizes physical objects and material by extracting and classifying their signals using a random forest, is often quoted as the pioneers of radar-based interaction. Yeo et al. [91] used radar to count, order, and identify physical objects in tangible interaction, to track their orientation, movement, and distance between objects. **GESTUREVLAD** [10] is a Doppler radar that recognizes poses of hands in real time and their variations in articulation with accuracy ($\tau \geq 96\%$). **PANTOMIME** [57], a fixed feet-based radar, accurately recognizes 21 gestures acquired by 45 participants from 3D point clouds using LSTM and **Pointnet++**. Wang et al. [83] recognize 2D stroke gestures, which do not require as many antennas as 3D. Short-range gestures are also recognized using 3D CNNs with a triplet loss [30].

The **GOOGLE SOLI** [45, 84] recognizes micro gestures (*e.g.*, finger wiggle, hand tilt, check mark, or thumb slide) in its first version and 11 gestures in its second version by combining deep convolutional and Recurrent Neural Networks (RNN). This chip initiated several works: real-time recognition of 10 hand gestures [21], swipe gesture recognition in any direction (*i.e.*, left, right, up, down, and omnidirectional swipes) [29], object classification by a robot [24], recognition of five gestures on the object using a 3D CNN and a spectrogram-based ConvNet [7], and **SOLIDS ON SOLI**, which identified the most distinctive features for recognizing gestures through various materials [78]. This radar works at a frequency that is about ten times larger than the Walabot device, thus resulting in a wavelength ten times smaller and a resolution about ten times finer than the Walabot. Although their radar is limited in the number of antennas (7), they can be oriented towards a better lateral resolution, which is not the case with rigid radars. In conclusion (also see the comparison table provided in supplementary material), we made the following observations to motivate our work:

- R₁. There are many types of radar [67] depending on their frequency band (see Table 2 for a non-exhaustive list of COTS radars), most of them custom [39] or assembled [29]. As the frequency band increases, the accuracy of its associated recognition improves and the number

Vendor	Model	Frequency band	Antennas	Interface	Raw data	Applications	Price
Vayyar	Care	V	24Tx/22Rx	WiFi	No	Fall detection	\$250
	Walabot Developer (EU/CE)	C	4Tx/15Rx	USB	Yes	Motion detection, wall scanning	\$600
	Walabot Developer (US/FCC)	S, C, X	4Tx/15Rx	USB	Yes	Motion detection, wall scanning	\$600
Novelda	Xethru X4M03	C	1Tx/1Rx	USB, UART, SPI, I2C	Yes	Radar development kit	\$400
	Xethru X4M200	C, X	1Tx/1Rx	USB, UART, USART	No	Breathing	\$400
	Xethru X4M300	C, X	1Tx/1Rx	USB, UART, USART	No	Presence	\$400
Innosent	IPM-170, IPM-365, IPM-165	K	1Tx/1Rx	/	No	Motion detection	\$5
	INS-333X	K	1Tx/1Rx	UART	No	Motion detection, proximity sensing	\$50
Infineon	BGT60LTR11AIP	V	1Tx/1Rx	SPI	Yes	Motion detection	\$15
	Distance2GoL BGT24LTR11	K	1Tx/1Rx	USB	Yes	Motion detection, proximity sensing	\$210
Inras	RadarBook2	X, K, W	8Tx/16Rx	Ethernet	Yes	Radar development kit	/
TI	IWR6843ISK	V	3Tx/4Rx	USB, UART, I2C	Yes	Motion detection, proximity sensing, people counting	\$250
Silicon Radar	EVALKIT SiRad Easy® r4	mm	1Tx/1Rx	USB, UART	Yes	Radar development kit	/

Table 2. Non-exhaustive list of COTS radars and their main characteristics.

of correctly recognized gesture classes increases, thus posing a challenge for low-frequency radars.

- R₂. Radars often remain stationary in the environment [7, 57], except for GOOGLE SOLI [45].
- R₃. The number of gestures recognized is moderate, usually 4 to 12 (e.g., 5 in Amin et al. [6], 6 in Zhang et al. [94], but 21 in PANTOMIME [57]).
- R₄. The number of samples per class is usually very high (e.g., 2,000 [39] and 2,750 in total for 11 classes [10]) to adequately train the model, such as CNNs. None of them seems to consider training a CNN with few samples. Few-Shot Learning (FSL) [49] allows the recognizer to learn from a few samples as humans do, especially when these samples are expensive to acquire. Furthermore, fewer training samples reduce the high dimensionality in the training dataset (e.g., to two dimensions only: distance and permittivity [66]).
- R₅. The most commonly recognized gestures are directional [59] along the three axes (e.g., horizontal left/right swipe) and their combinations (e.g., vertical up/down swipe [82]) with some movements (e.g., snap, clench [6]).



(a) Mobile context: with a handheld device.

(b) Stationary context: with a wall-placed device.

Fig. 2. Two contexts of use for the Walabot.

2.3 Walabot-based Interaction

In our work, we selected the [Walabot device](#) in order to overcome the five aforementioned observations as follows:

- R₁. This device is a low-frequency COTS ultra-wideband frequency modulated continuous wave radar (Fig. 1) that can be attached to a smartphone using a USB cable. However, the Walabot is shipped with undetermined and unmodifiable proprietary techniques (e.g., baseband property and digital processing), which therefore stems from an independent, replicable, and open method for gesture recognition.
- R₂. This device and its last version, the Walabot 2.0, which can be paired with a smartphone via Bluetooth and then used separately, make it suitable for both (semi)mobile (Fig. 2a) and stationary (Fig. 2b) contexts of use.
- R₃. By accessing its raw data, we create a dataset of 20 gesture classes.
- R₄. By optimizing a CNN for learning from a few samples, two samples from 5 participants per class will be enough.
- R₅. By covering multiple categories of hand gestures [3], we will go beyond simple directional gestures.

Furthermore, the Walabot is widely used in several domains of application, such as indoor human sensing radar [5], human activity recognition [8, 97], human position estimation [9], television remote control [64], ambient intelligence [79], material identification [1], and other [community applications](#).

Regarding hand gesture recognition with the Walabot, Zhang et al. [93] presented a CNN that recognizes eight hand gestures with 150 samples per gesture class; Sluÿters et al. [66] also adopted learning from a few samples with a template-based recognizer [71] to recognize sixteen gestures, but their electromagnetic modeling and inversion pipeline require sophisticated processing.

3 ACQUISITION OF A RADAR-BASED HAND GESTURE SET

A dataset of twenty different gestures (Tables 3 and 4) was designed based on data from multiple sources: literature reviews, prior work in radar-based interaction, and gesture elicitation studies [29, 45, 57, 79]. The dataset covers different categories of Aigner *et al.*'s taxonomy [3]: 14 semaphoric (1-10, 13-16 in Tables 3 and 4), 3 iconic (11, 18, 19), 1 pointing (20), and 2 pantomimic gestures (12, 17). Each gesture was performed ten times by 22 participants, resulting in a total of 20 (gestures) $\times$ 22 (participants) $\times$ 10 (repetitions) = 4,400 samples. Gestures were recorded with a custom C++ console application, which connected to the Walabot, configured it with PROF_SENSOR_NARROW, and captured data from 12 pairs of antennas at a rate of up to 40 slow time frames per second. The data from each pair of antennas were truncated to keep only the first 1024 fast-time samples out of 4096,

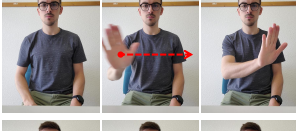
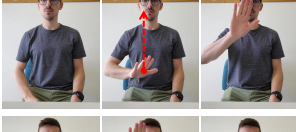
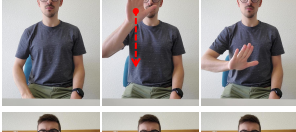
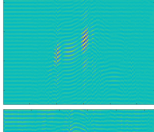
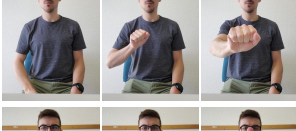
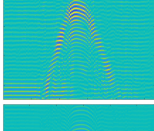
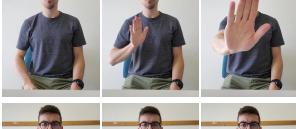
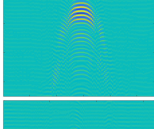
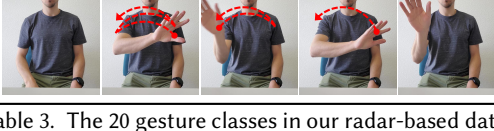
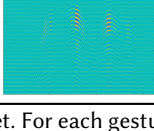
Gesture motion	Walabot image	Name [Reference(s)] (Type)
		1. Open hand [6, 22] (Dynamic semaphoric)
		2. Close hand [6, 30] (Dynamic semaphoric)
		3. Open, then close hand [2, 65] (Dynamic semaphoric)
		4. Swipe right [6, 19, 22, 65, 85, 96] (Stroke semaphoric)
		5. Swipe left [6, 19, 22, 85, 96] (Stroke semaphoric)
		6. Swipe up [6, 19, 85] (Stroke semaphoric)
		7. Swipe down [6, 19, 85] (Stroke semaphoric)
		8. Push with fist [/] (Dynamic semaphoric)
		9. Push with palm [22, 46, 48, 85, 96] (Dynamic semaphoric)
		10. Wave hand [22, 96] (Dynamic semaphoric)

Table 3. The 20 gesture classes in our radar-based dataset. For each gesture class, we provide an illustration of its motion (from left to right), the corresponding Walabot signal from one antenna pair after pre-processing, its name, a non-exhaustive list of references in which it is featured, and its classification according to Aigner et al. [3]’s taxonomy.

