electronics

MDPI

Article

RGB Fusion of Multiple Radar Sensors for Deep Learning-Based Traffic Hand Gesture Recognition

Hüseyin Üzen

Department of Computer Engineering, Faculty of Engineering and Architecture, Bingol University, Bingol 12000, Turkey; huzen@bingol.edu.tr

Abstract

Hand gesture recognition (HGR) systems play a critical role in modern intelligent transportation frameworks by enabling reliable communication between pedestrians, traffic operators, and autonomous vehicles. This work presents a novel traffic hand gesture recognition method that combines nine grayscale radar images captured from multiple millimeter-wave radar nodes into a single RGB representation through an optimized rotation–shift fusion strategy. This transformation preserves complementary spatial information while minimizing inter-image interference, enabling deep learning models to more effectively utilize the distinctive micro-Doppler and spatial patterns embedded in radar measurements. Extensive experimental studies were conducted to verify the model's performance, demonstrating that the proposed RGB fusion approach provides higher classification accuracy than single-sensor or unfused representations. In addition, the proposed model outperformed state-of-the-art methods in the literature with an accuracy of 92.55%. These results highlight its potential as a lightweight yet powerful solution for reliable gesture interpretation in future intelligent transportation and human–vehicle interaction systems.

Keywords: hand gesture recognition; radar gesture recognition; ensemble learning; pretraining

1. Introduction

In the last decade, human activity recognition (HAR) has gained much attention due to its numerous widespread applications in security monitoring, human–computer interaction, and healthcare [1–3]. HAR is often performed via wearable sensor-based methods or RGB cameras. Recently, radar-based applications have emerged in various fields, such as hand gesture recognition [4] and fall detection [5], occupancy sensing [6], activity classification [7], and driver assistance systems [8]. Hand gesture recognition (HGR) is another important radar-based application area where efficient human–computer interaction can be handled [9].

Numerous studies have investigated radar-based HGR. Wang et al. [10] proposed a method that used a mm-wave radar data sequence and a time-distributed–CNN–transformer (TD-CNNT) network to improve HGR accuracy. The approach used a TD wrapper and a CNN to extract localized characteristics from the data cube series. In addition, a position encoder was utilized to maintain temporal information inside the sequence, and a transformer network was employed to capture global attributes. Experimental results indicated that the proposed technique achieved a high HGR accuracy of 99.75%, indicating a significant improvement over the traditional approach. Jin et al. [11] proposed

Academic Editors: Rongye Shi, Qunbo Wang and Si Liu

Received: 1 December 2025
Revised: 26 December 2025
Accepted: 26 December 2025
Published: 28 December 2025

a CNN–Transformer network implementation for HGR based on frequency-modulated millimeter-wave (FM-MW) radar. The approach involves converting FM-MW radar echoes into 3D data. The CNN–Transformer (CNNT) network, which uses CNN to capture local characteristics and stacks Transformer modules for depth, achieved gesture recognition accuracy of 98% and 96% in settings without and with random dynamic interference, respectively. Liu et al. [12] introduced a dynamic HGR approach utilizing a two-step fusion deformable network with Gram Matching (GM). It involved preprocessing multimodal data for time synchronization, combining radar and vision data for gesture classification, and employing deformable convolution to capture gesture motion patterns. Additionally, Long Short-Term Memory (LSTM) units were used for temporal feature extraction, while GM served as a loss function to integrate radar and vision data effectively. Experiments demonstrated the proposed method's efficiency in complex environments and with diverse individuals. Mao et al. [13] developed an efficient HGR framework using multiple FM-MW radar-based sensing systems. The authors deployed two radars to capture diverse motion vectors. A 2D-FFT was employed, enabling the independent extraction of reflection points and providing crucial data for synthesizing motion velocity vectors. By integrating an LSTM network with synthesized motion vectors, 98.0% accuracy was obtained by the authors in HGR using a dataset of 1600 samples. Kern et al. [14] combined PointNet and LSTM for classification of the HGR for complex traffic gesture recognition. PointNet was used for feature extraction in the proposed method, and the LSTM network was used to classify the extracted features into the labeled traffic gesture classes. The experimental setup was designed to handle radar data from different angles. The experiments involved 35 people, and the authors reported a 92.2% accuracy score. Guo et al. [15] developed a methodology where the similarities between radar data representation and the self-attention model were used for efficient HGR. The authors investigated three self-attention models to classify the HGR. The adaptive clutter cancellation approach was used in the proposed model to determine the most efficient combination of various input variables. The experiments were conducted based on leave-one-out cross-validation, and the authors reported a macro F1 score of approximately 90%. Gao et al. [16] used a 60 GHz FM-MW radar to record violinists' bowing arm movement data, including seven gestures. The authors collected a dataset of 1200 bowing actions from three violinists and used raw radar signal data to generate Time–Doppler spectrograms. Two alternative techniques were considered for feature extraction from Time–Doppler data. In the first technique, a series of handcrafted features were extracted from the raw radar signals, and in the second one, CNNs were used to extract features from spectrogram radar images. Experimental works revealed that fine-tuning a pretrained SqueezeNet model produced a 95.00% accuracy score. Fırat et al. [17] presented a radar image-based, three-input deep learning model for traffic gesture recognition. The proposed approach extracts features by applying separate DenseNet-121 layers to three input images. Then, the Swin Transformer architecture is integrated and the three-input attention mechanism and feature fusion operations were performed on the radar image. The model achieved an average accuracy rate of 90.54% as a result of five-fold cross-validation. In addition to these studies, Chen et al. [18] developed a real-time multimodal human–robot collaboration (HRC) system, recognizing 16 dynamic hand/arm gestures via motion history images and convolutional neural networks; combining these with speech commands, they achieved a recognition accuracy exceeding 98% in industrial scenarios.

The reviewed literature indicates that various approaches have been proposed for radar-based HGR. While these methods offer distinct merits, their classification accuracy could be further optimized through novel and computationally efficient architectures. Consequently, this study introduces a high-precision, low-complexity method tailored for radar-based traffic HGR. Figure 1 illustrates the framework of the proposed approach.

