

Article

# Multimodal Hand Gesture Classification for the Human–Car Interaction

Andrea D’Eusanio <sup>1</sup>, Alessandro Simoni <sup>2</sup>, Stefano Pini <sup>2</sup>, Guido Borghi <sup>1,\*</sup>,  
Roberto Vezzani <sup>1,2</sup> and Rita Cucchiara <sup>2</sup>

<sup>1</sup> AIRI—Artificial Intelligence Research and Innovation Center, University of Modena and Reggio Emilia, 41125 Modena, Italy; andrea.deusanio@unimore.it (A.D.); roberto.vezzani@unimore.it (R.V.)

<sup>2</sup> DIEF—Department of Engineering “Enzo Ferrari”, University of Modena and Reggio Emilia, 41125 Modena, Italy; alessandro.simoni@unimore.it (A.S.); s.pini@unimore.it (S.P.); rita.cucchiara@unimore.it (R.C.)

\* Correspondence: guido.borghi@unimore.it

Received: 18 July 2020; Accepted: 20 August 2020; Published: 24 August 2020



**Abstract:** The recent spread of low-cost and high-quality RGB-D and infrared sensors has supported the development of Natural User Interfaces (NUIs) in which the interaction is carried without the use of physical devices such as keyboards and mouse. In this paper, we propose a NUI based on dynamic hand gestures, acquired with RGB, depth and infrared sensors. The system is developed for the challenging automotive context, aiming at reducing the driver’s distraction during the driving activity. Specifically, the proposed framework is based on a multimodal combination of Convolutional Neural Networks whose input is represented by depth and infrared images, achieving a good level of light invariance, a key element in vision-based in-car systems. We test our system on a recent multimodal dataset collected in a realistic automotive setting, placing the sensors in an innovative point of view, i.e., in the tunnel console looking upwards. The dataset consists of a great amount of labelled frames containing 12 dynamic gestures performed by multiple subjects, making it suitable for deep learning-based approaches. In addition, we test the system on a different well-known public dataset, created for the interaction between the driver and the car. Experimental results on both datasets reveal the efficacy and the real-time performance of the proposed method.

**Keywords:** hand gesture recognition; natural user interfaces; depth maps; infrared images; computer vision; deep learning; automotive

## 1. Introduction

The recent spread of cheap but high-quality RGB-D sensors, such as the Microsoft Kinect (<https://developer.microsoft.com/en-us/windows/kinect>), the pmdtec devices (<https://pmdtec.com/picofamily/>) and the Intel RealSense family (<https://www.intelrealsense.com>), has supported the development of Natural User Interfaces (NUIs). These interfaces allow us to set up a body-driven interaction, i.e., the human–computer interaction is provided by means of a contactless body capture (for instance, in terms of gestures [1] or voice [2]) in place of standard physical devices, e.g., the keyboard. In the last decade, they have gathered increasing attention of the researchers in the computer vision community [3–5].)

Among different solutions for the acquisition of 3D data, the usage of low-cost active depth, RGB-D or stereo infrared sensors [1,6–9], which are usually coupled with an infrared emitter and easily allow the acquisition of 2.5D data, has overcome other expensive and cumbersome 3D acquisition devices, such as Lidar and 3D scanners.

In this paper, we propose an NUI system based on hand gestures for the automotive context, in order to improve the ease and the speed of the interaction between the driver and the car. Specifically, we propose to use the NUI paradigm, aiming at reducing the driver inattention: since these interfaces are extremely user-friendly and intuitive [4], then they can increase the amount of the time in which the driver is focused on the driving activity, i.e., driver's hands are on, or next to, the steering wheel and the driver's eyes are looking to the road.

It is proved that distraction is one of the most crucial causes in fatal road crashes [10,11] and the presence of new technologies, like smartphones and tablets, has increased the distraction caused by secondary tasks during the driving activity (for instance, reading messages or texting) [12]. The American National Highway Traffic Safety Administration (<https://www.nhtsa.gov>) (NHTSA) defines the driver distraction as "an activity that could divert a person's attention away from the primary task of driving", and literature works [13,14] usually distinguish among three types of driver distraction.

The visual distraction is the first type and it is defined as not looking at the road. This is often caused by an incorrect use of a smartphone or the infotainment system during the driving. Several works [15,16] reveal that driver's inspection patterns on the forward view are strongly influenced by visual (and cognitive) distraction.

Another type of distraction is the manual distraction, which corresponds to the situation when the driver's hands are not on the steering wheel for a prolonged amount of time. This causes a lower reaction time and less capacity to avoid dangers. This type of distraction is often related to the previous one [17].

The last type of distraction is the cognitive distraction which corresponds to a driver whose attention is not directed to the driving activity. It is often caused by external factors, such as heavy cognitive load, bad physical conditions, or fatigue [18].

In this paper, we propose a system that aims to reduce the visual and manual distractions. Indeed, if drivers can interact with the car by performing dynamic gestures, they can be more focused on the driving activity, in terms of gaze direction and hands on the steering wheel [19], improving the road traffic safety.

However, developing a vision-based interaction system for the automotive context is challenging.

First of all, the system is required to be light invariant: the system reliability in presence of severe light changes (due to, for instance, tunnels, bad weather conditions, nighttime driving) and the ability to work without external light sources must be guaranteed. To this end, we investigate the use of a multimodal system based on the acquisition of active (i.e., employing an infrared emitter) infrared and depth sensors. This choice guarantees the acquisition of high-quality 2.5D data which does not depend on external light sources.

Moreover, the system must ensure real-time speed, in order to quickly detect the dynamic gesture and promptly provide a feedback to the user. In this regard, we propose the usage of acquisition devices with a high frame rate (from 10 up to 200 fps) and of fast deep learning-based algorithms that can assure real-time performance.