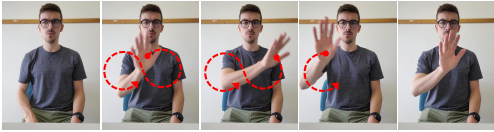
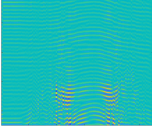
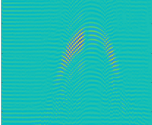
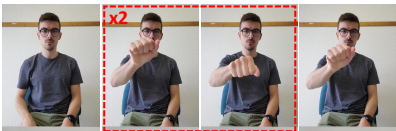
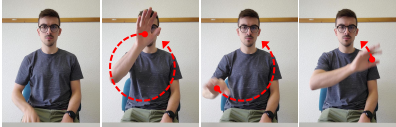
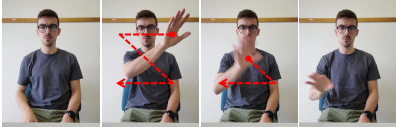
Motion gesture	Walabot image	Name [Reference(s)] (Type)
		11. Draw an infinity symbol [/] (Dynamic iconic)
		12. Barrier gesture [/] (Pantomimic)
		13. Extend one finger [2] (Static semaphoric)
		14. Extend two fingers [2] (Static semaphoric)
		15. Extend three fingers [2] (Static semaphoric)
		16. Extend four fingers [2] (Static semaphoric)
		17. Knock twice [46, 96] (Pantomimic)
		18. Draw a circle [19, 30, 46, 48] (Dynamic iconic)
		19. Draw a Z [46] (Dynamic iconic)
		20. Touch nose with index [60] (Pointing)

Table 4. The 20 gesture classes in our radar-based dataset (cont.).

which divided both the file size and the maximum range by four. For each registered sample, an output file is created. In this file, the data is organized so that each line represents one measurement. The first column represents the elapsed time between the first and the current measurement and the second column represents the amplitude of the measured signal. One frame consists of:

$$12 \text{ antenna pairs} \times 1024 \frac{\text{measurements}}{\text{antenna pairs}} = 12,288 \text{ lines} \quad (1)$$

Dividing the number of lines by 1024 gives the number of frames per gesture. Hence, each file is structured as follows.

Frame 1	Antenna pair 1 (1024 measurements)
	...
	Antenna pair 12 (1024 measurements)
...	...
Frame n	Antenna pair 1 (1024 measurements)
	...
	Antenna pair 12 (1024 measurements)

Table 5. Structure of the raw data output from the Walabot.

Therefore, the first 1024 lines represent the signal measured by the first antenna in the first frame, the next 1024 lines represent the signal measured by the second antenna in the first frame, etc. The number of frames n is variable and depends on the length of the gesture. The measurement provided by Walabot for each time instant is a real double precision number representing the voltage evaluated by the internal circuit normalized in the interval $[-1, 1]$.

The recording process of a sample was as follows: (1) the participant places both hands on their lap, (2) the experimenter triggers the recording and asks the participant to perform the gesture, (3) the participant performs the gesture, and (4) the experimenter stops the recording once the participant puts their hand back on their lap. The recording tool ran on a Dell XPS 17 9700 with an Intel i7-10875H CPU and 32GB or DDR4 RAM running Windows 10.

4 FORTE, OUR APPROACH FOR RADAR-BASED HAND GESTURE RECOGNITION WITH FEW SAMPLES

4.1 Raw Data Pre-processing

This subsection presents the main stages of the pre-processing needed to obtain effective data points to be used as input to our CNN. This pre-processing is performed off-line and is aimed at transforming the raw data, which are in the time domain, into data in the frequency domain to optimize the CNN process:

- *Raw data capture* (Fig. 3a): the raw data is acquired from each radar antenna according to Section 3. The output is a JSON file that is then processed to produce the Doppler images.
- *Fast Fourier Transform* (Fig. 3b): the radar signal is transformed from the time domain to the frequency domain through the Fast Fourier Transform algorithm [31], an operation required for the next stage.
- *Removal of radar source and antenna effects* (Fig. 3c): the radar source and antenna effects (e.g., internal reflections and transmissions) and antenna-target interactions are removed [66].
- *Removal of the background scene* (Fig. 3c): using the superposition principle, the first frame of a gesture is subtracted from the radar signal to remove the remaining reflections from static reflectors, such as walls, furniture, or other objects, but also body parts (e.g., the body of the end-user). This stage ensures appropriate feature extraction for later stages, as reflections from other sources could be confused with the user's hand.

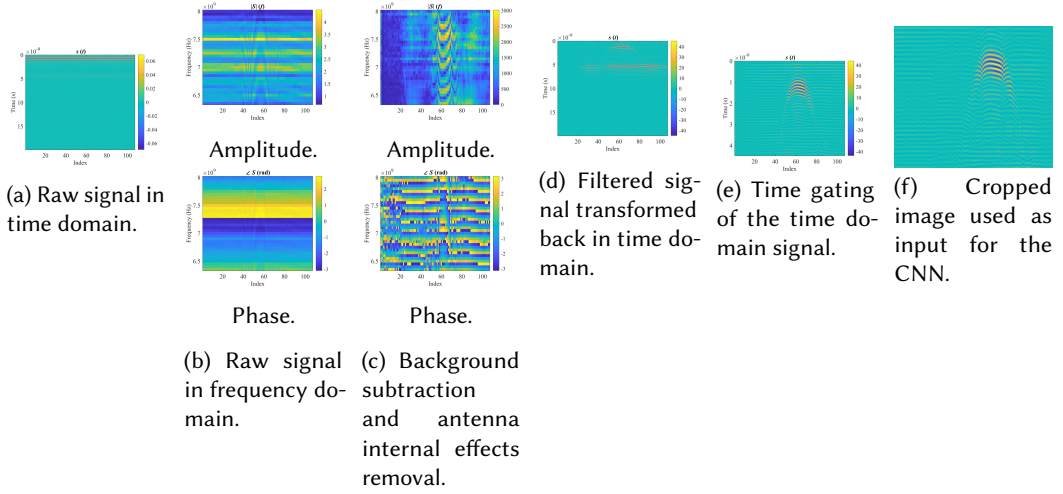


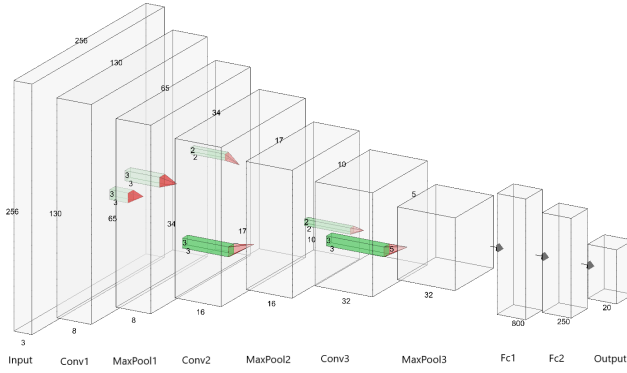
Fig. 3. Pre-processing pipeline.

- *Inverse Fast Fourier Transform (IFFT)* (Fig. 3d): the filtered radar signal is transformed from the frequency to the time domain with the Inverse FFT algorithm [31].
- *Time gating and cropping* (Figs. 3e, 3f): the time-domain data is truncated to keep only the portion relevant for gesture recognition. The signal received only within a given time window is kept to remove irrelevant information (e.g., objects that are too far from the radar). This improves accuracy and reduces the processing time of the recognition. The image is then cropped to 256×256 pixels, ready to feed the CNNs.

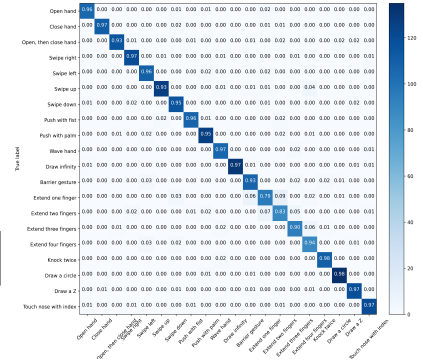
4.2 Design of three Convolutional Neural Networks

The pre-processing described in Section 4.1 produces 2D Doppler images, for which a ConvNet represents the best choice for this task [7]. A first model, hereafter referred to as MODEL 1, was developed as a CNN composed of 3 convolutional layers, each of them followed by a maximum-pooling layer, and completed by a 2-layered Fully Connected Neural Network used as a classifier, for a challenging subset of our dataset, consisting of 20 gesture classes with 5 participants giving only 2 samples per class (200 samples in total). A second model, hereafter referred to as MODEL 2, was developed with the full dataset consisting of 20 gesture classes with 22 participants giving 10 samples (4,400 samples in total, as acquired in Section 3), for which a more complex architecture was needed. A third model, hereafter referred to as MODEL 3, was developed for the same full dataset, but with one additional convolutional layer and different pooling options. These three models were designed to explore the possibilities, compare their results and identify the configuration that yields the best recognition accuracy. Basically, our three models involve the following layers:

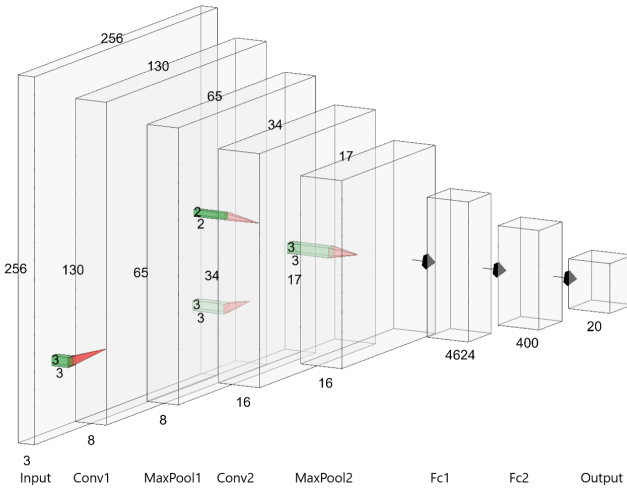
- *Convolutional layer*: this layer computes a dot product between two matrices: the *kernel*, which is the set of learnable parameters, and the other matrix, which is the restricted portion of the receptive field. The kernel is spatially smaller than an image. During the forward pass, the kernel slides across the height and width of the image-producing the image representation of that receptive region. This produces a 2D representation of the image (*activation map*) that gives the response of the kernel at each spatial position of the image. With convolutions, the output size is usually smaller than the input (e.g., 5×5 input convoluted with 3×3 kernel = 4×4 output).



