Journal of Intelligent & Robotic Systems (2023) 107:44
https://doi.org/10.1007/s10846-023-01832-3

SHORT PAPER

Real-Time Visual Recognition of Ramp Hand Signals for UAS Ground Operations

Miguel Ángel de Frutos Carro[1] · Fernando Carlos LópezHernández[2] · José Javier Rainer Granados[3]

Received: 16 February 2022 / Accepted: 7 February 2023 / Published online: 20 March 2023
© The Author(s) 2023

Abstract

We describe the design and validation of a vision-based system that allows the dynamic identification of ramp signals performed by airport ground staff. This ramp signals' recognizer increases the autonomy of unmanned vehicles and prevents errors caused by visual misinterpretations or lack of attention from the pilot of manned vehicles. This system is based on supervised machine learning techniques, developed with our own training dataset and two models. The first model is based on a pre-trained Convolutional Pose Machine followed by a classifier, for which we have evaluated two possibilities: A Random Forest and a Multi-Layer Perceptron based classifier. The second model is based on a single Convolutional Neural Network that classifies the gestures directly imported from real images. When experimentally tested, the first model proved to be more accurate and scalable than the second one. Its strength relies on a better capacity to extract information from the images and transform the domain of pixels into spatial vectors, which increases the robustness of the classification layer. The second model instead is more adequate for gestures' identification in low visibility environments, such as during night operations, conditions in which the first model appeared to be more limited, segmenting the shape of the operator. Our results support the use of supervised learning and computer vision techniques for the correct identification and classification of ramp hand signals performed by airport marshallers.

Keywords Gesture Recognition · Convolutional Pose Machines · Aircraft Marshalling Signals · UAS

MSC 68T45

Categories (6), (7).

✉ Miguel Ángel de Frutos Carro
mad.frutos@alumnos.upm.es

[1] Centro de Automática y Robótica, Universidad Politécnica de Madrid-Consejo Superior de Investigaciones Científicas, 28006 Madrid, Spain

[2] Department of Mathematical Analysis and Applied Mathematics, Universidad Complutense de Madrid (UCM), 28040 Madrid, Spain

[3] Universidad Internacional de La Rioja (UNIR), 26006 Logroño, Spain

1 Introduction

Unmanned Aerial Systems (UAS) are a reality nowadays. The next big technological and operational challenge is the coexistence of UAS with manned aircraft. Airports and aerodromes provide a series of visual aids (light signals, colours, gestures, etc.) to help human pilots to safely position themselves and transit in the environment. These aids, described in international handbooks [1], were originally designed for human interaction, but must now coexist with UAS in these spaces.

According to Tomaszewska et al. [2] more than 26% of reported air traffic accidents occur on the ground, with an estimated cost of 11 M$ per year. Furthermore, the number of ground accidents increased by 200% between 2012 and 2017, mostly due to pure human errors during manoeuvres' execution or in combination with other factors, such as high

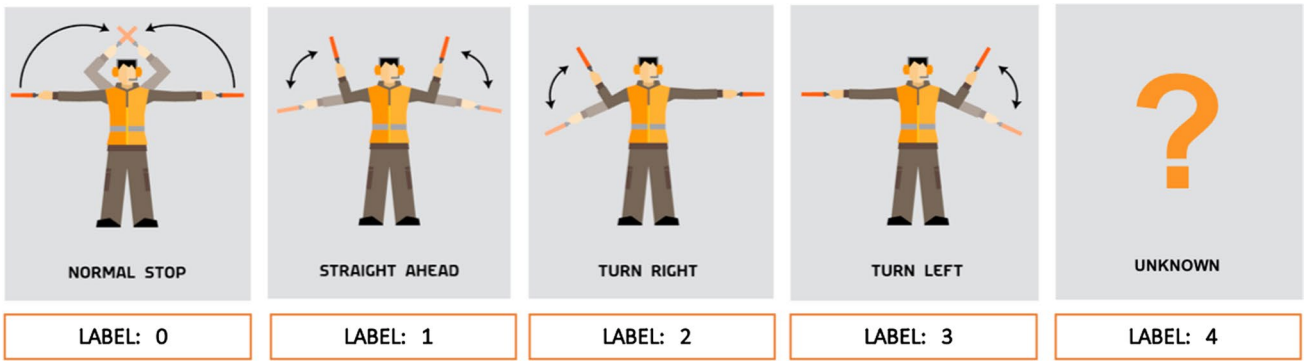


Fig. 1 Schematic representation of the five signal categories that the system needs to identify together with their corresponding label

workload, insufficient awareness of the situation or adverse weather [2]. Therefore, the presence of unmanned platforms in airports with autonomous navigation and their coexistence with current users, may decrease traffic accidents. Their implementation however is a fundamental problem in the sector. In this study, we attempted to solve this challenge by integrating visual recognition and interpretation of the main ramp hand signals -also known as aircraft marshalling signs-, performed by ground personnel (aircraft marshallers or ramp agents) equipped with visual identification aids, such as marshalling wands. The solution we reached represents a complete functional system that can be integrated in both unmanned aerial platforms and manned commercial aviation.

We evaluated and compared several strategies and models that allow the automatic and sufficiently successful identification and classification of ramp marshalling signals (Fig. 1). The device is intended for professional medium-sized UAS according to the classification table of the U.S. Army Roadmap for Group 2 category [3]. This category applies to systems that are too heavy to be operated by a single person but smaller than a light aircraft. They usually have a wingspan of about 5-10 m, landing gear, and can carry 100-200 kg payloads. Due to its size, they have restrictions for large or heavy equipment or requiring a lot of electrical power. We considered that, for being truly applicable, any proposed solution had to take advantage of the systems and signalling already present in the current airport environments, avoiding the adoption of new rules or technologies. It should also adapt to environmental situations (daylight, field of vision, etc.) comparable to those in which the human eye can recognize the gesture.

In this study we identified the most appropriate configuration for the system and elaborated two models based on visual processing and machine learning techniques, paying special attention to pre-trained neural networks. We compared the two models, identifying their specific advantages and limitations. To date, there is no properly labelled dataset for training and validation, and thus we developed our own dataset, which includes information taken from several individuals with different complexions. These individuals repeated each gesture several times in different lighting scenarios, including outdoors with real-life settings such as airfields with moving aircraft in the background.

Our solution has three potential applications:

1. *Unmanned Aerial Vehicles*: these vehicles need to operate in the same spaces of manned traffic and therefore they must be adapted to the current visual signage designed for humans. Providing UAS with a ramp signal recognizer would facilitate their introduction in existing airports and aerodromes without the need of large investment. As of today, only a small proportion of the larger unmanned vehicles (those with more than 1000 kg) have limited guidance assistance systems in airport environments based on heavy complex systems. However, the vast majority of professional UASs fall into the medium-size category [3], which is likely going to expand in the future. These UAVs currently lack "on-ground aid systems" and have restrictions to incorporate new equipment that penalizes their endurance and payload capacity.
2. *Manned aerial vehicles*: automatic visual recognition would provide the pilots with additional information, improving their situational awareness, alleviating their workload and even alerting of potential risky actions.
3. *Ground service vehicles*: there are many vehicles that operate on the airport apron providing ground assistance, including aircraft movement (tow tractors), passenger shuttling and baggage hauling. These vehicles operate daily on taxiways and are often involved in incidents that the proposed system would help to avoid.

The remaining of this manuscript is structured as follows: Sect. 2 reviews the state-of-the-art and currently available technology to address the problem of ramp hand

signal recognition. Section 3 describes our research strategy and presents the adopted materials and methods. The experimental analysis and evaluation of our classification architectures are described in Sect. 4, for the offline evaluation, and Sect. 5 for the online one. Section 6 presents a benchmarking against a standard CNN. Finally, Sect. 7 highlights the most important findings and describes future perspectives.

2 Background and Related Work

2.1 Aircraft Marshalling

Aircraft Marshaller is the official designation for the member of the ground personnel who helps the pilots to execute certain manoeuvres whenever there is a risk of incidents. These airport signals complement those the air traffic controller provides the pilot. A similar role exists in the defence sector under the name of flight deck personnel or Shooters when they perform their duties on aircraft carriers. These operators are equipped with a high visibility vest, ear protection helmets and two manoeuvring wands or flashlights. Their main functions are:

- Guiding of the aircraft in taxiing manoeuvres.
- Authorizing airplanes to perform certain manoeuvres.
- Coordinating the actions of all involved personnel.
- Managing and executing of emergency plans.
- Supervising regulation compliance and notifying any potential transgression.

Ramp signals refer to the combination of gestures that ground personnel perform with specific movements of their arms and hands. These gestures encode different messages directed to the human pilots located in the cockpit, from where they generally have a limited view of the aircraft's surroundings during taxiing and parking. There are different sets of gestures depending on their purpose. There are some discrepancies between the gestures used on the runway of a NATO aircraft carriers [4] and those used in international airports for civilian use [5]. Thus, the International Civil Aviation Organization (ICAO) dictated a series of mandatory recommendations for its adhering countries, in order to establish an international standard for all aspects of air transport [1]. Nowadays, these recommendations are the most popular and frequently used. The principal characteristics of ICAO gestures are:

- Only the arms are used.
- Although the gestures are dynamic, the spatial information prevails over the temporal one.
- Both arms encode information.
- Gestures are visually independent.