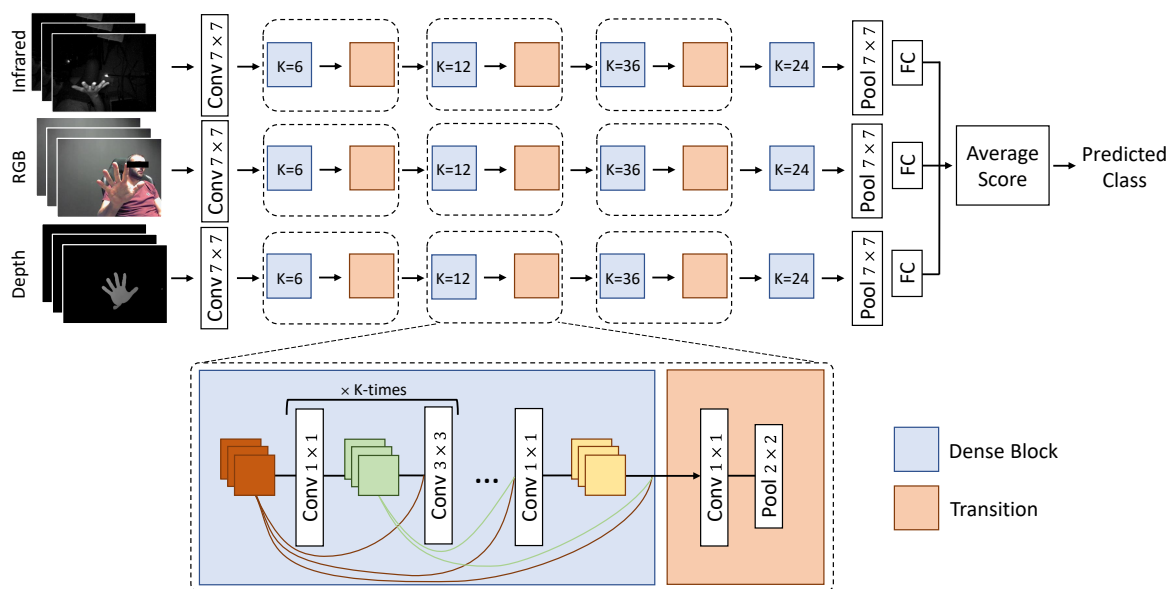
Finally, the system should not hinder the movements and the gaze of the driver for safety reasons. Thus, acquisition devices and embedded boards must have a limited form factor in order to be effectively integrated into the car cockpit. This is easily fulfilled by recent active infrared and depth sensors which are available on the market and have limited dimensions and weight.

Therefore, considering the aforementioned elements, in this paper:

- We propose a deep learning-based framework for the dynamic hand gesture recognition task. In particular, we follow the Natural User Interface paradigm: in this way, a driver could use hand gestures to safely interact with the infotainment system of the car.
- We extend the preliminary work proposed in [20]. Specifically, in this paper we investigate the use of multimodal data with the focus on light-invariant data (i.e., depth and infrared images).

- We propose and analyze the use of a multimodal deep learning-based architectures, as shown in Figure 1. Moreover, we conduct extensive tests about the use of several input data types (single, double and triple) and different training procedures.
- We test the proposed method on two public datasets, namely Briareo [20] and Nvidia Dynamic Hand Gesture [21] (here also referred as NVGestures). Results in terms of accuracy and confusion matrices confirm the high level of accuracy achieved, enabling the implementation of real-world human–car interaction application. We also report the computational performance.

The rest of the paper is organized as follows. In Section 2, we present the related work and datasets about the gesture recognition task focusing on the automotive setting. Then, in Section 3 we present and detail the proposed method. In particular, we detail the model architecture (Section 3.1) and the training procedure (Section 3.2) and present an investigation of multimodal fusion (Section 3.3). The exploited datasets and the experimental results are reported in Section 4, structured in dataset presentation, overall accuracy and comparison with the literature, analysis on the multimodal fusion, and speed performance analysis. Section 5 draws the final conclusions of the work.



**Figure 1.** Overview of the proposed multimodal architecture using a triple input (infrared, RGB and depth streams). Each branch is composed of a modified version of the DenseNet-161 [22] architecture, which combines a sequence of dense and transition blocks. The input is a stream of images from different modalities and the predictions of the three networks are combined with a late-fusion approach to obtain the predicted gesture.

## 2. Related Work

In the first part of this Section, we analyze methods available in the literature, focusing on works that use input data acquired through RGB-D sensors. In the last part, an investigation on publicly released datasets in the literature is conducted.

### 2.1. Methods

In recent years, the gesture classification or recognition task has been discussed frequently in the literature. From a general point of view, this task can be divided into two groups: non-vision based recognition, i.e., methods based on contact-based devices (e.g., gloves [23] and electronic bracelet [24]) and vision-based recognition, i.e., methods that rely on images or videos acquired by cameras.

In this paper, we focus our analysis on the second group which has drawn a continuous research interest in the last decade. In this context, one of the main challenges is to solve this task in a real-world

setting, in which occlusions and varying lighting conditions are usually present. In this way, the use of multimodal methods—which exploit data acquired by different sensors—and deep convolutional neural network has taken a step forward to obtaining great results, even though many methods based on machine learning techniques, such as Hidden Markov Models (HMM) [25] or Support Vector Machines (SVM) [26], have been widely used [8,27].

A 3D Convolutional Neural Network (3D-CNN) [28] is used in [29] to tackle this task. The 3D-CNN is based on the key aspects of standard Convolutional Neural Networks (CNNs) [30], but it is oriented to extract features from temporal sequences. Indeed, the 3D-CNN receives as input a single gesture sequence, then the extracted local spatial–temporal features are processed by a Recurrent Neural Network (RNN) [31] that outputs the final classification. We use this method, the winner of the competition on the VIVA challenge dataset [32], as main competitor. We note that, as reported by the authors, the use of a 3D-CNN and a RNN is demanding in terms of the amount of training data and procedure.

Taking inspiration from the same 3D architecture, Miao et al. [33] introduce a large-scale video-based gesture recognition technique, which considers both RGB and depth data. The extracted spatio-temporal features are then fed into a linear SVM classifier to obtain the final recognition output, however, showing a lower accuracy.

Since dynamic gestures are performed through time, the recognition task can be also approached using a Long-Short Term Memory (LSTM) [34] architecture such as the work presented in [20] in which the authors exploit information extracted by a Leap Motion controller (<https://developer.leapmotion.com>) and create different types of features that represent several hand motion characteristics. In addition, they investigate the use of a 3D-CNN on several data types, i.e., RGB, depth and infrared images. In [9] a Leap Motion controller is used to develop an interaction system based on a LSTM network; the system is oriented to the interaction for a CAD software. A Long Short-Term Memory (LSTM) and Gate Recurrent Unit (GRU) [35] are exploited, and the final classification is performed with a fully-connected layer. We observe that both these methods rely on data provided by the proprietary SDK of the Leap Motion controller.

## 2.2. Dataset for Hand Gesture Classification

Along with vision-based methods detailed in the previous Section, many datasets have been publicly released. In particular, we focus on multimodal datasets [8,20,21,32] in which the subjects are recorded by different acquisition sensors while performing a set of gesture classes created for different tasks. We report a list of these datasets in Table 1, showing, in particular, the number of subjects involved in the acquisition process, the number of gesture classes, if gestures are dynamic or static and the availability of 3D hand joints, and RGB, depth and infrared images.