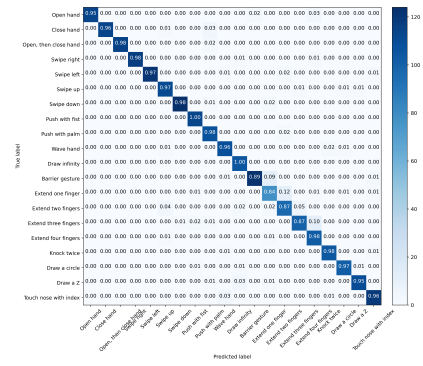
(a) Layered structure of MODEL 1.



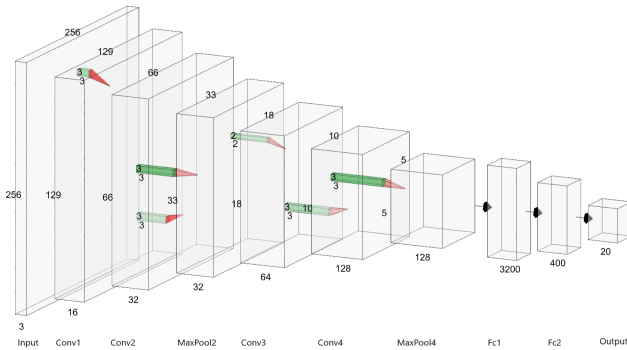
(b) Confusion matrix of MODEL 1.



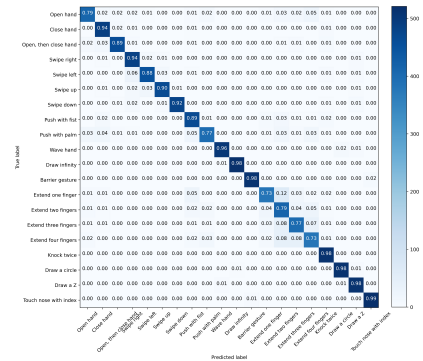
(c) Layered structure of MODEL 2.



(d) Confusion matrix of MODEL 2.



(e) Layered structure of MODEL 3.



(f) Confusion matrix of MODEL 3.

Fig. 4. Layered structure and confusion matrix obtained for the three CNN models.

- Some *padding*, i.e., the process of adding zeros to the input matrix can be added symmetrically to maintain the same size. Another parameter is the sliding size of the kernel, defined as *stride*, which denotes how many steps we are moving in each step of convolution, and by default, it is one.
- *Pooling layer*: this layer reduces the spatial size of the convolved feature to decrease the computational power required to process the data through dimensionality reduction. It is also useful for extracting dominant features which are rotational and positional invariant, thus maintaining the process of effectively training the model.
- *Max Pooling*: this layer returns the maximum value of the portion of the image covered by the kernel. Usually, only the kernel size and stride hyperparameters can vary.
- *Fully-connected layer*: this layer performs the classification based on the features extracted through the previous layers and their different filters. Although convolutional and pooling layers tend to use Rectified Linear Unit (ReLU) functions [26], fully connected layers generally leverage a Softmax activation function [27] to classify inputs appropriately, producing a probability from 0 to 1.

4.2.1 MODEL 1 and MODEL 2 Structures. These CNN-based classifiers follow a standard layered structure, i.e., a suite of convolutional layers to extract features and pooling layers to progressively reduce the size of the data, and of fully connected layers standing at the end of the network for the classification task. MODEL 1 takes as input RGB radar Doppler images. The different colors are needed to enrich the data and allow for a higher differentiation between the gestures. The image is resized, and its input size is (256,256,3). The layered structure, along with its parameters (Table 6), is structured (Fig. 4a) as follows:

- Input layer
- Convolutional layer 1 (Conv2D + BatchNorm + ReLU + MaxPool)
- Convolutional layer 2 (Conv2D + BatchNorm + ReLU + MaxPool)
- Convolutional layer 3 (Conv2D + BatchNorm + ReLU + MaxPool + Dropout)
- Flattening layer
- Classifier (FC1 + ReLU + Dropout + FC2 + ReLU + Output)

Batch normalization was applied to improve the robustness of training under varying hyperparameters [62], while two dropout layers were inserted to prevent overfitting problems, with a uniform dropout probability picked randomly between 0 and 0.25. MODEL 2 is exactly the same as MODEL 1 with one convolutional layer removed (Fig. 4c) to determine its impact on performance. Table 7 shows its parameters.

4.2.2 MODEL 3 structure. Starting from the first two models, a more sophisticated architecture was developed to try out new configurations applied to the full gesture set. This model considers an extra convolutional layer for finer feature extraction of the radar gestures. Together with a slightly different layering structure in terms of pooling and convolutions, the MODEL 3, along with its parameters (Table 8), is structured (Fig. 4e) as follows:

- Input layer
- Convolutional layer 1 (Conv2D + BatchNorm + ReLU)
- Convolutional layer 2 (Conv2D + BatchNorm + ReLU + MaxPool)
- Convolutional layer 3 (Conv2D + BatchNorm + ReLU + Dropout)
- Convolutional layer 4 (Conv2D + BatchNorm + ReLU + MaxPool)
- Flattening layer
- Classifier (FC1 + ReLU + Dropout + FC2 + ReLU + Output)

4.3 Implementation of FORTE

This section describes the implementation of FORTE into three stages:

- (1) *Raw data acquisition*: to access raw signal data directly from radar antennas, we used the [Walabot custom SDK](#) and develop a C++ script to automatically record and acquire gestures in JSON files.
- (2) *Pre-processing*: the JSON files were then pre-processed (see Section 4.1) using [MATLAB V.2021b](#) to obtain the input data for the CNN.
- (3) *Recognition*: the CNN was programmed in [Python V3.8](#), which was chosen for its wide range of ML oriented libraries, data manipulation easiness, versatility and readability, and very rich graphic options. In addition, the following libraries were used: [Pytorch](#) as the main framework for the deep learning part (due to its versatility and the many implementations with optimizers and regularizers, it was preferred over other libraries such as [Keras](#) or [Tensorflow](#) because it represented a good compromise between usability, stability, and performance); [scikit-learn](#), [pandas](#): two machine learning oriented libraries used to perform model validation and obtaining important metrics about it; [os](#), [shutil](#), [pickle](#), [json](#): libraries necessary to manage filenames, data, dictionaries; [numpy](#), [random](#): to work with arrays, data structures, generate random numbers; [matplotlib](#): to generate graphics and plots; and [datetime](#), [tdqm](#), [itertools](#): other utility libraries. Network training was performed on a cloud computing platform ([Google Cloud Platform](#)) with a virtual machine instance running characterized by the following elements: 8 N1 standard virtual CPUs; 30 GB RAM; 1 [NVIDIA Tesla T4 GPU](#).

Layer	Input size (channels, height, width)	Output size (channels, height, width)	Kernel size (height, width)	Padding	Stride	Number of output units
Input	(3, 256, 256)					
Conv1	(3, 256, 256)	(8, 130, 130)	(3, 3)	3	2	135,200
MaxPool1	(8, 130, 130)	(8, 65, 65)	(2, 2)	0	2	
Conv2	(8, 65, 65)	(16, 34, 34)	(3, 3)	3	2	18,496
MaxPool2	(16, 34, 34)	(16, 17, 17)	(2, 2)	0	2	
Conv3	(16, 17, 17)	(32, 10, 10)	(3, 3)	3	2	3,200
MaxPool3	(32, 10, 10)	(32, 5, 5)	(2, 2)	0	2	800
Flatten	(32, 5, 5)	(1, 800)				
Fc1	(1, 800)	(1, 800)				800
Fc2	(1, 800)	(1, 250)				250
Output	(1, 250)	(1, 20)				20

Table 6. MODEL 1 layers parameters.

Layer	Input size (channels, height, width)	Output size (channels, height, width)	Kernel size (height, width)	Padding	Stride	Number of output units
Input	(3, 256, 256)					
Conv1	(3, 256, 256)	(8, 130, 130)	(3, 3)	3	2	135,200
MaxPool1	(8, 130, 130)	(8, 65, 65)	(2, 2)	0	2	
Conv2	(8, 65, 65)	(16, 34, 34)	(3, 3)	3	2	18,496
MaxPool2	(16, 34, 34)	(16, 17, 17)	(2, 2)	0	2	
Flatten	(16, 17, 17)	(1, 4'624)				
Fc1	(1, 4'624)	(1, 400)				800
Fc2	(1, 400)	(1, 100)				250
Output	(1, 100)	(1, 20)				20

Table 7. MODEL 2 layers parameters.