2.2 Human–Robot Interaction

Human–Robot Interaction (HRI) is a multidisciplinary field that addresses effective and efficient communication mechanisms between humans and machines, enabling their collaboration in the execution of a given task. Understanding human interaction and implementing human communication abilities into machines is thus a fundamental aspect of HRI [6]. During human communication, the transmission of intentions, interests and feelings are also important. Therefore, it is necessary to develop an intelligent agent capable of understanding these nuances and to establish a solid human–robot communication. To this end, HRI uses several channels of communication, involving different scientific fields, such as natural language processing, computer vision, gesture recognition, etc.

This research uses HCI to solve the communication between ramp operators and the autonomous vehicles without using remote controls. These interactions are explicit communication, given that messages are predetermined, univocally identifiable and there is no space for subjective appreciations. Thus, the communication factors [7] presented in this model are:

- Emitter: Aircraft marshaller.
- Receptor: Unmanned or manned aircraft.
- Code: ICAO Ramp signals.
- Direction: Unidirectional from the emitter to the receptor.
- Channel: Visually encoded.
- Context-Free: Independence from the understanding of the previous gesture.
- With noise: Occasional interferences such as lack of luminosity or visibility.
- Redundant: The gesture is maintained for as long as it is valid.

2.3 Gesture Recognition

Gesture recognition has been intensely studied during the last two decades, thanks to the increasing amount of information that humans encode in gestures as a complement to verbal communication complement or even as its substitute in case of channel or agent limitations [8]. The current applications of gesture recognition techniques are many and varied, from interaction with autonomous systems, home automation, video games or socio-sanitary robots. It can be limited to specific body parts or comprising the whole body, with obvious differences in the expected precision and detail level. For example, a system that attempts to classify body gestures [9] will not analyse the position of the fingers,

whereas in a system designed for driving of autonomous cars [10], the position of the hands and fingers prevails over the corporal information, as the driver is seated.

Gestures can also be classified according to spatial/temporal information they provide. Spatial gestures are easy recognizable, as they do not have a temporal dimension, which usually requires a sequence of related data. The influence of speed and position in the dynamic recognition of gestures is reviewed in [6]. Numerous techniques have been developed to recognise and classify gesture´s sequences, including the Hidden Markov Model (HMM) [11], the Longest Common Subsequence Problem (LCS) [12] or the Viterbi algorithm [13]. All of them are suited to identify dynamic gestures of the whole body [14] or only of the hands [15]. However, the techniques based on Dynamic Time Warping (DTW) outstand because they allow to compare two sequences with different time scales and determine their degree of similarity. New techniques based on the use of artificial neural networks have also emerged especially those used for video classification, which treat the problem sequentially, and not as an isolated classification of frames. For instance, Donahue et al. [16] proposes to extract the characteristics of each video frame with a Convolutional Neural Network (CNN) and then move the sequence to a second independent Recurring Neural Network (RNN). A similar solution is found in [17], in which the features extracted by the convolutional network are moved to a Multi-Layer Perceptron (MLP). This is based on the hypothesis that a MLP will naturally infer the temporal characteristics of the sequence without knowing that it is a sequence. Furthermore, the work based on three-dimensional convolutional networks (3D-CNN) can extract graphic characteristics directly from a set of temporally related frames, identifying a low-level representation with 3D convolutional operations. This approach has yielded excellent results [10] but has some important disadvantages [18] as compared to architectures based on 2D networks. Indeed, 3D-CNNs demand much more computational resources and memory, which limits their use for real-time applications or embedded platforms. On the other hand, there are few available pre-trained 3D-CNN networks for extracting features in videos, but there are numerous and very powerful networks trained using 2D-CNN architectures.

These techniques can also be classified into collaborative and non-collaborative. Techniques in which the user performing the gesture relies on the use of additional hardware are classified as collaborative or invasive. Examples are those based on gloves [19] or vests that help capture the motion through different sensors (inertial, position encoders, etc.). Other techniques, more relevant for this study, are based on computer vision without the aid of hardware (non-collaborative), such as those described in the seminal work of Paul Viola and Michael Jones [20] or Dalal and Trigs [21]. These techniques are frequently used because they are widely accessible and not very computationally demanding, though they do present some disadvantages, such as the number of false positives or the lack of detections of different human poses if they are not perpendicular to the camera. More recently, AlexNet showed the potential of Convolutional Neural Networks (CNN) for image classification [22]. Since then, numerous applications have used this approach. The Convolutional Pose Machines (CPM) are of particular interest for the gestures and postures recognition [23]. CPM are a set of pre-trained and deeply convolutional networks that identify the main joints of the body, hands, feet and even facial features through a series of coordinates or singular points reproducing an artificial skeleton. The recognition of traditional traffic police gestures has been successfully tested using a novel method of vision-based pose estimation [24][24]. Both works combine the CPM with a Long Short-Term Memory (LSTM) network, given that the gestures of the Chinese police are not context-free and require the integration of temporal features to achieve a successful recognition.

The problem of gesture recognition comprises the segmentation and the recognition of singular points [26]. The architecture can be implemented in a top-down or bottom-up manner. In the top-down, the segmentation identifies all individuals present in the image and then extracts the poses [27]. The bottom-up approach identifies the singular points and then groups them to form skeletons, producing the actual segmentation [28]. The model developed by the Computational Perception Laboratory of the Carnegie Mellon University is particularly interesting. The authors designed and trained an architecture based on several deep networks divided in two steps [29]. The first step, composed of 10 layers, identifies a map of the image characteristics. The second and more complex step consists of two CNNs in parallel, in which the first branch predicts a set of confidence maps based on the probability that the identified zone corresponds to one of the points of interest. The second convolutional branch instead infers a confidence map that encodes the degree of affinity between these individual points, forming the vectors that will make up the skeleton. The results obtained by the two networks are crossed using a greedy algorithm, finally identifying all singular points and vectors that will make up the skeleton (Fig. 2). Usually, a vector with at least three components represents every individual point on this skeleton: the first two correspond to the spatial coordinates (x and y) in a given reference frame, and the third measures the confidence of the body area identified by that point.

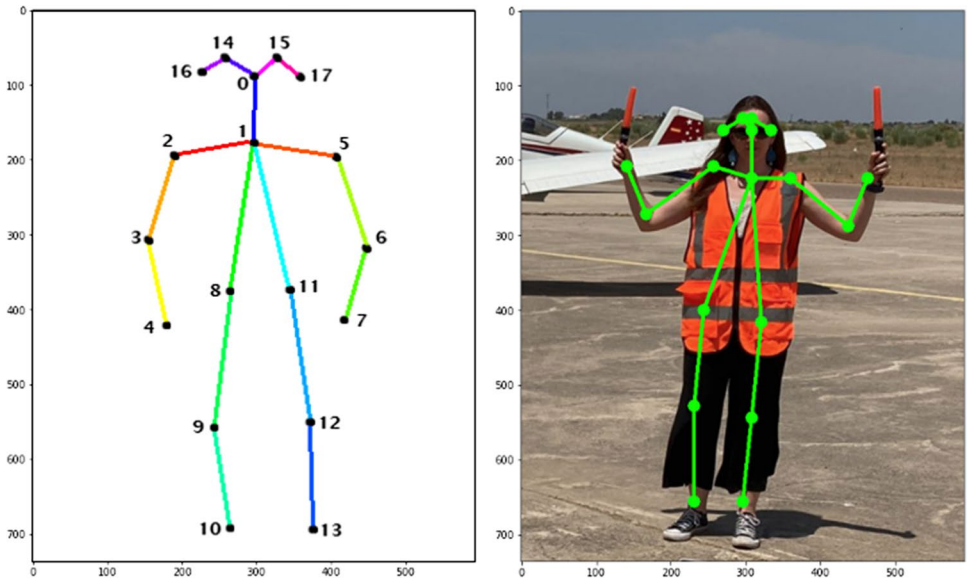


Fig. 2 CPM output in the COCO format

The models presented so far are bi-dimensional but there are models that expand the information to a third dimension. These models require the use of a camera equipped with depth sensors (as the Microsoft-Kinect®) for the simultaneous acquisition of visual and depth information or a mathematical model to infer a 3D representation from a 2D image [14]. One of such models is the Human Mesh Recovery (HMR) [30], based on the training of antagonistic networks with 2D images labelled for spatial information. The authors of the "Skepxels" concept (skeleton picture element) have followed this approach, adding time and speed information to the usual spatial coordinates, thereby obtaining a temporal representation of the skeleton [31].

When using a deep network-based architecture, the resulting model largely depends on the training dataset used. Numerous high-quality datasets have recently emerged, among which the Common Objects in Context – COCO is of particular interest [32].

There are few references related to the recognition of ramp signals. In two previous studies [12] [33], the strategy used to tackle the problem was based on images captured with an RGB camera, which were then used for classifying the gestures with either the Longest Common Subsequence Method [12] or the Radon Transform [33]. A third study addresses the recognition of signals used by NATO aircraft carriers [4] with depth sensors and the application of the Motion History Images (MHI) algorithm to classify both hands and whole-body gestures.

