IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 23, 2021, accepted December 3, 2021, date of publication December 23, 2021, date of current version January 20, 2022.

Digital Object Identifier 10.1109/ACCESS.2021.3138167

E-Pilots: A System to Predict Hard Landing During the Approach Phase of Commercial Flights

DEBORA GIL[1], AURA HERNÀNDEZ-SABATÉ[1], JULIEN ENCONNIERE[2], SARYANI ASMAYAWATI[2], PAU FOLCH[3], JUAN BORREGO-CARAZO[1], AND MIQUEL ÀNGEL PIERA[4], (Member, IEEE)

[1]Computer Vision Center, Computer Science Department, Universitat Autònoma de Barcelona, Campus de la UAB, Bellaterra, 08193 Barcelona, Spain
[2]School of Aerospace Transport and Manufacturing, Cranfield University, Bedford MK43 0AL, U.K.
[3]Aslogic, Rubí, 08191 Barcelona, Spain
[4]Telecommunications and Systems Engineering Department, Universitat Autònoma de Barcelona, Campus de la UAB, Bellaterra, 08193 Barcelona, Spain

Corresponding author: Debora Gil (debora@cvc.uab.cat)

This work was supported in part by the Horizon 2020 (H2020) Clean Sky 2 Project ''E-PILOT: Evolution of cockPIt operations Levering on cOgnitive compuTing Services'' under Grant 831993, in part by the Spanish Project under Grant RTI2018-095209-B-C21, in part by the Generalitat de Catalunya under Grant 2017-SGR-1597 and Grant 2017-SGR-1624, and in part by the Centres de Recerca de Catalunya (CERCA)-Program.

ABSTRACT More than half of all commercial aircraft operation accidents could have been prevented by executing a go-around. Making timely decision to execute a go-around manoeuvre can potentially reduce overall aviation industry accident rate. In this paper, we describe a cockpit-deployable machine learning system to support flight crew go-around decision-making based on the prediction of a hard landing event. This work presents a hybrid approach for hard landing prediction that uses features modelling temporal dependencies of aircraft variables as inputs to a neural network. Based on a large dataset of 58177 commercial flights, the results show that our approach has 85% of average sensitivity with 74% of average specificity at the go-around point. It follows that our approach is a cockpit-deployable recommendation system that outperforms existing approaches.

INDEX TERMS Decision support systems, hard landing prediction, machine learning, neural networks.

I. INTRODUCTION

Between 2008-2017, 49% of fatal accidents involving commercial jet worldwide occurred during final approach and landing, and this statistic has not changed in several decades [1]. A considerable proportion of approach and landing accidents/incidents involved runway excursions, which has been identified as one of the top safety concerns shared by European Union Aviation Safety Agency (EASA) member states [2], as well as US National Transportation Safety Board and US Federal Aviation Administration [3].

According to EASA [2], there are several known precursors to runway excursions during landing. These include unstable approach, hard landing, abnormal attitude or bounce at landing, aircraft lateral deviations at high speed on the ground, and short rolling distance at landing. Some precursors can occur in isolation, but they can also cause the other precursors, with unstable approach being the predominant one. Boeing reported that whilst only 3% of approaches in commercial aircraft operation met the criteria of an unstable approach, 97% of them continued to landing rather than executing a go-around [4]. A study conducted by Blajev and Curtis [5] found that 83% of runway excursion accidents in their 16-year analysis period could have been avoided by a go-around decision. Therefore, making timely decision to execute a go-around manoeuvre could therefore potentially reduce the overall aviation industry accident rate [4].

A go-around occurs when the flight crew makes the decision not to continue an approach or a landing, and follows procedures to conduct another approach or to divert to another airport. Go-around decision can be made by either flight crew members, and can be executed at any point from the final approach fix point to wheels touching down on the runway (but prior to activation of brakes, spoilers, or thrust reversers). In addition to unstable approaches, traffic, blocked runway, or adverse weather conditions are other reasons for a go-around. Despite a clear policy and training on go-around policies in most airlines, operational data show that flight crew decision-making process in deciding for a go-around could be influenced by many other factors.

The associate editor coordinating the review of this manuscript and approving it for publication was Massimo Cafaro.

These include fatigue, flight schedule pressure, time pressure, excessive a head-down work, incorrect anticipation of aircraft deceleration, visual illusions, organizational policy/culture, inadequate training or practice, excessive confidence in the ability to stabilize approach, and Crew Resource Management issues [5]. It is for these reasons that on-board real-time performance monitoring and alerting systems that could assist the flight crew with the landing/go-around decision are needed [5], [6].

Such on-board systems could utilize the huge and ever-increasing amount of data collected from aircraft systems and the exponential advances in machine learning methods and artificial intelligence. EASA is anticipating a huge impact of machine learning on aviation, including helping the crew to take decisions in particular in high workload circumstances (e.g. go-around, or diversion [7]. Artificial Intelligence in aviation is considered one of the strategic priorities in the European Plan for Aviation Safety 2020-2024 [8].

Under the hypothesis that a hard-landing (HL) occurrence has precursors and, thus, it can be predicted, this paper presents a cockpit deployable machine learning system to predict hard landings considering the aircraft dynamics and configuration. In particular, this paper evaluates three main hypothesis. A primary hypothesis is to assess to what extend HL can be predicted at DH for go-around recommendation from the analysis of the variables recorded from FMS. A second hypothesis is to analyze if precursors are particular to aircraft types. A third hypothesis is to validate if the variability on the aircraft state variables can provide enough information to predict a HL regardless of the operational context (like environmental conditions and automation factors).

II. RELATED WORK

Although there is a lot of work addressing the prediction of flight landing incidents [9]–[12] and other unsafety situations [13]–[16], the prediction of hard landing accidents have been less researched. Furthermore, most of the existing works focus on the prediction of HL for unmanned aerial vehicles (UAV), which dynamical features and flying protocols are completely different from the ones of commercial flights.

A Hard Landing (HL) is a phenomenon in which the airplane has an excessive impact on the ground at the moment of landing. This impact is directly related to the vertical (or normal) acceleration, therefore, HL can be defined as flights where the vertical acceleration exceeds the limited value of the aircraft type during the landing phase. A threshold on such normal acceleration (Airbus uses vertical acceleration $> 2G$ at Touch Down, TD) triggers maintenance requirement, so that can be considered as a criterion for HL detection.

Under the former definition of HL, existing approaches for HL prediction can be split into two groups: those based on a classifier to discriminate flights with normal acceleration at TD above a given threshold from other flights and those based on a regressor that predicts the normal acceleration with the aim of using this predicted value as the HL detector.

Classifiers can be categorized into machine learning and deep learning approaches. Machine learning methods [17]–[19] apply a classifier to UAV flight data recorded using the Quick Access Recorder (QAR) sampled at a discrete set of heights that define the feature space. Most methods [17], [19] use the values of variables describing aircraft dynamics sampled between 9 and 2 meters before TD. Others, like [18], use statistical descriptors (panel data) of such variables also sampled at the very last meters before TD. On one hand, it is not clear what is the capability of these approaches to capture time-sequence dependencies that variables might have across the approach phase. On the other hand, the temporal window (9-2 meters before landing) used for predictions in UAV flights might not be appropriate for HL predictions in commercial flights. The approximate limit altitude (known as Decision Height -DH-) in commercial flights to decide a go around is 100 feet (38 meters). Thus, regardless of their accuracy in predicting HL, these ML methods are not applicable for commercial flights due to the altitude range required.

Deep learning approaches are mainly based on Long Short-Term Memory Recurrent Neural Network (LSTM) architectures. Proposed by [20], these networks are a variant of Recurrent Neural Networks (RNN) [21] able to model long term dependencies within temporal data. In particular, the very recent work in [22] used the signals of 3 kinds of landing related features (aircraft dynamics, atmospheric environment, and pilot operations) as inputs to a LSTM network predicting HL. Their comparison to classic machine learning approaches in terms of precision and recall of HL events of A320 flights indicates a potentially higher performance in terms of HL recall with 70% of HL detection while keeping with a percentage (76%) of precision similar to the one obtained by classic machine learning approaches. Despite the promising results, we consider that the experimental design of [22] lacks in some aspects for properly assessing the potential for deployment in the cockpit. First, the test set used is balanced with almost the same number of HL and non HL cases. However, in a real situation, HL cases are rare events that represent only 3-4% of flights. By balancing the test set, precision might be too optimistic and, even unrealistic. In order to guarantee a useful decision support system, the number of false alarms should be properly estimated. Second, the authors conducted an analysis that showed that the optimal temporal window for doing predictions was between 10 and 2 seconds before landing. This temporal window corresponds to heights between 164 and 13 feet, which are below the decision height (100 feet) of commercial flights. Finally, the data only include a single aircraft type (A320). Given that aircraft aerodynamics are strongly related to aircraft design, the generalisation of the approach remains unknown.