**Table 1.** A list of datasets commonly used for hand gesture classification task. We report the number (#) of subjects involved and the gestures, taking into consideration also the presence of dynamism and 3D hand joints (3DJoints). Furthermore, we include also the types of the collected data: RGB images, depth maps (from both Structured Light (SL) and Time-of-Flight (ToF) devices) and infrared images.

Dataset	Year	# Subjects	# Gestures	Dynamic	3DJoints	RGB	Depth	Infrared
Unipd [8]	2014	14	10		✓	✓	SL	
VIVA [32]	2014	8	19	✓		✓	SL	✓
Nvidia [21]	2015	20	25	✓		✓	SL	✓
LMDHG [36]	2017	21	13	✓	✓			✓
Turms [19]	2018	7	-	✓				✓
CADGestures [9]	2019	30	8	✓	✓			✓
Briareo [20]	2019	40	12	✓	✓	✓	ToF	✓

Some of these datasets collect samples for hand gesture analysis for a particular language or environment; for example, the Chalearn dataset [37] contains a high number of subjects and samples,

but it is specific for the Italian Sign Language and it is acquired in indoor scenarios. On the automotive setting the Turms dataset [19] reproduces a real automotive context, but it is focused on driver's hand detection and tracking without gestures annotations. Authors from [8] present one of the first attempts to improve the hand gestures recognition task using other types of data instead of RGB images or videos. Therefore, they propose a dataset with depth maps and 3D hand joints information, acquired by the first version of Microsoft Kinect and a Leap Motion device. The dataset contains the recordings of 14 people performing 10 different types of static gestures which are repeated 10 times each. Similarly to Chalearn, it was collected in an indoor environment, with the acquisition devices placed in front of the subjects-and it is specific for the American Sign Language. Therefore, these datasets are not suitable for our experimental evaluation.

The VIVA Hand Gestures [32] dataset is acquired in an automotive setting, during real driving situations, and it has been released for the challenge organized by the Laboratory for Intelligent and Safe Automobiles (LISA). Frequent occlusions and variable external light sources make this dataset challenging. Gestures are acquired through the first version of the Microsoft Kinect device, a Structured Light device able to acquire both RGB and depth data. The 19 classes of gestures are performed by eight subjects. Each gesture is repeated two times, once from the driver's point of view and once from the passenger's one. Unfortunately, this dataset, even though collected during the driving activity, lacks in realism due to the presence of a flat green surface placed on the infotainment area on which gestures are performed.

The Leap Motion Dynamic Hand Gesture (LMDHG) dataset [36] collects unsegmented dynamic gestures, executed by either one or two hands. It consists of 3D coordinates of 23 hand joints, collected through the public SDK of the Leap Motion controller. The 13 types of gestures are performed by 21 people. A total of 50 sequences were released, including the no-gesture class.

In the CADGestures dataset [9] gestures are encoded in an 18-dimensional vector exploiting the 3D joints of fingers, palm and arm, which enable the computation of significant angles and translation values. Using their dominant hand, 30 performed a gesture from a set of 8 classes twice, obtaining a total of 480 gestures.

The largest dataset in terms of the number of gestures is the Nvidia Dynamic Hand Gesture dataset [21]; composed of recordings from multiple sensors, i.e., the SoftKinetic DS325 and the DUO 3D stereo camera, placed in frontal and top positions with the respect of the driver, respectively. This dataset contains 25 classes of gestures performed by 20 different subjects with the right hand while the left one handles the steering wheel. Each gesture is repeated three times and acquired in 5 s video samples. Several types of data are available, such as RGB, optical flow, infrared and depth images. We select this challenging dataset for our experimental evaluation, in addition to the Briareo dataset [20].

The Briareo dataset consists in data collected in a car cockpit, placing the acquisition devices in the tunnel console between the driver and the passenger, looking upwards toward the car ceiling. Due to this position, sensors can be easily integrated and gestures can be performed near the steering wheel.

These two datasets are further detailed in Section 4.1.

### 3. Proposed Method

In this Section, we present and detail the proposed method. Specifically, we describe the architecture of the model and its training procedure. Then, we introduce the multimodal fusion technique and the corresponding multimodal architecture.

An overview of the proposed multimodal method is shown in Figure 1.

#### 3.1. Model

In this Section, we describe the proposed deep learning approach for the dynamic hand gesture recognition task. Our model is based on DenseNet [22], a deep convolutional network which adds connections between each layer block and all following blocks.

In particular, we modify the DenseNet-161 architecture in order to take as input sequence clips of 40 frames, which are sufficient to contain the most relevant movements of a gesture and to predict a probability distribution over the  $C$  gestures of the specific dataset. To this end, we replace the first convolutional layer with another one that matches the 40-frame input tensor. In addition, we modify the last fully-connected layer (followed by a softmax layer) to predict a distribution over  $C$  output classes. With this straightforward modification, we can predict an entire gesture with an efficient and effective network which does not require heavy recurrent modules or 3D convolutions while obtaining remarkable results, as shown in Section 4.2.

Considering the input data type, we adapt the number of input channels accordingly to the channel of the single frames. That is, we used  $40 \times 3$  input channels for the RGB frames and  $40 \times 1$  for the infrared and the depth frames. The input resolution is unchanged and thus set to  $224 \times 224$  pixels.

A detailed definition of the proposed model is reported in Table 2. The first and second columns contain the name of the block and its definition in terms of layers and repetitions; the third and fourth columns contain the input size and the output size for each block.

A visual representation of this architecture is shown in Figure 1 (in the multimodal configuration presented in Section 3.3).

**Table 2.** Architectural details of the proposed model. It is derived from DenseNet-161 [22] to take sequence clips of 40 frames as input and to predict a distribution over  $C$  gestures.

Block	Definition	Input Size	Output Size
Convolution	$7 \times 7$ conv, stride 2	rgb: $120 \times 224 \times 224$ infrared: $40 \times 224 \times 224$ depth: $40 \times 224 \times 224$	$96 \times 112 \times 112$
Pooling	$3 \times 3$ max pool, stride 2	$96 \times 112 \times 112$	$96 \times 56 \times 56$
Dense Block (1)	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$96 \times 56 \times 56$	$384 \times 56 \times 56$
Transition Block (1)	$1 \times 1$ conv $2 \times 2$ avg pool, stride 2	$384 \times 56 \times 56$	$192 \times 28 \times 28$
Dense Block (2)	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$192 \times 28 \times 28$	$768 \times 28 \times 28$
Transition Block (2)	$1 \times 1$ conv $2 \times 2$ avg pool, stride 2	$768 \times 28 \times 28$	$384 \times 14 \times 14$
Dense Block (3)	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 36$	$384 \times 14 \times 14$	$2112 \times 14 \times 14$
Transition Block (3)	$1 \times 1$ conv $2 \times 2$ avg pool, stride 2	$2112 \times 14 \times 14$	$1056 \times 7 \times 7$
Dense Block (4)	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$	$1056 \times 7 \times 7$	$2208 \times 7 \times 7$
Pooling	$7 \times 7$ global avg pool	$2208 \times 7 \times 7$	$2208 \times 1 \times 1$
Classification	$C$ -class fully connected softmax	2208	$C$

### 3.2. Training

In the experiments, the network is not trained from scratch. On the contrary, it is initialized with the weights pre-trained on the ImageNet dataset [38] provided by the PyTorch library [39].