The results presented in this report go beyond these previous studies, providing a scalable system that has a direct industrial application.

3 Materials and methods

3.1 Research Framework

To generate a solution that could be applicable to the environment of medium-sized UAS, we have identified the following requirements:

a. *Work automatically* to detect and classify 5 ramp signals (Fig. 1) recognized by the civil authorities [5]: normal stop, straight ahead, turn right, turn left and, unknown or no-gesture when none of the previous is detected.
b. *Let immediate interaction by recognizing gestures with a latency below 3 s* [34].
c. *Use signalling and systems* already available in airport environments.
d. *Work properly* in different ambient condition (daylight, field of vision, etc.) in which the human eye recognise the gesture.
e. *Work correctly* in all positions with respect to the UAS in which a human observer is able to identify the gesture.
f. *Have a size* and weight that allows its installation in a professional medium-sized UAS [3] with landing gear and capable of taxiing through an aerodrome.

3.2 Approach for Ramp Hand Signals recognition

Our approach uses exclusively visual processing to recognize the upper body gestures, without relying on other sensors (e.g., rangefinders, depth sensors, etc.) or technologies (e.g., fingers or facial expressions). Our solution consists of an RGB camera and a small embedded

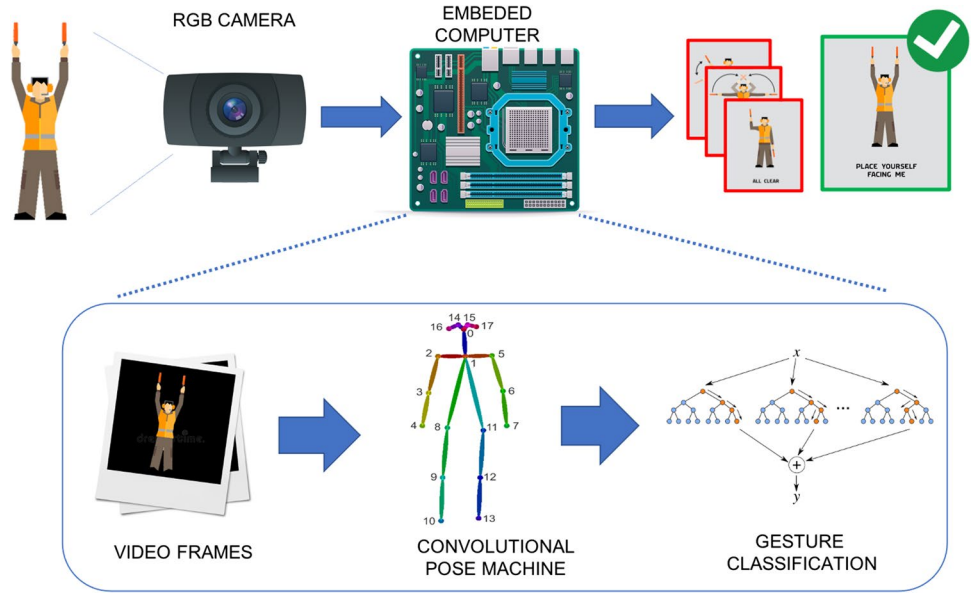


Fig. 3 Functional high-level diagram of the proposed device

processing unit with limited computational power, which is easy to integrate in different platforms. The software for the dynamic identification and classification of the ramp signals runs iteratively. In the following sections, we describe the evaluation of a combination of different supervised machine learning techniques that seemed to be the most appropriate for our goal.

3.3 Proposed Architecture

We used a CPM (a very deep 2D CNN pre-trained and optimized with large general datasets) for the identification and extraction of the marshaller extremities followed by gesture classification (i.e., supervised-classification, Fig. 3). The output of the CPM is a set of coordinates identifying the different human joints [29]. This transforms a set of pixels into spatial vectors, which simplifies subsequent calculations and classification. These numerical values, duly normalized and processed, are the input for the next step: gesture classification. We evaluated two different supervised classifiers for this step.

The architecture is based on six clearly differentiated iterative functions with separate objectives. The inputs/outputs are linked as shown in the flow chart depicted in Fig. 4 and are described below:

1. *Capturing and pre-processing the image*: We use an onboard RGB camera to capture the image. These cameras have sufficient image quality and capture speed to provide an acceptable spatial and temporal resolution. This step encompasses video capture tasks and initial image pre-processing.
2. *Segmentation and pose extraction:* The CPM used for posture segmentation and extraction is based on the ResNet-18 model, an effective 18-layer deep neural network that follows a Residual [35] type architecture pre-trained on the MS-COCO dataset [32]. This represents a compromise between complexity and spatial resolution. Its environment's independence has several advantages like the reduction of the training dataset for ramp hand signal classification and enhanced robustness. The selected output is a skeleton composed of 18 points. Each point is composed of 2 Cartesian coordinates, x and y, indicating the position of the joints relative to the origin of the coordinates.
3. *Normalization:* To ensure the independence of the coordinates from the position of the subject in respect to the camera, the image coordinates are transformed using one of the points within the skeleton as the reference of origin.
4. *Differentiation and reduction of dimensionality:* only the points corresponding to the arms and trunk are relevant, whereas the rest of the information can be discarded.
5. *Gesture classification.* We tested two supervised learning approaches for selecting the classification method: Random Forest (RF) and a densely connected MLP. Both approaches have been proven to efficiently recognize other kinds of human gestures with limited training data [6]. The choice of hyperparameters and the discussion of the results obtained are analysed in the next section.
6. *Filtering and representation:* This step executes the weighting logic and low-pass filters to maintain a continuous output between the different categories and min-

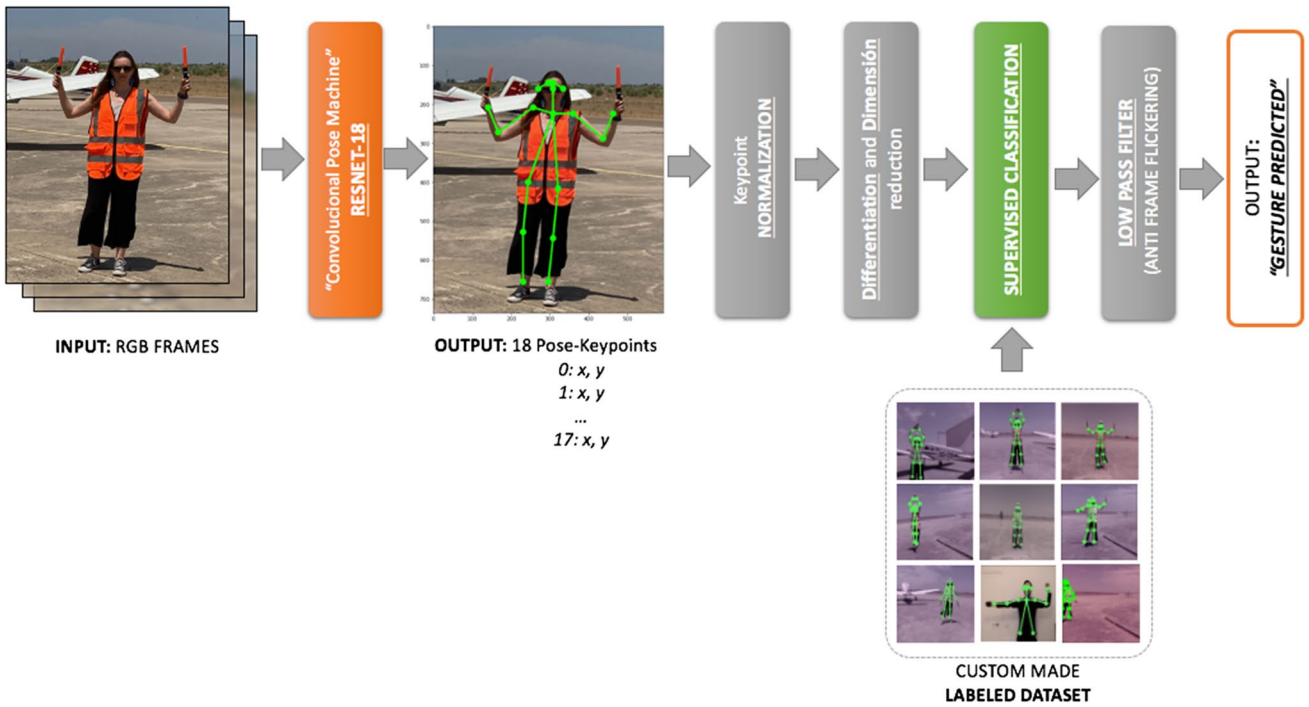


Fig. 4 Flow diagram with the main functional blocks of the proposed architecture

imize the effects of flickering by classifying different categories of consecutive frames.

Of note, the reduced size of the available training data does not map this problem to a pure image classification problem. However, this alternative has been tested in the second architecture proposed, and explained in Sect. 6.