Regression approaches predicting normal acceleration are also mostly based on deep learning LSTM strategies [23], [24]. Both works use the values of a selection of QAR variables describing aircraft dynamics recorded at a time t to predict the vertical acceleration at time t + 1. In order to accelerate the convergence of networks, there is a previous selection of QAR variables using classic machine learning feature selection methods (aerodynamic theory and correlation analysis in the case of [23] and random forest followed by Principal Component Analysis in the case of [24]). This might be limiting the capability of the system for fully exploring time dependencies and might discard discriminative features. Although both works obtain accurate predictions with an average Mean Squared Error (MSE) of the order of 10^{-3}, LSTM is not trained to predict the vertical acceleration at TD at the next time interval after the current observation. In fact, a recurrent network can only predict acceleration at the immediate time interval from the current observation and its capability for long term predictions is not clear. Since HL depends on the values of such vertical acceleration in a tight temporal window at the time of TD, this limits the deployability of system in a cockpit.

A. CONTRIBUTIONS

This paper presents an analysis of approaches for early prediction of hard-landing events in commercial flights. Unlike previous works, experiments are designed to analyze to what extend methods can be deployable in the cockpit as go-around recommendation systems. With this final goal, we contribute to the following aspects:

1) **Hybrid model with optimized net architecture**. We propose a hybrid approach that uses features modelling temporal dependencies of aircraft variables as input to a neural network with an optimized architecture. In order to avoid any bias caused by a lack of convergence of complex models (like LSTM), we use a standard network and model potential temporal dependencies associated with unstable approaches as the variability of different types of aircraft variables at a selected set of altitudes. The concatenation of such variability for variables categorized into 4 main types (physical, actuator, pilot operations and all of them) are the input features of different architectures in order to determine the optimal subset.
2) **Exhaustive comparison to SoA in a large database of commercial flights.** A main contribution compared to existing works is that our models have been tested and compared to SoA methods on a large database of Flight Management System (FMS) recorded data of an airline no longer in operation that includes 3 different aircraft models (A319, A320, A321). Results show that the optimal classification network when all variable types are considered achieves an average recall of HL events of 85% with a specificity of 75% in average, which outperforms current LSTM methods found in the literature. Regarding regression networks, our hybrid model performs similarly to LSMT methods with an average MSE of the order of 10^{-3} in accelerations estimated at TD.
3) **Analysis of the performance of classifiers and regressors.** With the final goal of developing a cockpit deployable recommendation system we have conducted a study of the performance of classification and regression models in terms of the flight height and different aircraft variables including the impact of automation and pilot manoeuvres. Results on our large dataset of commercial flights, show that although our regression networks performs similarly to SoA methods (with MSE of 10^{-3} in estimations at TD), the accuracy for detecting HL is very poor (46% of sensitivity). This indicates that regression models might not be the most appropriate for the detection of HL events in a cockpit deployable support system.
4) **Sources of errors and capability for go-around recommendation.** Unlike previous approaches, we analyse the capability of networks for the detection of HL before the decision height, as well as, the influence of the operational context. We have also performed an analysis of the sources of errors, including selection of the best variable type, optimal altitude range used for predictions, biases due to aircraft type and capability of regressors for HL prediction.

The paper is organized as follows. Section 2 describes the methodology, including the description of variables, analysis of automation factors and network models. Section 3 reports the experiments conducted to assess the performance of models and error analysis. Section 4 discusses the results obtained and compares them to existing methods.

III. METHODS

A. DATASET DESCRIPTION

The authors have access to a large database of Flight Monitoring System (FMS) recorded data of an airline no longer in operation. This database has the following information:

- Fleet: A319/A320/A321.
- Various airports.
- 377,446 flights.
- 370 parameters available at various sampling frequencies.

Several primary criteria were defined to limit the data to what is considered meaningful for the hard landing predictions and the evaluation of the 3 hypothesis posed in this paper:

- All (A319/A320/A321).
- LHR - Heathrow Airport.
- Start of data: Final Approach Fix (FAF).
- End of data: 20 seconds after touch down.
- 58 parameters selected.

Heathrow airport was chosen as the sole airport to ease flight comparison and training of ML. Moreover, aircraft landing at Heathrow must follow a straight corridor further easing the landing comparison. This drops the number of available flights to 178,654. The data retrieved from the FMS starts at the FAF defined as 3 minutes before touching down and ends 20 seconds after touching down to capture the maximum G, labelled maxG, at touch down. A binary variable, labelled Wheel_on_Ground, was added to indicate the time of touch down when set to 1. Then, maxG was computed as the maximum value of Normal_acc_g in a window of $+/-5$ seconds around Touch Down (TD) time as the maximum time Wheel_on_Ground equals 1.

Parameters linked to characterizing unstable approaches are selected for the study. These parameters are linked to the aircraft dynamics (e.g. accelerations, rates, angle of attack), the position relative to the runway (glideslope and localizer), the aircraft configuration (landing gear state, control surfaces position) and the cockpit activity with the stick and throttle inputs. This reduces the number of raw parameters from 370 to 58. Additionally, dropouts and a significant amount of noise and data quantisation were identified. The poor data quality led to a reduction in the number of flights to approximately 58,177. Flights with maxG higher than the Mean plus 2x Standard Deviation of the normal acceleration at TD are classified as HL. This defines the threshold at 1.4037g and 2673 flights are flagged as HL. This represents approximately 4,6% of the total number of flights, which is consistent with the numbers reported [25].

The selected dataset allows to validate the 3 hypothesis posed in this paper. The temporal window always includes the decision height in order to validate to what extend the analysis of the aircraft dynamic state variables is enough for a go-around recommendation. The inclusion of the 3 types of aircraft allows to evaluate if HL precursors are particular to aircraft types, which is the second hypothesis of the paper. Finally, in order to validate the impact of environmental conditions (third hypothesis) data did not included the weather measurements rather its impact on the aircraft parameter features.

The selected parameters were recorded at sampling frequencies between 0.25 and 8 Hz. However, since pilots make decisions according to altitude, we resampled all numerical variables as a function of altitude. To do such a change of variables, we used a linear interpolation of the values sampled at the frequencies to obtain values sampled at a uniform sampling of altitudes.

The final set of selected parameters were split into four different categories: 1) **actuators**, linked to actuators states, 2) **pilot**, related to pilot activity in the cockpit, 3) **physical**, as those parameters related to physical magnitudes as well as other factors such as 4) **automation factors**, as those binary parameters indicate whether an automatic system or guidance is engaged. The final set of selected parameters is described in Table 1. Aircraft weight is not listed, as the parameter was deemed unreliable. Those parameters posteriori computed are indicated in the description.

B. IMPACT OF AUTOMATION FACTORS IN HL

In order to explore the impact of automation in HL, the correlation between maxG and the following pilot decision making variables: autopilot, flight director, speed break, landing gear, and autothrust is evaluated. Autopilot, autothrust and flight director are computed as the last time/ altitude they are engaged. Landing gear and speed break are computed as the time/altitude they are first engaged. To better explore the impact of the above factors in HL, the data has been split into hard landing (labelled HL) and non-hard landing (labelled NHL) events to detect any bias in the factors associated with HL.

Figure 1 shows the boxplots for the factors grouped according to their label. Notice that there are no significant differences between the values obtained in HL and NHL. Therefore automation factors do not seem to have an impact on the maxG and do not favour HL. Consequently, they will not be included in prediction models.

C. HL PREDICTION MODELS

A hard landing (HL) is defined as an event where vertical (or normal) acceleration exceeds a threshold value specific to the airplane type during the landing phase. A threshold on such normal acceleration (Airbus uses vertical acceleration $> 2G$ at touch down, TD) triggers maintenance requirement and, thus, can be considered as a criterion for HL detection. Under this criterion, a Machine Learning System (ML) for HL prediction could be a classifier to discriminate flights with normal acceleration at TD above a given threshold from other flights. However, the values of the normal acceleration at TD follow a continuous unimodal probabilistic distribution. This fact also suggests using a regressor to predict the normal acceleration at TD and use either its value or a threshold on it as the HL predictor. In this work, we have considered both approaches:

- *Regressors:* The dependent variable to be predicted is the maximum normal acceleration (labelled maxG) at TD. This variable is computed as the maximum value of Normal_acc_g in a window of ±5 seconds around TD time set as the maximum time Wheel_on_Ground = 1.
- *Classifiers:* We have considered a binary problem to classify hard landing (labelled HL) from non-hard-landing (labelled NHL). In our dataset flights with maxG > 1.4037 at TD are classified as HL.

For all ML methods (both regressors and classifiers) the input features are the concatenation of the variability of the continuous variables described in subsection III-A at a discrete set of flight altitudes which include the decision height, DH. The discrete sampling altitudes are $[1500, 1000, 500, 400, 300, 200, 150, 100, 50, 40, 30]$ and the decision height was set to 100 feet. The lower altitude of 30 feet was selected as the limit point the pilot can safely avoid a HL event.

For each variable, the variability is computed as the standard deviation in a temporal window of size $2w$ seconds centred at each sampling altitude. If $X(t)$ denotes a time-dependant aircraft variable and ta is the first time that the sampling altitude is achieved, then the variability at ta, noted $Xstd(ta)$ can be computed as the integral:

$$Xstd(ta) = \frac{1}{2w} \int_{-w}^{w} (X(ta-t) - \overline{X})^2 \, dt \tag{1}$$

TABLE 1. Set of selected parameters.