Layer	Input size (channels, height, width)	Output size (channels, height, width)	Kernel size (height, width)	Padding	Stride	Number of output units
Input	(3, 256, 256)					
Conv1	(3, 256, 256)	(16, 129, 129)	(3, 3)	2	2	266,256
Conv2	(16, 129, 129)	(32, 66, 66)	(3, 3)	2	2	139,392
Maxpool2	(32, 66, 66)	(32, 33, 33)	(2, 2)	0	2	
Conv3	(32, 33, 33)	(64, 18, 18)	(3, 3)	2	2	20,736
Conv4	(64, 18, 18)	(128,10,10)	(3, 3)	2	2	12,800
MaxPool4	(128, 10, 10)	(128, 5, 5)	(2, 2)	0	2	3,200
Flatten	(128, 5, 5)	(1, 3200)				
Fc1	(1, 3200)	(1, 400)				400
Fc2	(1, 400)	(1, 100)				100
Output	(1, 100)	(1, 20)				20

Table 8. MODEL 3 layers parameters.

5 TRAINING AND VALIDATION

5.1 Training

While MODEL 1, MODEL 2, and MODEL 3 were trained in separate sessions and in different moments, they share the main characteristics in terms of optimization techniques for model training:

- *No data augmentation.* The application of geometrical transformations to augment the dataset would distort the input data in an undesirable way. Each gesture radar representation is correlated with its spatial and temporal features, therefore image operations such as stretching, cropping, zooming, and flipping, usually employed to extend the set of images and strengthen the training, here would generate an opposite effect. Contrarily to standard pixel signal processed by CNNs for object recognition, image augmentation methods toward rotation, scaling, and translation invariance are inappropriate as our radar images are captured as a time-domain signal: any modification of the source signal implies a shift in the frequency pattern of the gesture. For the same reason, articulation invariance cannot be ensured by this method, as we can for 2D stroke gestures [53]. However, advanced techniques can be used to augment data: Generative adversarial networks (GANs) could in theory generate new images, even without requiring existing data; Neural Style Transfer (NST) could introduce in the layered structure of our models (Fig. 4a to 4e) some new convolutional layers trained to deconstruct the radar images and separate context from the style [33].
- *Optimizers.* Stochastic Gradient Descent (SGD) [12] with momentum, and ADaptive Moment Estimation (ADAM) [37] optimizers were implemented to improve speed convergence of the learning algorithm.
- *Regularizers.* Weight decay, batch normalization and dropout were applied to regularize the model, improve its performance, and prevent overfitting situations. Early stopping was also used to stop model training with the *patience* parameter to avoid overfitting.
- *Loss function.* As the task is a multi-class classification problem, the categorical cross-entropy loss was used for the training, where the loss is computed between each pair of classes, and then all the contributions are summed up.

Table 9 shows the hyperparameters used for the two training sessions of the CNNs. No automatic tuning was performed; rather, these parameter configurations are the result of an empirical trial-and-error process carried out during the development of the models.

Model	Optimizer	Learning Rate	Momentum	Weight Decay	Dropout prob. 1	Dropout prob. 2	Epochs	Patience	Batch size
1	SGD	1e-4	0.7	-	6.61e-3	9.68e-3	15	-	16
2	Adam	2e-3	-	1e-4	0.2	0.1945	43	4	16

Table 9. Model training hyperparameters.

5.2 Cross-Validation Results

After the training was completed, the performance of the three models was tested using a *k-fold cross-validation* technique with $k=5$. This technique is more reliable than the holdout method, in which the data set is separated into two sets, called the *training set* and the *testing set* (e.g., 50%-50% in [10]). We chose $k=5$, which is a common value that should provide a good insight into the performance of the model while not being too computationally intensive, as a higher value of “k” means a more difficult validation task. The dataset is randomly divided into k folds of approximately equal size. The first fold is kept for testing and the model is trained on the other $k-1$ folds. The process is repeated k times, and each time different folds or a different group of data points are used for validation. Using this validation technique, it was possible to test the strength of the models and reduce the variability in the dataset, thus obtaining a better estimate of the classification performances. The evaluation metrics used in the assessment of the models are the following and are to be intended as the average values over the 5 folds: classification accuracy, precision, recall, and confusion matrix over the 20 gestures. Table 10 shows the results obtained for each model in the aforementioned metrics, while Figs. 4b, 4d, and 4f reproduce the confusion matrices obtained for each model. These results suggest the following comments:

Model	Classification Accuracy	Precision	Recall
1	0.9496	0.9592	0.9603
2	0.9460	0.9416	0.9427
3	0.8813	0.8881	0.8760

Table 10. Cross-validation results.

- MODEL 1 achieved the best results for all evaluation metrics in this cross-validation. Although MODEL 1 was trained with much fewer participants than MODEL 2 and MODEL 3 (5 vs. 22), less variability in gesture articulation and the reduced number of model parameters could explain the higher accuracy, suggesting that learning with a few samples is supported by simpler, more regularized models. This configuration supports this learning to some extent, as a few samples are used to train the classifier. In comparison, Zhang et al. [93] obtained an excellent average accuracy of 98.96%, but only for 8 simple gestures and with 120 samples as opposed to 20 various gestures with only 5 samples. FORTE even outperforms the average accuracy of 84.5% obtained by Sluÿters et al. [66], for 16 gestures with 5 samples.
- MODEL 1 and MODEL 2 obtained approximately identical results. The difference in architecture is imperceptible; therefore, a lower number of convoluted layers could be preferred for computational reasons.
- Results in the 5 folds were substantially the same, even though only the average scores are reported. This suggests a well-balanced dataset whose pre-processing is adequate to benefit from the CNN potential.
- The lower classification results are obtained for some particular gesture classes, such as *Extend one finger*, *Extend two fingers*, *Extend three fingers*, *Extend four fingers*. This behavior, which occurs for all models, shows how similar classes are more difficult to recognize, possibly because of the low resolution of the signal.

6 DISCUSSION AND IMPLICATIONS FOR DESIGN

Based on our results, we discuss and suggest some implications for designing hand gesture interaction with radars, keeping in mind that the study represents an instance of the general problem.

Maximize the surface exposure of gestures. The gestures that are best recognized and differentiated from each other are those that are distinguished by sufficiently different surface exposures. For example, the gestures “Extend one finger”, “Extend two fingers”, “Extend three fingers”, etc. (Table 4) vary very little in terms of exposure surface and are therefore complicated to distinguish for a radar. On the contrary, gestures that significantly differ in terms of surface exposure, either at a given moment or over time, are those that are best recognized. For example, the gestures “Push with fist” and “Push with palm” (Table 3) performed with the flat hand or with the wrist, respectively, are well differentiated.

Keep the radar-hand distance short. The accuracy of radar-based hand gestures is better at short distances (e.g., from 1 foot≈30 cm to 4 feet≈122 cm) than at long distances (e.g., from 3 feet≈91 cm to 7 feet≈213 cm). Our study investigated the Walabot operating between 3 and 10 GHz (represented in green in Fig. 1), which represents a low, and therefore challenging, frequency band. For this reason, gestures were acquired and accurately tested at short distances.

Keep the radar-hand angle and elevation short. The radar’s range resolution is usually better than the angular resolution, which means that gestures in front of the radar are often recognized more accurately than gestures performed close to the border of its cone, such as with extreme elevations and extreme angles. For example, directional gestures are appropriately recognized as long as they are kept within the angle and elevation limits. More particularly, directional gestures, such as swipes, are well recognized if they do not extend too much either vertically or horizontally. Similarly, other gestures involving a significant motion should preserve the motion envelope with a range vertically and horizontally. For example, the “Wave hand” gesture (10 in Table 3) should not exceed the maximum angles and elevations.

Keep familiar gestures first. Familiar gestures and frequently produced gestures were issued more consistently than unfamiliar, unusual ones. For example, the “Barrier” gesture, which became a frequent gesture during the pandemic, is consistently issued by participants. The “Draw an infinity symbol” (11 in Table 4) is considered less familiar than the “Draw a Z” gesture (19 in Table 4).

7 LIMITATIONS

This section discusses some limitations that are intrinsic to the setup used throughout our study.

Only one Walabot at a short distance. The frequency band of the Walabot used in this study represents a narrow range that could easily represent a serious challenge due to its low resolution. Although the pre-processing defined in Section 4.1 is independent of any radar and thus should accommodate other versions of this radar or radars with a higher frequency band with minor changes, our study is limited to only one Walabot at a short distance. Testing the same 20 gestures at a long distance, perhaps with two or more radars, has not been achieved. When two radars are used instead of one, the accuracy also improves as already noticed [59].

Only a limited set of participants. While the entire data set comprises 10 samples from 22 participants, MODEL 1 obtained the best results with only 5 participants producing 2 samples. This obviously represents an advantage but does not consider a large set of participants. Participants may vary [81] in terms of hand dominance, hand dexterity, gesture articulation, hand palm surface, finger mobility, and arm size. These variations could influence the recognition process. While

acquiring a set covering all articulation variations is probably utopian, even if they are known theoretically (by formulas) and/or empirically (e.g., the range of the human palm surface is known), it is still worth testing to what extent the model is resistant to some of these variations and to what extent these variations might disturb the recognition of which gestures. In particular, gestures performed by right-handed participants are not equal to symmetric gestures performed by left-handed participants. New gestures are needed.

Only convolutional neural networks. Although CNNs were reported to be the most widely adopted deep learning method and although we compared three of them, no other DL method was investigated for external comparison. For example, Avrahami *et al.* [8, 9] compared different methods, ranging from classical ones to more modern ones to identify which method is the most suitable for radar-based gestures.

Only one cross-validation method. Although the same k-fold cross-validation method was consistently used to test and compare all models, only one value $k=5$ was used (other values such as $k=10$ could be considered additionally) and one method type was used. Other methods [28] could examine to what extent different methods would produce similar results. For example, Berenguer *et al.* [10] compared their models using the holdout method [28], but also with cross-session validation beyond cross-user validation and using split testing, perhaps with different distributions (e.g., 80%-20%, 70%-30%, and their symmetric distributions).

