Enabling the Automated Assessment of Precision Aerobatic Manoeuvres

Thomas David, Robert Clarke
Department of Aerospace Engineering, University of Bristol, United Kingdom

Artur Uzieblo
Australian Precision Aerobatics, Australia

Dimitris Panayiotou, Tom Richardson
Department of Aerospace Engineering, University of Bristol, United Kingdom

This paper describes a flight path assessment process for precision aerobatic manoeuvres. The work forms part of 'Flight Coach', a project providing technological improvements to pilot feedback and judging in precision aerobatics. An on-board flight logger is used to record information during a precision aerobatic flight. A template data set representing the flown sequence is then constructed and aligned to the flight. The template is adjusted to match the scale of the recorded flight, and a comparative assessment is performed. This work offers a significant contribution to the sport of precision aerobatics, but also develops tools and processes that could be applied more widely in areas such as pilot training.

I. Introduction

Precision Aerobatics is a sport where competitors control aircraft through predefined sequences of aerobatic manoeuvres. A group of judges observe the sequence and provide scores by subtracting points for perceived errors in the execution of each manoeuvre. The downgrades that judges should apply to different types of error are described in well established rule books for each aerobatic competition discipline. A common frustration for competitors is with the ability of judges to apply these rules consistently. Automating the assessment of precision aerobatic manoeuvres has the potential to transform the sport, whilst the methods developed as part of the solution can make significant contributions to the wider community. The Flight Path Assessment (FPA) process described in this paper forms part of Flight Coach [1]. Flight Coach provides data driven tools to precision aerobatic pilots and judges with the aim of reducing subjectivity in the scoring.

Among the wider applications of the FPA process, a central one is machine learning based flight control, which is attracting growing interest. For example, the work in [2] demonstrates the use of Reinforcement Learning (RL) to train a flight controller to perform an agile perched landing manoeuvre. In the case described in the paper the desired outcome, and so the reward function, is clearly defined by the objective. On the other hand [3] describes autonomous aerobatic manoeuvres performed with a rotary wing UAV, where a reward function is derived through trajectory learning from a number of flights performed by an expert human pilot. Imitation learning limits the performance of the trained controller to that of the expert pilot and is not desirable for an automated scoring system as the expert pilots influence could be considered a potential source of bias. The work in [4] generates template trajectories by numerically solving optimal control problems. These optimal control problems are generally solved by saturating certain parameters and so are inherently high risk. By calculating reward from standard manoeuvre definitions the FPA process has the potential to offer a safer, albeit less optimal approach. Additionally, the FPA process offers an alternative to a direct trajectory deviation reward, as it can allow the trajectory to vary whilst still meeting higher level criteria.

Part of the FPA process involves recording and labelling flight data using a temporal alignment algorithm. The data is unique to precision aerobatics as many repeat attempts at pre-defined schedules are available. This has the potential to make a significant contribution to Flight Regime Recognition (FRR), a well researched area for military aircraft and helicopter load estimations for fatigue analysis. The work in [5] shows that FRR using low cost flight data recorders is an effective method of estimating loads on aircraft components, and can even be preferable to directly mounting strain sensors on the component being analysed. A variety of methods are used for FRR, a common one is to train a neural network with labelled training data to predict the regime based on state observations [6].

The FPA process has a potential application as a debriefing tool for use in pilot training. Various pilot performance metrics derived from instrumented pilot proficiency check flights are used in [7] to augment subjective scoring by flight instructors. The paper highlights the limitations of a human evaluators quantitative observation capabilities and the labour intensive nature of the one to one evaluator-evaluatee ratio. The Flight Coach Plotter is a free web app developed as part of the Flight Coach project which is already receiving significant uptake by aerobatic pilots for its presentation of objective post flight feedback. A view of the flight coach plotter, displaying some recorded manoeuvres from the F3A P23 aerobatic sequence is shown in Figure 1