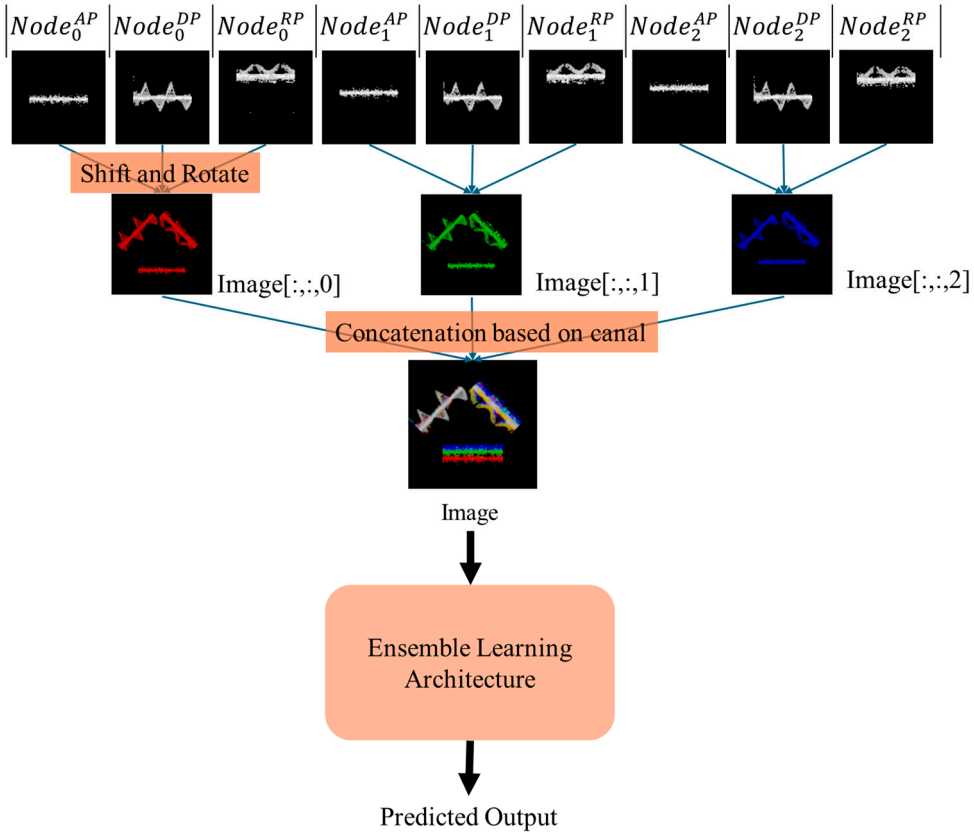


Figure 1. Illustration of the proposed method, showing fusion of nine grayscale radar images (AP: Angular Profile, DP: Doppler Profile, RP: Range Profile from Nodes 0 to 2) into a 128 × 128 RGB image, followed by ensemble classification using pretrained CNNs.

As illustrated in Figure 1, the proposed method takes two-dimensional grayscale radar images with dimensions of 128 × 128 as input. The classification is based on radar data from three sensors (nodes), each providing Doppler, range, and angular profiles. Consequently, a total of nine inputs are utilized to construct a single RGB radar image. Specifically, the input grayscale images from $Node_0$, $Node_1$, and $Node_2$ are initially shifted, rotated, and combined to form the red, green, and blue channels of the final image, respectively. These color channel images are then concatenated along the depth dimension to produce the final color radar image.

After the construction of the input color radar image, an ensemble learning architecture is developed for the classification stage of the proposed model. Two pretrained CNN models are considered in the proposed ensemble-based learning model, and a search mechanism is employed to determine the most appropriate couple of pretrained CNN models. To this end, ResNet18, ResNet34, ResNet50, ResNet101, VGG16, VGG19, DenseNet121, DenseNet169, InceptionV3, and MobileNetV2 models are used. Among these models, the ensemble learning model based on ResNet50 and VGG19 gave the most successful accuracy value of 92.55%. The main contributions of this paper are:

1. A novel fusion strategy for the combination of different radar sensors and measured parameters is proposed. In this strategy, nine different grayscale radar images are converted into an RGB image by shifting, rotating, and adding approaches. In this way, data loss is minimized and provides a powerful input image for the deep learning model.
2. A simple and effective ensemble learning approach was developed using the synthesized RGB radar image. This approach achieved high performance by combining the strengths of previously proposed deep learning models.
3. Comprehensive experimental validation through systematic comparison of individual grayscale radar images, multiple CNN architectures, and ensemble configurations is carried out, demonstrating that the proposed VGG19 + ResNet50 ensemble achieves 92.55% accuracy, surpassing existing literature benchmarks.

The remainder of the paper is arranged as follows. The next part presents a detailed overview of the proposed strategy, including brief introductions to the relevant ideas. We also expound on the related dataset. Section 3 goes into the experimental methodologies used, giving a full overview of the work carried out and presenting the results achieved. Moving on to Section 4, we present discussions based on the results of the trials. In addition, we compare the acquired results to some of the previous results in the discussion section. Finally, Section 5 summarizes the study with conclusions based on our findings and insights into potential directions for additional research and development.

2. Proposed Method

As seen in Figure 1, the proposed method comprises two main parts: construction of the color radar images and ensemble learning based on the pairs of the pretrained CNN models. These parts are introduced in detail in the next subsections.

2.1. Construction of the Color Radar Images

As mentioned earlier, $\text{Node}_0^{\text{AP}}$, $\text{Node}_0^{\text{DP}}$, $\text{Node}_0^{\text{RP}}$, $\text{Node}_1^{\text{AP}}$, $\text{Node}_1^{\text{DP}}$, $\text{Node}_1^{\text{RP}}$, $\text{Node}_2^{\text{AP}}$, $\text{Node}_2^{\text{DP}}$, and $\text{Node}_2^{\text{RP}}$ are grayscale radar representations for three radar sensors (Node), and AP, DP, and RP are used to indicate the Doppler, range, and angular profiles, respectively. The contents of these images have similar characteristics, but they may differ in their signal distribution. Therefore, a novel methodology is considered to combine all grayscale radar images to construct a single colored radar image. However, directly combining or summing all grayscale radar images to create a single radar image can result in significant data loss due to the potential for overlapping of grayscale radar images. To alleviate this issue, a simple but effective approach is developed to combine all grayscale radar images into one image without loss of information. Thus, shifting and rotating operations are applied to the grayscale radar images of each Node. In other words, as shown in Figure 2a, the shifting operation is applied to the AP of the grayscale radar images, and the rotating operations are applied to the DP and RP of the grayscale images, respectively.

As seen in Figure 2, three grayscale radar images are first synthesized and then combined at the channel level. The applied rotation and shifting operations prevent the overlap of significant data within the pixels. This strategy is employed since overlapping pixels in radar images can potentially lead to destructive or constructive interference, thereby dampening or amplifying the signal. Additionally, the primary rationale for adopting an RGB image format is that established models in the literature are optimized for RGB inputs. This approach facilitates the development of a streamlined yet effective model, avoiding the complexities of multi-input architectures. The processing steps of the proposed model are summarized in Algorithm 1.

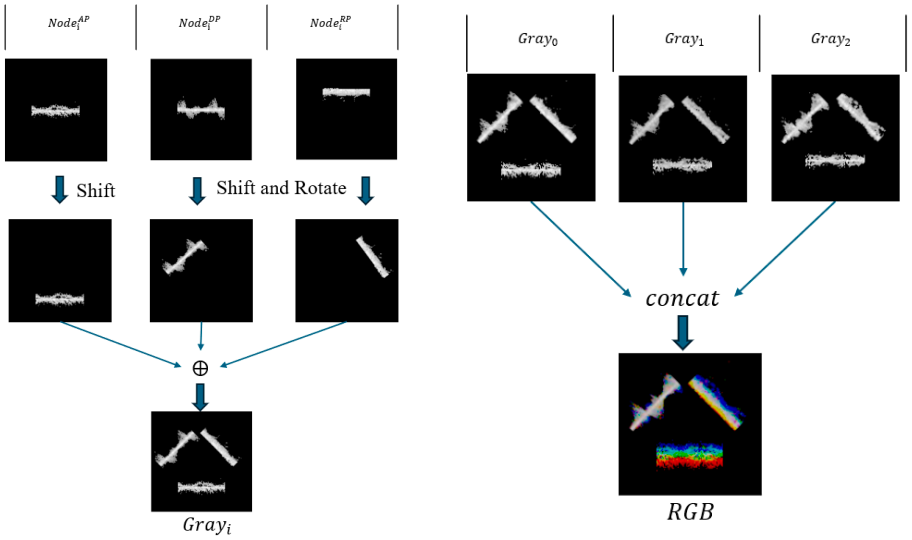


Figure 2. (**a**) Combining AP (shifted), DP (rotated and shifted), and RP (rotated and shifted) images via element-wise addition (⊕) to form a grayscale image per node. (**b**) Concatenating (*concat*) grayscale images from Nodes 0 (Red), 1 (Green), and 2 (Blue) along the depth dimension to create the final 128 × 128 × 3 RGB image.