We have discarded the use of 3D-CNN, which can extract graphic characteristics directly from a set of frames with a temporal relationship. This is because of its high computational demand for real-time applications [18], although it has given excellent results in other gesture recognition applications [10].

For the classification layer, we have considered only models based on supervised training. Models based on unsupervised clusters, such as K-means, have been widely used [14] because new gestures can be added easily. However, our proposed scenario is limited to the five most representative gestures out of the 20 international gestures and does not contemplate new additions or variations. This justifies our selection of a supervised model, with a better classification performance and less uncertainty, assuming that the dataset is unbiased, sufficiently large and without outliers.

The inherent advantages of using a previously trained model for first pose estimation does not overcome the need of a ramp hand signal dataset at the classification stage. Indeed, there are no such public datasets. We thus initiated the generation of a manually labelled, small dataset of these gestures. When capturing the images, we considered different body sizes and lighting conditions. Section 3.7 describes in detail the development and validation of the model.

3.4 Contextual conditions

We assumed the following hypotheses in the experimental setup:

1. The subject remains in a position facing the camera.
2. The image quality is such that the human eye can identify the executed gesture.
3. The duration of the gesture is similar to the real ones, avoiding rapid changing or contradictory gesticulation.
4. The gestures and poses are executed so that the arms overlap with the torso as in real conditions.
5. The illumination is sufficient to recognize of the marshaller silhouette.
6. The marshaller has a collaborative attitude.
7. The issues related to mechanical or electrical integration in the aerial platform are not considered.
8. The issues related with software development strategies, or level of guarantees, needed in all software/hardware development for aviation are not addressed.
9. The selected solution should fit for a state-of-the-art embedded computer.

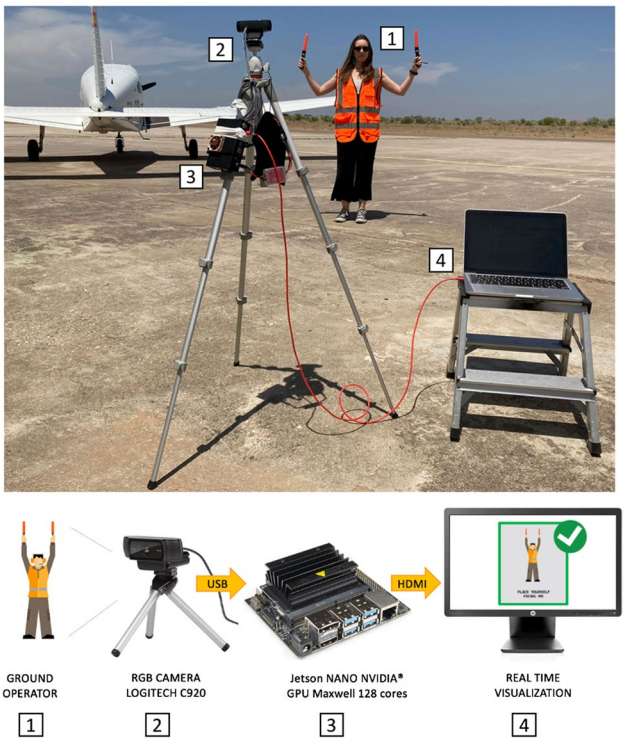


Fig. 5 General description of the test setup used in the course of this work

- *A RGB camera* with automatic focus and a 720p / 30FPS resolution, located on a tripod and connected via USB to the embedded computer.
- *The Embedded Computer, a* NVIDIA® Jetson Nano (Linux Tegra L4T) executes the software used for both the generation of the dataset and the demonstration in real-time.
- *A Monitor / External PC* is used as auxiliary validation and testing tool.

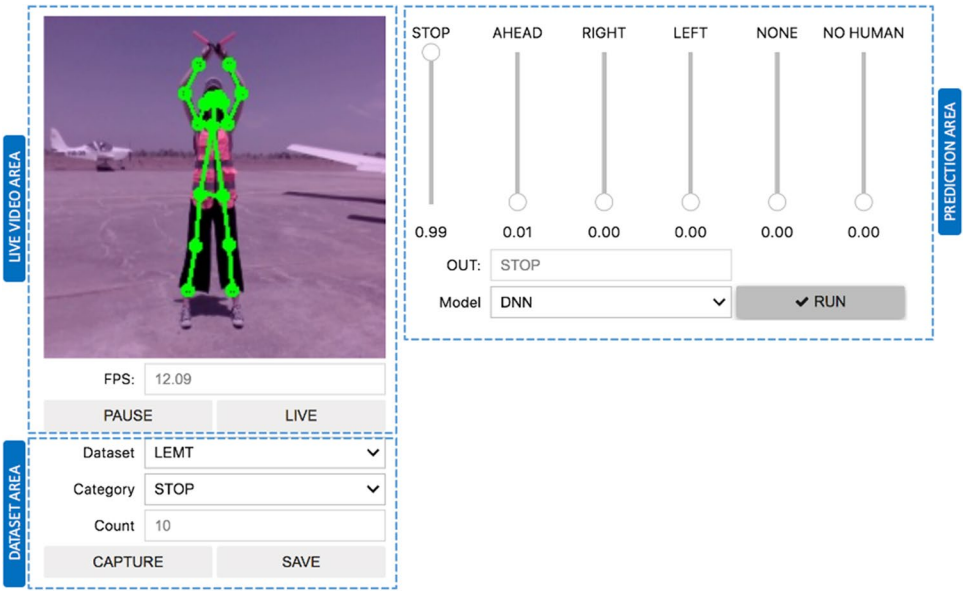
The custom software tools developed are publicly available in an online repository (see Sect. 8). They have been built using the following frameworks and software packages:

- *TensorFlow 2.0.*
- *NVIDIA TensorRT®.*
- *Tensor RT Pose Estimation.*[1]
- *Scikit-Learn 1.1.3.*

3.6 Experiment description

We have developed a software application to facilitate the operations through a Graphical User Interface (GUI). This was especially useful for dataset generation and experimentation. The GUI consists of a single tab with three main panels (Fig. 6):

Fig. 6 Outline of the developed GUI with the different areas labelled in the blue boxes



3.5 Research Resources

The setup for dataset generation (capturing and labelling the different images) and real-time evaluation consists of 3 main elements (Fig. 5):

1. *The Live video panel* shows the real-time RGB images captured by the camera with the superposed CPM output (the inferred skeleton) and shows the time to execute the

[1] Open-source project that implements a pre-trained CPM: https://github.com/NVIDIA-AI-IOT/trt_pose

Table 1 Total number of samples registered from each of the three volunteers who participated in the dataset acquisition

Category	Number of samples
STOP	1548
AHEAD	1507
RIGHT	1419
LEFT	1381
NONE	1372
Total	7 227

3. *The Prediction panel* contains all elements related to the selection of the classification model (Models menu) and the corresponding output. The output is encoded in 6 probability bars that indicate the probability with which each category is predicted (from 0 to 1). We have also included a "NO HUMAN" category that appears when the presence of a person in the image is highly unlikely. The most probable category is shown in the OUT field, and the execution of the classification can be stopped or resumed at any time with the RUN button.

Fig. 7 Sample image gallery of Ramp Hand signal dataset, manually collected and labelled

main loop, measured in frames per second (FPS). The PAUSE/LIVE button stops and resumes the output of the camera and the CPM superposition.

2. *The Dataset panel* includes all the tools needed for capturing and labelling images for training and validation purposes (Category menu), as well as the tools for classifying the images into different subjects (Dataset menu). The Count field keeps track of captured number of samples per category and subject. The CAPTURE and SAVE buttons respectively capture the CPM output and save it to the array of samples.

3.7 Data Collection Procedure

The data were captured in a variety of environmental and lighting conditions, including controlled indoor environments and real-life outdoors scenarios, such as an airfield with manned traffic or people and other support vehicles moving freely around. We used the camera described in Sect. 3.5 to capture the footage. We produced a total of 7227 labelled samples, involving 2 males and 1 female. Each gesture was repeated

about 1400 times, generating the distribution shown in Table 1.

Figure 7 shows a sample of raw images used to generate the datasets. Each volunteer was individually instructed to correctly perform the different signs based on a video executed by a professional. Volunteers were asked to move within the field of view of the camera, without following a predefined position of the body angle with respect to the camera, and complying with the assumptions defined in Sect. 3.4. This was performed to teach the classifier to dismiss useless information and focus only on the arms. The volunteers were also wearing the characteristic equipment (reflective vest and marshalling wands) to determine if their reflective nature prevented the system to operate normally and to verify slight gesture deviations imposed by the equipment. The subject's privacy was kept at all times by saving the mentioned coordinates and discarding the real images from the public dataset available in an online repository (see Sect. 8).

The data curation, performed on a general-purpose computer, included importing and grouping all data in a single object per category, further detecting and deleting any outlier values to avoid a negative impact on the training process. The data were then exported to.csv file for convenience, and a list of headers to identify each coordinate was added. We prepare the data for model training, by appending the corresponding labels to each sample, grouping them in a single dataset and shuffling to obtain a homogeneous random set. The labels were then converted into categories, resulting in low density and dispersed vectors. The samples were finally divided into two groups with 80% of them as training data and the remaining 20% as validation data. The source code is available at the repository indicated in Sect. 8.

