600177

Classification of Aircraft Maneuvers for Fault Detection

Nikunj C. Oza, Irem Y. Tumer, Kagan Tumer, and Edward M. Huff
Computational Sciences Division
NASA Ames Research Center
Mail Stop 269-3
Moffett Field, CA 94035-1000
{oza,itumer,kagan,huff}@email.arc.nasa.gov

Abstract

Automated fault detection is an increasingly important problem in aircraft maintenance and operation. Standard methods of fault detection assume the availability of either data produced during all possible faulty operation modes or a clearly-defined means to determine whether the data is a reasonable match to known examples of proper operation. In our domain of fault detection in aircraft, the first assumption is unreasonable and the second is difficult to determine. We envision a system for online fault detection in aircraft, one part of which is a classifier that predicts the maneuver being performed by the aircraft as a function of vibration data and other available data. We explain where this subsystem fits into our envisioned fault detection system as well as experiments showing the promise of this classification subsystem.

1 Introduction

A critical aspect of the operation and maintenance of aircraft is detecting problems in their operation when they occur in flight. This allows maintenance and flight crews to fix problems before they become severe and lead to significant aircraft damage or even a crash. Fault detection systems designed for this purpose are becoming a standard requirement in most aircraft. However, most systems are inundated with false alarms, mainly due to an inability to match modeled behavior with real signatures, making their reliability questionable in practice [CITE fault detection lit]. Because of the highly critical nature of the aircraft domain application, most fault detection systems are faced with the task of functioning for systems for which fault data are non-existent. Models are typically used to predict the effect of damage and failures on otherwise healthy (baseline) data [3, 5]. However, while models are a necessary first start, the modeled system response often doesn't take the operational variability and noise into account, hence resulting in the high

rates of false alarms. Novelty detection is one approach to overcome this problem, addressing the problem of modeling the proper operation of a system and detecting when its operation deviates significantly from normal operation [2, 4].

In this paper, we present an approach to novelty detection based on in-flight aircraft data. The data were collected as part of a research effort to understand the sources of variability present in the actual flight environment, with the purpose of eliminating the high rates of false alarms [3, 5, 6]. The fundamental idea is the use of multiple sources of information to predict aspects of system state, such as the maneuver being performed, and predicting faults when the system state predictions are incompatible. In this paper, we present several maneuver classifiers. These classifiers take vibration data from various accelerometers and/or other available data as input and predict the maneuver being performed. Multiple subsystems that predict the maneuver may be present in the system. Models of aircraft operation that generate predictions of vibration signatures may also be included in this system. An overall fault predictor would compare the maneuver predictions from the various subsystems and uses other appropriate data to diagnose whether a fault is present based on these predictions. For example, if the vibration data-based classifier predicts that the helicopter is flying forward at high speed, but other data and/or subsystems indicate that the aircraft is on the ground, then the probability that a fault is present is high.

In the following, Section 2 discusses the aircraft under study and the data generated from them. We discuss the machine learning methods that we used and the data preparation that we performed in order to use these methods in Section 3. We discuss our experimental results in Section 4. We summarize the results of this paper and discuss ongoing and future work in Section 5.

2 Aircraft Data

Data used in this work were collected from two helicopters: an AH1 Cobra and OH58c Kiowa [3]. The data were collected by having two pilots each fly two designated sequences of steady-state maneuvers according to a predetermined test matrix [3]. The test matrix used a modified Latin-square design to counterbalance changes in wind conditions, ambient temperature, and fuel depletion. Each of the four flights consisted of an initial period on the ground (Maneuver G) with the helicopter blades at flat pitch, a low hover (Maneuver H), a sequence of maneuvers drawn from the 12 primary maneuvers, a low hover, and finally a return to ground. Each maneuver was scheduled to last 34 seconds in order to allow a sufficient number of cycles of the main rotor and planetary gear assembly to apply the signal decomposition techniques used in the previous studies.