Algorithm 1. Proposed Rotation Shift-Based RGB Radar Image Construction Pipeline

Input: Nine 64 × 64 grayscale radar images:
AP_0, DP_0, RP_0 (Node 0)
AP_1, DP_1, RP_1 (Node 1)
AP_2, DP_2, RP_2 (Node 2)
Output: Single 128 × 128 × 3 RGB radar image
Δx = [0,−32,32]; Δy = [32,−16,−32]
For each node i ∈ {0,1,2}:
1. Resize AP_i, DP_i, RP_i to 128 × 128 using zero-padding
2. Apply fixed shifts to AP_i to avoid overlap:
$AP_{shifted}$ = shift(AP_i, Δx[0], Δy[0]) // empirically Δx, Δy determined
3. Rotate DP_i and RP_i by predefined angles:
DP_{rot} = rotate(DP_i, 45)
$DP_{rot\_shifted}$ = shift(DP_{rot}, Δx[1], Δy[1])
RP_{rot} = rotate(RP_i,135)
$RP_{rot\_shifted}$ = shift(RP_{rot}, Δx[2], Δy[2])
4. Combine node-wise grayscale image:
$Gray_i$ = $AP_{shifted}$ + $DP_{rot\_shifted}$ + $RP_{rot\_shifted}$
5. Assign to RGB channels (fixed mapping):
if i == 0 → R = Gray_0
if i == 1 → G = Gray_1
if i == 2 → B = Gray_2
I_RGB ← Concatenate(R, G, B; axis = 2)

As shown in Algorithm 1, first, nine 64 × 64 grayscale profiles (AP, DP, RP) belonging to three radar nodes are expanded to 128 × 128 pixels using zero-padding while preserving their center points. To take advantage of the natural sparsity of radar spectrograms and to ensure that the information-carrying signal regions are combined without destructive interference, experimentally optimized constant affine transformations are applied to the

profiles. The spatial transformation operation (rot_shifted) on an input image is formulated as in Equation (1) with the rotation angle θ and the translation vector parameters [Δx, Δy]:

$$\text{rot\_shifted}(x, y, \theta, \Delta x, \Delta y) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \quad (1)$$

where R(θ) represents the 2 × 2 standard rotation matrix that enables rotation relative to the center of the image, while θ represents the rotation angle. In addition, the vector [Δx, Δy] represents the shifting of pixel coordinates along the horizontal and vertical axes.

In the hyperparameter optimization process, angles of 0, 45, 90, and 135 degrees were considered for θ, and values in the range of −64 to +64 were considered for the translation components. As a result of the analyses, the most suitable parameter set that minimizes signal overlap was determined as follows: For the AP signal, θ = 0 degrees, Δx = 0, Δy = 32 (downward shift only along the vertical axis); for the DP signal, θ = 45 degrees, Δx = −32, Δy = −16 (rotation and placement in the upper left quadrant); and for the RP signal, θ = 135 degrees, Δx = 32, Δy = −32 (rotation and placement in the upper right quadrant).

Thanks to these transformations, the energy density of each profile is focused on different and empty regions of the canvas. The combined grayscale image ($Gray_i$) for radar node i is obtained by summing the transformed profiles element by element as in Equation (2):

$$Gray_i = AP_{shifted} + DP_{rot\_shifted} + RP_{rot\_shifted} \quad (2)$$

This process is repeated for each node, preserving the original signal characteristics without loss. In the final stage, the grayscale images obtained from Node 0, Node 1, and Node 2 are assigned to the Red (R), Green (G), and Blue (B) channels, respectively, creating the final 128 × 128 × 3 RGB radar image to be fed into the deep learning model.

2.2. Ensemble Learning Paradigm for RGB Image Classification

The resulting RGB image is classified with an ensemble learning-based classifier in the ensemble learning phase. The proposed ensemble learning model is fed with two pretrained architectures at this stage. To identify these models, extensive analysis was conducted and comparisons were made between various architectures. During the analyses, ResNet [19] (ResNet18 [19], ResNet34 [20], ResNet50 [21], ResNet101 [22]), VGG16-VGG19 [23], DenseNet121-DenseNet169 [24], InceptionV3 [25,26], and MobileNetV2 [27,28] models were used. All models were first put through the training and testing process at this stage alone. These models were then tuned for the ensemble learning process. The proposed model is given in Figure 3.

For ensemble learning, the weights of all layers of the network architecture except the last fully connected layers were frozen. Then, the last layers of the two architectures discussed were combined. The averaging layer was used in the merging process. This layer produces the output by taking the point average of the two layers in the input. As a result of experimental studies, the proposed model was examined in detail, and its performance was evaluated. In addition to the pre-trained architectures, the layers and features of the proposed model itself were also meticulously analyzed. As a result of these analyses, the most suitable architecture for the targeted task of the proposed model was obtained by combining the ResNet50 and VGG19 architectures.