These initial weights are then fine-tuned on the training set of the selected dataset (presented in Section 4.1) through Stochastic Gradient Descent (SGD) [40,41] with momentum [42] set to 0.5 for Briareo, and through Adam [43] with weight decay  $1 \times 10^{-4}$  for NVGestures. The model is trained for

50 epochs. The network is optimized by minimizing the Categorical Cross-Entropy (CCE) [44] loss, defined as

$$L_{CCE} = - \sum_i^N \sum_c^C (y_{i,c} \log(p_{i,c})), \quad (1)$$

where  $N$  is the number of samples in the mini-batch,  $C$  is the number of classes,  $y_{i,c}$  is 1 if the class  $c$  is the ground truth class of the sample  $i$  and 0 otherwise. Finally,  $p_{i,c}$  is the predicted class distribution over the  $C$  classes for the  $i$ -th sample. We set a learning rate of  $1 \times 10^{-2}$  for Briareo,  $1 \times 10^{-4}$  for Adam, and a batch size of 8 clips.

Input data is augmented at run-time through random flip and random crop (to the input resolution) with a probability of 0.5. Since there is variability in the execution time between different gestures and between the execution of the same gesture computed by different subjects, we extract 40 contiguous frames from the center of each recording session for training and testing the network.

We point out that this is not a limitation, since the same cut can be done in a real-world application by detecting the hand in the field of view of the camera. A trivial, but effective way to detect it is a simple threshold on the average value of the depth map: if a hand is over the cameras, the average depth distance is different and this decrease/increase can be used as beginning and ending of the sequence.

Input frames are normalized to have, on average, 0 mean and unit variance by subtracting the average pixel value of the training set and dividing by its standard deviation.

### 3.3. Multimodal Fusion

In literature, many works show that deep learning architectures can benefit from multimodal fusion (e.g., [27,45–47] among others). However, a general and effective fusion strategy is still not found, while it has been shown that the best fusion scheme is often task and dataset dependent [48].

Among different possible configurations, we have selected the late fusion strategy, also called decision-level fusion, to combine the different data types available in the selected datasets. In details, we train a different unimodal network for each available data type, as described in the previous Sections. Then, during testing, the unimodal networks are fed with synchronized data from different modalities and their predictions are combined with an average layer to obtain the final estimation.

We show the proposed multimodal architecture in Figure 1. It is composed of multiple branches where each branch corresponds to the unimodal model presented in Section 3.1. It takes a clip of 40 frames in a different modality and predicts a probability distribution over  $C$  classes. Then, the single predictions are averaged to obtain the final predicted gesture.

In the following section, we show that this setting is the most suitable for the gesture recognition task with the proposed deep architecture. Moreover, we compare different combinations of input data and different training strategies and analyze the computational cost of the proposed method.

## 4. Experimental Evaluation

In this Section, we firstly described the datasets on which the proposed model is trained. Then, we evaluate through experiments the unimodal and multimodal architectures for the proposed system for in-car dynamic gesture recognition.

In detail, we present the datasets Briareo [20] and NVGestures [21] and analyze the performance of the single-modality networks and their combination with the late-fusion approach presented in Section 3.3. Then, we report the results obtained on the two datasets and compare them to other literature approaches. Finally, we report a comparison of different fusion and training schemes for the Briareo dataset and an analysis of the computational requirements and performance of the proposed architecture.

#### 4.1. Datasets

##### 4.1.1. Briareo Dataset

This dataset has been presented in [20] and it has been acquired placing the acquisition devices in an innovative point of view, i.e., in the central tunnel between the driver and the passenger seats. This particular position has been selected, aiming to: (i) reduce the visual occlusions that can be generated by the driver or passengers movements; (ii) facilitate the integration of the acquisition devices in the car cockpit; (iii) protect the infrared sensors from the direct sunlight that could critically compromise the operation of this kind of sensors.

From a technical point of view, the dataset has been acquired through three different cameras:

- Pico Flexx (<https://pmdtec.com/picofamily/flexx>): this is a Time-of-Flight (ToF) depth sensor. As reported in [49], ToF devices assure a better quality with the respect of Structured Light (SL) depth sensors, for instance, reducing the presence of visual artifacts (visually represented as black pixels or missing values). It has a spatial resolution of  $224 \times 171$  pixels, acquiring 16-bit depth images. This sensor is suitable for the automotive context due to its very limited form factor (only  $68 \times 17 \times 7.35$  mm) and weight (8 g), making it easy to be integrated in a car cockpit. Moreover, the acquisition range allows to perform gesture next to the device, a crucial element in an indoor environment like a car. In particular, there are two possible depth resolutions and two possible ranges: 0.5–4 m and 0.1–1 m. During the acquisition, the second one is used. The frame rate for the acquisition is set to 45 frame per seconds.
- Leap Motion (<https://www.leapmotion.com>): an infrared stereo camera specifically designed for the human–computer interaction. It is suitable for the automotive context due to the high frame rate (up to 200 frame per seconds) and limited size ( $70 \times 12 \times 3$  mm) and weight (32 g). In addition, the presence of two cameras with a good spatial resolution ( $640 \times 240$ ) is remarkable. A fish-eye lens guarantees a proper acquisition range for in-car applications. These sensors are equipped with a proprietary SDK able to detect the 3D location of hand joints, together with the bone lengths and their orientations, with real-time performance.

Sample frames of the dataset captured with the two acquisition devices are shown in Figure 2. The frames acquired with the RGB camera and with the Pico Flexx device are reported in the first three rows, while the acquisition of the right camera of the Leap Motion sensor is reported in the last one.