Summary matrices were created from the raw data by averaging the data produced during each revolution of the planetary gear. The summarized data consists of 31168 revolutions of data for the AH-1 and 34144 revolutions of data for the OH58c. Each row, representing one revolution, indicates the maneuver being performed during that revolution as well as columns representing the following 30 quantities: Revolutions per minute of the planetary gear, Torque (four columns: average, standard deviation, skew, and kurtosis), Vibration data from six accelerometers (four columns per accelerometer: root-mean-square, skew, kurtosis, and a binary variable indicating whether signal clipping occurred), Pilot (binary variable). For the AH-

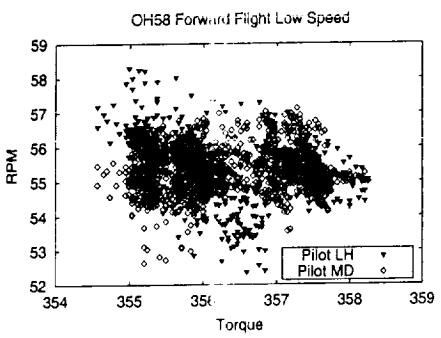

Figure 1: OH58 Maneuver 1 (Forward Flight Low Speed)

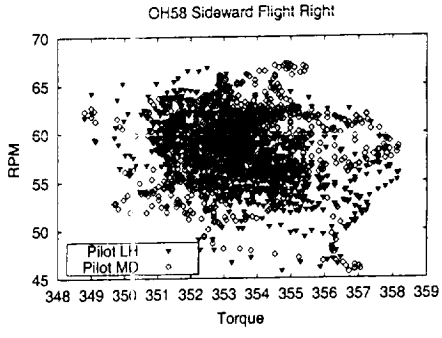

Figure 2: OH58 Maneuver 4 (Sideward Flight Right)

1, the following additional data (14 columns) were available for collection from a 1553 bus: Altitude (average and standard deviation), Speed (average and standard deviation), Rate of climb (average and standard deviation), Heading (average and standard deviation), Bank Angle (average and standard deviation), Pitch (average and standard deviation), Slip (average and standard deviation).

3 Approach

Sample data from two selected maneuvers are shown in Figure 2. The highly-variable nature of the data, as well as differences due to different pilots and different days when the aircraft were flown, are clearly visible, making this a challenging classification problem. To perform the necessary mapping for this problem, we chose multilayer perceptrons (MLPs) with one hidden layer and radial basis function (RBF) networks as our base classifiers. The first was selected due to its relative ease to use whereas the second for its potential ability to focus on specific "areas" of the feature space [CITE kagan and nikunj's paper]. Furthermore, we constructed ensembles of each type of classifier, as well as ensembles consisting of half MLPs and half RBF networks, because ensembles have been shown to improve upon the performance of their constituent or base classifiers, particularly when the correlation among those base classifiers can be kept low [1, 9].

We used data sets consisting of all the available features as inputs (44 for the AH1, 30 for the OH58) and one output for each maneuver (14 possible maneuvers in both cases) gathered from the 176 summary matrices.[1] This resulted in 31168 patterns (revolutions) for the AH1 and 34144 for the OH58. Both types of classifiers were trained using a randomly-selected two-thirds of the data (21000 examples for the AH1, 23000 for the OH58) and were tested on the remainder for the first set of experiments.

For both data sets and for both types of classifiers, we determined the number of hidden units/kernels experimentally. For MLPs, we explored hidden layer sizes ranging from 5 to 100 in increments of 5, and settled on 25 hidden units for the AH1 and 65 units for the OH58. We used a learning rate and momentum term of

[1] We linearly transformed all the input features to be in the $[-2, 2]$ range.

Table 1: Sample confusion matrix for OH58 (MLP).