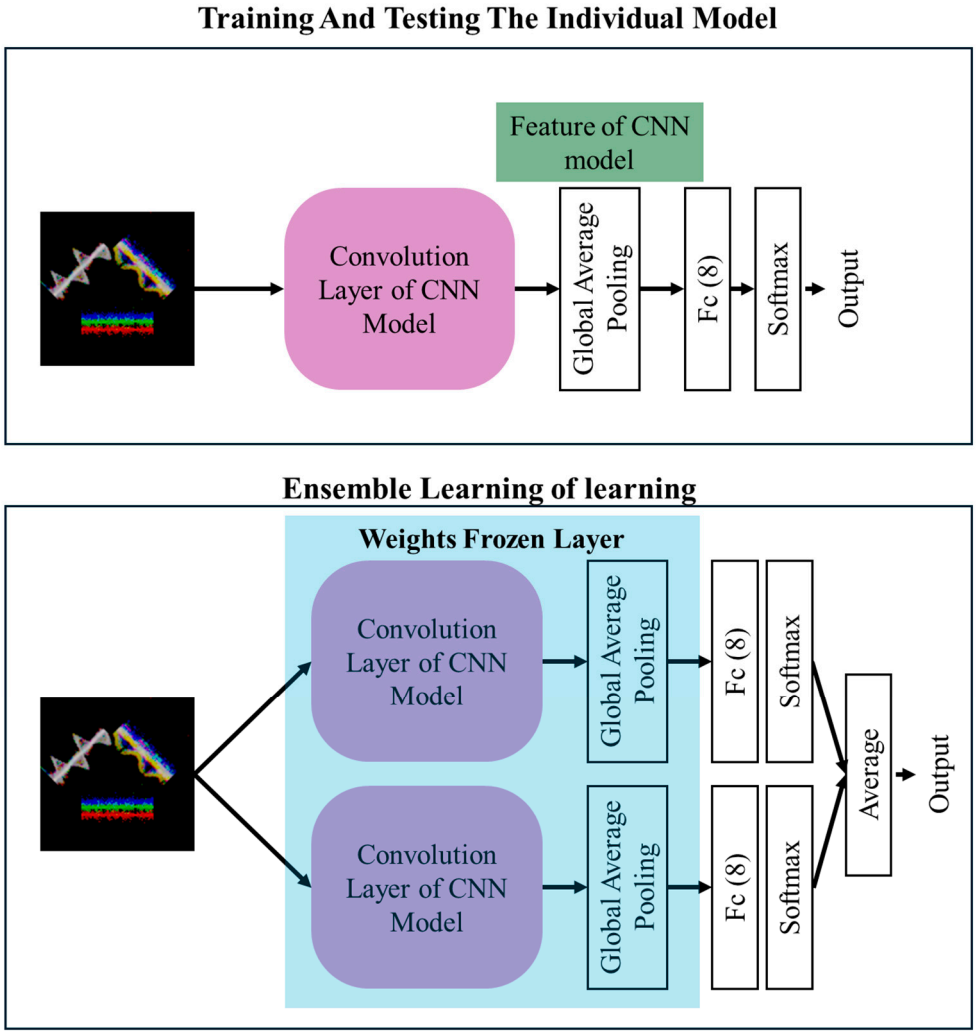


Figure 3. Proposed ensemble learning model structure.

3. Experimental Works

3.1. Dataset

The radar image dataset, created by Kern et al. [14], has been used for traffic gesture recognition and made available as an open dataset for scientific research. The dataset was collected using a sensor network consisting of three 77 GHz automotive chirp sequence radar sensors. The experimental configuration required the installation of three chirp array radar sensors on a rail. Here, there was a distance of 55 cm between Node 0 and Node 1 and a distance of 140 cm between the outermost nodes. This data set includes "Fly" (0), "Approach" (1), "Slow" (2), "Wave" (3), "Move Away" (4), "Transition" (5), "Start" (6), and "Stop" (7). The measurement protocol was strategically designed to accommodate the natural variability of movement orientations in real traffic scenarios, allowing movements in different orientations to be recorded. In total, 35 participants contributed to the dataset, with position specifications loosely determined. Each participant performed movements at nominal orientations of approximately 0° and 90°, optionally adding three additional orientations. The measurement campaign lasted several days and was conducted in two different locations. Specifically, measurements were taken outdoors only for 17 participants and indoors only for 15 participants, while 3 participants took part in both environmental settings. After applying the signal processing and sample creation processes in the MATLAB R2023a (Windows) environment, a dataset consisting of 15,700 samples was available.

3.2. Experimental Setup

This study conducted all experimental research on a computer equipped with RTX 3080 Ti GPU, 64 GB RAM, and an i9 processor. Experimental procedures were based on TensorFlow-Keras libraries. During the processes of training and testing the models, the batch size was set to 32, the learning rate was set to 0.0001, and the standard configuration was 100 epochs. In addition, the Adam optimization approach was used to train the models. Categorical cross-entropy loss function was used for backpropagation. During the process of training the models, the ReducLROnPlateau method was used to select the optimum learning rate. This method works by monitoring the loss value during training, and if no improvement is observed for a certain period, the learning rate decreases by a factor of 0.1. In this way, it ensures that the model captures an effective training procedure. The radar image dataset was used in the experimental studies following the methodology described in Refs. [14,29]. Here, this dataset uses five-fold cross-validation. Additionally, the authors shared the same five-fold division technique as open data for fair comparisons. Only the images in the dataset were resized to 128 x 128 pixels as input to the model. Additionally, performance evaluation in the experimental studies was based on accuracy, F1 score, precision, and recall metrics.

3.3. Results

In this section, the experimental studies are presented. Within this scope, firstly, the performance of the proposed RGB radar images was compared with standard radar images. Secondly, the performance of pre-trained CNN models using RGB radar images was evaluated. Lastly, the classification results of the ensemble learning models were provided.

3.3.1. Performance Comparison of the Proposed RGB Radar Images with the Standard Radar Images

Initially, experiments were carried out to discuss the performance comparison of the proposed RGB radar images with the individual radar images ($Node_0^{AP}$, $Node_0^{DP}$, $Node_0^{RP}$, $Node_1^{AP}$, $Node_1^{DP}$, $Node_1^{RP}$, $Node_2^{AP}$, $Node_2^{DP}$, and $Node_2^{RP}$) on traffic HGR. To this end, pretrained CNN models, namely, ResNet18, ResNet34, ResNet50, ResNet101, DenseNet121, DenseNet169, and InceptionV3, were considered. The obtained classification results are presented in Table 1. As seen in Table 1, the columns represent accuracy scores for the constructed RGB radar image, and the other images that were obtained from the three radar sensors and the rows of Table 1 represent the accuracy scores for the considered pretrained CNN models.

Table 1. Performance (%) comparison of the developed RGB and the other radar images on traffic HGR with pretrained CNN models.

Model	RGB	$Node_0^{AP}$	$Node_0^{DP}$	$Node_0^{RP}$	$Node_1^{AP}$	$Node_1^{DP}$	$Node_1^{RP}$	$Node_2^{AP}$	$Node_2^{DP}$	$Node_2^{RP}$
ResNet18	87.83	47.79	85.05	75.56	44.67	82.88	76.26	46.84	83.98	76.86
Resnet34	88.74	49.68	86.63	77.30	47.82	84.33	76.60	47.47	85.02	73.23
Resnet50	90.88	48.36	86.19	74.36	46.87	83.63	76.89	49.84	85.43	74.62
Resnet101	88.43	48.64	85.90	76.13	47.13	83.66	75.03	50.12	84.70	82.09
Vgg16	91.04	53.62	87.35	81.80	50.91	86.79	80.07	52.96	85.93	83.13
Vgg19	89.97	54.19	87.16	81.80	50.69	85.68	80.42	53.15	85.84	79.57
Densenet121	89.12	50.66	86.94	80.26	49.36	85.84	76.04	47.95	85.24	76.51
Densenet169	89.4	52.23	86.79	79.25	48.58	85.68	79.41	49.93	85.62	77.45
InceptionV3	90.32	50.34	86.50	78.84	47.32	85.30	76.63	54.82	84.77	70.80
MobileNet	88.43	38.11	82.62	72.22	34.58	79.16	67.90	35.05	82.40	76.86

When the accuracy values given in Table 1 are examined, it is seen that the use of the recommended RGB radar images in traffic HGR generally produces higher accuracy. Particularly in deep learning models such as VGG16 and InceptionV3, the accuracy rates achieved using RGB radar images significantly exceed the performance of individual images. For example, in the VGG16 model, the highest accuracy rate obtained with RGB images is 91.04%, while the rate obtained with individual images is between 50.91% and 87.35%. Similarly, the Inception V3 model achieved an accuracy score of 90.32% with RGB images, while the rate achieved with individual images was between 47.32% and 86.5%. This also applies to many other models. For example, the accuracy rates obtained using RGB images in models such as ResNet18, ResNet34, ResNet50, ResNet101, DenseNet121, and DenseNet169 are significantly higher than those obtained with individual images. On the one hand, the highest accuracy score in Table 1 (91.04%) was obtained when the VGG 16 model was used with RGB images, while the highest score (87.35%) when using individual images was obtained with $\text{Node}_0^{\text{DP}}$ and the VGG16 model.