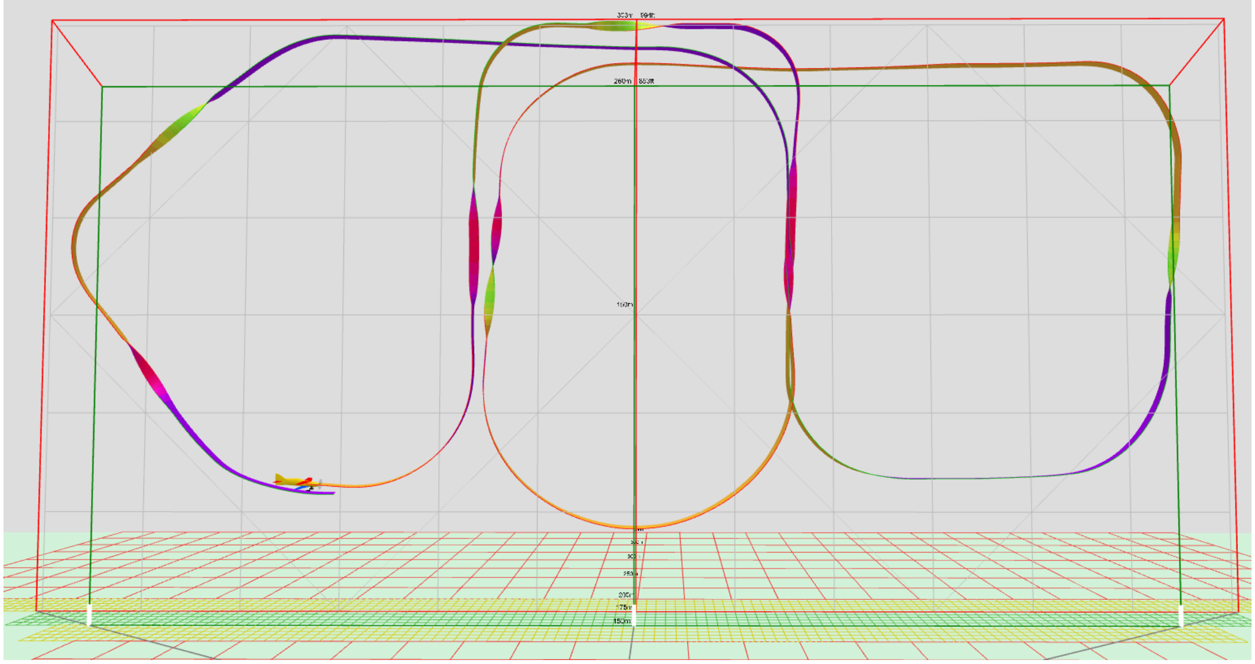


Fig. 1 Extract from the Flight Coach plotter

Some work already exists on electronic judging aids, such as [8], where the applicability of a data logging system aimed at general aviation safety was discussed. The FPA process described here follows a similar approach, but instead uses widely available, low cost UAV flight controllers running open source firmware for its data logging device. This allows the system to be adopted more quickly as well as allowing the developers to focus on post processing the flight data using open source tools.

The work in this paper is based on flight logs recorded using a UAV flight controller running ArduPilot firmware [9] mounted in a model aircraft performing the F3A P21 aerobatic sequence [10]. The flight logger records a time history of state information that is compared to an artificially generated template set of data constructed from the sequence definition. Temporal alignment of the two time series is performed to identify the individual manoeuvres, and the primitive elements that make up each manoeuvre, within the flown data. This segmentation of the recorded data allows the grading criteria applicable to each manoeuvre and element to be applied.

The objective of this work is to record a sequence of manoeuvres flown by an aerobatic aircraft, then to align and compare them to a generated template set of data based on the sequence definition. The aircraft and systems used to record the flight data is described in Section II. The sequence used for the study and the process of generating the template data set is described in Section III. The temporal alignment process is discussed in Section IV. Section V describes the process of constructing a new, scaled template based on the aligned data-set and exposing the comparisons for grading.