Class	Classification													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	693	0	7	6	79	0	0	0	0	0	0	0	0	0
2	0	679	0	0	0	0	0	0	0	0	0	0	47	0
3	55	1	568	64	31	6	0	11	9	1	11	7	0	3
4	26	0	43	691	15	0	0	3	0	0	0	2	0	1
5	196	0	68	41	412	0	0	0	0	0	2	16	0	0
6	0	0	0	0	0	719	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	1079	0	0	0	0	0	0	0
8	0	9	22	16	0	0	0	**748**	**177**	**97**	11	6	3	0
9	0	1	1	6	0	0	0	**172**	**381**	**162**	4	7	6	0
10	0	4	1	6	0	0	0	**186**	**170**	**376**	0	8	13	0
11	4	0	15	4	3	0	0	2	1	0	**494**	**217**	0	0
12	3	0	7	6	4	0	0	2	1	0	**200**	**531**	0	0
13	0	63	0	0	0	0	0	4	1	0	0	0	712	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	685

0.2, and we trained for 100 epochs. The performance of both types of classifiers was fairly insensitive to both the hidden unit size/number of kernels and learning parameters. We created 280 MLPs for each helicopter, and we report results as averages over these 280 runs. These 280 MLPs were given different random initial weights before training, but were trained using the same training sets.

For the RBF networks, we used 100 centers for the OH58 data and determined each kernel's center and width using the nearest 300 patterns.[2] For the AH1 data, we used 55 kernels with the centers and widths determined by the nearest 500 patterns. For each helicopter, we created 100 RBF networks, each of which had a different set of centers, and report results as averages over these 100 runs.[3]

For both data sets and classifiers, we used simple averaging ensembles. Though simple to apply, such ensembles perform remarkably well on a variety of data sets [1, 7, 8]. We experimented with ensembles consisting of 2 to 100 base classifiers for our MLP and MLP/RBF ensembles, and 2 to 50 base classifiers for our RBF ensembles, although performance improvements after 10 base classifiers were marginal. These ensembles consisted of random samples drawn from the 280 MLPs and 100 RBF networks that we created for our single-network experiments. For each size of ensemble, we drew 20 random samples and report the results as averages over these runs.

In addition, we calculated the *confusion matrix* of every classifier we created. Entry (i, j) of the *confusion matrix* of a classifier states the number of times that an example of class i is classified as class j. In examining the confusion matrices of our classifiers (see Table 1 for an example of a confusion matrix—entry $(1, 1)$ is in the upper left corner), we noticed that particular maneuvers were continually being confused with one another. In particular, the three hover maneuvers (8-Hover, 9-Hover Turn Left, and 10-Hover Turn Right) were frequently confused with one another and the two coordinated turns (11-Coordinated Turn Left) and (12-Coordinated Turn Right) were also frequently confused (the counts associated with these errors are shown in bold in Table 1. These sets of maneuvers are similar enough to one another that misclassifications within these groups are unlikely to imply the

[2]That is, for each center, the 300 training cases closest to it in Euclidian distance were used to determine its radius. Therefore, the radius increases with the number of points.

[3]Due to the large computation time needed to obtain the centers and widths of the kernels on such large data sets, we only used 100 RBFs as opposed to 280 MLPs.

Table 2 OH58c and AH1 Single Revolution Test Set Results.

Base Type	N	OH58 Results		AH1 Results	
		Single Rev	Single Rev Confusion	Single Rev	Single Rev Confusion
MLP	1	80.533 ± 0.110	93.098 ± 0.073	96.161 ± 0.138	98.643 ± 0.094
	4	83.114 ± 0.063	94.307 ± 0.038	97.747 ± 0.071	99.583 ± 0.064
	10	83.578 ± 0.047	94.470 ± 0.025	98.089 ± 0.042	99.737 ± 0.041
	100	83.960 ± 0.018	94.683 ± 0.010	98.225 ± 0.008	99.818 ± 0.003
RBF	1	77.650 ± 0.142	90.860 ± 0.104	95.811 ± 0.098	99.106 ± 0.060
	4	78.408 ± 0.089	91.384 ± 0.052	96.272 ± 0.032	99.390 ± 0.035
	10	78.550 ± 0.039	91.607 ± 0.027	96.441 ± 0.021	99.472 ± 0.013
	50	78.729 ± 0.018	91.638 ± 0.011	96.438 ± 0.009	99.493 ± 0.005
MLP/ RBF	2	81.851 ± 0.087	93.548 ± 0.053	97.392 ± 0.069	99.515 ± 0.053
	4	82.724 ± 0.084	94.097 ± 0.047	97.715 ± 0.063	99.646 ± 0.056
	10	83.308 ± 0.041	94.346 ± 0.031	97.899 ± 0.019	99.764 ± 0.011
	100	83.798 ± 0.023	94.548 ± 0.014	97.989 ± 0.007	99.791 ± 0.003