Category	Variable	Unit	Description
Physical Variables	Distance2TD	m	Computed distance to touch down
	Airspeed_kt	kt	Airspeed
	V_app_kt	kt	Target approach speed
	Deltaspeed2target_kt	kt	Delta Airspeed to target speed - computed
	Deltaspeed2ground_kt	kt	Delta Airspeed to ground speed - computed
	Normal_acc_g	G	Normal acceleration
	Long_acc_g	G	Longitudinal acceleration
	Lat_acc_g	G	Lateral acceleration
	Pitch_deg	deg	Pitch attitude
	Roll_deg	deg	Roll attitude
	Mag_hdg_deg	deg	Heading from runway direction - computed
	AoA_corr_deg	deg	Calibrated angle of attack is computed from AoA_LH_deg
	EPR_1	-	Engine Pressure Ratio 1
	EPR_2	-	Engine Pressure Ratio 2
	Localiser_dots	dot	Localiser deviation
	Glideslope_dots	dot	Glideslope deviation is computed
	Drift_deg	deg	Difference between heading and track
Pilot	Sidestick_pitch	deg	Total longitudinal sidestick input (1+2) - computed
	Sidestick_roll	deg	Total lateral sidestick input (1+2) - computed
	THR_lever_1	deg	Thrust lever angle engine 1
	THR_lever_2	deg	Thrust lever angle engine 2
Actuators	Flap_lever_pos	1/2/3/4	Flap lever position
	Flap_angle_deg	deg	Flap position
	Slat_angle_deg	deg	Slat position
	Rudder_deg	deg	Rudder position
	Elev_pos_LH_deg	deg	Elevator position (LH)
	Elev_pos_RH_deg	deg	Elevator position (RH)
	Aileron_pos_LH_deg	deg	Aileron position (LH)
	Aileron_pos_RH_deg	deg	Aileron position (RH)
Automation Factors	AP_engaged	0/1	Autopilot operation computed
	Spd_brk_cmd	0/1	Speed break position (0: not engaged, 1: engaged)
	Gear_sel_down	0/1	Landing gear position (0: up, 1: down)
	AT_engaged	0/1	Auto thrust operation
	FltDir_eng	0/1	Flight Director operation is computed

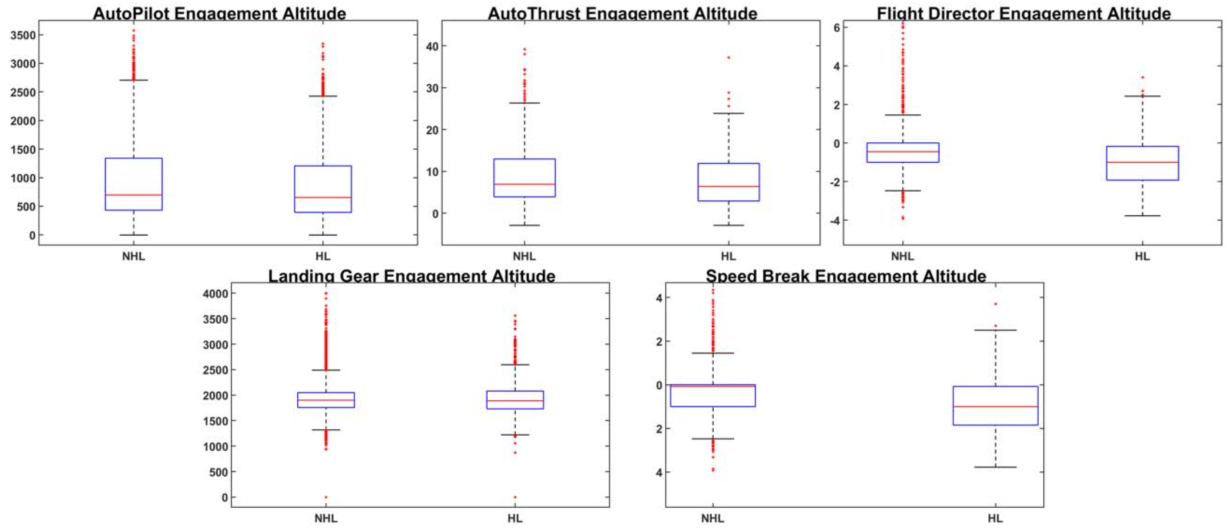

FIGURE 1. Impact of automation factors.

where $\overline{X}$ denotes the expected value in the temporal window:

$$\overline{X} = \frac{1}{2w}\int_{-w}^{w} X(ta - t)\,dt \tag{2}$$

Since, according to subsection III-B, the automation factors did not have any impact on neither maxG values nor HL, they were, thus, not considered in the training of the models. In order to account for the difference in units and magnitudes, variables were normalized to follow a Gaussian normal distribution of 0 mean and standard deviation 1. The normalization parameters were learned from the training set.

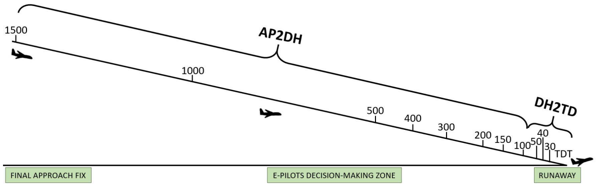

FIGURE 2. Altitude sampling.

TABLE 2. Network models.

	AP2TD	AP2DH	DH2TD
Physical	187	136	51
Actuator	88	64	24
Pilot	44	32	12
All	341	248	93

In order to effectively deploy the ML in a cockpit as a decision support system it is essential to obtain accurate predictions as soon as possible to deliver instructions at the most appropriate time. In particular, methods should be able to predict HL before the decision height, DH. In order to explore the full capability of the ML systems, as well as the capability for early detection of HL, variables were categorised into 3 groups of altitude ranges:

- *AP2TD Range:* This includes all sampling altitudes, from the beginning of the approach phase to touchdown. Models trained with this set of altitudes set the maximum accuracy that the system can achieve.
- *AP2DH Range:* This includes altitudes from the beginning of the approach phase to the decision height: [1500, 1000, 500, 400, 300, 200, 150, 100]. Models trained with this set of altitudes set the actual capability for HL early detection and the usefulness of the system for a go-around recommendation.
- *DH2TD Range:* This set includes altitudes from the decision height to 30 feet before touchdown: [50, 40, 30]. Models trained with this altitude range will assess the capability to predict HL just in time to safely avoid it.

Figure 2 shows the 3 ranges of altitude sampling. The range from the beginning of the approach phase to the decision height takes the sampling from 1500 to 100 meters, while the sampling from decision height to touch down only contains 3 samples, although they are closer together. As well, the range from approach phase to touch down is also considered.

A different network was trained for each variable category (Physical, Actuator, Pilot) and range of altitudes (AP2TD, AP2DH and DH2TD). We also trained a model having as input the concatenation of the 3 categories. This model was labelled as AllÂI. Table 2 reports the dimensionality of each of the networks input features for the 9 models considered, as well as the concatenation of all of them.

Regarding the neural networks architectures, several configurations were explored. We considered the same architectures for, both, the regressor and the classification networks. For each case, we have implemented architectures increasing dimensionality as well as architectures reducing it. The number of layers was kept relatively low, since according to the literature [23], very deep architectures do not significantly improve results. The number of neurons per layer was varied from a low number to a large one (including architectures with several neurons linked to the variable category dimensionality, noted dim). Table 3 summarizes the different architectures that have been considered, each vector contains the number of neurons for each hidden layer of the network together with the label that will be used, from now on, to reference them.

For classification networks, we used a softmax activation function for the output of the last layer, a cross-entropy loss and a balanced class sampling for training. Meanwhile, for regression networks, we used a linear activation function for the output layer, the quadratic error as loss function and no class balancing for training.

IV. EXPERIMENTS

A. EXPERIMENTAL DESIGN

The performance of the different approaches for detection of HL events was assessed using sensitivity and specificity measures, which are common metrics in classification assessment. The sensitivity measures the capability of the system to detect HL events, while the specificity measures the capability for detection NHL.

Let us note TP the number of true positives (i.e. HL correctly detected by the system), FP, the number of false positives (NHL detected as HL by the IA system), TN the number of true negatives (NHL detected by the system) and FN the number of false negatives (HL missed by the system), then sensitivity and specificity are given by equations in (3) and (4).

$$Sensitivity = \frac{TP}{TP + FN} \tag{3}$$

$$Specificity = \frac{TN}{TN + FP} \tag{4}$$

For regression models, we have considered the mean squared error (MSE) of the predicted maxG:

$$MSE = \frac{1}{N} \sum_i (maxG_i - \widehat{maxG_i})^2 \tag{5}$$

for $\widehat{maxG_i}$ the values of $maxG$ predicted by the regressor for the i-th sample of the test set, $maxG_i$ the true values of $maxG$ for the i-th test sample and N the size of the test set. Since a sample is considered a HL if the maxG predicted by the model is above 1.4037, in order to assess to what extent a good regression also ensures good detection of HL, the predicted maxG was binarized with a threshold equal to 1.4037 to also compute sensitivity and specificity.

For all models, the optimization of the loss functions was done with the scaled conjugate gradient back-propagation with default parameters sigma = 5e-5 and lambda = 5e-7. The batch size was equal to the number of training samples and training was stop if the performance gradient was below 1e-06 for 6 epochs.

TABLE 3. Architecture of the neural networks.