The Briareo dataset contains the following 12 dynamic gestures, visually represented in Figure 3:

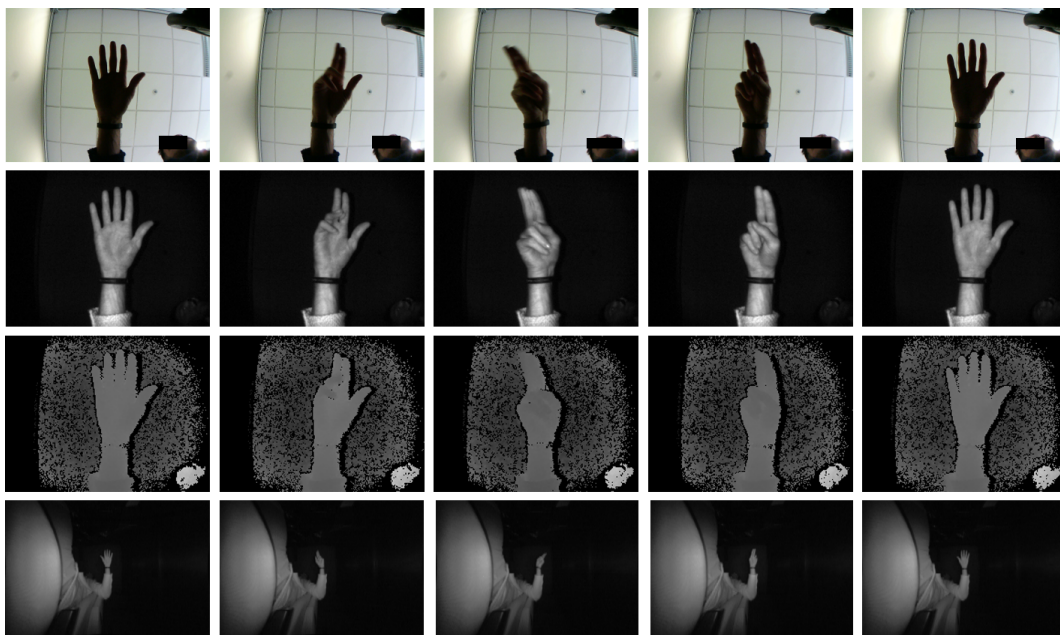
- Fist
- Pinch
- Flip
- Phone
- Right swipe
- Left swipe
- Top-down swipe
- Bottom-up swipe
- Thumb up
- Point
- Clockwise rotation
- Counterclockwise rotation

All the gestures were oriented to the interaction between the driver and a hypothetical infotainment system. Some of them are directly related to common actions that can be performed during the driving activity, such as making a phone call (“phone”) or skipping a song (“right/left swipe”).

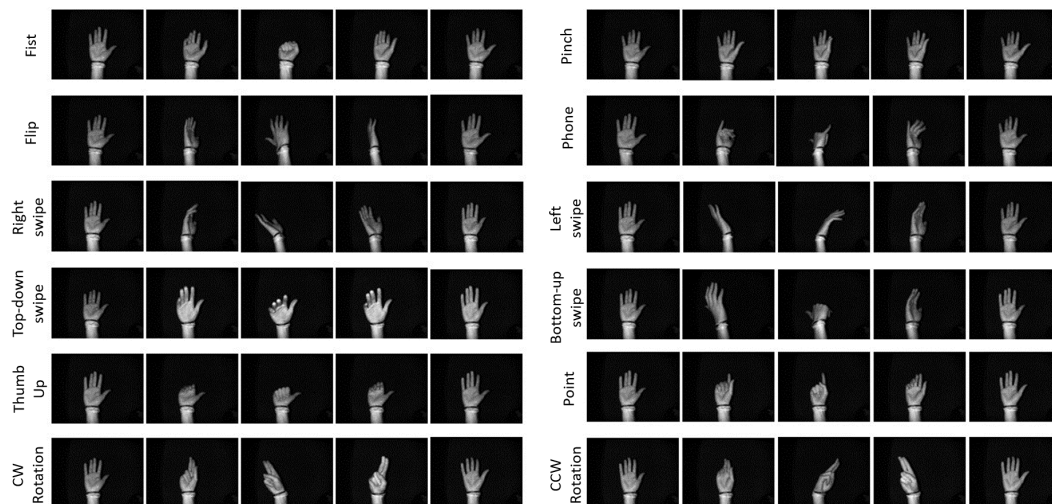
Gestures were performed by 40 subjects (33 males and 7 females) and each subject repeats the gesture 3 times, for a total of 120 collected sequences. Each sequence lasted for at least 40 frames.



An additional sequence was then added containing all gestures performed sequentially without interruptions. All the acquisition devices were synchronized.



**Figure 2.** Sample of multimodal data included in the Briareo dataset [20]. The first row contains RGB frames, the second contains Infrared (IR) frames, while the third contains depth maps. The last row reports the rectified frame of the infrared stereo camera (right view). As shown, RGB frames suffer the lack of an additional light source, while infrared and depth data clearly collect the driver’s hand. Frames are sampled from the gesture “clockwise rotation”.



**Figure 3.** Dynamic gesture classes contained in the Briareo dataset. All gestures are designed for the interaction between an user, i.e., the driver, and a traditional infotainment system, in which it is possible for instance skipping song (“right/left swipes”) or make a phone call (“phone”). Here, frames are taken from the infrared domain. Image taken from [20].

#### 4.1.2. Nvidia Dynamic Hand Gesture dataset

As mentioned above, we tested our system also on the Nvidia Dynamic Hand Gesture dataset [21], referred as NVGestures in the following. The setting was an indoor car simulator with the recording devices placed frontally and top-mounted with respect to the driver position. Specifically, we collected the experimental results on all 25 dynamic gestures contained in this dataset, such as showing “thumb

up” or “OK”, moving either the hand or two fingers up, down, left or right, showing the index finger, or two or three fingers, clicking with the index finger, beckoning, opening or shaking the hand, pushing the hand up, down, out or in, rotating two fingers clockwise or counter-clockwise, pushing two fingers forward, closing the hand twice. For further details about this dataset, please see Section 2 or refer to the original paper [21].

#### 4.2. Experimental Results

In Table 3 we analyzed the different contributions of the input data types to the overall accuracy on the Briareo dataset. In particular, we tested the system with a single input, represented by the individual use of RGB, infrared or depth images, with the combination of two inputs and using all the available modalities. From the single input case, it is possible to note that the network trained on RGB data achieved the worst results. Indeed, as shown in Figure 2, intensity frames suffered the lack of external light sources: the hand appears dark and several details about hand pose and fingers were not visible. Furthermore, when it was used in combination with other inputs, the RGB data negatively affected the combined accuracy. Nevertheless, RGB-based model still achieved a good level of accuracy. Higher results were obtained when using infrared or depth data. In fact, both of them relied on external light sources which resulted in clear IR and depth frames, overcoming the brightness issue of the RGB data. Moreover, depth maps, encoding the 2.5D content of the hand pose, were more discriminative with respect to other data types and provide the highest accuracy in the single-input setting. Finally, the combined usage of infrared and depth data led to the highest accuracy. That is, depth maps and infrared images represented the best combination in terms of overall accuracy: these experimental results showed the effectiveness of the proposed multimodal method and its applicability in real-world applications requiring light-invariance gesture recognition.