presence of faults. Therefore, for our second set of experiments, we recalculated the classification accuracies after consolidating these maneuvers (e.g., all three hovers into one maneuver and both left and right turns into one maneuver).

Finally, we used the knowledge that a helicopter needs some time to change maneuvers. That is, two sequentially close patterns are unlikely to come from different maneuvers. To obtain results that use this "prior" knowledge, we tested on sequences of revolutions by averaging the classifiers' outputs on a window of examples surrounding the current one. In one set of experiments, we averaged over windows of size 17 (8 revolutions before the current one, the current one, and 8 revolutions after the current one) which corresponds to about three seconds. Note that, because the initial training and test sets were randomly chosen from this sequence, this averaging could not be performed on the test set alone. Instead it was performed on the full data set for both helicopters. To allow meaningful comparisons of these results, we also computed the "full set error" (training and test errors) on the original, segmented data and these results are presented in Tables 3,4.[4]

4 Results

In this section we describe the experimental results that we have obtained so far. In Table 2, the column marked "Single Rev" shows the results of running individual networks and ensembles of various sizes on the summary matrices randomly split into training and test sets. We only present results for some of the ensembles we constructed due to space limitations and because the ensembles exhibited relatively small gains beyond 10 base models. N is the number of base models used for the classification. MLPs and ensembles of MLPs outperform RBFs and ensembles of RBFs consistently. The ensembles of MLPs improve upon single MLPs to a greater extent than ensembles of RBF networks do upon single networks, indicating that the MLPs are more diverse than the RBF networks. Mixed ensembles have

[4]We performed this windowed averaging as though the entire data were collected over a single flight. However, it was in fact collected in stages, meaning that there are no transitions between maneuvers. We show these results to demonstrate the applicability of this method to sequential data obtained in actual flight after training the network on "static" single revolution patterns.

Table 3: OH58c Single Revolution and Windowing Results on Full Data Set.

Base Type	N	Single Rev	Single Rev Consolidated	Window of 17	Window of 17 Consolidated
MLP	1	82.724 ± 0.121	94.067 ± 0.049	89.813 ± 0.191	96.799 ± 0.142
	4	85.466 ± 0.073	95.020 ± 0.034	91.287 ± 0.130	96.956 ± 0.043
	10	86.035 ± 0.050	95.243 ± 0.034	91.550 ± 0.081	97.006 ± 0.044
	100	86.414 ± 0.015	95.420 ± 0.007	91.621 ± 0.022	97.067 ± 0.008
RBF	1	79.484 ± 0.053	91.313 ± 0.099	84.670 ± 0.212	95.008 ± 0.115
	4	79.127 ± 0.094	91.786 ± 0.045	84.739 ± 0.131	95.026 ± 0.058
	10	79.297 ± 0.047	91.975 ± 0.020	84.977 ± 0.070	95.232 ± 0.045
	50	79.460 ± 0.014	92.014 ± 0.008	85.086 ± 0.021	95.103 ± 0.017
MLP/ RBF	2	83.740 ± 0.093	94.212 ± 0.063	89.935 ± 0.163	96.508 ± 0.084
	4	84.710 ± 0.075	94.748 ± 0.048	90.493 ± 0.125	96.779 ± 0.069
	10	85.280 ± 0.038	95.012 ± 0.030	90.755 ± 0.068	96.869 ± 0.043
	100	85.681 ± 0.017	95.147 ± 0.012	90.838 ± 0.029	96.822 ± 0.014