4 Results and Offline Evaluation

A previous study evaluated more than 126 classification models for the recognition of dynamic gestures, concluding that the neural network and the RF classifier were the most accurate for gesture recognition [6]. After preliminary experimentation, we agree with this conclusion. We found that the supervised learning models RF and MLP were the best performing classifiers for our purpose. As already mentioned, both models have been trained on a general-purpose computer and then transferred to the embedded processing unit dedicated only to real-time execution. Data preparation is the same for both classifiers. The 18 two-dimensional vectors that make up the skeleton must be serialized from the labelled dataset into a single vector with 36 fields. Each model uses this training dataset together with the ground truth vector to minimize the loss function. Considering that our data are numerical and the datasets small, the training process was rapid, of the order of seconds in pairs for the RF and of a few minutes for the MLP.

The confusion matrix was the best metric for our analyses because it shows on a table the relationship between prediction and ground truth and enables the easy calculation of other metrics. Among other available metrics, we also used:

- *Accuracy or success rate* that indicates the ratio between the final correct model prediction and the total number of predictions. This metric could be misleading when the class distribution is unbalanced, which is not the case of our dataset.
- *F-Score. This* widely used parameter boils down the performance of the model to a single metric; in particular, combining accuracy and recall in a harmonic mean [6]. It also serves to assess the trade-offs adopted by the model during classification.

As in any supervised learning model, the correct choice of hyperparameters plays a fundamental role. We did not evaluate in details the possible different hyperparameters and their potential optimization, but we have taken the first solution that allows for the evaluation of the entire system.

4.1 Random Forest

The RF is a well-known learning method that uses multiple decision trees during the training time [36]. This method has been extensively used in a variety of applications owing to its prediction capacity and easiness in training and implementation. By combining several trees and choosing the final result as the most voted in each individual tree, this method achieves the greatest generalisation and prediction, with low noise and instability, even without using large datasets. In this method, each new tree is built with a random fraction of the available samples and of the variables. Increasing the number of trees raises the possibility that the most descriptive variables appear several times in different trees. This characteristic is essential since the final variance is highly reduced, even though the model bias increases with respect to a single tree. On the other hand, the fact that each tree leaves a fraction of the samples unused (out-of-the-bag) allows for sequential training without the need for cross-validation or leave-one-out technique, as they achieve similar effects. The most important hyperparameters to choose are the number of trees and the number of variables. To analyse the impact of these two hyperparameters, the root-mean-square error (RMSE) against the test dataset was used.

When evaluating the number of samples to extract to train each base estimator, we concluded (as Fig. 8b shows)

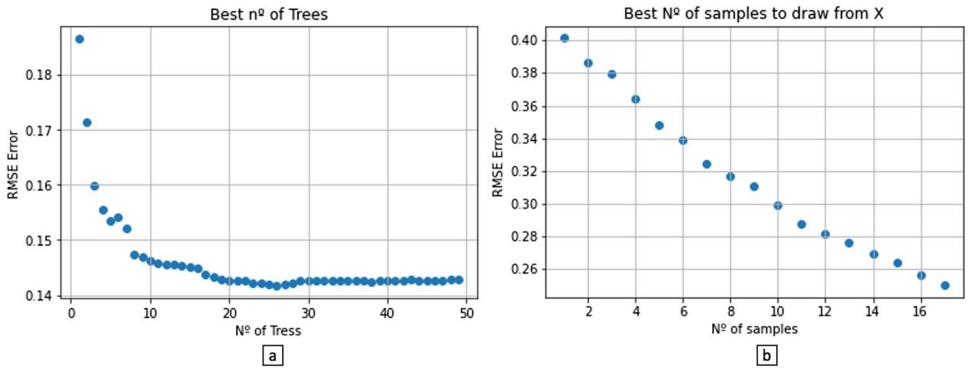


Fig. 8 RMSE against hyperparameters number of trees (**a**) and number of samples (**b**) for the RF method

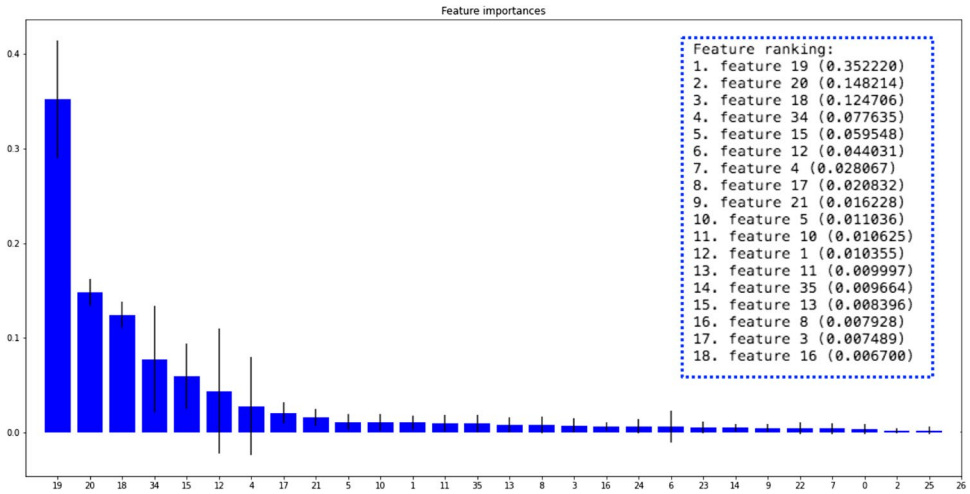


Fig. 9 Classification of the most descriptive variables according to the improvement obtained in the cut-off criterion for each variable in each of the trees

that the error decreases by increasing the number of samples in each estimator and finally decided to keep the default value that was set equal to the maximum number of independent variables. On the other hand, the error of the test clearly depends on the function of the maximum number of trees allowed. After analysing the results, we found that the error stabilizes when we reach 50 trees, that is, the error does not reduce by increasing the number of trees (Fig. 8a). Finally, we established the value of this parameter in 26 trees, which value corresponds to the minimum error achieved (Fig. 8a).

Once the combination of hyperparameters that best fits the characteristics of our data set were analysed, we trained our model. One of the advantages of this method is that it allows for the elaboration of graphs in a simple way that shows the importance of each variable (Fig. 9), as the aggregate result for all trees of the improvement obtained in the cut-off criterion for the variable.

Figure 9 shows an exponential distribution in which the first 5 variables account for the 75% of the decision information. In particular, the first 3 variables (19, 20 and 18) correspond to the wrists' coordinates. This seems logical given that these are the joints that move the most in each gesture and thus best contribute to their differentiation.

Figure 10a depicts the results of the RF model as a confusion matrix. The obtained values are remarkable, with a 97% success rate. Among the classification failures, the stop ('STOP') and the forward ('AHEAD') gestures stand out. This seems logical if we consider that both gestures are based on the movement of both arms near the head.

4.2 Multi-Layer Perceptron

This is the second artificial neuronal network (ANN) selected for this research. The first ANN model we used (i.e., the CPM) was pre-trained with the MS-COCO dataset [32] to identify the coordinates of the body extremities. This model follows a completely different architecture and

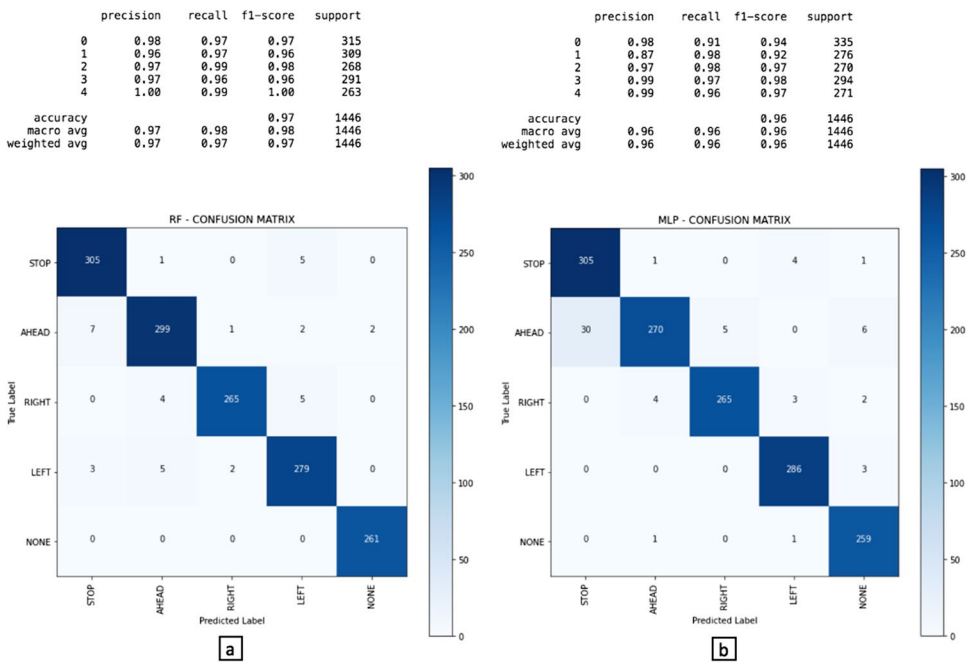


Fig. 10 Confusion matrix, F-score, accuracy and derived metrics for the offline performance evaluation of the RF model (**a**) and MLP model (**b**)

uses an input vector with the coordinates of the body to calculate the probability of each image to fall into to one of the 5 classes. Its topology is based on an MLP, a class of feed forward and densely connected neural networks. The information follows a sequential path from the input to the output layer. Unlike the first model, pretrained on a large dataset, this second ad hoc MLP classifier was trained with our smaller dataset.