**Table 3.** Accuracy of the proposed multimodal system. Specifically, we report results individually exploiting one type of input (“single input”) and all the possible combinations of multiple modalities. As reported, the use of depth maps and infrared images represents the best choice in terms of gesture recognition accuracy.

	Input Data						
	Single Input			Double Input		Triple Input	
RGB	✓			✓		✓	✓
Infrared		✓		✓	✓		✓
Depth			✓		✓	✓	✓
Accuracy	0.833	0.861	0.903	0.864	<b>0.920</b>	0.895	0.909

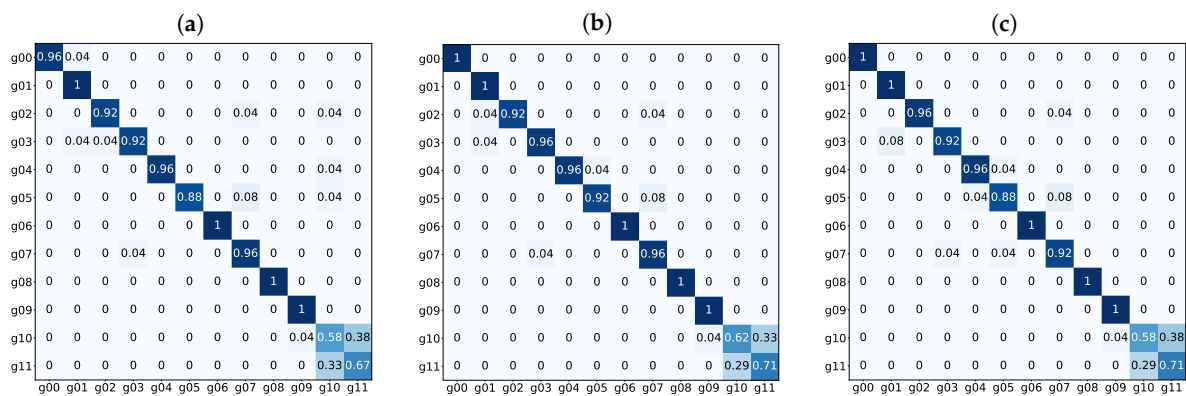
The bold of the number is to highlight the highest result. Same applies to the subsequent tables.

In Table 4, we report the results collected on the Briareo dataset. The unimodal approach presented in [20], based on the well-known C3D network [28] is compared with the best combination obtained from the previous experiment, i.e., the late fusion of infrared- and depth-based models. For a better comparison, we report the overall accuracy on the whole test set, as well as the accuracy for each gesture of the dataset. Moreover, Figure 4 depicts the confusion matrices calculated on the results of our method. They represent the results of the unimodal depth network (left), the multimodal depth+infrared network (center) and the multimodal depth+infrared+RGB network (right).

The confusion matrices reveal that the proposed method had, in general, a high level of accuracy. They also show that the “clockwise” and “counterclockwise rotations” were the two most challenging gestures, probably due to their similarity from a spatial point of view (hand pose) and their temporal symmetry.

**Table 4.** Comparison with competitors on the Briareo dataset [20]. Both overall accuracy and accuracy per gesture are reported.

Gesture	C3D [20]			Ours
	RGB	Depth	Infrared	Depth + Infrared
Fist	0.542	0.708	0.750	<b>1.000</b>
Pinch	0.833	0.875	0.958	<b>1.000</b>
Flip-over	0.792	0.750	0.875	<b>0.917</b>
Telephone call	0.625	0.792	<b>1.000</b>	0.958
Right swipe	0.833	0.833	0.917	<b>0.958</b>
Left swipe	0.833	<b>0.917</b>	0.792	<b>0.917</b>
Top-down swipe	0.917	0.750	0.958	<b>1.000</b>
Bottom-up swipe	0.750	0.833	0.875	<b>0.958</b>
Thumb up	0.917	0.625	<b>1.000</b>	<b>1.000</b>
Point	0.667	0.708	<b>1.000</b>	<b>1.000</b>
CW Rotation	0.542	0.375	<b>0.750</b>	0.625
CCW Rotation	0.417	<b>0.958</b>	0.635	0.708
<b>Overall Accuracy</b>	<b>0.722</b>	<b>0.760</b>	<b>0.875</b>	<b>0.920</b>

**Figure 4.** Confusion matrices of the proposed method. From the left, we report the performance of the systems using respectively depth; depth and infrared; depth, infrared and RGB data as input. (a) Depth; (b) Depth + Infrared; (c) Depth + Infrared + RGB.

In Table 5 we report results obtained on the Nvidia Dynamic Gesture dataset. In this case, we compared with the results obtained by the 2D Convolutional Neural Networks (2D CNN), which were the most similar to our setting. Unfortunately, authors did not report multimodal results for this setting. Therefore, we compared using the proposed unimodal networks only. For the sake of comparison, we also report results obtained by the use of a 2D Recurrent Neural Network (2D RNN) and a 2D RNN, which makes use of a CTC cost function, detailed in [21] (2D RNN + CTC).

**Table 5.** Comparison with competitors on the NVGestures dataset [21]. The overall accuracy for the RGB and the depth data type is reported.

Model	Input Type	
	RGB	Depth
2D CNN [21]	0.556	0.681
2D RNN [21]	0.579	0.647
2D RNN + CTC [21]	<b>0.656</b>	0.691
<b>Ours</b>	0.520	<b>0.761</b>

#### 4.2.1. Multimodal Fusion Analysis

Then, we investigated the role of the mid- and late-fusion strategies in the network architecture implementation for different types of data. The mid-fusion was implemented by fusing the unimodal networks at feature level, after the last convolutional layer. Results are reported in Table 6: when using infrared and depth data, the late-fusion represented the best choice and guaranteed an absolute improvement of about 8% with respect to the mid-fusion.

Moreover, we analyzed different training procedures for the mid-level fusion strategy in Table 7. The “end-to-end” training refers to a training from scratch of the multimodal system, i.e., the unimodal networks are joint together and simultaneously trained from scratch. In the “fine-tuning” experiment we exploited the pre-trained models and then we performed a fine-tuning of the whole combined network. Finally, the “frozen” version consists of an individual train of each network, followed by their union and a training of the joint fully connected layers, not updating the weights of the convolutional layers of the single networks. As it can be seen, the “frozen” approach yielded to the higher accuracy. However, it was yet lower that the accuracy obtained with a late fusion, confirming that a late fusion was the best strategy for this combination of task and dataset.