performances superior to the pure-MLP for small numbers of base models, but have worse performances for larger numbers of models. Mixed ensembles perform better than pure-RBF ensembles for all numbers of base models. In the smaller ensembles, the diversity provided by including RBF networks helped relative to pure-MLP ensembles. However, in the larger ensembles, replacing half the MLPs with RBFs degrades performance—the RBFs are different from the MLPs but not different enough from each other to warrant having such a large number of them. Note that the column marked "Single Rev Confusion" shows the single revolution results after allowing for confusions among the hover maneuvers and among the coordinated turns. As expected, the performances improved dramatically.

Table 3 shows the results of performing the windowed averaging described in the previous section in the column marked "Window of 17." The column "Window of 17 Confusion" gives the results allowing for the confusions mentioned earlier. The columns marked "Single Rev" and "Single Rev Confusion" are the average of the training and test errors, weighted by their sizes. We can clearly see the benefits of this windowed averaging, which serves to smooth out some of the noise present in the data.

Table 4: AH1 Single Revolution and Windowing Results on Full Data Set.

Base Type	N	Single Rev	Single Rev Confusion	Window of 17	Window of 17 Confusion
MLP	1	96.567 ± 0.115	98.789 ± 0.081	97.821 ± 0.111	98.744 ± 0.086
	4	98.007 ± 0.064	99.561 ± 0.060	98.933 ± 0.080	99.374 ± 0.082
	10	98.313 ± 0.041	99.769 ± 0.042	99.179 ± 0.040	99.621 ± 0.039
	100	98.438 ± 0.006	99.852 ± 0.003	99.268 ± 0.004	99.700 ± 0.002
RBF	1	96.023 ± 0.093	99.209 ± 0.051	97.120 ± 0.114	98.931 ± 0.064
	4	96.480 ± 0.031	99.469 ± 0.029	97.495 ± 0.044	99.141 ± 0.023
	10	96.638 ± 0.015	99.535 ± 0.011	97.636 ± 0.019	99.194 ± 0.011
	50	96.649 ± 0.008	99.558 ± 0.005	97.624 ± 0.005	99.187 ± 0.003
MLP/ RBF	2	97.664 ± 0.059	99.611 ± 0.045	98.564 ± 0.062	99.327 ± 0.053
	4	97.957 ± 0.052	99.699 ± 0.046	98.725 ± 0.056	99.390 ± 0.055
	10	98.092 ± 0.017	99.796 ± 0.010	98.818 ± 0.021	99.516 ± 0.012
	100	98.144 ± 0.014	99.810 ± 0.008	98.852 ± 0.006	99.546 ± 0.003

Table 4 shows the analogous results for the AH1 helicopter. The performances are substantially better here than for the OH58. We expected this because the AH1

Table 5: AH1 Bus and Non-Bus Results

Inputs	Single Rev	Single Rev Confusion	Window of 17	Window of 17 Confusion
Bus	90.380 ± 0.110	95.871 ± 0.091	91 209 ± 0.126	96.027 ± 0.086
Non-Bus	87.884 ± 0.228	93.731 ± 0.171	92 913 ± 0.355	96.110 ± 0.236
$P(agree)$	79.523 ± 0.247	90.063 ± 0.202	85 609 ± 0.320	93.393 ± 0.247

is a heavier helicopter, so it is less affected by conditions that tend to introduce noise such as wind changes. Just as with the OH58, on the AH1, the mixed ensembles outperform the pure ensembles for small numbers of base models but perform worse than the MLP ensembles for larger numbers of base models. Once again, we can see that ensembles of MLPs outperform single MLPs to a greater extent than ensembles of RBFs outperform single RBFs, so the RBFs are not as different from one another. Because of this, it does not help to add large numbers of RBF networks to an MLP ensemble. Note that the same sets of maneuvers that were frequently confused on the OH58 were confused on the AH1. Taking this confusion into account boosted performance significantly. The windowed averaging approach did not always yield improvement when allowing for the maneuver confusions, but helped when classifying across the full set of maneuvers. However, in all cases when windowed averaging did not help, the classifier performance was at least 98.93%, so there was very little room for improvement.