No incorporation into a real-world application. The validation was carried out offline: the pre-processing was performed once and for all and the test was also performed computationally, without incorporating gesture recognition MODEL 1 into a real-world interactive application, running in real-time in a real context of use. This limitation represents for us the best opportunity for future work in order to compare the computation testing with respect to gesture recognition in a real-world application.

Loss of customization. Template-based recognizers are often praised for their ease of training, testing, and implementation. They also intrinsically support gesture personalization in a straightforward way: modifying the dataset (e.g., by adding a new template to an existing gesture class or by replacing an existing gesture class with a personal one) does not change the algorithm and its implementation. In this paper, lost this ability to customize since the model needs to be re-trained after each modification to the data set [86].

8 CONCLUSION AND FUTURE WORK

To overcome the five limitations mentioned in Sections 1 and 2.2, this paper addressed the problem of radar-based hand gesture recognition as follows: with a COTS smartphone-attached radar (R_1) that can be used in both (semi-)mobile and stationary contexts of use (R_2); a definition and a comparison of three optimized CNNs based on pre-processed radar signals on a significant dataset of 20 (R_3) challenging motion gesture classes (R_5). Among the three CNNs, MODEL 1 gives the best average accuracy of 94.96% with a precision of 95.92% and a recall of 96.03% on the 20 gesture classes with solely 5 participants producing 2 samples (R_5), which is “few samples”.

In future work, we plan to address the last aforementioned limitation, *i.e.*, to incorporate the best CNN into a real-world interactive application in a real-world context of use. For this purpose, we are envisioning an application for scheduling a meeting with somebody, either in front of the person (as in the mobile context represented in Fig. 2a) or behind the office door of this person (as in the stationary context represented in Fig. 2b). In this way, we could compare gesture recognition as computationally tested in this article with gesture recognition operated “in situ”. The environmental conditions could then be imposed on the first scenario by altering the distance and

lighting conditions and on the second scenario by tolerating different door materials (e.g., wood, glass, and PVC) to test radar-based interaction through materials. Since FORTE was developed with different programming and scripting languages for off-line pre-processing and recognition (see Section 4.3), we are working towards developing a fully integrated FORTE version in C++ to enable nearly real-time pre-processing and recognition, which is feasible at first glance.

OPEN SCIENCE

Our GitHub repository and its companion website are publicly accessible at <https://sites.uclouvain.be/ingenious/2023/04/18/forte-few-samples-for-recognizing-hand-gestures-on-a-smartphone-attached-radar/> with the dataset (Section 3), the CNNs presented in Section 4, and the detailed results reported in Section 5.

ACKNOWLEDGMENTS

The authors of this paper are very grateful to anonymous EICS reviewers and Associate Chair whose suggestions helped improve and clarify this manuscript. They also thank the participants of the gesture study reported in the article for their participation. The authors acknowledge funding received by [Wallonie-Bruxelles-International](#) (WBI), Belgium, under grant SUB/2021/519018 and [UEFISCDI](#), Romania, under grant PN-III-CEI-BIM-PBE-2020-0001/IBM/2021 (Project “[RadarSense](#)”). Arthur Sluÿters is funded by the “[Fonds de la Recherche Scientifique - FNRS](#)” under Grant n°40001931 and n°40011629.

REFERENCES

- [1] Gianluca Agresti and Simone Milani. 2019. Material Identification Using RF Sensors and Convolutional Neural Networks. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing* (Brighton, UK) (ICASSP 2019). 3662–3666. <https://doi.org/10.1109/ICASSP.2019.8682296>
- [2] Shahzad Ahmed, Faheem Khan, Asim Ghaffar, Farhan Hussain, and Sung Ho Cho. 2019. Finger-Counting-Based Gesture Recognition within Cars Using Impulse Radar with Convolutional Neural Network. *Sensors* 19, 6 (2019), 1–14. <https://doi.org/10.3390/s19061429>
- [3] Roland Aigner, Daniel Wigdor, Hrvoje Benko, Michael Haller, David Lindbauer, Alexandra Ion, Shengdong Zhao, and Jeffrey Tzu Kwan Valino Koh. 2012. *Understanding Mid-Air Hand Gestures: A Study of Human Preferences in Usage of Gesture Types for HCI*. Technical Report MSR-TR-2012-111. <https://www.microsoft.com/en-us/research/publication/understanding-mid-air-hand-gestures-a-study-of-human-preferences-in-usage-of-gesture-types-for-hci/>
- [4] Ahmad Akl and Shahrokh Valaee. 2010. Accelerometer-based gesture recognition via dynamic-time warping, affinity propagation, & compressive sensing. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing* (Dallas, TX, USA) (ICASSP 2010). 2270 – 2273. <https://doi.org/10.1109/ICASSP.2010.5495895>
- [5] Mohammed Alloulah, Anton Isopoussu, and Fahim Kawsar. 2018. On Indoor Human Sensing Using Commodity Radar. In *Proceedings of the ACM International Joint Conference and International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers* (Singapore, Singapore) (UbiComp ’18). Association for Computing Machinery, New York, NY, USA, 1331–1336. <https://doi.org/10.1145/3267305.3274180>
- [6] Moeness G. Amin, Zhengxin Zeng, and Tao Shan. 2019. Hand Gesture Recognition based on Radar Micro-Doppler Signature Envelopes. In *Proceedings of the IEEE Radar Conference* (Boston, MA, USA) (RadarConf ’19). 1–6. <https://doi.org/10.1109/RADAR.2019.8835661> ISSN: 2375-5318.
- [7] Nuwan T. Attygalle, Luis A. Leiva, Matjaz Kljun, Christian Sandor, Alexander Plopski, Hirokazu Kato, and Klen Copic Pucihar. 2021. No Interface, No Problem: Gesture Recognition on Physical Objects Using Radar Sensing. *Sensors* 21, 17 (2021), 5771. <https://doi.org/10.3390/s21175771>
- [8] Daniel Avrahami, Mitesh Patel, Yusuke Yamaura, and Sven Kratz. 2018. Below the Surface: Unobtrusive Activity Recognition for Work Surfaces Using RF-Radar Sensing. In *Proceedings of the 23rd ACM International Conference on Intelligent User Interfaces* (Tokyo, Japan) (IUI ’18). Association for Computing Machinery, New York, NY, USA, 439–451. <https://doi.org/10.1145/3172944.3172962>
- [9] Daniel Avrahami, Mitesh Patel, Yusuke Yamaura, Sven Kratz, and Matthew Cooper. 2019. Unobtrusive Activity Recognition and Position Estimation for Work Surfaces Using RF-Radar Sensing. *ACM Trans. Interact. Intell. Syst.* 10, 1, Article 11 (Aug. 2019), 28 pages. <https://doi.org/10.1145/3241383>