**Table 6.** Comparison of different fusion strategies for different types of data.

	Input Data			
	Double Input		Triple Input	
RGB	✓		✓	✓
Infrared	✓	✓		✓
Depth		✓	✓	✓
Mid-Fusion	0.882	0.837	0.885	0.878
Late-Fusion	0.864	<b>0.920</b>	<b>0.895</b>	<b>0.909</b>

**Table 7.** Comparison of different training procedures for the mid-level fusion strategy combining RGB, infrared and depth data.

	Training Procedures		
	End-to-End	Fine-Tuning	Freezed
Mid-Fusion	0.722	0.774	<b>0.878</b>

#### 4.2.2. Computational Performance

Thanks to its simplicity, as explained in Section 3.1, the system was able to run in real-time at about 36 frames per second on a system with an Intel Core i7-7700K and a Nvidia GeForce GTX 1080 Ti. The model had 28 M parameters and required just about 1 GB of GPU memory in the unimodal setting.

Regarding the multimodal combination with the late fusion strategy, the architecture required a different network for every data type, but they can run in parallel on the same GPU maintaining real-time speed. In the best setting (see Table 3), the system ran in real-time at about 27 fps on the same machine on which the unimodal test was carried out. The model had 56 M parameters and required about 2.7 GB of GPU memory. Even if the worst-case scenario was considered, i.e., using RGB, Infrared and depth data on a device with limited memory (requiring running the models sequentially), the proposed approach was able to run at about 10 fps which was enough to give a real-time feedback to the user.

## 5. Conclusions

In this paper, we propose a multimodal hand gesture recognition system for the automotive context. Following the Natural User Interface paradigm, we focus on dynamic gestures that can help in reducing the driver’s manual and visual distractions during the driving activity. We investigate the use

of different input types, i.e., RGB, infrared and depth data, and different multimodal fusion strategies to combine data from multiple sources. Through an extensive experimental validation on two publicly released datasets, we show that the infrared and depth data represent the best combination in terms of hand gesture recognition accuracy in an automotive setting. Finally, an analysis of the computational performance confirms the real-time speed of the proposed framework and thus its feasibility for real-world in-car applications.

**Author Contributions:** Conceptualization, A.D., A.S., S.P., G.B.; methodology, A.D., S.P., G.B.; software, A.D.; validation, A.D., A.S., S.P., G.B.; formal analysis, S.P., G.B., R.V.; investigation, A.D., A.S., S.P., G.B.; resources, R.V., R.C.; data curation, A.D., S.P., G.B.; writing—original draft preparation, A.D., A.S., S.P., G.B.; writing—review and editing, A.D., A.S., S.P., G.B., R.V.; visualization, A.D.; supervision, G.B., R.V., R.C.; project administration, R.V., R.C.; funding acquisition, R.V., R.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work has been partially supported by the project “PREVUE (Predicting activities and Events by Vision in an Urban Environment)” funded by the MIUR Progetti di Ricerca di Rilevante Interesse Nazionale (PRIN) 2017.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

NUI	Natural User Interface
SL	Structured Light
ToF	Time-of-Flight
CNN	Convolutional Neural Network
SVM	Support Vector Machine
LSTM	Long Short-Term Memory
GRU	Gate Recurrent Unit
SGD	Stochastic Gradient Descent

## References

1. Borghi, G.; Vezzani, R.; Cucchiara, R. Fast gesture recognition with multiple stream discrete HMMs on 3D skeletons. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016; pp. 997–1002.
2. Vidakis, N.; Syntychakis, M.; Triantafyllidis, G.; Akoumianakis, D. Multimodal natural user interaction for multiple applications: The gesture—Voice example. In Proceedings of the 2012 International Conference on Telecommunications and Multimedia (TEMU), Chania, Greece, 30 July–1 August 2012; pp. 208–213.
3. Saba, E.N.; Larson, E.C.; Patel, S.N. Dante vision: In-air and touch gesture sensing for natural surface interaction with combined depth and thermal cameras. In Proceedings of the 2012 IEEE International Conference on Emerging Signal Processing Applications, Las Vegas, NV, USA, 12–14 January 2012; pp. 167–170.
4. Liu, W. Natural user interface-next mainstream product user interface. In Proceedings of the 2010 IEEE 11th International Conference on Computer-Aided Industrial Design & Conceptual Design 1, Yiwu, China, 17–19 November 2010; Volume 1, pp. 203–205.
5. Rodríguez, N.D.; Wikström, R.; Lilius, J.; Cuéllar, M.P.; Flores, M.D.C. Understanding movement and interaction: an ontology for Kinect-based 3D depth sensors. In *Ubiquitous Computing and Ambient Intelligence. Context-Awareness and Context-Driven Interaction*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 254–261.
6. Boulabiar, M.I.; Burger, T.; Poirier, F.; Coppin, G. A low-cost natural user interaction based on a camera hand-gestures recognizer. In Proceedings of the International Conference on Human-Computer Interaction, Orlando, FL, USA, 9–14 July 2011; Springer: Berlin/Heidelberg, Germany, 2011; pp. 214–221.
7. Villaroman, N.; Rowe, D.; Swan, B. Teaching natural user interaction using OpenNI and the Microsoft Kinect sensor. In Proceedings of the 2011 Conference on Information Technology Education, New York, NY, USA, 20–22 October 2011; pp. 227–232.