3.3.2. The Performance Evaluation of Pretrained CNN Models Using RGB Radar Images

In this section, five-fold cross-validation results of various deep learning models frequently used in the literature are presented. In these results, RGB images obtained by combining $\text{Node}_0^{\text{AP}}$, $\text{Node}_0^{\text{DP}}$, $\text{Node}_0^{\text{RP}}$, $\text{Node}_1^{\text{AP}}$, $\text{Node}_1^{\text{DP}}$, $\text{Node}_1^{\text{RP}}$, $\text{Node}_2^{\text{AP}}$, $\text{Node}_2^{\text{DP}}$, and $\text{Node}_2^{\text{RP}}$ images were used. The data in Table 2 was examined in terms of accuracy, F1 score, precision, and sensitivity to evaluate the performance of the models.

Table 2. Individual performance (%) results of deep learning models in the classification of RGB radar images. These results were averaged over five folds.

Model Name	Accuracy (%)	F1 Score (%)	Precision (%)	Recall (%)
ResNet18	89.77	89.58	89.91	89.67
ResNet34	90.14	89.96	90.33	90.03
ResNet50	90.45	90.29	90.67	90.36
ResNet 101	90.40	90.19	90.70	90.28
VGG16	91.25	91.13	91.47	91.21
VGG19	91.35	91.20	91.56	91.29
DenseNet121	90.48	90.29	90.63	90.40
DenseNet169	90.67	90.53	90.90	90.61
InceptionV3	90.81	90.66	90.97	90.72
MobileNet	88.54	88.38	88.85	88.46

As shown in Table 2, the VGG19 model achieved the best results in the five-fold average performance tests. Specifically, the VGG19 model had an accuracy rate of 91.35%, an F1 score of 91.20%, a precision of 91.56%, and a sensitivity of 91.29%. These metrics indicate that the VGG19 model is highly successful at classifying RGB images. In contrast, MobileNet architecture produced the lowest performance results.

3.3.3. Classification Results of Ensemble Learning Models

Ensemble learning is a technique used to achieve better classification results by combining multiple machine learning models. The goal is to leverage the strengths of different models to offset their weaknesses and enhance overall performance. When selecting the most suitable models for combination, it is crucial to consider both their individual success and their structural differences. Based on the results mentioned above, the most suitable

candidates for ensemble learning are typically models with high accuracy and performance. Models like VGG16 and VGG19, which have higher accuracy rates compared to others, are therefore good candidates for ensemble learning. However, other models, such as ResNet50 and DenseNet169, also perform well and can be considered for this technique. In this experimental study, binary combinations of all models listed in Table 3 were tested using ensemble learning.

Table 3. Combining binary models results (%) of ensemble learning models. Bold indicates best result.

Model	Metric	ResNet34	ResNet50	ResNet101	VGG16	VGG19	DenseNet121	DenseNet169	InceptionV3	MobileNet
ResNet18	Accuracy	91.02	91.34	91.30	91.79	92.05	91.39	91.46	91.54	90.82
	F1 score	90.84	91.18	91.11	91.66	91.91	91.22	91.34	91.41	90.65
	Precision	91.17	91.50	91.50	91.94	92.19	91.52	91.66	91.70	90.99
	Recall	90.91	91.26	91.20	91.73	91.99	91.30	91.41	91.47	90.74
ReSNet34	Accuracy		91.45	91.32	91.95	92.09	91.44	91.32	91.61	91.06
	F1 score		91.31	91.14	91.82	91.96	91.27	91.18	91.47	90.90
	Precision		91.64	91.55	92.17	92.26	91.57	91.56	91.78	91.28
	Recall		91.37	91.21	91.88	92.03	91.34	91.25	91.53	90.98
ReSNet50	Accuracy			91.41	92.39	**92.55**	91.70	91.57	91.96	91.50
	F1 score			91.25	92.27	**92.44**	91.56	91.44	91.84	91.36
	Precision			91.62	92.55	**92.73**	91.89	91.79	92.14	91.70
	Recall			91.33	92.32	**92.50**	91.62	91.50	91.88	91.42
ReSNnet101	Accuracy				91.90	92.04	91.59	91.68	91.82	91.32
	F1 score				91.74	91.88	91.40	91.52	91.67	91.12
	Precision				92.10	92.20	91.73	91.89	92.00	91.53
	Recall				91.81	91.98	91.50	91.60	91.74	91.21
Vgg16	Accuracy					91.97	91.97	92.03	92.45	92.02
	F1 score					91.84	91.83	91.90	92.35	91.93
	Precision					92.16	92.13	92.25	92.61	92.22
	Recall					91.92	91.92	91.97	92.39	91.98
Vgg19	Accuracy						92.19	92.16	92.30	92.05
	F1 score						92.05	92.03	92.17	91.91
	Precision						92.33	92.36	92.46	92.21
	Recall						92.12	92.10	92.23	91.97
Densenet121	Accuracy							91.42	91.84	91.17
	F1 score							91.26	91.70	91.02
	Precision							91.60	91.97	91.31
	Recall							91.34	91.76	91.11