Config1	Config2	Config3	Config4
[20,20,30]	[10,10,20,20]	[20,20,30,30]	[dim,dim,dim*2]
Config5	Config6	Config7	Config8
[16,32,64,128]	[dim, dim*2, dim*4]	[dim,dim,round(dim/2)]	[128,128,64]

We followed a 15-fold approach to split data into training and test to allow statistical comparison of methods performance. To account for data unbalancing, a different fold was computed for each class and NHL train fold was randomly sub-sampled to the size of the HL training fold set. The testing fold was not altered to measure performance in a realistic setting. The number of samples of each train fold is 6066 (half of them being HL cases) and the size of each test fold is 3280 with around 100 being HL. For each score, confidence intervals (CI) at 95% of confidence were computed in order to detect significant differences. Significance in performance across factors and methods was further assessed using a Student t-test for paired data in case of pairwise comparisons and an analysis of variance (ANOVA) in case of multiple comparisons. A p-value under 0.01 was considered significant.

The following experiments have been conducted:

1) *Predictive Power of Models:* Optimal architectures were chosen as the ones that achieved better quality scores (sensitivity, specificity for classifiers and MSE for regressors) in training. The optimal regression neural network is compared to the optimal classification nets in terms of sensitivity, specificity in testing.
2) *Cockpit Deployable Potential:* In order to assess to what extent models can be effectively deployed in the cockpit, we have analyzed their performance according to the categorization of variables to determine the minimum set of variables and according to the altitude ranges to assess their capability for early detection of HL and for recommending a go-around.

B. RESULTS

1) PREDICTIVE POWER OF MODELS

Boxplots in figures 3 and 4 show, respectively, sensitivity and specificity for classification networks and boxplots in figure 5 show MSE for regression networks grouped according to network architecture. We show a different boxplot for each altitude range and variable type.

Visual analysis of the boxplots for sensitivity indicate that all architectures seem to perform equally for models trained using the 3 categories of variables for any altitude range. For models trained using the concatenation of all variables, Config5 and Config7 could perform worst for some models. This is confirmed by an ANOVA test which detects a significant lower sensitivity of Config5 and Config architectures for all altitude ranges.

Visual analysis of the boxplots for specificity indicate that all architectures seem to perform equally for models seem to perform equally for models trained using the Pilot and Actuator variables for any altitude range. ANOVA test confirms this fact and the multicomparison for the remaining cases show that Config1, Config3, Config4 and Config6 are significantly worse in all altitude ranges when either Physical or All variables are considered.

Visual analysis of the boxplots for MSE indicates that Config6 and Config4 are the worst performers for most models (with the exception of AP2DH for Pilot variables and DH2TD for Actuators), but Config5 and Config8 also show limitations (for all models using DH2TD ranges and AP2TD using Actuators). ANOVA detects a significant higher MSE for Config6 in all cases, Config4 in AP2DH and DH2TD ranges when Physical and All variables are used and, Config5 in DH2TD ranges and Config8 in DH2TD ranges when Pilot and Physical variables are used. Although it is not significant, Config7, Config2 and Config3 seem to have lower MSE for all cases.

Tables 4 and 5 report, respectively, the confidence intervals of sensitivity and specificity in testing data for Config6, which is one of the best classification networks. The variable categories correspond to columns, while the altitude ranges correspond to rows. The average detection of HL events reaches 85% when either physical or all variables are considered, with average specificity of 72%.

Table 6 reports MSE confidence intervals in the test set for Config2, one of the best architectures for regression models. Config7 achieves its optimal MSE when only physical variables are considered in the range DH2TD with average MSE of 2.6×10^{-3}. Like classification networks, an increasing number of layers and neurons does not seem to improve performance.

Tables 7 and 8 report confidence intervals for sensitivity and specificity for a classification obtained by thresholding the maxG estimated by Config2 regression network according to the limit set for HL. In spite of an average MSE of 2.6×10^{-3}, the sensitivity is quite low with an average of 46% at most.

2) COCKPIT DEPLOYABLE POTENTIAL

The analysis of Tables 4-6 indicates that the performance of models (both, classifiers and regressors) depends on the type of aircraft variable used to train models. Figure 6 show boxplots for sensitivity and specificity for the classifier and MSE for the regressor grouped according to the 4 types of variables. For both approaches, we show results for models trained with altitudes in the range AP2TD. Boxplots for sensitivity clearly show a better performance of models trained with Physical and All variables, while boxplots for specificity indicate a worse performance of these variables. Finally, boxplots for

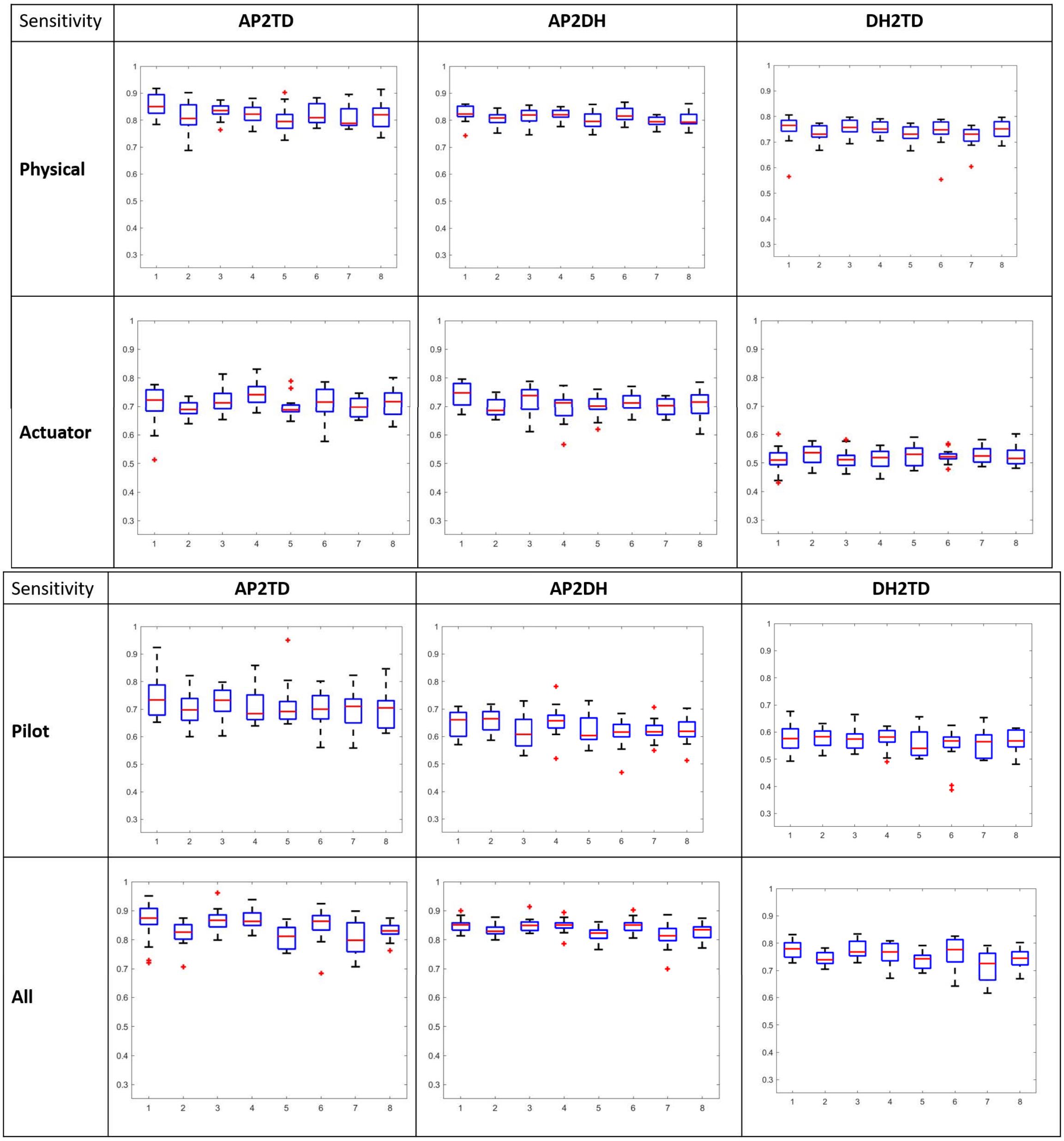

FIGURE 3. Sensitivity for classification networks.

TABLE 4. Confidence intervals of sensitivity for the classification network with Config6 architecture.

	Physical	Actuator	Pilot	All
AP2TD	[0.7950, 0.8850]	[0.6710, 0.7421]	[0.6700, 0.7385]	[0.8187 , 0.8845]
AP2DH	[0.8034, 0.8359]	[0.6967, 0.7345]	[0.5840 , 0.6409]	[0.8356 , 0.8622]
DH2TD	[0.7078, 0.7721]	[0.5114, 0.5363]	[0.5112, 0.5862]	[0.7334 , 0.7924]

TABLE 5. Confidence intervals of specificity for the classification network with Config6 architecture.