- [10] Abel Díaz Berenguer, Meshia Cédric Oveneke, Habib-Ur-Rehman Khalid, Mitchel Alioscha-Perez, André Bourdoux, and Hichem Sahli. 2019. GestureVLAD: Combining Unsupervised Features Representation and Spatio-Temporal Aggregation for Doppler-Radar Gesture Recognition. *IEEE Access* 7 (2019), 137122–137135. <https://doi.org/10.1109/ACCESS.2019.2942305>
- [11] Abhishake Kumar Bojja, Franziska Mueller, Sri Raghu Malireddi, Markus Oberweger, Vincent Lepetit, Christian Theobalt, Kwang Moo Yi, and Andrea Tagliasacchi. 2019. HandSeg: An Automatically Labeled Dataset for Hand Segmentation from Depth Images. In *Proceedings of the 16th IEEE Conference on Computer and Robot Vision* (Kingston, QC, Canada) (CRV '19). IEEE Press, Piscataway, NJ, USA, 151–158. <https://doi.org/10.1109/CRV.2019.00028>
- [12] Léon Bottou. 1998. *Online Algorithms and Stochastic Approximations*. Cambridge University Press, Cambridge, UK.
- [13] Sanders Brandon. 2014. *Mastering Leap Motion*. Packt Publishing, Birmingham.
- [14] Alexandre Calado, Paolo Roselli, Vito Errico, Nathan Magrofuoco, Jean Vanderdonckt, and Giovanni Saggio. 2022. A Geometric Model-Based Approach to Hand Gesture Recognition. *IEEE Trans. Syst. Man Cybern. Syst.* 52, 10 (2022), 6151–6161. <https://doi.org/10.1109/TSMC.2021.3138589>
- [15] Necati Cihan Camgöz, Ahmet Alp Kindiroglu, and Lale Akarun. 2015. Gesture Recognition Using Template Based Random Forest Classifiers. In *Computer Vision - ECCV 2014 Workshops*, Lourdes Agapito, Michael M. Bronstein, and Carsten Rother (Eds.). Springer International Publishing, Cham, 579–594. https://doi.org/10.1007/978-3-319-16178-5_41
- [16] Ariel Caputo, Andrea Giachetti, Simone Soso, Deborah Pintani, Andrea D'Eusano, Stefano Pini, Guido Borghi, Alessandro Simoni, Roberto Vezzani, Rita Cucchiara, Hai-Dang Nguyen, Minh-Triet Tran, and Andrea Giachetti. 2021. SHREC 2021: Skeleton-based hand gesture recognition in the wild. *Computer Graphics* 99 (2021), 201–211. <https://doi.org/10.1016/j.cag.2021.07.007>
- [17] Fabio M. Caputo, S. Burato, Gianni Pavan, Théo Voillemin, Hazem Wannous, Jean-Philippe Vandeborbe, Mehran Maghoumi, Eugene M. Taranta II, Alaleh Razmjoo, Joseph J. LaViola Jr., Fabio Manganaro, Stefano Pini, Guido Borghi, Roberto Vezzani, Rita Cucchiara, Hai-Dang Nguyen, Minh-Triet Tran, and Andrea Giachetti. 2019. Online Gesture Recognition. In *Eurographics Workshop on 3D Object Retrieval*, Silvia Biasotti, Guillaume Lavoué, and Remco Veltkamp (Eds.). The Eurographics Association, 93–102. <https://doi.org/10.2312/3dor.20191067>
- [18] Fabio M. Caputo, Pietro Prebianca, Alessandro Carcangiu, Lucio D. Spano, and Andrea Giachetti. 2018. Comparing 3D trajectories for simple mid-air gesture recognition. *Computers & Graphics* 73 (2018), 17 – 25. <https://doi.org/10.1016/j.cag.2018.02.009>
- [19] Zhaoxi Chen, Gang Li, Francesco Fioranelli, and Hugh Griffiths. 2019. Dynamic Hand Gesture Classification Based on Multistatic Radar Micro-Doppler Signatures Using Convolutional Neural Network. In *2019 IEEE Radar Conference (RadarConf)*. 1–5. <https://doi.org/10.1109/RADAR.2019.8835796> ISSN: 2375-5318.
- [20] Hong Cheng, Lu Yang, and Zicheng Liu. 2016. Survey on 3D Hand Gesture Recognition. *IEEE Transactions on Circuits and Systems for Video Technology* 26, 9 (2016), 1659–1673. <https://doi.org/10.1109/TCSVT.2015.2469551>
- [21] Jae-Woo Choi, Si-Jung Ryu, and Jong-Hwan Kim. 2019. Short-Range Radar Based Real-Time Hand Gesture Recognition Using LSTM Encoder. *IEEE Access* 7 (2019), 33610–33618. <https://doi.org/10.1109/ACCESS.2019.2903586>
- [22] Chuan Du, Lei Zhang, Xiping Sun, Junxu Wang, and Jialian Sheng. 2020. Enhanced Multi-Channel Feature Synthesis for Hand Gesture Recognition Based on CNN With a Channel and Spatial Attention Mechanism. *IEEE Access* 8 (2020), 144610–144620. <https://doi.org/10.1109/ACCESS.2020.3010063>
- [23] Matthias G. Ehrnsperger, Henri L. Hoese, and Uwe Siart and Thomas F. Eibert. 2019. Performance Investigation of Machine Learning Algorithms for Simple Human Gesture Recognition Employing an Ultra Low Cost Radar System. In *2019 Kleinheubach Conference*. 1–4.
- [24] Zak Flintoff, Bruno Johnston, and Minas Liarokapis. 2018. Single-Grasp, Model-Free Object Classification using a Hyper-Adaptive Hand, Google Soli, and Tactile Sensors. In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS '18)*. 1943–1950. <https://doi.org/10.1109/IROS.2018.8594166>
- [25] Bogdan-Florin Gheran, Jean Vanderdonckt, and Radu-Daniel Vatavu. 2018. Gestures for Smart Rings: Empirical Results, Insights, and Design Implications. In *Proceedings of the 2018 Designing Interactive Systems Conference* (Hong Kong, China) (DIS '18). Association for Computing Machinery, New York, NY, USA, 623–635. <https://doi.org/10.1145/3196709.3196741>
- [26] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep Sparse Rectifier Neural Networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 15)*, Geoffrey Gordon, David Dunson, and Miroslav Dudík (Eds.). PMLR, Fort Lauderdale, FL, USA, 315–323. <https://proceedings.mlr.press/v15/glorot11a.html>
- [27] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Softmax Units for Multinoulli Output Distributions*. MIT Press, New York, NY, USA, 180–184. <http://www.deeplearningbook.org>.
- [28] Trevor Hastie, Robert Tibshirani, and Jerome H. Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edition*. Springer. <https://doi.org/10.1007/978-0-387-84858-7>

- [29] Eiji Hayashi, Jaime Lien, Nicholas Gillian, Leonardo Giusti, Dave Weber, Jin Yamanaka, Lauren Bedal, and Ivan Poupyrev. 2021. RadarNet: Efficient Gesture Recognition Technique Utilizing a Miniature Radar Sensor. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) (CHI '21). Association for Computing Machinery, New York, NY, USA, Article 5, 14 pages. <https://doi.org/10.1145/3411764.3445367>
- [30] Souvik Hazra and Avik Santra. 2019. Short-Range Radar-Based Gesture Recognition System Using 3D CNN With Triplet Loss. *IEEE Access* 7 (2019), 125623–125633. <https://doi.org/10.1109/ACCESS.2019.2938725>
- [31] Michael T. Heideman, Don H. Johnson, and C. Sidney Burrus. 1984. Gauss and the history of the fast fourier transform. *IEEE ASSP Magazine* 1, 4 (oct 1984), 14–21. <https://doi.org/10.1109/MASSP.1984.1162257>
- [32] Michael Hoffman, Paul Varcholik, and Joseph J. LaViola. 2010. Breaking the status quo: Improving 3D gesture recognition with spatially convenient input devices. In *Proceedings of the IEEE Virtual Reality Conference* (Boston, MA, USA) (VR '10). IEEE Computer Society Press, Los Alamitos, USA, 59–66. <https://doi.org/10.1109/VR.2010.5444813>
- [33] Yuan Hu, Lei Chen, Zhibin Wang, Xiang Pan, and Hao Li. 2022. Towards a More Realistic and Detailed Deep-Learning-Based Radar Echo Extrapolation Method. *Remote Sensing* 14, 1 (2022). <https://doi.org/10.3390/rs14010024>
- [34] Cloe Huesser, Simon Schubiger, and Arzu Çöltekin. 2021. Gesture Interaction in Virtual Reality. In *Human-Computer Interaction – INTERACT 2021*. Springer International Publishing, Cham, 151–160.
- [35] Interuniversity Microelectronics Centre (IMEC). 2019. 140 GHz radar for gesture recognition technology and driver monitorin. <https://www.imec-int.com/en/expertise/radar-sensing-systems/140ghz-radar-modules>
- [36] Suzanne Kieffer, Ugo Braga Sangiorgi, and Jean Vanderdonckt. 2015. ECOVAL: A Framework for Increasing the Ecological Validity in Usability Testing. In *Proceedings of 48th Hawaii International Conference on System Sciences* (Kauai, Hawaii, USA) (HICSS 2015), Tung X. Bui and Ralph H. Sprague Jr. (Eds.). IEEE Computer Society, 452–461. <https://doi.org/10.1109/HICSS.2015.61>
- [37] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- [38] Utkarsh Kunwar, Sheetal Borar, Moritz Berghofer, Julia Kylmä, İlhan Aslan, Luis A. Leiva, and Antti Oulasvirta. 2022. Robust and Deployable Gesture Recognition for Smartwatches. In *Proceedings of 27th International Conference on Intelligent User Interfaces* (Helsinki, Finland) (IUI '22), Giulio Jacucci, Samuel Kaski, Cristina Conati, Simone Stumpf, Tuukka Ruotsalo, and Krzysztof Gajos (Eds.). ACM, 277–291. <https://doi.org/10.1145/3490099.3511125>
- [39] Shengchang Lan, Zonglong He, Kai Yao, and Weichu Chen. 2018. Hand Gesture Recognition using a Three-dimensional 24 GHz Radar Array. In *Proceedings of the IEEE/MTT-S International Microwave Symposium (IMS '18)*. 138–140. <https://doi.org/10.1109/MWSYM.2018.8439658> ISSN: 2576-7216.
- [40] Gierad Laput and Chris Harrison. 2019. Sensing Fine-Grained Hand Activity with Smartwatches. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (Glasgow, Scotland, UK) (CHI '19). Association for Computing Machinery, New York, NY, USA, 1–13. <https://doi.org/10.1145/3290605.3300568>
- [41] Michalis Lazarou, Bo Li, and Tania Stathaki. 2021. A novel shape matching descriptor for real-time static hand gesture recognition. *Computer Vision and Image Understanding* 210 (2021), 103241. <https://doi.org/10.1016/j.cviu.2021.103241>
- [42] Luis A. Leiva, Matjaz Kljun, Christian Sandor, and Klen Copic Pucihar. 2020. The Wearable Radar: Sensing Gestures Through Fabrics. In *22nd International Conference on Human-Computer Interaction with Mobile Devices and Services* (Oldenburg, Germany) (MobileHCI '20). Association for Computing Machinery, New York, NY, USA, Article 17, 4 pages. <https://doi.org/10.1145/3406324.3410720>
- [43] Feifei Li, Yujun Li, Baozhen Du, Hongji Xu, Hailiang Xiong, and Min Chen. 2019. A Gesture Interaction System Based on Improved Finger Feature and WE-KNN. In *Proceedings of the 2019 4th International Conference on Mathematics and Artificial Intelligence* (Chengdu, China) (ICMAI 2019). Association for Computing Machinery, New York, NY, USA, 39–43. <https://doi.org/10.1145/3325730.3325759>
- [44] Tianxing Li, Xi Xiong, Yifei Xie, George Hito, Xing-Dong Yang, and Xia Zhou. 2017. Reconstructing Hand Poses Using Visible Light. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 3, Article 71 (sep 2017), 20 pages. <https://doi.org/10.1145/3130937>
- [45] Jaime Lien, Nicholas Gillian, M. Emre Karagozler, Patrick Amihood, Carsten Schwesig, Erik Olson, Hakim Raja, and Ivan Poupyrev. 2016. Soli: Ubiquitous Gesture Sensing with Millimeter Wave Radar. *ACM Trans. Graph.* 35, 4 (July 2016). <https://doi.org/10.1145/2897824.2925953>
- [46] Haipeng Liu, Yuheng Wang, Anfu Zhou, Hanyue He, Wei Wang, Kunpeng Wang, Peilin Pan, Yixuan Lu, Liang Liu, and Huadong Ma. 2020. Real-time Arm Gesture Recognition in Smart Home Scenarios via Millimeter Wave Sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 4 (Dec. 2020), 140:1–140:28. <https://doi.org/10.1145/3432235>
- [47] Jiayang Liu, Lin Zhong, Jehan Wickramasuriya, and Venu Vasudevan. 2009. UWave: Accelerometer-Based Personalized Gesture Recognition and Its Applications. *Pervasive Mob. Comput.* 5, 6 (Dec. 2009), 657–675. <https://doi.org/10.1016/j.pmcj.2009.07.007>