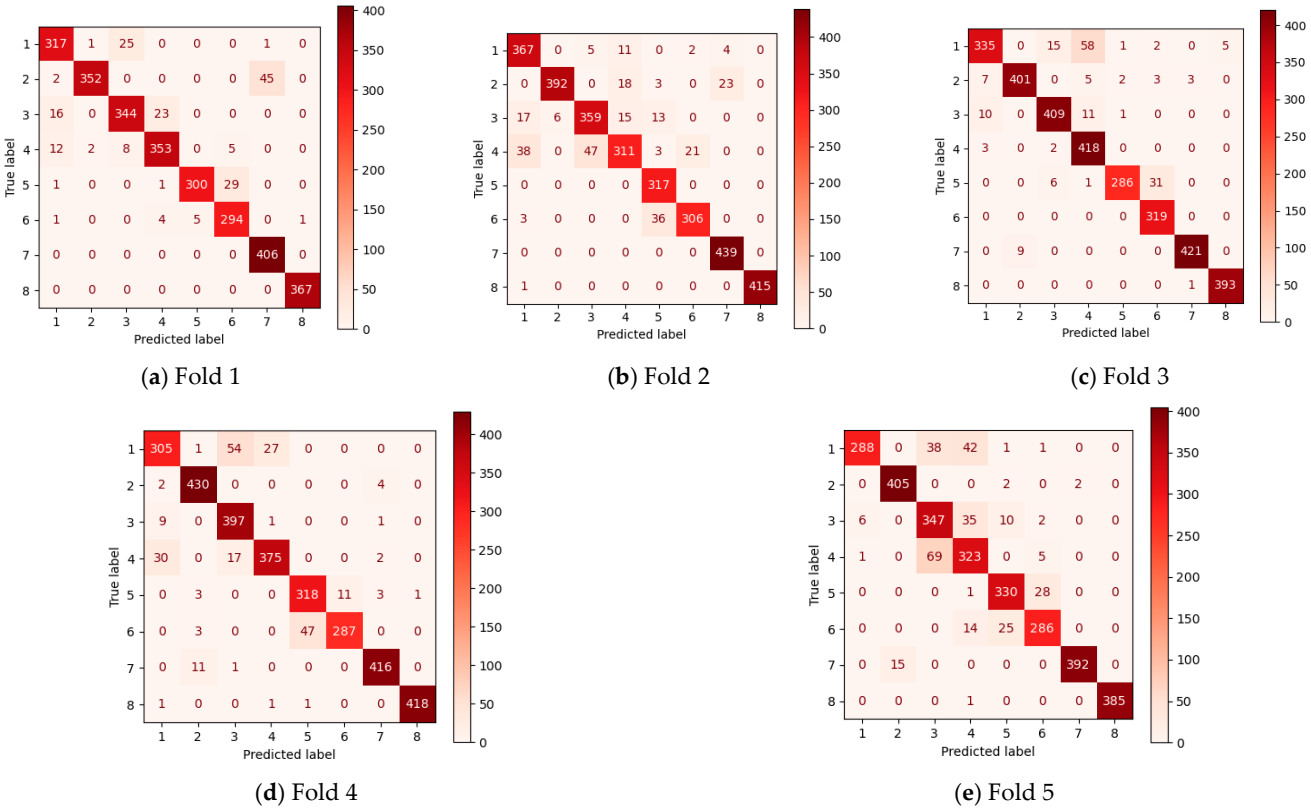
Table 3. *Cont.*

Model	Metric	ResNet34	ResNet50	ResNet101	VGG16	VGG19	DenseNet121	DenseNet169	InceptionV3	MobileNet
Densenet169	Accuracy								91.93	91.07
	F1 score								91.81	90.92
	Precision								92.11	91.25
	Recall								91.87	91.00
InceptionV3	Accuracy									91.44
	F1 score									91.31
	Precision									91.64
	Recall									91.36

In Table 3, we present the performance scores of various binary combinations of models, including ResNet34, ResNet50, ResNet101, VGG16, VGG19, DenseNet121, DenseNet169, InceptionV3, and MobileNet. The highest scores were achieved with the VGG19 and ResNet50 models. By combining these two models, we obtained an accuracy rate of 92.55%, an F1 score of 92.44%, a precision of 92.73%, and a recall of 92.5%. Compared to the individual results of VGG19 and ResNet50 in Table 3, the combination provided an accuracy 1.2 percentage points higher for VGG19 and an accuracy 2.1 percentage points higher for ResNet50. Additionally, all models listed in Table 3 had accuracy scores above 90%. These findings indicate that the ensemble learning model is highly effective for classifying radar images.

Confusion matrices for all five folds are presented in Figure 4. The ensemble model demonstrates considerable robustness for gestures exhibiting distinct kinematic signatures, such as "wave" and "wave_through," achieving consistently high classification performance across all folds. These gestures exhibit periodic Doppler shifts and distinct spatial trajectories, which make them highly separable in the fused RGB representation. In contrast, movements with more subtle or partially overlapping kinematic profiles—i.e., "come closer," "push away," and "slow down"—exhibit varying degrees of confusion. This pattern is particularly evident in folds 1, 4, and 5, where bidirectional misclassifications are repeated among these classes. Closer inspection reveals that errors mostly occur between movements with similar kinematic properties. For example, "approach" and "push away" are performed along the same radial axis but in opposite directions; depending on the participant's speed, amplitude, or posture, the resulting radar reflections may be partially similar to each other. Similarly, the "slow down" movement, characterized by moderate and repetitive arm oscillations, occasionally overlaps with the "come closer" or "push away" transition phases, producing similar Doppler signatures. Another consistent confusion pattern is observed between "stop" and "thank you." Both movements produce limited Doppler activity and have quasi-static properties, leading to similar spectral representations in the combined RGB image. Class imbalance also contributes to some of the observed errors. The dataset distribution is as follows: "come closer," 1906 samples; "fly," 2101 samples; "push away," 2032 samples; "slow down," 2045 samples; "stop," 1667 samples; "thank you," 1631 samples; "wave," 2110 samples; and "wave through," 1984 samples. The classes "stop" and "thank you," which each account for approximately 10.5% of the 15,476 valid samples, are underrepresented compared to the remain-

ing six classes. This relative scarcity partially explains the mutual misclassifications observed between these two gestures. Consequently, the proposed model achieved overall high accuracy and robustness across different gesture profiles, indicating that the remaining classification errors are largely due to inherent similarities and class imbalance between gestures.



(**a**) Fold 1 (**b**) Fold 2 (**c**) Fold 3

(**d**) Fold 4 (**e**) Fold 5

Figure 4. Confusion matrices for the proposed model in each fold. Labels from 1 to 8 given in the confusion matrices represent "come_closer," "fly," "push_away," "slow_down," "stop," "thank_you," "wave," and "wave_through," respectively.

3.3.4. Ablation Study: Contribution of Each Image Type

Tables 1 and 2 compare the performance of individual grayscale images and the combined RGB image. The RGB approach was found to be highly effective in this comparison. Additionally, an additional experiment was conducted to understand the contribution of each type of image (AP, DP, and RP) to the proposed ablation method. In this experiment, the VGG16 model, which demonstrated the highest individual performance, was used, and new RGB images were created by completely removing the AP, DP, or RP images from the ablation process. The results are presented in Table 4.

The ablation study presented in Table 4 demonstrates the contribution of the proposed RGB fusion method to the classification success of each image type. While the original RGB image achieved 91.04% accuracy, when all Doppler images (DP) were removed, accuracy dropped to 82.05%, a dramatic loss of 6.99 points. This result demonstrates that Doppler images are the most discriminative source of information for hand gesture recognition. When RP images were removed, accuracy dropped to 87.90% (−3.14 points), while when AP images were removed, the decrease was only 2.02 points. These findings confirm that the applied rotation and shift operations do not distort the signal content; on the contrary, they preserve the complementary information from each image type to synthesize an RGB representation.

Table 4. Ablation results by image type (VGG16, five-fold average accuracy).

Image Used	Accuracy (%)	Drop Rate According to RGB
Full RGB–AP, DP, and RP	91.04	–
DP and RP only (all APs removed)	89.02	−2.02
AP and RP only (all DPs removed)	82.05	−6.99
AP and DP only (all RPs removed)	87.90	−3.14

4. Discussion

This study presents a two-stage approach for classifying radar images for human hand gesture recognition. In the first stage, different radar images—namely, $\text{Node}_0^{\text{AP}}$, $\text{Node}_0^{\text{DP}}$, $\text{Node}_0^{\text{RP}}$, $\text{Node}_1^{\text{AP}}$, $\text{Node}_1^{\text{DP}}$, $\text{Node}_1^{\text{RP}}$, $\text{Node}_2^{\text{AP}}$, $\text{Node}_2^{\text{DP}}$, and $\text{Node}_2^{\text{RP}}$—were processed using image-processing techniques to generate a new RGB image through rotation, shifting, addition, and merging operations. The main purpose of this fusion approach is to minimize data loss by processing nine different images simultaneously into a single image, in contrast to multi-input models. In the second stage of the proposed model, an ensemble learning-based approach was employed using RGB images. In this approach, several different pre-trained deep models were evaluated, and the highest accuracy scores were achieved with the VGG19 and ResNet50 models. These results are presented in Table 5, along with the results of other models reported in the literature. The comparisons in Table 5 were obtained by reproducing the studies of Kern et al. [14] and Fırat et al. [17] on the same dataset, using the same preprocessing steps and five-fold cross-validation protocol as closely as possible. We utilized the exact five-fold cross-validation indices provided by the dataset creators [14] to ensure that all models were tested on identical data splits.