We identified 3 main types of layers in the architecture of our ramp signal classifier:

- *Input layers:* each of the neurons in this layer can process one of the dataset variables. In our case, the input layer contained 36 neurons, that is, 18 pairs of x, y coordinates, corresponding to the joints the CPM identified in the previous step.
- *Output layers:* the number of possible results in the classifier determines the number of neurons in the output layer. In our case, the layer is formed by 5 neurons with a SoftMax activation function, which provides a normalized vector with the probabilities of belonging to each of the mutually exclusive classes (one-hot encoding).
- *The Hidden layers* have an intermediate position in our network between the input to the output layers. The number of neurons in each layer is a hyperparameter of the model. Its choice will be discussed later.

We chose a Rectified Linear Unit (RELU) as the activation function for all the neurons in the hidden layers of the network, because it is computationally efficient and has given excellent results in different situations. We have chosen a categorical cross-entropy for the loss function given the nature of our problem and how we prepared the labels. We opted for an ADAM type optimizer (Adaptive Moment Estimation), which, based on the same principle as the calculation of the descent through gradients, introduces the advanced concepts of moment and RMSProp, which translates into a more robust and faster convergence method.

Two important hyperparameters were considered:

1. The number of epochs that help to defining how many times the data of our training set need to pass through the network. This value is critical to prevent the model from overfitting.
2. The size of the training mini-batches, which defines the number of samples that we use in each iteration.

We used a model composed of two densely connected hidden layers. We have inserted regularization filters based on the Dropout technique between the layers of neurons. These filters allow 50% of the neurons to be randomly deactivated only between iterations of the learning cycle, thus preventing the memorisation of the training data and the possibility of generalizations. During the training phase, we have calculated a value for each one of the 21,893 parameters (weights and biases) that compose our network.

After some training iterations, we concluded that there is no advantage in extending their number beyond 300 epochs, given that this number is sufficient to reach a minimum and stable value of the learning curves (Fig. 11). Our initial results showed 96% success over

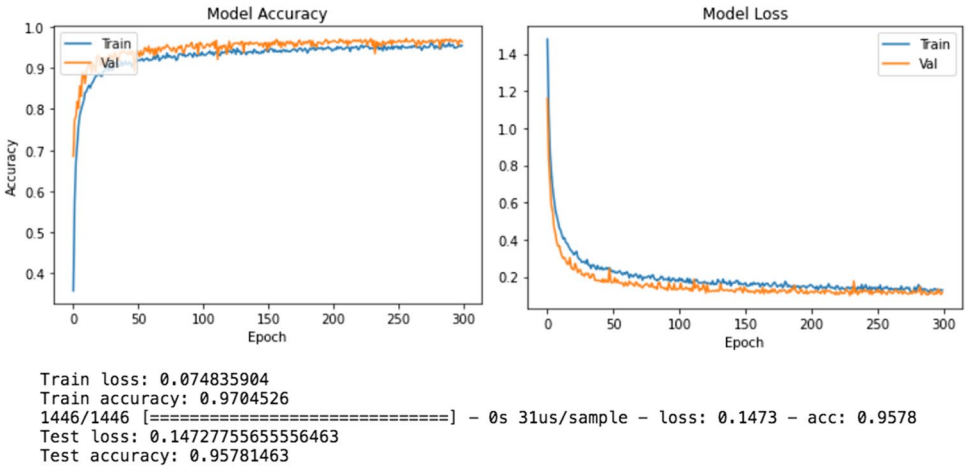


Fig. 11 Accuracy and Loss Function metrics obtained during the training phase of the MLP-based model

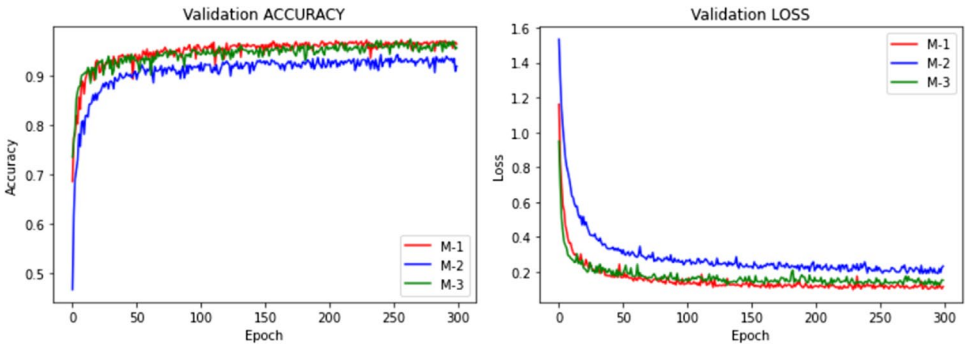


Fig. 12 Comparison of the Accuracy and Loss Function metrics for the 3 set of evaluated hyperparameters evaluated

the validation set (i.e., samples that the network had not seen before), with a stable curve of the validation data that closely follows that of the training data (Fig. 11), indicating that the network does not overfit. It should be noted that the Dropout technique was disabled at the time of the test, and therefore the validation loss and the accuracy values improved with respect to the performance obtained during the training phase.

We thus tested if adjusting the hyperparameters would improve the classification. However, comparison of our original model (Model-1) with two similar models in which the number of neurons in all layers has been either reduced to 32 (Model-2) or increased to 256 neurons (Model-3) showed no significant improvement (Fig. 12). Indeed, reducing the number of trainable parameters (2405 trainable parameters) deteriorate the performance, whereas increasing them (142,341 trainable parameters) does not significantly improve it (Fig. 12). In conclusion, in both cases, there were no significant increase of the computational cost or the execution time and the differences were small and attributable to the number of neurons presented in each network. We thus selected Model-1 as the best solution for our purpose.

We next compared the generated RF and MLP models by calculating the confusion matrix and other performance indicators for the selected MLP model. As shown in Fig. 10b, most errors can be assigned to the misclassification of several 'AHEAD' gestures into the 'STOP' category. As we indicated, these RF model errors may proceed from the standard position of the hands close to the head and above the neck.

Comparing the results of both classifiers for the offline evaluation (Fig. 10), we observe that both RF and MLP achieve similar accuracy and performance with the other metrics used: confusion matrix and f-score. In addition, both models tend to produce the similar mistakes when classifying similar gestures.

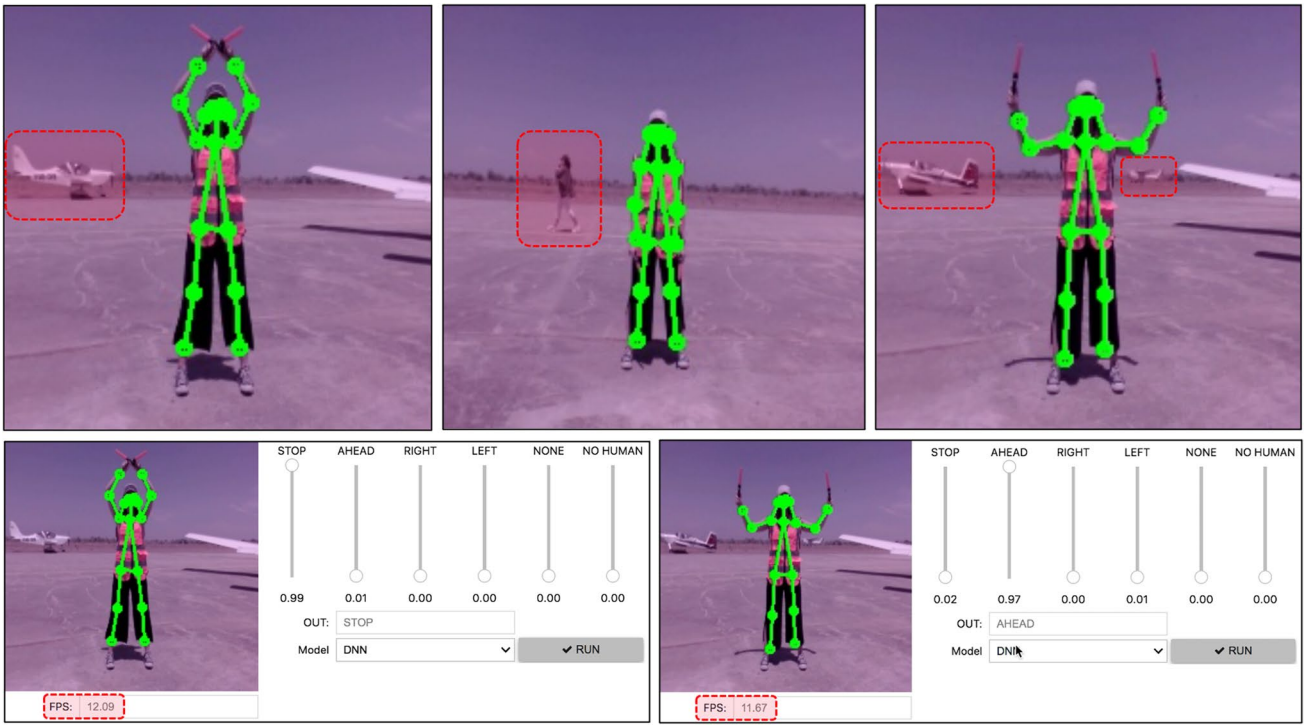


Fig. 13 Samples images of SW tool used for online validation during real test scenarios. FPS field, vehicles and people in the background have been highlighted in red