	Physical	Actuator	Pilot	All
AP2TD	[0.7110, 0.7332]	[0.7714, 0.8035]	[0.7372 , 0.7828]	[0.7182 , 0.7313]
AP2DH	[0.7029, 0.7161	[0.7822 , 0.7931]	[0.7475 , 0.7739]	[0.7164 , 0.7311]
DH2TD	[0.6397 , 0.6507]	[0.6620 , 0.6809]	[0.6457 , 0.6804]	[0.6515 , 0.6653]

MSE indicate that actuators and pilot related variables perform worse.

According to ANOVA test, differences are clearly significant for, both, sensitivity, specificity and MSE with

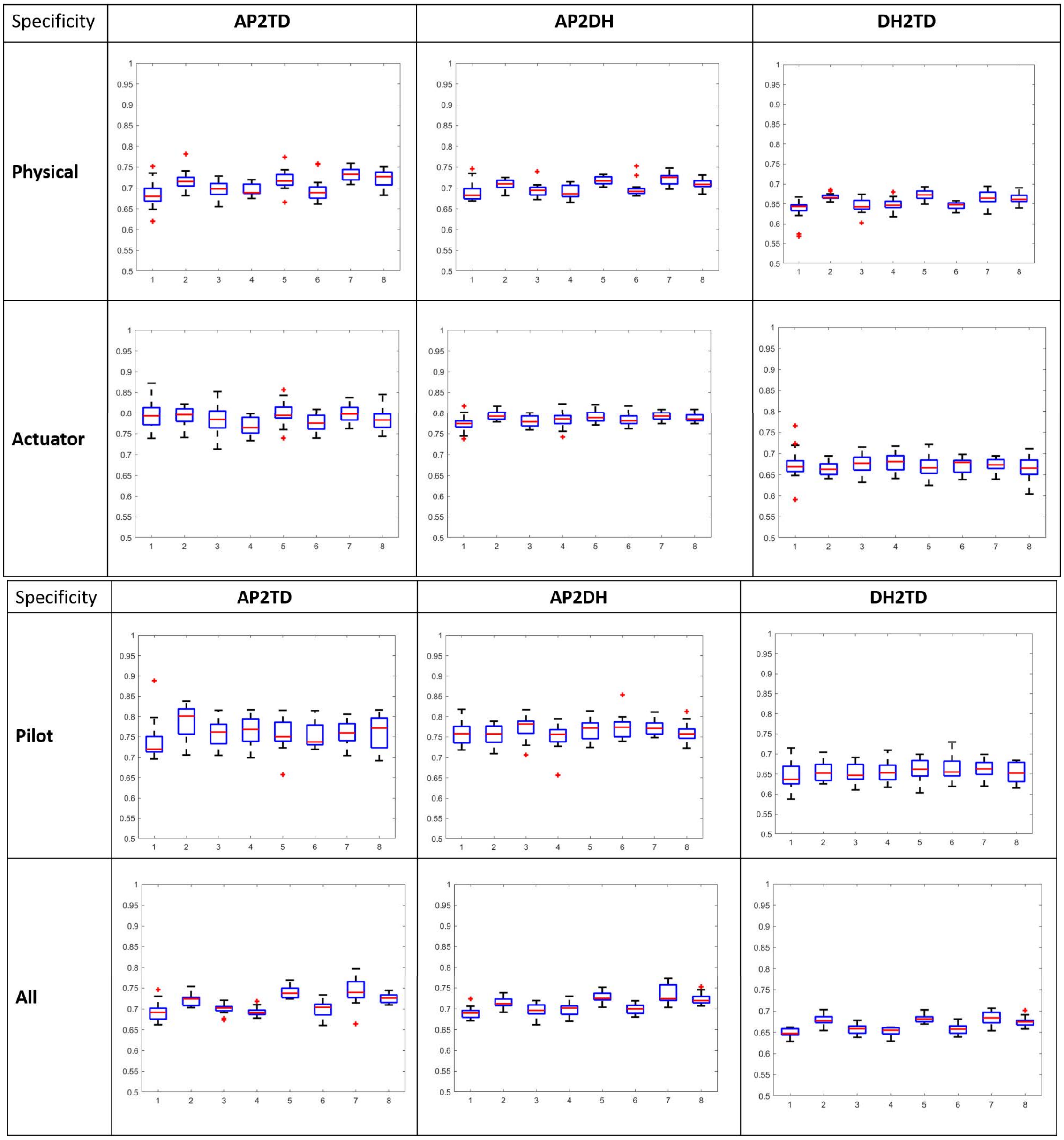

FIGURE 4. Specificity for classification networks.

TABLE 6. Confidence intervals of MSE for the regression network with Config2 architecture.

	Physical	Actuator	Pilot	All
AP2TD	[0.0022 , 0.0037]	[0.0067, 0.0069]	[0.0067, 0.0070]	[0.0021, 0.0040]
AP2DH	[0.0053, 0.0064]	[0.0067, 0.0070]	[0.0067, 0.0072]	[0.0058, 0.0067]
DH2TD	[0.0025, 0.0027]	[0.0068, 0.0071]	[0.0067, 0.0070]	[0.0023, 0.0034]

TABLE 7. Confidence intervals of sensitivity for regression neural network with Config2 architecture.

	Physical	Actuator	Pilot	All
AP2TD	[0.3751, 0.5072]	[0.0128, 0.0243]	[0.0126, 0.0260]	[0.3290, 0.4661]
AP2DH	[0.0646, 0.1179]	[0.0115, 0.0320]	[0.0097, 0.0243]	[0.0507, 0.1034]
DH2TD	[0.4439, 0.4836]	[0.0075, 0.0155]	[0.0096, 0.0162]	[0.4161, 0.4594]

p-val < 0.01. A multi-comparison across variables detects that Physical variables are the ones that have the highest sensitivity for predicting HL, at the cost of a decrease in specificity. Average differences for sensitivity are

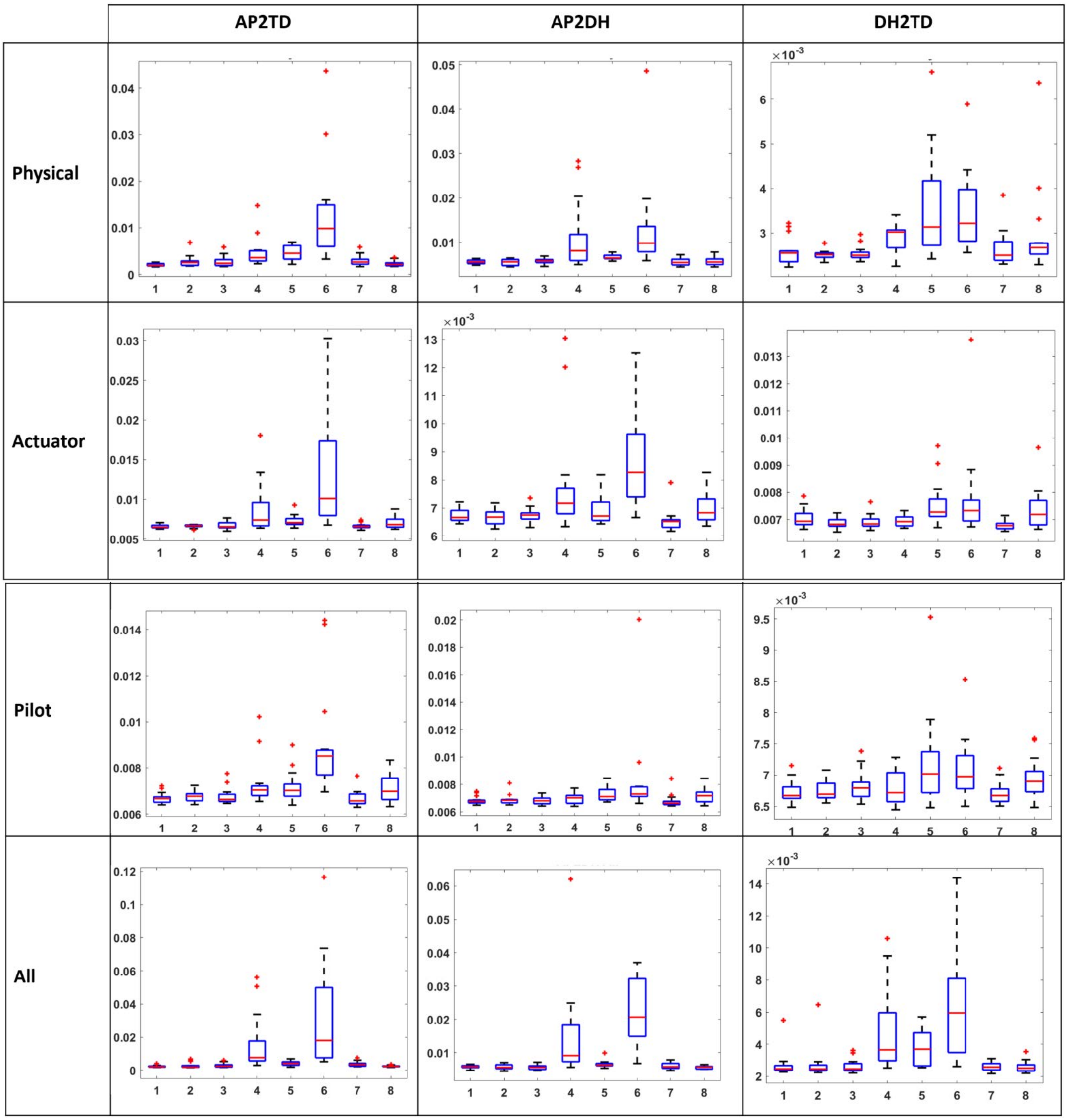

FIGURE 5. MSE for regression networks.