As observed in the study's introduction and Table 5, limited studies are available in the field of HGR. Current approaches typically employ structures such as CNNs and LSTM. Some studies have also utilized the PointNet model for feature extraction. In the study by Kern et al. [14], a 92.2% accuracy score was achieved using PointNet features with the LSTM technique. On the other hand, the model proposed by Firat et al. [17] reached 90.54% accuracy rate using DenseNet121 and Swin Transformer architectures. This model used a three-input architecture to classify radar images. Although this structure produced effective results, it requires a high processing cost. In contrast, our proposed model offers a more economical and efficient approach by converting nine input images into a single RGB image. It is important to clarify the distinction between the complexity of the fusion strategy and the classifier backbone. While the proposed RGB fusion method relies on computationally inexpensive affine transformations (low algorithmic complexity), the chosen ensemble classifier (VGG19 + ResNet50) is computationally intensive due to its high parameter count. The proposed fusion strategy allows for flexibility; it can be paired with lightweight models for real-time applications or heavy models for maximum accuracy. In this study, we prioritized accuracy. As a result, our model, which processes RGB images based on Doppler Spectrogram (DS), Range Spectrogram (RS), and Azimuth Spectrogram (AS) features, achieved 92.55% accuracy using the ResNet50- and VGG19-based ensemble learning method.

Analyses were conducted to validate the rationale for preferring the normal average approach over more complex fusion strategies such as weighted voting or attention-based mechanisms. Since the VGG19 and ResNet50 architectures exhibited very similar accuracy rates of 91.35% and 90.45%, respectively, and similar error variances, the equal weighting strategy offered the most robust error reduction performance by minimizing the risk of overfitting. Accordingly, a comparative performance analysis was conducted between the "normal average," "weighted average" (with 0.1-step grid search), and "max voting"

methods. The results showed that the weighted average method provided only a negligible increase of 0.08% compared to the normal average, while increasing the computational complexity of the model. Consequently, considering the balance between computational efficiency and generalization ability, the normal average method was adopted as the final fusion strategy.

Table 5. Comparison of the proposed model and models in the literature for the classification of HGR images.

Reference	Features	Classifier	Accuracy (%)
Kern et al. [14] (2022)	DS, RS	CNN	88.8
Kern et al. [14] (2022)	DS, RS, and AS	CNN	89.7
Kern et al. [14] (2022)	PointNet features with R, v, SNR, and i_n	LSTM	90.0
Kern et al. [14] (2022)	PointNet features with R, v, SNR, and i_n without spatial filtering	LSTM	87.0
Kern et al. [14] (2022)	PointNet features with R, v, θ, SNR, and i_n	LSTM	89.5
Kern et al. [14] (2022)	PointNet features with x, y, SNR, and i_n	LSTM	80.7
Kern et al. [14] (2022)	PointNet features with R, v, θ, x^{norm}, y^{norm}, SNR, and i_n	LSTM	92.2
Fırat et al. [17] (2025)	DenseNet121 and Swin transformer	Transformer	90.54
Proposed	RGB image based on DS, RS, and AS	Ensemble learning based on ResNet50 and VGG19	92.55

While the proposed system demonstrates high accuracy in controlled environments, real-world application presents certain challenges. In complex traffic scenarios with numerous pedestrians and vehicles, radar signals can be affected by inter-target interference and jamming, making it difficult to isolate the hand movement of a specific target. Although millimeter-wave radars are generally more robust to lighting conditions compared to cameras, extreme weather conditions such as heavy rain or dense fog can cause signal attenuation, potentially degrading the quality of Doppler signatures. Furthermore, the placement of radar nodes is critical; deviations from the calibrated geometry used in this study can alter the angles of movement, potentially requiring an adaptive calibration mechanism or a more rotation-insensitive coupling strategy.

In light of these results, the advantages and limitations of the proposed model are outlined below:

Advantages:

- The RGB image transformation technique used in the proposed model minimizes data loss by merging nine different radar images into a single image. This preserves various features while reducing the processed data volume.
- By consolidating grayscale radar images from multiple nodes into a single-color radar image, we effectively retain diverse radar image features while simplifying subsequent processes. The ensemble learning approach further enhances classification accuracy by leveraging the strengths of multiple pre-trained CNN models.

- An ensemble learning-based approach was employed by combining VGG19 and ResNet50 deep learning models, producing robust and high scores. Additionally, the community learning approach is more cost-efficient compared to other approaches in the literature.

Weaknesses:

- The RGB fusion process currently uses empirically determined fixed shifts and rotations, making it semi-manual and potentially requiring recalibration if radar node placement or movement protocols change.
- Although the results are promising, strengthening the data calibration used is needed for real-time applications. Furthermore, the model is designed to be adaptable to different radar hardware configurations or new detection features.

While the proposed method achieves state-of-the-art performance (92.55% accuracy), some limitations must be acknowledged. In terms of limitations, while the dataset containing 15,700 samples is substantial, it includes data from only 35 participants. This limits the model's ability to generalize to a broader population with diverse physical characteristics and movement habits. Additionally, the current semi-manual fusion strategy relies on fixed empirical shifts, which may not be optimal for all deployment scenarios. Future work will focus on three main directions to address these issues: (i) collecting a large-scale "in-the-wild" dataset including diverse weather conditions, (ii) developing learnable spatial transformation modules to automate the fusion process, and (iii) implementing lightweight versions of the ensemble model for real-time embedded systems to handle multi-target tracking and gesture recognition simultaneously. Finally, future work will investigate emerging 3D feature extraction and multi-modal depth fusion techniques [30,31] to enhance real-time perception in latency-sensitive, dynamic environments.

5. Conclusions

In this paper, a novel RGB radar image construction method from three different radar sensor DS, AS, and RS images was introduced and its application on traffic HGR was carried out. To handle the efficient traffic HGR, ensemble learning over the pretrained CNN models was carried out by using a grid search mechanism in the space of ResNet34, ResNet50, ResNet101, VGG16, VGG19, DenseNet121, DenseNet169, InceptionV3, and MobileNet models. Extensive experimental studies have revealed that RGB color radar image construction demonstrates promising results for traffic HGR applications. In addition, it was observed that there was a notable improvement in classification performance when using the constructed RGB radar images compared to individual grayscale images. These findings point to the usability of the proposed method in radar-based applications such as traffic movement recognition and demonstrate its potential for adaptability to real-world scenarios. Future research directions may include exploring additional ensemble learning strategies, investigating the transferability of the proposed approach to other radar datasets and applications, and further improving the radar image generation process for optimal feature retention. In particular, fully automatic image fusion strategies will be investigated. Finally, new datasets will be explored, and data collection for real-time scenarios is planned.

Funding: This research received no external funding.