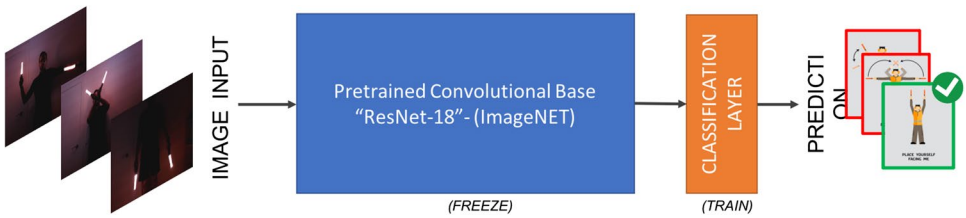


Fig. 14 Second proposed architecture based on a single Convolutional Neural Network (CNN) for gesture classification directly from the images

4.3 Online Performance Evaluation in a Relevant Environment

After training and evaluation offline, both models were transferred to the embedded computer to test their real-time recognition capacity under operational conditions similar to those expected if installed in a UAS. To this end, we moved the test setup (Fig. 5) to the trafficked apron of the aerodrome with ICAO code LEMT, in which manned and unmanned aviation could soon coexist. We tested a single subject dressed in the characteristic equipment (see Sect. 3.7). The software tool described in Sect. 3.6 was used to analyse the performance and the success and execution rate (measured in FPS) of the two models. Figure 13 shows examples of these validations in a real environment.

The high dynamism and the lack of control of the real environment make it difficult to quantify the success rate of each classifier under equal conditions. Nevertheless, the predictions for the classification of both models were similar to those observed during the offline validations as long as the CPM could extract the joints' coordinates. We should note that the camera (see Sect. 3.5) had problems in capturing an adequate image in some lighting conditions, such as direct lighting. This impaired the CPM task and gesture classification. Given the small training dataset, the low dependency on the background is significant. This may be mainly due to the previous training on a large dataset of the CPM network, responsible for the first segmentation and extraction of the individual points.

The execution rate was rather consistent for both models, with image capturing and running at an average of 12 FPS, which translates into a response time of approximately 85 ms. Given that an aircraft movement on an aerodrome surface is less than 10 m/s [1], the achieved response seems in both cases sufficient for fluid interaction of the ground

guidance tasks. Of note, an intermediate optimization action was required to run the MLP model on the embedded GPU hardware efficiently, otherwise the system was, on average, three times slower than the RF model run in the same conditions.

On the basis of the qualitative run-time analysis, we conclude that both models can correctly recognize the different gestures. Nevertheless, in general, the MLP performed better, with a less noisy response and prediction uncertainty.

4.4 Benchmarking the CNN

Although the proposed architecture gave excellent results, showing little dependency on the environment, we evaluated a possible alternative (Fig. 14), based on a pure image classification task, using a classical convolutional classification network (CCN). This architecture was pre-trained on a general large dataset, in which the last layer was replaced with a classifier customized for the gesture recognition needed for our study.

Although we started from a network in which most of the parameters had already been pre-calculated, we needed to generate a new dataset for training the additional classification layer. We thus generated a completely new but smaller dataset (450 images labelled) and used data augmentation techniques to artificially increase the amount of data. We took care that these transformations did not result in images inconsistent with their label or with the problem itself (i.e., symmetries in which the position of the arms is horizontally reflected or the marshaller appears upside-down).

Unlike the previous architecture in which there were two models in series with differentiated roles, this new architecture presents a single network, in which the matrix of the pixels of the captured images is the input parameter and the probability of belonging to each one of the categories is the output.

We chose a ResNet-18 network, pre-trained over the ImageNet dataset (with 14 million annotated common objects images) [22]. The last layer was modified to fit only 5 categories according to our study. Using the application of the Transfer Learning technique, we could reuse all the parameters calculated in the previous 17 frozen layers, which are in charge of extracting the low-level characteristics of the image. Thus, this last layer is a network connecting the 512 neurons of the ResNet-18 to each one of the proposed categories. Each neuron of this layer has a SoftMax activation function. We used again the ADAM optimization algorithm. The loss function is defined as categorical cross-entropy, given that this is still a classification problem of self-excluding category.

After implementing and training the new model, we obtained good results in the offline evaluation phase. However, the model overfitted and cannot generalize well new data, due to the small number of images used in the training phase. In other words, the model does not behave properly when the operator or the background diverge from the training images.

5 Conclusion and Future Work

Here, we have proposed and evaluated two vision-based architectures for the dynamic identification of ramp hand signals used by airport ground staff. The first architecture is based on a first segmentation phase using a pre-trained CPM to extract the coordinates of the operator extremities. This is followed by a second gesture classification step, based on supervised learning, to classify each executed gesture. In this configuration, the RF and the MLP classifiers achieved similar accuracy for the offline evaluation. We obtained a similar performance with the other used metrics: confusion matrix and f-score. We also have performed an online evaluation against a practical operating environment, an aerodrome with characteristics in which an UAS would operate. Both models can process a new image every 85 ms; however, the model based on MLP appeared to be more robust but requires a GPU optimization. In fact, the MLP is 3 times slower than the RF model when both are executed on CPU. Thus, a non-optimized deep neural network-based model should be avoided if the hardware has insufficient computational power. The greatest advantage of this architecture is its adaptability to the environment, which makes it more robust and scalable.

The second architecture is based on a pure image classification task using a CNN pre-trained on a large but general dataset. The last neuronal layer of this CNN architecture was replaced with our own classifier. Due to the small number of images used in the training phase, the model becomes overfitted and thus could not generalize new data. This difficulty could be overcome only by generating a really big dataset, of several orders of magnitude larger than the one we used. This new dataset should include operators with different complexions, clothing and, more importantly, diverse backgrounds. Due to the complexity and magnitude of that process, we opted for using the first CPM approach given its scalability, the use of summarized data, the high performance and suitability for real dynamic backgrounds. Nevertheless, the second architecture could be very useful for night operations. Indeed, we tested this possibility by generating a third dataset with dark images in which the operator uses the characteristic light wands to signal gestures in the dark, confirming its suitability in these conditions. The model based on the CPM instead cannot perform a proper segmentation in the dark.

As a next step, we would like to transform the current prototype into a reliable industrial solution, which implies improving the robustness and scalability of the system and the possibility to combine the different architectures presented here. This solution could enable the system to operate in any type of illumination. The first layer would assess if there were dark pixels in the image, progressively giving more weight to the CNN model prediction, as the background gets darker. We should also add more information or features to the coordinates calculated by the CPM. Indeed, adding variables that compute the speed of the gesture might solve the problem of classifying similar gestures in the wrong category, as we have observed. These additions would also enable the system to identify if the manoeuvre is executed at a higher or lower speed, encoding more information. This extra information could be estimated by the frequency of oscillation of each arm using, for example, an FFT analysis on the coordinates produced by the CPM. Finally, we could encode this quantitative information into qualitative information using some kind of fuzzy logic, instructing the system if the movement is slower or quicker than usual.

These are some solutions that could be developed. However, the results we present here altogether represent a proof of the concept that it is possible to use supervised learning and computer vision techniques for the correct identification and classification of ramp hand signals performed by airport marshallers.

Author Contributions Miguel Ángel de Frutos (MAdF) initiated the present project during his Master of Sciences at the Universidad Internacional de la Rioja (UNIR) and performed the subsequent data analysis and manuscript elaboration while pursuing his doctoral research at the Universidad Politécnica de Madrid (UPM). Fernando López Hernández (UCM) and J. Javier Rainer (UNIR) supervised the elaboration of the manuscript. All authors discussed and approved the final manuscript.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature. The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Data Availability A repository with the dataset used, the software, the figures and the demonstration videos, is publicly available at: https://github.com/astromaf/ramp_hand_signals_recognition

Declarations

Ethics Approval The authors declare that this work is original and does not include experiments with animals.

Consent to Participate All individuals participating in the study provided an informed consent. The captured information has been nonetheless adequately anonymized.

Consent for Publication The participants in the experiments provided informed consent for publication of the related images. Nevertheless, their faces or any other biometric data can be recognised in the relevant images.