8. Marin, G.; Dominio, F.; Zanuttigh, P. Hand gesture recognition with leap motion and kinect devices. In Proceedings of the 2014 IEEE International Conference on Image Processing (ICIP), Paris, France, 27–30 October 2014; pp. 1565–1569.
9. Mazzini, L.; Franco, A.; Maltoni, D. Gesture Recognition by Leap Motion Controller and LSTM Networks for CAD-oriented Interfaces. In Proceedings of the International Conference on Image Analysis and Processing, Trento, Italy, 9–13 September 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 185–195.
10. Wilson, F.A.; Stimpson, J.P. Trends in fatalities from distracted driving in the United States, 1999 to 2008. *Am. J. Public Health* **2010**, *100*, 2213–2219. [[CrossRef](#)]
11. Dong, Y.; Hu, Z.; Uchimura, K.; Murayama, N. Driver inattention monitoring system for intelligent vehicles: A review. *IEEE Trans. Intell. Transp. Syst.* **2011**, *12*, 596–614. [[CrossRef](#)]
12. McKnight, A.J.; McKnight, A.S. The effect of cellular phone use upon driver attention. *Accid. Anal. Prev.* **1993**, *25*, 259–265. [[CrossRef](#)]
13. Ranney, T.A.; Garrott, W.R.; Goodman, M.J. *NHTSA Driver Distraction Research: Past, Present, and Future*; SAE Technical Paper; SAE: Warrendale, PA, USA, 2001.
14. Borghi, G.; Gasparini, R.; Vezzani, R.; Cucchiara, R. Embedded recurrent network for head pose estimation in car. In Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 11–14 June 2017.
15. Harbluk, J.L.; Noy, Y.I.; Trbovich, P.L.; Eizenman, M. An on-road assessment of cognitive distraction: Impacts on drivers' visual behavior and braking performance. *Accid. Anal. Prev.* **2007**, *39*, 372–379. [[CrossRef](#)] [[PubMed](#)]
16. Recarte, M.A.; Nunes, L.M. Mental workload while driving: effects on visual search, discrimination, and decision making. *J. Exp. Psychol. Appl.* **2003**, *9*, 119. [[CrossRef](#)] [[PubMed](#)]
17. Young, K.L.; Salmon, P.M. Examining the relationship between driver distraction and driving errors: A discussion of theory, studies and methods. *Saf. Sci.* **2012**, *50*, 165–174. [[CrossRef](#)]
18. Sharwood, L.N.; Elkington, J.; Stevenson, M.; Wong, K.K. Investigating the role of fatigue, sleep and sleep disorders in commercial vehicle crashes: a systematic review. *J. Australas. Coll. Road Saf.* **2011**, *22*, 24.
19. Borghi, G.; Frigieri, E.; Vezzani, R.; Cucchiara, R. Hands on the wheel: a dataset for driver hand detection and tracking. In Proceedings of the 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), Xi'an, China, 15–19 May 2018.
20. Manganaro, F.; Pini, S.; Borghi, G.; Vezzani, R.; Cucchiara, R. Hand Gestures for the Human-Car Interaction: the Briareo dataset. In Proceedings of the International Conference on Image Analysis and Processing, Trento, Italy, 9–13 September 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 560–571.
21. Molchanov, P.; Yang, X.; Gupta, S.; Kim, K.; Tyree, S.; Kautz, J. Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4207–4215.
22. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
23. Weissmann, J.; Salomon, R. Gesture recognition for virtual reality applications using data gloves and neural networks. In Proceedings of the IJCNN'99, International Joint Conference on Neural Networks, Proceedings (Cat. No. 99CH36339), Washington, DC, USA, 10–16 July 1999; Volume 3, pp. 2043–2046.
24. Shull, P.B.; Jiang, S.; Zhu, Y.; Zhu, X. Hand gesture recognition and finger angle estimation via wrist-worn modified barometric pressure sensing. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2019**, *27*, 724–732. [[CrossRef](#)] [[PubMed](#)]
25. Rabiner, L.R. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* **1989**, *77*, 257–286. [[CrossRef](#)]
26. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [[CrossRef](#)]
27. Wu, D.; Pigou, L.; Kindermans, P.J.; Le, N.D.H.; Shao, L.; Dambre, J.; Odobez, J.M. Deep dynamic neural networks for multimodal gesture segmentation and recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 1583–1597. [[CrossRef](#)] [[PubMed](#)]
28. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning spatiotemporal features with 3d convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 13–16 December 2015; pp. 4489–4497.

29. Molchanov, P.; Gupta, S.; Kim, K.; Kautz, J. Hand gesture recognition with 3D convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Boston, MA, USA, 7–12 June 2015; pp. 1–7.
30. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
31. Graves, A.; Schmidhuber, J. Offline handwriting recognition with multidimensional recurrent neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 7–10 December 2009; pp. 545–552.
32. Ohn-Bar, E.; Trivedi, M.M. Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 2368–2377. [[CrossRef](#)]
33. Miao, Q.; Li, Y.; Ouyang, W.; Ma, Z.; Xu, X.; Shi, W.; Cao, X. Multimodal gesture recognition based on the resc3d network. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Venice, Italy, 22–29 October 2017; pp. 3047–3055.
34. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)] [[PubMed](#)]
35. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078.
36. Boulahia, S.Y.; Anquetil, E.; Multon, F.; Kulpa, R. Dynamic hand gesture recognition based on 3D pattern assembled trajectories. In Proceedings of the 2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA), Montreal, QC, Canada, 28 November–1 December 2017.
37. Escalera, S.; Baró, X.; Gonzalez, J.; Bautista, M.A.; Madadi, M.; Reyes, M.; Ponce-López, V.; Escalante, H.J.; Shotton, J.; Guyon, I. Chalearn looking at people challenge 2014: Dataset and results. In Proceedings of the Workshop at the ECCV, Zurich, Switzerland, 6–12 September 2014; Springer: Berlin/Heidelberg, Germany, 2014; pp. 459–473.
38. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
39. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 8026–8037.
40. Robbins, H.; Monro, S. A stochastic approximation method. *Ann. Math. Stat.* **1951**, 400–407. [[CrossRef](#)]
41. Kiefer, J.; Wolfowitz, J. Stochastic estimation of the maximum of a regression function. *Ann. Math. Stat.* **1952**, *23*, 462–466. [[CrossRef](#)]
42. Sutskever, I.; Martens, J.; Dahl, G.; Hinton, G. On the importance of initialization and momentum in deep learning. In Proceedings of the International Conference on Machine Learning, Atlanta, GA, USA, 17–19 June 2013; pp. 1139–1147.
43. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
44. Zhang, Z.; Sabuncu, M. Generalized cross entropy loss for training deep neural networks with noisy labels. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; pp. 8778–8788.
45. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 568–576.
46. Pini, S.; Ahmed, O.B.; Cornia, M.; Baraldi, L.; Cucchiara, R.; Huet, B. Modeling multimodal cues in a deep learning-based framework for emotion recognition in the wild. In Proceedings of the 19th ACM International Conference on Multimodal Interaction, Glasgow, UK, 13–17 November 2017; pp. 536–543.
47. Gao, Q.; Ogenyi, U.E.; Liu, J.; Ju, Z.; Liu, H. A two-stream CNN framework for American sign language recognition based on multimodal data fusion. In Proceedings of the UK Workshop on Computational Intelligence, Portsmouth, UK, 11–13 September 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 107–118.

48. Ramachandram, D.; Taylor, G.W. Deep multimodal learning: A survey on recent advances and trends. *IEEE Signal Process. Mag.* **2017**, *34*, 96–108. [[CrossRef](#)]
49. Sarbolandi, H.; Lefloch, D.; Kolb, A. Kinect range sensing: Structured-light versus Time-of-Flight Kinect. In *Computer Vision and Image Understanding*; Elsevier: Amsterdam, The Netherlands, 2015; pp. 1–20.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).