Data Availability Statement: Data available on reasonable request.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Li, X.; He, Y.; Jing, X. A Survey of Deep Learning-Based Human Activity Recognition in Radar. *Remote Sens.* **2019**, *11*, 1068. [CrossRef]
2. Wan, Q.; Li, Y.; Li, C.; Pal, R. Gesture Recognition for Smart Home Applications Using Portable Radar Sensors. In Proceedings of the 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Chicago, IL, USA, 26–30 August 2014; Volume 2014, pp. 6414–6417. [CrossRef]
3. Bulling, A.; Blanke, U.; Schiele, B. A Tutorial on Human Activity Recognition Using Body-Worn Inertial Sensors. *ACM Comput. Surv.* **2014**, *46*, 1–33. [CrossRef]
4. Sharma, R.R.; Kumar, K.A.; Cho, S.H. Novel Time-Distance Parameters Based Hand Gesture Recognition System Using Multi-UWB Radars. *IEEE Sens. Lett.* **2023**, *7*, 6002204. [CrossRef]
5. Amin, M.G.; Zhang, Y.D.; Ahmad, F.; Ho, K.C.D. Radar Signal Processing for Elderly Fall Detection: The Future for in-Home Monitoring. *IEEE Signal Process. Mag.* **2016**, *33*, 71–80. [CrossRef]
6. Santra, A.; Ulaganathan, R.V.; Finke, T. Short-Range Millimetric-Wave Radar System for Occupancy Sensing Application. *IEEE Sens. Lett.* **2018**, *2*, 7000704. [CrossRef]
7. Li, Z.; Fioranelli, F.; Yang, S.; Zhang, L.; Romain, O.; He, Q.; Cui, G.; Le Kernec, J. Multi-Domains Based Human Activity Classification in Radar. In Proceedings of the IET Conference Proceedings, Online Conference, 4–6 November 2020; Volume 2020, pp. 1744–1749. [CrossRef]
8. Sun, S.; Petropulu, A.P.; Poor, H.V. MIMO Radar for Advanced Driver-Assistance Systems and Autonomous Driving: Advantages and Challenges. *IEEE Signal Process. Mag.* **2020**, *37*, 98–117. [CrossRef]
9. Ahmed, S.; Kallu, K.D.; Ahmed, S.; Cho, S.H. Hand Gestures Recognition Using Radar Sensors for Human-Computer-Interaction: A Review. *Remote Sens.* **2021**, *13*, 527. [CrossRef]
10. Wang, C.; Zhao, X.; Li, Z. DCS-CTN: Subtle Gesture Recognition Based on TD-CNN-Transformer via Millimeter-Wave Radar. *IEEE Internet Things J.* **2023**, *10*, 17680–17693. [CrossRef]
11. Jin, B.; Ma, X.; Zhang, Z.; Lian, Z.; Wang, B. Interference-Robust Millimeter-Wave Radar-Based Dynamic Hand Gesture Recognition Using 2D CNN-Transformer Networks. *IEEE Internet Things J.* **2023**, *11*, 2741–2752. [CrossRef]
12. Liu, H.; Liu, Z. A Multimodal Dynamic Hand Gesture Recognition Based on Radar–Vision Fusion. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 8001715. [CrossRef]
13. Mao, Y.; Zhao, L.; Liu, C.; Ling, M. A Low-Complexity Hand Gesture Recognition Framework via Dual MmWave FMCW Radar System. *Sensors* **2023**, *23*, 8551. [CrossRef]
14. Kern, N.; Grebner, T.; Waldschmidt, C. PointNet + LSTM for Target List-Based Gesture Recognition with Incoherent Radar Networks. *IEEE Trans. Aerosp. Electron. Syst.* **2022**, *58*, 5675–5686. [CrossRef]
15. Guo, Z.; Guendel, R.G.; Yarovoy, A.; Fioranelli, F. Point Transformer-Based Human Activity Recognition Using High-Dimensional Radar Point Clouds. In Proceedings of the IEEE Radar Conference, San Antonio, TX, USA, 1–5 May 2023; pp. 1–6. [CrossRef]
16. Gao, H.; Li, C. Automated Violin Bowing Gesture Recognition Using FMCW-Radar and Machine Learning. *IEEE Sens. J.* **2023**, *23*, 9262–9270. [CrossRef]
17. Fırat, H.; Üzen, H.; Atila, O.; Şengür, A. Automated Efficient Traffic Gesture Recognition Using Swin Transformer-Based Multi-Input Deep Network with Radar Images. *Signal Image Video Process.* **2025**, *19*, 35. [CrossRef]
18. Chen, H.; Leu, M.C.; Yin, Z. Real-Time Multi-Modal Human–Robot Collaboration Using Gestures and Speech. *J. Manuf. Sci. Eng.* **2022**, *144*, 101007. [CrossRef]
19. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition; IEEE Computer Society: Los Alamitos, CA, USA, 2016; Volume 2016, pp. 770–778.
20. Gao, L.; Zhang, X.; Yang, T.; Wang, B.; Li, J. The Application of ResNet-34 Model Integrating Transfer Learning in the Recognition and Classification of Overseas Chinese Frescoes. *Electronics* **2023**, *12*, 3677. [CrossRef]
21. Poudel, S.; Kim, Y.J.; Vo, D.M.; Lee, S.W. Colorectal Disease Classification Using Efficiently Scaled Dilation in Convolutional Neural Network. *IEEE Access* **2020**, *8*, 99227–99238. [CrossRef]
22. Üzen, H.; Altın, M.; Balıkçı Çiçek, İ. Bal Arı Hastalıklarının Sınıflandırılması Için ConvMixer, VGG16 ve ResNet101 Tabanlı Topluluk Öğrenme Yaklaşımı. *Fırat Üniversitesi Mühendislik Bilim. Derg.* **2024**, *36*, 133–145. [CrossRef]
23. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015—Conference Track Proceedings, San Diego, CA, USA, 7–9 May 2015.
24. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR, Honolulu, HI, USA, 21–26 July 2017; Volume 2017, pp. 2261–2269. [CrossRef]

25. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; Volume 2016, pp. 2818–2826. [CrossRef]
26. Üzen, H.; Türkoğlu, M.; Ari, A.; Hanbay, D. InceptionV3 Based Enriched Feature Integration Network Architecture for Pixel-Level Surface Defect Detection. *Gazi Üniversitesi Mühendislik-Mimar. Fakültesi Derg.* **2022**, *2*, 721–732. [CrossRef]
27. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520. [CrossRef]
28. Firat, H.; Üzen, H. Detection of Pneumonia Using A Hybrid Approach Consisting of MobileNetV2 and Squeeze- and-Excitation Network. *Turk. J. Nat. Sci.* **2024**, *13*, 54–61. [CrossRef]
29. Traffic Gesture Dataset. Available online: https://www.uni-ulm.de/in/mwt/forschung/online-datenbank/traffic-gesture-dataset/ (accessed on 10 December 2023).
30. Guo, T.; Lu, B.; Wang, F.; Lu, Z. Depth-aware super-resolution via distance-adaptive variational formulation. *J. Electron. Imaging* **2025**, *34*, 053018. [CrossRef]
31. Chen, Z.; Yang, J.; Chen, L.; Li, F.; Feng, Z.; Jia, L.; Li, P. RailVoxelDet: An Lightweight 3D Object Detection Method for Railway Transportation Driven by on-Board LiDAR Data. *IEEE Internet Things J.* **2025**, *12*, 37175–37189. [CrossRef]