Conflict of Interest The authors have no relevant financial or non-financial interests to disclose.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. ICAO, Annex 2 - Rules of the Air - Tenth Edition, no. November. (2005)
2. Tomaszewska, J., Zieja, M., Woch, M., Krzysiak, P.: Statistical analysis of ground-related incidents at airports. J. KONES **25**(3), 467–472 (2018). https://doi.org/10.5604/01.3001.0012.4369
3. Dempsey, M. E., Rasmussen, S.: "Eyes of the army--US Army roadmap for unmanned aircraft systems, 2010--2035," (2010)
4. Song, Y., Demirdjian, D., Davis, R.: "Tracking body and hands for gesture recognition: NATOPS aircraft handling signals database," 2011 IEEE Int. Conf. Autom. Face Gesture Recognit. Work. FG 2011, pp. 500–506 (2011).https://doi.org/10.1109/FG.2011.5771448
5. Civil Aviation Authority (CAA), "Visual aids handbook," Aids.**10**(6), 690–691, (1996). https://doi.org/10.1097/00002030-199606000-00024
6. Castillo, J.C., Alonso-Martín, F., Cáceres-Domíngue, D., Malfaz, M., Salichs M. Malfaz, A., Salichs, M.A.: "The Influence of Speed and Position in Dynamic Gesture Recognition for Human-Robot Interaction," J. Sensors., (2019). https://doi.org/10.1155/2019/7060491
7. Shannon, C.E.: "The Mathematical Theory of Communication," M.D. Comput., (1997). https://doi.org/10.2307/410457
8. Demarco, K.J., West, M.E., Howard, A.M.: "Underwater human-robot communication: A case study with human divers," Conf. Proc. - IEEE Int. Conf. Syst. Man Cybern., vol. 2014-Janua, no. January, pp. 3738–3743, (2014). https://doi.org/10.1109/smc.2014.6974512
9. Baek, T., Lee, Y.G.: Traffic control hand signal recognition using convolution and recurrent neural networks. J. Comput. Des. Eng. **9**(2), 296–309 (2022). https://doi.org/10.1093/jcde/qwab080
10. 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 Networks," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Decem, 4207–4215 (2016). https://doi.org/10.1109/CVPR.2016.456
11. Kapuscinski, T., Oszust, Wysocki, M.,D. Warchol.: "Recognition of hand gestures observed by depth cameras," Int. J. Adv. Robot. Syst., vol. 12, (2015). https://doi.org/10.5772/60091
12. Choi, C., Ahn, J.H., Byun, H.: "Visual recognition of aircraft marshalling signals using gesture phase analysis," IEEE Intell. Veh. Symp. Proc., pp. 853–858 (2008). https://doi.org/10.1109/IVS.2008.4621186

13. Waldherr, S., Romero, R., Thrun, S.: Gesture based interface for human-robot interaction. Auton. Robots **9**(2), 151–173 (2000). https://doi.org/10.1023/A:1008918401478
14. Ribó, A., Warchol, D., M. prz edu pl Oszust: An approach to gesture recognition with skeletal data using dynamic time warping and nearest neighbour classifier". Int. J. Intell. Syst. Appl. **8**(6), 1–8 (2016). https://doi.org/10.5815/ijisa.2016.06.01
15. Raheja, J.L., Minhas, M., Prashanth, D., Shah, T., Chaudhary, A.: Robust gesture recognition using Kinect: A comparison between DTW and HMM. Optik (Stuttg) (2015). https://doi.org/10.1016/j.ijleo.2015.02.043
16. Donahue, J., et al.: Long-Term Recurrent Convolutional Networks for Visual Recognition and Description. IEEE Trans. Pattern Anal. Mach. Intell. **39**(4), 677–691 (2017). https://doi.org/10.1109/TPAMI.2016.2599174
17. Zhou, B., Andonian, A., Oliva, A., Torralba, A.: "Temporal Relational Reasoning in Videos," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), **11205** LNCS,831–846 (2018). https://doi.org/10.1007/978-3-030-01246-5_49
18. Hara, K., Kataoka, H., Satoh, Y.: "Can Spatiotemporal 3D CNNs Retrace the History of 2D CNNs and ImageNet?," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., pp. 6546–6555, (2018). https://doi.org/10.1109/CVPR.2018.00685
19. L. Abraham, A. Urru, N. Normani, M. P. Wilk, M. Walsh, and B. O'flynn, "Hand tracking and gesture recognition using lensless smart sensors," Sensors (Switzerland), vol. **18**, no. 9, (2018). https://doi.org/10.3390/s18092834
20. Viola, P., Jones, M.: "Rapid object detection using a boosted cascade of simple features," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, (2001).https://doi.org/10.1109/cvpr.2001.990517
21. Dalal, N., Triggs, B.: "Histograms of oriented gradients for human detection," in Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR (2005). https://doi.org/10.1109/CVPR.2005.177
22. Krizhevsky, A., Sutskever, I., Hinton, G.E.: "ImageNet classification with deep convolutional neural networks," in Advances in Neural Information Processing Systems (2012)
23. Wei, S.E., Ramakrishna, V., Kanade, T., Sheikh, Y.: "Convolutional pose machines," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-Decem, pp. 4724–4732 (2016). https://doi.org/10.1109/CVPR.2016.511
24. He, J., Zhang, C., He, X., Dong, R.: Visual Recognition of traffic police gestures with convolutional pose machine and handcrafted features. Neurocomputing **390**, 248–259 (2020). https://doi.org/10.1016/j.neucom.2019.07.103
25. Wang, S., et al.: Skeleton-based traffic command recognition at road intersections for intelligent vehicles. Neurocomputing **501**, 123–134 (2022). https://doi.org/10.1016/j.neucom.2022.05.107
26. Schneider, P., Memmesheimer, R., Kramer, I., Paulus, D.: "Gesture Recognition in RGB Videos Using Human Body Keypoints and Dynamic Time Warping," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 11531 LNAI, pp. 281–293, (2019). https://doi.org/10.1007/978-3-030-35699-6_22
27. Fang, H.S., Xie, S., Tai, Y.W., Lu, C.: RMPE: Regional Multi-person Pose Estimation. Proc. IEEE Conf. Comput. Vis. (2017). https://doi.org/10.1109/ICCV.2017.256
28. Cao, Z., Hidalgo Martinez, G., Simon, T., Wei, S.-E., Sheikh, Y.A.: "OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," IEEE Trans. Pattern Anal. Mach. Intell., (2019). https://doi.org/10.1109/tpami.2019.2929257.
29. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y : "Realtime multi-person 2D pose estimation using part affinity fields," Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017, vol. 2017-Janua, pp. 1302–1310 (2017). https://doi.org/10.1109/CVPR.2017.143
30. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: "End-to-End Recovery of Human Shape and Pose," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2018). https://doi.org/10.1109/CVPR.2018.00744
31. Liu, J., Akhtar, N., Mian, A.: "Skepxels: Spatio-temporal Image Representation of Human Skeleton Joints for Action Recognition," pp. 10–19, (2017), [Online]. Available: http://arxiv.org/abs/1711.05941
32. Lin, T.Y., et al : "Microsoft COCO: Common objects in context," Lect. Notes Comput. Sci.(including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), **8693**(5)740–755 (2014). https://doi.org/10.1007/978-3-319-10602-1_48
33. Singh, M., Mandal, M., Basu, A.: "Visual gesture recognition for ground air traffic control using the radon transform," 2005 IEEE/RSJ Int. Conf. Intell. Robot. Syst. IROS, pp. 2850–2855, (2005). https://doi.org/10.1109/IROS.2005.1545408
34. Blackett, C., Fernandes, A., Teigen, E., Thoresen, T.: Effects of Signal Latency on Human Performance in Teleoperations. Lect. Notes Networks Syst. **319**(August), 386–393 (2022). https://doi.org/10.1007/978-3-030-85540-6_50
35. He, K., Zhang, X., Ren, S., Sun, J.: "Deep residual learning for image recognition," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-Decem, pp. 770–778, (2016). https://doi.org/10.1109/CVPR.2016.90
36. Breiman, L.: "Random forests," *Random For.*, pp. 1–122, (2001), doi: https://doi.org/10.1201/9780367816377-11

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Miguel Ángel de Frutos is currently pursuing his Ph.D. in Automatic Control and Robotics at the Centro de Automática y Robótica, (UPM-CSIC). He completed his undergraduate studies in Aerospace Engineering and holds a Master of Science degree in Artificial Intelligence. His research interests lie in the areas of sensor fusion, unmanned aircraft systems, robotics, computer vision, and machine learning.

Fernando Carlos LópezHernández is a full-time associate professor at the Mathematical Analysis and Applied Mathematics Department at Universidad Complutense de Madrid (UCM). His current research interests lie in neural networks, dynamic systems, computer vision, statistical machine learning and data-driven science.

José Javier Rainer Granados is a full-time associate professor at School of Engineering and Technology, and Director of the Knowledge Transfer Office at Universidad Internacional de La Rioja (UNIR). His current research interests lie in robotics, cognitive systems, fuzzy logic and machine learning.