TABLE 8. Confidence intervals of specificity for regression neural network with Config2 architecture.

	Physical	Actuator	Pilot	All
AP2TD	[0.9778, 0.9842]	[0.9915, 0.9959]	[0.9918, 0.9957]	[0.9823, 0.9865]
AP2DH	[0.9882, 0.9961]	[0.9916, 0.9961]	[0.9900, 0.9970]	[0.9928, 0.9951]
DH2TD	[0.9720, 0.9789]	[0.9943, 0.9979]	[0.9963, 0.9976]	[0.9723, 0.9836]

$[6.97\%, 14.36\%]$ with only $[-8.21\%, -4.85\%]$ drop in specificity. Finally, in the case of regression, MSE Physical variables perform better than Actuators and Pilot, which do not have a significant difference in average errors. The combination of all categories by straight concatenation of features does not significantly improve the performance of models trained with the Physical variables alone in any of the approaches.

Regarding the impact of altitudes, boxplots for sensitivity shown in Figure 7 do not present any clear difference in performance across the 3 altitude ranges. However, ANOVA was significant for both measures. A multi-comparison test detected that, both, sensitivity and specificity for AP2DH and AP2TD altitudes were comparable and significantly better than the one obtained for DH2TD altitudes. The confidence intervals for the differences in sensitivity and specificity between AP2TD and DH2TD were, respectively, [10.03 %, 13.44%] and [7.32%, 9.45%]. This drop in sensitivity is not unexpected given the dynamic nature of a landing and the fact that a stable approach can quickly turn unstable due to

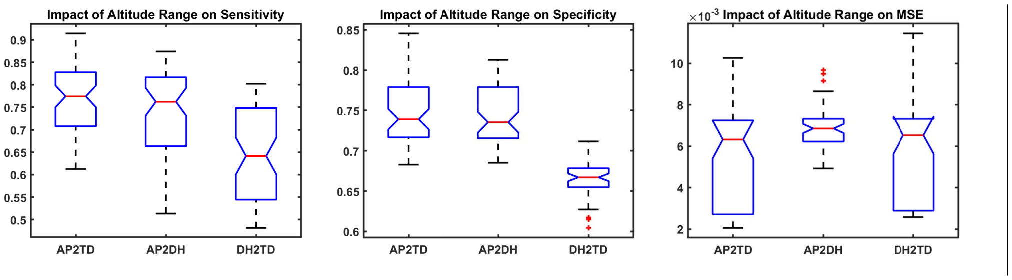

FIGURE 6. Impact of variable type on performance.

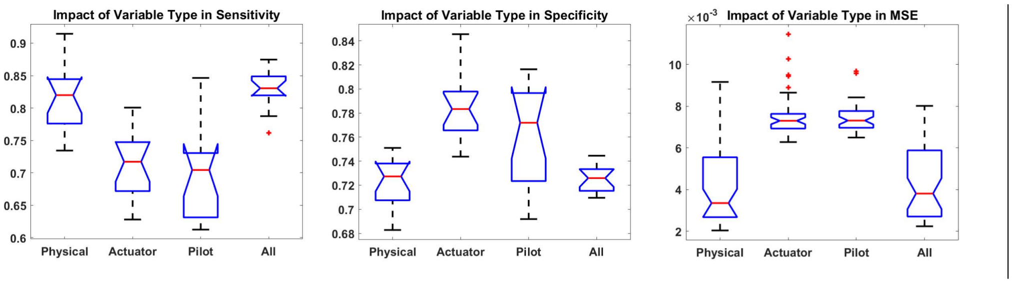

FIGURE 7. Impact of altitude range on performance.

changes in environmental conditions close to TD (like sudden tail win).

In the case of the regressor MSE, ANOVA detected significant differences between the range AP2DH and the ones that used data until TD. This indicates that regressors might only accurately predict maxG if data close to TD is taken into account. This together with their poor performance for actually detecting HL events, discards regression models as the approach to use in a cockpit deployable system for early detection of HL.

V. DISCUSSION

A. COMPARISON TO EXISTING METHODS

We have compared our method when all variables are considered to the LSTM model of [22] and two typical models (Support Vector Machine (SVM) and Logistic Regression (LR)) also reported in [22].

We have re-trained from scratch LSTM, SVM and LR in our data set using the variables and metrics proposed in our study. Following the same procedure as in [22], we build a LSTM network with one fully connected layer for classification, and train it using 9 sampled seconds of data from second 2 to 10 before TD. As there is no indication for the values of hyperparameters in the aforementioned work, we manually tuned the batch size and learning rates to 8 and 0.0001, respectively. We used an Adam optimizer and train for 55 epochs. To be able to handle overfitting, at each fold we divided the training set into training and validation using 5% of training data, and saved the model only when the validation loss decreases. As in the original study the authors do not use any regularization term, we also avoided using one.

We fine-tuned the number of neurons of the LSTM by performing a 15-fold grid search over the same values as in the mentioned study, [20, 30, 40, 50, 60], and obtain metric values over the validation set. Finally, once we have selected the best performing value, we perform 15 fold training for the specific value and test it on the test set, obtaining the definitive results. The SVM kernel was also optimized using grid search. LR has not any hyperparameters.

Barplots in figure 8 graphically compare average specificity and sensitivity achieved by our method at the 3 ranges of altitudes, the LSTM model of [22], SVM and LR. For the AP2TD, AP2DH altitude ranges our method has a sensitivity 5% higher than the best performer LSTM. Regarding specificity, AP2TD, AP2DH have average precision 7.7% higher than LSTM.

Regarding regression models, the MSE error achieved by our model in training and testing is comparable to the one obtained by the LSTM model reported in [23]. However, in spite of an error in maxG estimation of 2.6×10^{-3} in testing the regressor performs poorly in the detection of HL events with a 46% sensitivity. Such poor performance, can be attributed to two main issues. First, MSE is of the order of the squared absolute error, so $MSE = 2.6 \times 10^{-3}$ corresponds to a deviation in maxG predictions of $\sqrt{MSE} = \pm 0.05$. Second, we have observed (as the plots in 9 illustrate) that the regression error (residuals) is a decreasing function of maxG. This implies that models are underestimating maxG for the HL class and discourages us from using the prediction of maxG to detect HL events.

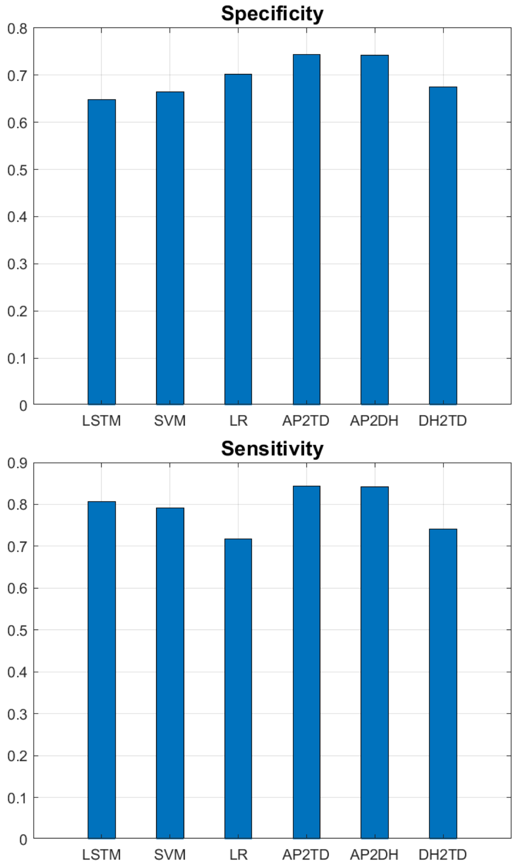

FIGURE 8. Accuracy results of different models.

B. IMPROVEMENTS

The proposed models only use the selected parameters on the final approach (below 2000 ft above ground). Wind direction and amplitude, the level of turbulence and the risk of gusts can have a significant impact on the possibility of an unstable approach resulting into a hard landing. The current analysis capture some of these effects as the variability the aircraft physical parameters are directly linked to the aforementioned weather conditions. Low visibility conditions, such as fog, and icing conditions can also impair the quality of the landing thus increasing the risk of hard landings but they are not considered in the study. The distance of flight has no specific influence on the results but the quality of the approach before reaching the final straight will have a significant impact. The aircraft might not be correctly configured or still have a significant level of energy to dissipate. This can be the result of ATC commands such as delayed descent instructions.

Another issue to be considered is that models did not include some key parameters that could have an impact in predictions.