II. Flight Data Recording

Flight data was recorded using a Pixhawk 4 mini UAV flight controller running ArduPlane firmware [9], with a Ublox M8N external GPS and Magnetometer. In order to reduce impact on the aircraft the flight recording system was powered by a dedicated supply with no connection to the flight critical systems. For the same reason no pitot tube was used and airspeed information was estimated by the ArduPilot firmware. Competition rules do not allow any hardware associated with feedback control to be present in the aircraft, so all flight logging systems mounted in aircraft that are used for competition need to be easily removable. The aircraft used to record the data presented in this paper is shown in Figure 2. This is a typical F3A aerobatic aircraft; span 1.85 m, length 2 m, weight 4500 g and powered by a 3.4 kW electric motor. Figure 3 shows a visualisation of the example section of P21 sequence recorded with the aircraft. The data is visualised in a 3D plot with red and blue lines representing traces of port and starboard wing tips, along with mesh representations of the aircraft position and orientation at regular time intervals.

Fig. 2 Aircraft used for example data acquisition

The ArduPilot log is parsed and the relevant position, orientation, body axis rates, body speeds and accelerations are extracted or calculated as necessary. The recorded position and attitude data is rotated to a standard coordinate frame that is used throughout the process. The sequence frame is defined with the origin on the pilot position, Y axis to the pilots right, parallel to the maneuvering plane and Z axis up. The transformation between the ArduPilot home coordinate frame and the sequence frame is calculated based on the GPS data recorded in the log and a predefined flight line, which positions the aerobatic box in the world.

The visualisations of the recorded data shown in this paper are based on three manoeuvres from a recorded F3A P21 sequence. F3A is a discipline governed by the Fédération Aéronautique Internationale (FAI) and is the principal Radio Control (RC) aerobatic discipline. The Flight Coach project offers potential for the collection of considerable amounts of data from hundreds of different pilots of different standards, flying different sequences, aircraft types and competition disciplines. Already we have over 100 pilots using the plotter from at least 14 different countries.

III. Template Data Generation

Aerobatic manoeuvres are referenced for competition by the FAI in the Aresti catalog [11]. Figures from the catalog are assembled into a sequence according to the competition rules. Generally a competition sequence is used for all competitions within a discipline for a period of one or two years. In some cases ’unknown’ sequences may be written specifically for a competition and given to the pilots on the day to ensure that they they do not have the opportunity to practise the manoeuvres. The figure sequences are typically provided to competitors as a series of Aresti catalog numbers alongside a set of diagrams using Aresti notation, a graphical notation for aerobatic figures. Figure 4 shows an extract of the Aresti sequence definition and Figure 5 shows a visualisation of the constructed template dataset for the first three manoeuvres of the F3A P21 sequence.

While relatively easy for humans to understand, the graphical nature of Aresti notation makes it challenging to use in a computer directly. Michael Golan developed One Letter Aerobatic Notation (OLAN) to provide a compact, text-based representation of a figure sequence and some associated layout information to use for encoding Aresti diagrams. At present, the notation is designed for full-scale aerobatics and future work is required to expand the OLAN notification to include all possible manoeuvres in the model aerobatic disciplines. The OpenAero project has developed a web-based