- [48] Xinye Lou, Zhiwen Yu, Zhu Wang, Kaijie Zhang, and Bin Guo. 2018. Gesture-Radar: Enabling Natural Human-Computer Interactions with Radar-Based Adaptive and Robust Arm Gesture Recognition. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. 4291–4297. <https://doi.org/10.1109/SMC.2018.00726> ISSN: 2577-1655.
- [49] Naveen Madapana and Juan P. Wachs. 2021. ZF-SSE: A Unified Sequential Semantic Encoder for Zero-Few-Shot Learning. In *Proceedings of the 16th IEEE International Conference on Automatic Face and Gesture Recognition* (Jodhpur, India) (FG 2021). IEEE, 1–8. <https://doi.org/10.1109/FG52635.2021.9667025>
- [50] Mehran Maghoumi and Joseph J. LaViola Jr. 2019. DeepGRU: Deep Gesture Recognition Utility. In *Proceedings of the 14th International Symposium on Visual Computing, ISVC 2019, Lake Tahoe, NV, USA, October 7-9, 2019, Part I (Lecture Notes in Computer Science, Vol. 11844)*, George Bebis, Richard Boyle, Bahram Parvin, Darko Koracin, Daniela Ushizima, Sek Chai, Shinjiro Sueda, Xin Lin, Aidong Lu, Daniel Thalmann, Chaoli Wang, and Panpan Xu (Eds.). Springer, 16–31. https://doi.org/10.1007/978-3-030-33720-9_2
- [51] Mehran Maghoumi, Eugene Matthew Taranta, and Joseph J. LaViola. 2021. DeepNAG: Deep Non-Adversarial Gesture Generation. In *Proc. of 26th International Conference on Intelligent User Interfaces* (College Station, TX, USA) (IUI '21), Tracy Hammond, Katrien Verbert, Dennis Parra, Bart P. Knijnenburg, John O'Donovan, and Paul Teale (Eds.). ACM, 213–223. <https://doi.org/10.1145/3397481.3450675>
- [52] Nathan Magrofuoco, Paolo Roselli, and Jean Vanderdonckt. 2021. Two-Dimensional Stroke Gesture Recognition: A Survey. *ACM Comput. Surv.* 54, 7, Article 155 (jul 2021), 36 pages. <https://doi.org/10.1145/3465400>
- [53] Nathan Magrofuoco, Paolo Roselli, and Jean Vanderdonckt. 2022. μV : An Articulation, Rotation, Scaling, and Translation Invariant (ARST) Multi-stroke Gesture Recognizer. *Proc. ACM Hum. Comput. Interact.* 6, EICS (2022), 150:1–150:25. <https://doi.org/10.1145/3532200>
- [54] Antigoni Mezari and Ilias Maglogiannis. 2017. Gesture Recognition Using Symbolic Aggregate Approximation and Dynamic Time Warping on Motion Data. In *Proceedings of the 11th EAI International Conference on Pervasive Computing Technologies for Healthcare* (Barcelona, Spain) (PervasiveHealth '17). Association for Computing Machinery, New York, NY, USA, 342–347. <https://doi.org/10.1145/3154862.3154927>
- [55] Tumuganti NagaKarthik, Eun Hye Ahn, Yun Sik Bae, and Jun Rim Choi. 2013. TCAM based pattern matching technique for hand gesture recognition. In *2013 International SoC Design Conference (ISOCC)*. 368–369. <https://doi.org/10.1109/ISOCC.2013.6864052>
- [56] Institute of Electrical and Electronics Engineers. 2020. IEEE Standard Letter Designations for Radar-Frequency Bands. *IEEE Std 521-2019 (Revision of IEEE Std 521-2002)* (2020), 1–15. <https://doi.org/10.1109/IEEESTD.2020.8999849>
- [57] Sameera Palipana, Dariush Salami, Luis A. Leiva, and Stephan Sigg. 2021. Pantomime: Mid-Air Gesture Recognition with Sparse Millimeter-Wave Radar Point Clouds. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 1 (March 2021), 27:1–27:27. <https://doi.org/10.1145/3448110>
- [58] Joseph Paradiso, Craig Abler, Kai-yuh Hsiao, and Matthew Reynolds. 1997. The Magic Carpet: Physical Sensing for Immersive Environments. In *Proceedings of the ACM Extended Abstracts on Human Factors in Computing Systems* (Atlanta, Georgia) (CHI EA '97). Association for Computing Machinery, New York, NY, USA, 277–278. <https://doi.org/10.1145/1120212.1120391>
- [59] Avishek Patra, Philipp Geuer, Andrea Munari, and Petri Mähönen. 2018. mm-Wave Radar Based Gesture Recognition: Development and Evaluation of a Low-Power, Low-Complexity System. In *Proceedings of the 2nd ACM Workshop on Millimeter Wave Networks and Sensing Systems (mmNets '18)*. Association for Computing Machinery, New York, NY, USA, 51–56. <https://doi.org/10.1145/3264492.3264501>
- [60] Jorge-Luis Pérez-Medina, Santiago Villarreal, and Jean Vanderdonckt. 2020. A Gesture Elicitation Study of Nose-Based Gestures. *Sensors* 20, 24 (2020). <https://doi.org/10.3390/s20247118>
- [61] Takuya Sakamoto, Xiaomeng Gao, Ehsan Yavari, Ashikur Rahman, Olga Boric-Lubecke, and Victor M. Lubecke. 2018. Hand Gesture Recognition Using a Radar Echo I–Q Plot and a Convolutional Neural Network. *IEEE Sensors Letters* 2, 3 (Sept. 2018), 1–4. <https://doi.org/10.1109/LSSENS.2018.2866371>
- [62] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. 2018. How does batch normalization help optimization? *Advances in neural information processing systems* 31 (2018).
- [63] Alexandru-Ionuț Slean, Cristian Pamparău, Arthur Sluțters, Radu-Daniel Vatavu, and Jean Vanderdonckt. 2023. Flexible gesture input with radars: systematic literature review and taxonomy of radar sensing integration in ambient intelligence environments. *Journal of Ambient Intelligence and Humanized Computing* (2023). <https://doi.org/10.1007/s12652-023-04606-9>
- [64] Alexandru-Ionuț Slean, Cristian Pamparău, and Radu-Daniel Vatavu. 2022. Scenario-Based Exploration of Integrating Radar Sensing into Everyday Objects for Free-Hand Television Control. In *Proceedings of ACM International Conference on Interactive Media Experiences* (Aveiro, JB, Portugal) (IMX '22). Association for Computing Machinery, New York, NY, USA, 357–362. <https://doi.org/10.1145/3505284.3532982>
- [65] Sruthy Skaria, Da Huang, Akram Al-Hourani, Robin J. Evans, and Margaret Lech. 2020. Deep-Learning for Hand-Gesture Recognition with Simultaneous Thermal and Radar Sensors. In *2020 IEEE Sensors*. 1–4. <https://doi.org/10.1109/Sensors48622.2020.9352982>