First, aircraft weight (mass) was missing in the study because the aircraft mass dataset from the flight database is unreliable. However, aircraft weight has several potential impacts:

- The potential energy needed to be dissipated to land is directly proportional to aircraft weight. In other words, the heavier the aircraft, the more energy needs to be dissipated to land at an acceptable speed and descent rate.
- Similarly, the energy that the landing gear must dissipate at TD is also proportional to the potential energy.
- The aircraft weight will have an impact on the aircraft dynamics. The aircraft inertias are directly linked to the aircraft weight. The aircraft centre of gravity is another parameter impacting aircraft stability and controllability and thus takeoff and landing performance. That is the reason why an aircraft will have load planner as well as in-flight centre of gravity target system with a trim tank to maintain it within operational ranges. The more forward the centre of gravity is, the higher the minimum speed, the higher the landing speed. This is mitigated by the presence of a stability augmentation system found on aircraft such as the Airbus A320 studied herein. The centre of gravity is not normally known but the weight is.

Secondly, the aircraft centre of gravity, or the point at which the total weight of the aircraft is centred, is also a key parameter in the vehicle stability and control margins calculation which can greatly impact aircraft take-off and landing performance. Although commercial aircraft have trim-tank with centre of gravity target systems, the centre of gravity can vary within a range of certified positions within its airworthiness requirements. Therefore, including the centre of gravity and mass within the model could substantially improve the accuracy of the hard landing prediction.

The machine learning approach can also be improved in several aspects. Although results appear superior to existing methods, our models would benefit from a more complex analysis of temporal dependencies using a convolutional neural network to extract deep dependencies. The impact in predictions of meteorological conditions affecting visibility or aircraft aerodynamics should also be investigated to assess the benefits of their incorporation into our models. Given that the combination of all categories by straight concatenation of features does not significantly improve the performance of models trained with any single category, alternative architectures for their combination should be further investigated. Finally, the percentage of HL due to condition changes at TD should be determined to properly assess the capability of systems for early prediction of HL.

Finally, for a cockpit-deployable machine learning system to support flight crew go-around decision, some results regarding the hardware and software requirements, especially for the speed of networks should be investigated. Although such aspects have not been directly addressed in the main corpus of the work, given the size (number of parameters) and nature of the models (fully connected models), we do not expect any hardware or software constraints with regards to latency in the deployment in a cockpit environment. The deployment of fully connected networks is already available even for low resource microcontrollers [26] and the latency in such cases [27], and with similar models as ours, is below 50 ms o 1 s, which are our main sampling rates. Hence,

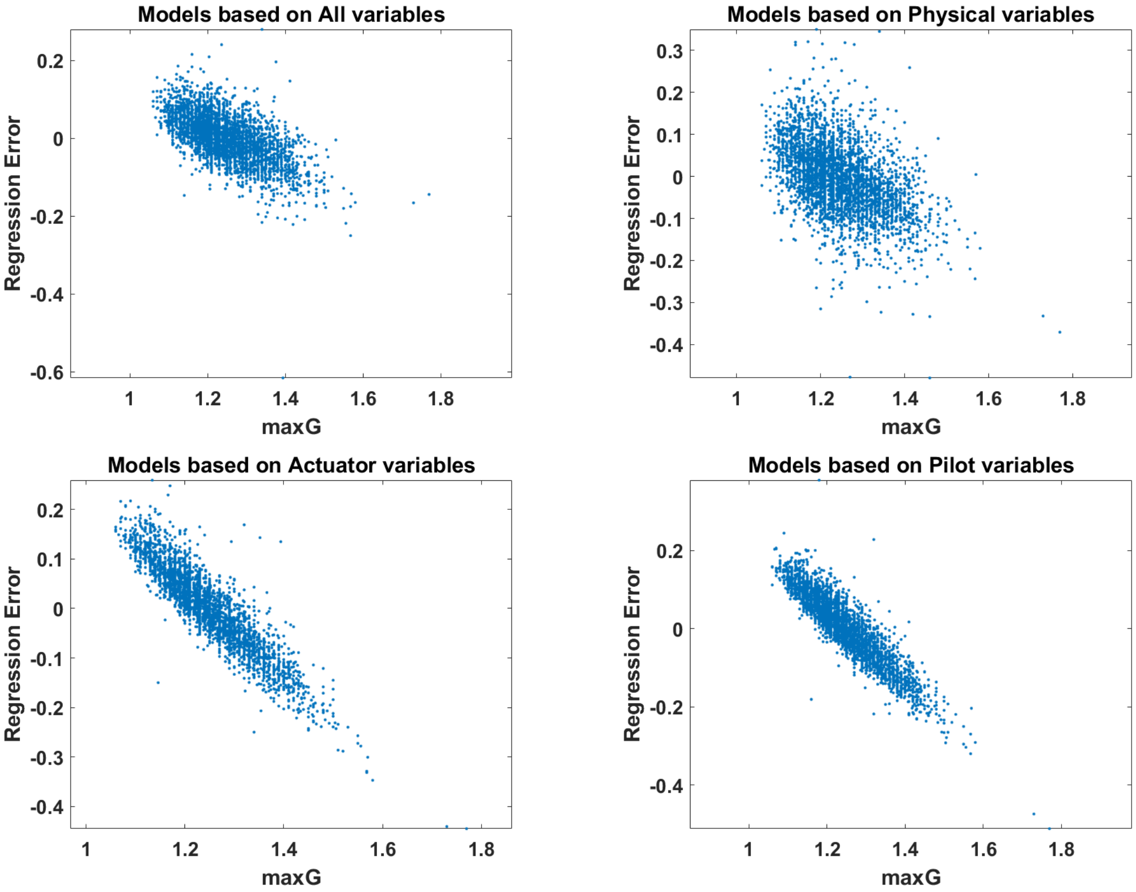

FIGURE 9. Point clouds of MaxG against regression error for the different variable types.

deployment software and latency are not considered as strong impediments for the future deployment in a cockpit.

VI. CONCLUSION

The following conclusions can be extracted from the analysis carried out in this paper.

The analysis of automation factors (autopilot, flight director and auto-thrust) suggests that these factors do not have any influence on the probability of a HL event and, thus, it might not be necessary to incorporate them into models.

Experiments for the optimization of architectures show that the configurations that achieve higher sensitivity are the ones with the lowest number of neurons. As reported in the literature [23] increasing the number of layers and neurons does not improve the performance of neither classifiers nor regressors.

Models using only Physical variables achieve an average recall of 94% with a specificity of 86% and outperform state-of-the-art LSTM methods. This brings confidence into the model for early prediction of HL in a cockpit deployable system. Regarding capability for go-around recommendation before DH, even if we perform better than existing methods, there is a significant drop in recall and specificity due to the dynamic nature of a landing approach and factors influencing HL close to TD.

Comparing classifiers and regression approaches, experiments show that a low MSE error in estimation of maxG does not guarantee accurate HL predictions. Experiments for assessing the capability of models for early detection of HL show that classifiers are able to accurately predict HL before DH. This is not the case of regressors, which predict maxG more accurately if data close to TD is considered into the model. The study suggests that classifiers are a better approach for early prediction of hard landing.

Neural networks performance could be increased if they were used to extract deep learning features from continuous signals by using one dimensional convolutional networks and different architectures for a better combination of the three categories of variables. Also, models should incorporate additional parameters such as aircraft mass and centre of gravity position which are known to impact vehicle dynamics.

Finally, there are some issues that have not been covered in this work, that remain as future work, and should be further developed. Among such cases, stand out the robustness of the classifier (regressor) to unseen cases and its behavior under a drifting data environment. In a safety demanding environment as aviation, it surely be needed to investigate such issues and we expect to do in further works. In the future, such a system could be expanded to also include Air Traffic Management in which the information is shared with the Air

Traffic Controller in order to anticipate the likely scenario and optimize runway use.

ACKNOWLEDGMENT

The authors wish to thank Dr. David Barry of Cranfield University for providing them with the flight data and for his technical support throughout this work. The author Debora Gil would like to dedicate this work to her mother, Esther Resina Enfedaque. DGil is a Serra Hunter Fellow. Opinions expressed in this article reflect the authors' views only.