5 Discussion

In this paper, we presented an approach to fault detection that contains a subsystem to classify an operating aircraft into one of several states. More specifically, the proposed system determines the maneuver being performed by an aircraft as a function of vibration data and any other available data. Through experiments with two helicopters, we demonstrated that the system is able to determine the maneuver being performed with good reliability (at least 95% when allowing for confusions among very similar system states and smoothing by combining predictions from short sequences of data). The initial results show great promise in classifying the correct maneuver with high certainty. Future work will involve applying this approach to "free-flight data", where the maneuvers are not static or steady-state, and transitions between maneuvers exist.

The results presented in this paper address the maneuver classification portion of the online fault detection system envisioned in this research. To address the overall novelty detection problem, future work will involve experiments to determine the probabilities of agreement between different classification results, to detect possible faults when there is a mismatch. For example, for the AH1 helicopter, we have data from a 1553 bus as described in Section 2. We trained some classifiers using just the bus data as inputs and other classifiers using all except the bus data. Table 5 shows just the results of training 20 single MLPs on these data using the same network topology as for the other MLPs trained on all the AH1 data. They performed much worse than the single MLPs trained with all the inputs presented at once. The last line in the table indicates the percentage of maneuvers for which the two types of classifiers agreed.

Recall from Section 1 that we would like classifier disagreement to indicate the presence of a fault; therefore, we would like these agreement probabilities to be much higher. However, we hypothesize that we can use the bus data in a much simpler way. For example, if the vibration data-based classifier predicts that the aircraft is performing a high-speed forward flight, but the bus data indicates that airspeed is near zero, then the probability of a fault is high. We do not necessarily need a system that returns the maneuver as a function of all the variables that constitute the bus data. In this example, we merely need to know that a near-zero airspeed is inconsistent with a high-speed forward flight. We plan to perform a detailed study of the collected bus data so that we may construct simple classifiers representing knowledge of the type just mentioned and use them to find inconsistencies such as what we just described. We are confident that using the different types of system models, metrics, and classifiers mentioned in this paper, we can obtain a reliable fault detector.

References

[1] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[2] Paul Hayton, Bernhard Schölkopf, Linel Tarassenko, and Paul Anusiz. Support vector novelty detection applied to jet engine vibration spectra. In Todd K. Leen, Thomas G. Dietterich, and Volker Tresp, editors, *Advances in Neural Information Processing Systems-13*, pages 946–952. Morgan Kaufmann, 2002.

[3] Edward M. Huff, Irem Y. Tumer, Eric Barszcz, Mark Dzwonczyk, and James McNames. Analysis of maneuvering effects on transmission vibration patterns in an AH-1 cobra helicopter. *Journal of the American Helicopter Society*, 2002.

[4] Nathalie Japkowicz, Catherine Myers, and Mark Gluck. A novelty detection approach to classification. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 518–523, Montreal, Canada, 1995. Morgan Kaufmann.

[5] D.A. McAdams and I.Y. Tumer. Towards failure modeling in complex dynamic systems: impact of design and manufacturing variations. In *ASME Design for Manufacturing Conference*, volume DETC2002/DFM-34161, September 2002.

[6] I.Y. Tumer and E.M. Huff. On the effects of production and maintenance variations on machinery performance. *Journal of Quality in Maintenance Engineering*, To appear. 2002.

[7] K. Tumer and J. Ghosh. Analysis of decision boundaries in linearly combined neural classifiers. *Pattern Recognition*, 29(2):341–348, February 1996.

[8] K. Tumer and J. Ghosh. Error correlation and error reduction in ensemble classifiers. *Connection Science, Special Issue on Combining Artificial Neural Networks: Ensemble Approaches*, 8(3 & 4):385–404, 1996.

[9] D. H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.