- 1109/SENSORS47125.2020.9278683 ISSN: 2168-9229.
- [66] Arthur Sluÿters, Sébastien Lambot, and Jean Vanderdonckt. 2022. Hand Gesture Recognition for an Off-the-Shelf Radar by Electromagnetic Modeling and Inversion. In *Proceedings of 27th International ACM Conference on Intelligent User Interfaces* (Helsinki, Finland) (IUI '22). Association for Computing Machinery, New York, NY, USA, 506–522. <https://doi.org/10.1145/3490099.3511107>
 - [67] Arthur Sluÿters, Sébastien Lambot, Jean Vanderdonckt, and Radu-Daniel Vatavu. 2023. RadarSense: Accurate Recognition of Mid-Air Hand Gestures with Radar Sensing and Few Training Examples. *ACM Transactions on Interactive Intelligent Systems* (March 2023). <https://doi.org/10.1145/3589645>
 - [68] Arthur Sluÿters, Quentin Sellier, Jean Vanderdonckt, Vik Parthiban, and Pattie Maes. 2022. Consistent, Continuous, and Customizable Mid-Air Gesture Interaction for Browsing Multimedia Objects on Large Displays. *International Journal of Human-Computer Interaction* (2022), 1–32. <https://doi.org/10.1080/10447318.2022.2078464> arXiv:<https://doi.org/10.1080/10447318.2022.2078464>
 - [69] Quentin De Smedt, Hazem Wannous, Jean-Philippe Vandeborre, Joris Guerry, Bertrand Le Saux, and David Filliat. 2017. 3D Hand Gesture Recognition Using a Depth and Skeletal Dataset. In *Eurographics Workshop on 3D Object Retrieval*, Ioannis Pratikakis, Florent Dupont, and Maks Ovsjanikov (Eds.). The Eurographics Association. <https://doi.org/10.2312/3dor.20171049>
 - [70] Adrian Tang, Robert Carey, Gabriel Virbila, Yan Zhang, Rulin Huang, and Mau-Chung Frank Chang. 2020. A Delay-Correlating Direct-Sequence Spread-Spectrum (DS/SS) Radar System-on-Chip Operating at 183–205 GHz in 28 nm CMOS. *IEEE Transactions on Terahertz Science and Technology* 10, 2 (2020), 212–220. <https://doi.org/10.1109/TTHZ.2020.2969105>
 - [71] Eugene M. Taranta II, Amirreza Samiei, Mehran Maghoumi, Pooya Khaloo, Corey R. Pittman, and Joseph J. LaViola Jr. 2017. Jackknife: A Reliable Recognizer with Few Samples and Many Modalities. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) (CHI '17). ACM, New York, NY, USA, 5850–5861. <https://doi.org/10.1145/3025453.3026002>
 - [72] Radu-Daniel Vatavu. 2011. The Effect of Sampling Rate on the Performance of Template-Based Gesture Recognizers. In *Proceedings of the 13th International Conference on Multimodal Interfaces* (Alicante, Spain) (ICMI '11). Association for Computing Machinery, New York, NY, USA, 271–278. <https://doi.org/10.1145/2070481.2070531>
 - [73] Radu-Daniel Vatavu. 2013. The impact of motion dimensionality and bit cardinality on the design of 3D gesture recognizers. *International Journal of Human-Computer Studies* 71, 4 (2013), 387 – 409. <https://doi.org/10.1016/j.ijhcs.2012.11.005>
 - [74] Radu-Daniel Vatavu. 2023. Gesture-Based Interaction. In *Handbook of Human-Computer Interaction*, Jean Vanderdonckt, Philippe Palanque, and Marco Winckler (Eds.). Springer International Publishing, Cham, 1–47. https://doi.org/10.1007/978-3-319-27648-9_20-1
 - [75] Radu-Daniel Vatavu. 2023. IFAD Gestures: Understanding Users' Gesture Input Performance with Index-Finger Augmentation Devices. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) (CHI '23). Association for Computing Machinery, New York, NY, USA, Article 576, 17 pages. <https://doi.org/10.1145/3544548.3580928>
 - [76] Radu-Daniel Vatavu, Lisa Anthony, and Jacob O. Wobbrock. 2012. Gestures As Point Clouds: A \$P Recognizer for User Interface Prototypes. In *Proceedings of the 14th ACM International Conference on Multimodal Interaction* (Santa Monica, California, USA) (ICMI '12). ACM, New York, NY, USA, 273–280. <https://doi.org/10.1145/2388676.2388732>
 - [77] Radu-Daniel Vatavu, Lisa Anthony, and Jacob O. Wobbrock. 2018. \$Q: A Super-quick, Articulation-invariant Stroke-gesture Recognizer for Low-resource Devices. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services* (Barcelona, Spain) (MobileHCI '18). ACM, New York, NY, USA, Article 23, 12 pages. <https://doi.org/10.1145/3229434.3229465>
 - [78] Klen Čopić Pucihar, Nuwan T. Attygalle, Matjaz Kljun, Christian Sandor, and Luis A. Leiva. 2022. Solids on Soli: Millimetre-Wave Radar Sensing through Materials. *Proc. ACM Hum.-Comput. Interact.* 6, EICS, Article 156 (jun 2022), 19 pages. <https://doi.org/10.1145/3532212>
 - [79] Santiago Villarreal-Narvaez, Alexandru-Ionuț Șiean, Arthur Sluÿters, Radu-Daniel Vatavu, and Jean Vanderdonckt. 2022. Informing Future Gesture Elicitation Studies for Interactive Applications That Use Radar Sensing. In *Proceedings of the 2022 International Conference on Advanced Visual Interfaces* (Frascati, Rome, Italy) (AVI 2022). Association for Computing Machinery, New York, NY, USA, Article 50, 3 pages. <https://doi.org/10.1145/3531073.3534475>
 - [80] Oleh Viunytskyi and Alexander Totsky. 2017. Novel bispectrum-based wireless vision technique using disturbance of electromagnetic field by human gestures. In *2017 Signal Processing Symposium (SPSymposium)*. 1–4. <https://doi.org/10.1109/SPS.2017.8053684>
 - [81] Jinqiang Wang, Dianguo Cao, Yang Li, Jiashuai Wang, and Yuqiang Wu. 2022. Multi-user motion recognition using sEMG via discriminative canonical correlation analysis and adaptive dimensionality reduction. *Frontiers in Neurorobotics* 16, 997134 (2022). <https://doi.org/10.3389/fnbot.2022.997134>

- [82] Liying Wang, Zongjie Cao, Zongyong Cui, Changjie Cao, and Yiming Pi. 2020. Negative Latency Recognition Method for Fine-Grained Gestures Based on Terahertz Radar. *IEEE Transactions on Geoscience and Remote Sensing* 58, 11 (Nov. 2020), 7955–7968. <https://doi.org/10.1109/TGRS.2020.2985421>
- [83] Pengcheng Wang, Junyang Lin, Fuyue Wang, Jianping Xiu, Yue Lin, Na Yan, and Hongtao Xu. 2020. A Gesture Air-Writing Tracking Method that Uses 24 GHz SIMO Radar SoC. *IEEE Access* 8 (2020), 152728–152741. <https://doi.org/10.1109/ACCESS.2020.3017869>
- [84] Saiwen Wang, Jie Song, Jaime Lien, Ivan Poupyrev, and Otmar Hilliges. 2016. Interacting with Soli: Exploring Fine-Grained Dynamic Gesture Recognition in the Radio-Frequency Spectrum. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) (UIST '16). Association for Computing Machinery, New York, NY, USA, 851–860. <https://doi.org/10.1145/2984511.2984565>
- [85] Yong Wang, Aihu Ren, Mu Zhou, Wen Wang, and Xiaobo Yang. 2020. A Novel Detection and Recognition Method for Continuous Hand Gesture Using FMCW Radar. *IEEE Access* 8 (2020), 167264–167275. <https://doi.org/10.1109/ACCESS.2020.3023187>
- [86] Jacob O. Wobbrock, Meredith Ringel Morris, and Andrew D. Wilson. [n.d.]. User-defined Gestures for Surface Computing. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (New York, NY, USA, 2009) (CHI '09). ACM, 1083–1092. <https://doi.org/10.1145/1518701.1518866>
- [87] Christian Wolff. 2022. Waves and Frequency Ranges. <https://www.radartutorial.eu/07.waves/WavesandFrequencyRanges.en.html>
- [88] Hao Xu and Anbin Xiong. 2021. Advances and Disturbances in sEMG-Based Intentions and Movements Recognition: A Review. *IEEE Sensors Journal* 21, 12 (2021), 13019–13028. <https://doi.org/10.1109/JSEN.2021.3068521>
- [89] Mais Yassen and Shaidah Jusoh. 2019. A systematic review on hand gesture recognition techniques, challenges and applications. *PeerJ Computer Science* 5 (Sept. 2019), e218. <https://doi.org/10.7717/peerj-cs.218>
- [90] Hui-Shyong Yeo, Gergely Flamich, Patrick Schrempf, David Harris-Birtill, and Aaron Quigley. 2016. RadarCat: Radar Categorization for Input & Interaction. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) (UIST '16). Association for Computing Machinery, New York, NY, USA, 833–841. <https://doi.org/10.1145/2984511.2984515>
- [91] Hui-Shyong Yeo, Ryosuke Minami, Kirill Rodriguez, George Shaker, and Aaron Quigley. 2018. Exploring Tangible Interactions with Radar Sensing. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 4, Article 200 (Dec. 2018), 25 pages. <https://doi.org/10.1145/3287078>
- [92] Hui-Shyong Yeo and Aaron Quigley. 2017. Radar Sensing in Human-Computer Interaction. *Interactions* 25, 1 (Dec. 2017), 70–73. <https://doi.org/10.1145/3159651>
- [93] Bo Zhang, Lei Zhang, Mojun Wu, and Yan Wang. 2021. Dynamic Gesture Recognition Based on RF Sensor and AE-LSTM Neural Network. In *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*. 1–5. <https://doi.org/10.1109/ISCAS51556.2021.9401065>
- [94] Jiajun Zhang, Jinkun Tao, and Zhiguo Shi. 2019. Doppler-Radar Based Hand Gesture Recognition System Using Convolutional Neural Networks. In *Communications, Signal Processing, and Systems*, Qilian Liang, Jiasong Mu, Min Jia, Wei Wang, Xuhong Feng, and Baoju Zhang (Eds.). Vol. 463. Springer Singapore, Singapore, 1096–1113. https://doi.org/10.1007/978-981-10-6571-2_132 Series Title: Lecture Notes in Electrical Engineering.
- [95] Kaijie Zhang, Zhiwen Yu, Dong Zhang, Zhu Wang, and Bin Guo. 2020. RaCon: A gesture recognition approach via Doppler radar for intelligent human-robot interaction. In *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. 1–6. <https://doi.org/10.1109/PerComWorkshops48775.2020.9156109>
- [96] Zhenyuan Zhang, Zengshan Tian, Ying Zhang, Mu Zhou, and Bang Wang. 2019. u-DeepHand: FMCW Radar-Based Unsupervised Hand Gesture Feature Learning Using Deep Convolutional Auto-Encoder Network. *IEEE Sensors Journal* 19, 16 (Aug. 2019), 6811–6821. <https://doi.org/10.1109/JSEN.2019.2910810>
- [97] Shangyue Zhu, Junhong Xu, Hanqing Guo, Qiwei Liu, Shaoen Wu, and Honggang Wang. 2018. Indoor Human Activity Recognition Based on Ambient Radar with Signal Processing and Machine Learning. In *2018 IEEE International Conference on Communications (ICC)*. 1–6. <https://doi.org/10.1109/ICC.2018.8422107>

Received October 2022; revised February 2023; accepted April 2023