REFERENCES

[1] *Statistical Summary of Commercial Jet Airplane Accidents–Worldwide Operations| 1959–2017*, Boeing Commercial Airplanes, Aviation Saf., Seattle, WA, USA, 2018.
[2] "Developing standardised FDM-based indicators," Eur. Aviation Saf. Plan 2012-2015, Cologne, Germany, 2016.
[3] "Advisory circular ac no: 91-79a mitigating the risks of a runway overrun upon landing," Federal Aviation Admin., Washington, DC, USA, 2016.
[4] M. Coker and L. S. Pilot, "Why and when to perform a go-around maneuver," *Boeing Edge*, vol. 2014, pp. 5–11, 2014.
[5] T. Blajev and W. Curtis, "Go-around decision making and execution project: Final report to flight safety foundation," Flight Saf. Found., Alexandria, VA, USA, Mar. 2017.
[6] "European action plan for the prevention of runway excursions," Eurocontrol, Brussels, Belgium, 2013.
[7] "Artificial intelligence roadmap—A human-centric approach to ai in aviation," Eur. Union Aviation Saf. Agency, Cologne, Germany, 2020.
[8] "The European plan for aviation safety (EPAS 2020–2024)," Eur. Union Aviation Saf. Agency, Cologne, Germany, 2019.
[9] L. Wang, C. Wu, and R. Sun, "Pilot operating characteristics analysis of long landing based on flight QAR data," in *Proc. Int. Conf. Eng. Psychol. Cognit. Ergonom.* Berlin, Germany: Springer, 2013, pp. 157–166.
[10] L. Li, J. Hansman, R. Palacios, and R. Welsch, "Anomaly detection via a Gaussian mixture model for flight operation and safety monitoring," *Transp. Res. C, Emerg. Technol.*, vol. 64, pp. 45–57, Mar. 2016.
[11] M. Miwa, T. Tsuchiya, S. Yonezawa, N. Yokoyama, and A. Suzuki, "Real-time flight trajectory generation applicable to emergency landing approach," *Trans. Jpn. Soc. Aeronaut. Space Sci.*, vol. 52, no. 175, pp. 21–28, 2009.
[12] G. Holmes, P. Sartor, S. Reed, P. Southern, K. Worden, and E. Cross, "Prediction of landing gear loads using machine learning techniques," *Struct. Health Monitor.*, vol. 15, no. 5, pp. 568–582, Sep. 2016.
[13] D. Zhou, X. Zhuang, H. Zuo, H. Wang, and H. Yan, "Deep learning-based approach for civil aircraft hazard identification and prediction," *IEEE Access*, vol. 8, pp. 103665–103683, 2020.
[14] F. M. Vieira, C. de Oliveira Bizarria, C. L. Nascimento, and K. T. Fitzgibbon, "Health monitoring using support vector classification on an auxiliary power unit," in *Proc. IEEE Aerosp. Conf.*, Mar. 2009, pp. 1–7.
[15] Y. Luo and X. Zhang, "Aircraft electronic board fault detection based on infrared thermal imaging and integrated SVM," *Meas. Control Technol.*, vol. 12, p. 12, 2012.
[16] A. O. Alkhamisi and R. Mehmood, "An ensemble machine and deep learning model for risk prediction in aviation systems," in *Proc. 6th Conf. Data Sci. Mach. Learn. Appl. (CDMA)*, Mar. 2020, pp. 54–59.
[17] S. Qian, S. Zhou, and W. Chang, "An improved aircraft hard landing prediction model based on panel data clustering," in *Proc. 29th Chin. Control Decis. Conf. (CCDC)*, May 2017, pp. 438–443.
[18] L. Zheng, J. Xie, and S. Qian, "Risk prediction method of aircraft hard landing based on flight data," in *Proc. Saf. Rel.-Safe Soc. Changing World, 28th Int. Eur. Saf. Rel. Conf. (ESREL)*, 2018, pp. 1827–1832.
[19] C. Hu, S.-H. Zhou, Y. Xie, and W.-B. Chang, "The study on hard landing prediction model with optimized parameter SVM method," in *Proc. 35th Chin. Control Conf. (CCC)*, Jul. 2016, pp. 4283–4287.
[20] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[21] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Oct. 2015.
[22] H. Zhang and T. Zhu, "Aircraft hard landing prediction using LSTM neural network," in *Proc. 2nd Int. Symp. Comput. Sci. Intell. Control (ISCSIC)*, Sep. 2018, pp. 1–5.
[23] C. Tong, X. Yin, J. Li, T. Zhu, R. Lv, L. Sun, and J. J. P. C. Rodrigues, "An innovative deep architecture for aircraft hard landing prediction based on time-series sensor data," *Appl. Soft Comput.*, vol. 73, pp. 344–349, Dec. 2018.
[24] C. Tong, X. Yin, S. Wang, and Z. Zheng, "A novel deep learning method for aircraft landing speed prediction based on cloud-based sensor data," *Future Gener. Comput. Syst.*, vol. 88, pp. 552–558, Nov. 2018.
[25] 4International Air Transport Association, "Unstable approaches, risk mitigation policies, procedures and best practices," Final Rep., 2016.
[26] C. Banbury *et al.*, "Micronets: Neural network architectures for deploying tinyml applications on commodity microcontrollers," in *Proc. Mach. Learn. Syst.*, vol. 3, 2021.
[27] K. Dokic, M. Martinovic, and D. Mandusic, "Inference speed and quantisation of neural networks with TensorFlow lite for microcontrollers framework," in *Proc. 5th South-East Eur. Design Automat., Comput. Eng., Comput. Netw. Social Media Conf. (SEEDA-CECNSM)*, Sep. 2020, pp. 1–6.

DEBORA GIL received the Ph.D. degree in mathematics from the Computer Science Department, UAB. She is currently a Professor with the Computer Science Department, UAB, and in charge of the Interactive and Augmented Modeling (IAM, iam.cvc.uab.es) Research Group, Computer Vision Center (CVC). She has a wide multidisciplinary experience and is an expert in mathematical and statistical modeling of heterogeneous data in clinical (diagnosis and intervention) decision support systems. She has coauthored more than 60 articles in journals (41 indexed in JCR) and over 100 international conferences. She has been IP of several national (seven competitive and five with private companies) and international (WP Leader in E-pilots, H2020-CS2-CFP08-2018-01 and a Co-ordinator of H2020 Topiomics-ATTRACT).

AURA HERNÀNDEZ-SABATÉ received the degree in mathematics, in 2002, and the Ph.D. degree in computer science, in 2009. She is currently a member of the Interactive and Augmented Modeling Research Group, Computer Vision Center, and an Associate Professor with the Computer Science Department, Universitat Autonoma de Barcelona. She has published more than 50 works in refereed journals and international conferences and has participated in several national and international research projects. Her main research interests include develop personalized support systems using computer vision algorithms. In particular, she is focused on the analysis and integration of multimodal data for personalized and adapted support systems for means of transport on groups with high workload. She received an Extraordinary Doctorate Award for her Ph.D. thesis.

JULIEN ENCONNIERE received the M.Eng. degree in aerospace engineering from the ISAE ENSICA, Toulouse, France in 2013, the M.Sc. degree in thermal power from Cranfield University, in 2013, and the Ph.D. degree in the field of rotorcraft performance where his thesis specifically focused on the performance analysis of hybrid electric powerplants and novel vehicle designs for rotorcraft applications.

From 2014 to 2017, he worked as a Simulation Engineer with CleanSky Green Rotorcraft Technology Evaluator. From 2018 to 2019, he was a Performance Engineer at Rolls-Royce. He is currently a Research Fellow in aerospace cognitive computing at Cranfield University. His research interests include modeling, simulation, and control of EVTOL vehicles, and applications of machine learning in aerospace.

SARYANI ASMAYAWATI received the B.Eng. degree in aeronautical engineering from the Institut Teknologi Bandung, Indonesia, and the M.Sc. degree in human factors and safety assessment in aeronautics from Cranfield University, U.K., where she is currently pursuing the Ph.D. degree in transport systems. She worked as an Air Safety Investigator with the Indonesian National Transportation Safety Committee, then as a Senior Consultant Safety Engineer at RGW Cherry and Associates, U.K. She joined Cranfield University, in 2009, where she is also a Senior Lecturer and a Course Director in safety and accident investigation. Her research interests include aviation human factors, particularly in organization safety culture and cognitive engineering. Her more recent work with Rolls-Royce were focused on flight crew decision making and human–machine interface, particularly in dealing with powerplant system malfunctions and interface development for a novel powerplant systems.

PAU FOLCH received the degree in mathematics, in 2016, and the M.Sc. degree in modeling for science and engineering from the Universitat Autonoma de Barcelona (UAB), in 2018. He is currently a member of Aslogic 2011 SL, a company which belongs to the UAB Research Park (PRUAB) and its Cluster of Technological Innovation in Aeronautical Management. He has been working with Aslogic more than five years as a Mathematical Modeler and Data Science, during this period he has been participating as a Researcher in several international consortia innovation research and development projects mainly in the healthcare and aeronautics fields. His main research interests include modeling complex systems by means of mathematical and statistical tools.

JUAN BORREGO-CARAZO was born in Soria, Spain, in 1993. He graduated in physics from the Universitat AutÃšnoma de Barcelona (UAB), Spain, in 2016, and the M.Sc. degree in data science from the University of Barcelona (UB). He is currently pursuing the Ph.D. degree (industrial) with the UAB devoted to optimization methods for deploying neural networks to resource-constrained devices. During his Ph.D., he has collaborated with several companies, including Kostal Group and Samsung R + D, U.K. He is also working at the Computer Vision Center (CVC) on deep learning-based bronchoscopy calibration and tracking methods. His main interests include deep learning and neural networks, computer vision, and machine learning.

MIQUEL ÀNGEL PIERA (Member, IEEE) graduated the degree (Hons.) in computer engineering from the Universitat Autonoma de Barcelona (UAB), in 1988, the M.Sc. degree in control engineering from the University of Manchester Institute of Science and Technology, in 1991, and the Ph.D. degree from the UAB, in 1993. He is currently a full-time Professor with the System Engineering Department, UAB. He is a member of LogiSim, a recognized research group on modeling and simulation of complex systems, and a Co-Founder of two spinoff companies. He is the coauthor of four patents/IPR currently in industrial applications. He is also an expert in air transportation, production technologies/logistics, intelligent transport systems, industrial collaboration, and technology transfer. His awards and honors include the Institution of Mechanical Engineers Award, U.K., Aerospace Division: William Sweet Smith Prize 2015, and the Society for Computer Simulation "Outstanding Professional Contribution Award, 2013."