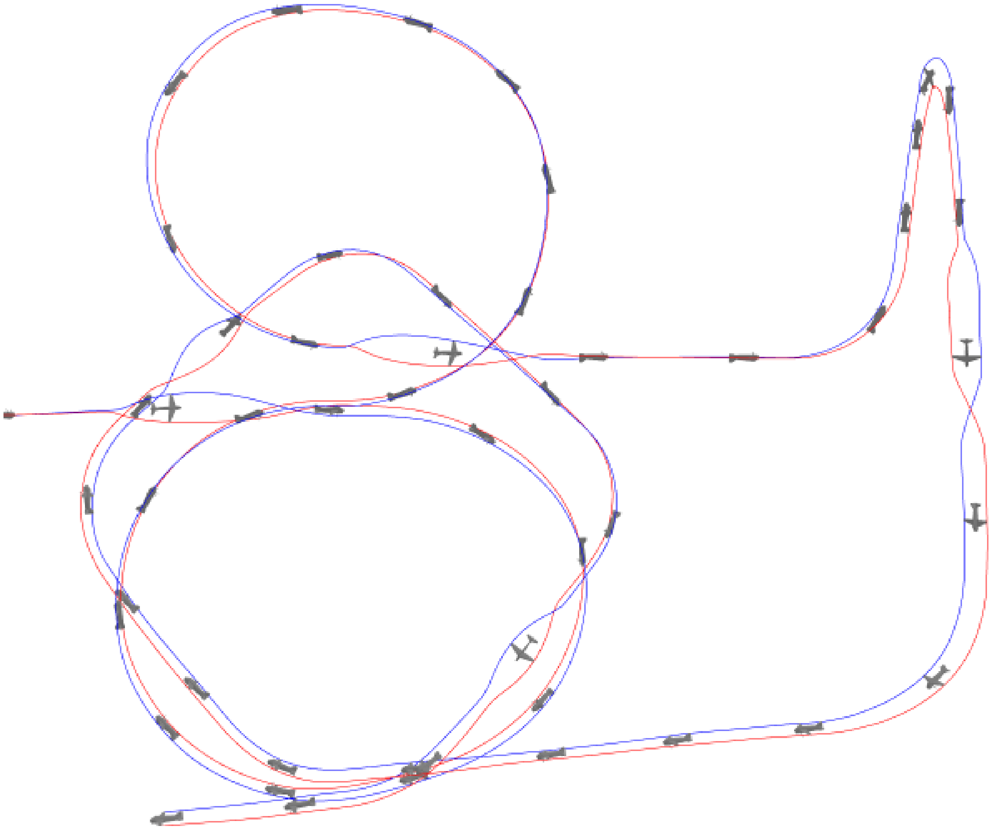


Fig. 3 Visualisation of actual flight data for the first three manoeuvres of the P21 sequence

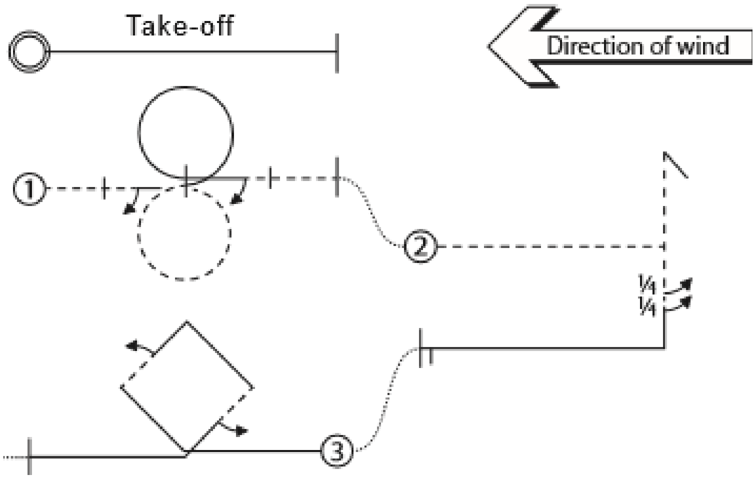


Fig. 4 Extract from the Aresti diagram for the F3A P21 aerobatic sequence [10]

tool [12] to generate and edit Aresti notation diagrams. This effectively serves as a graphical interface to edit OLAN strings.

Once the OLAN string for a sequence is obtained, it can be handled more easily by a computer. In our work, a library of figure definitions and some post-processing is used to parse the OLAN string and generate a series of primitive elements that together define the sequence. The primitive elements are one of:

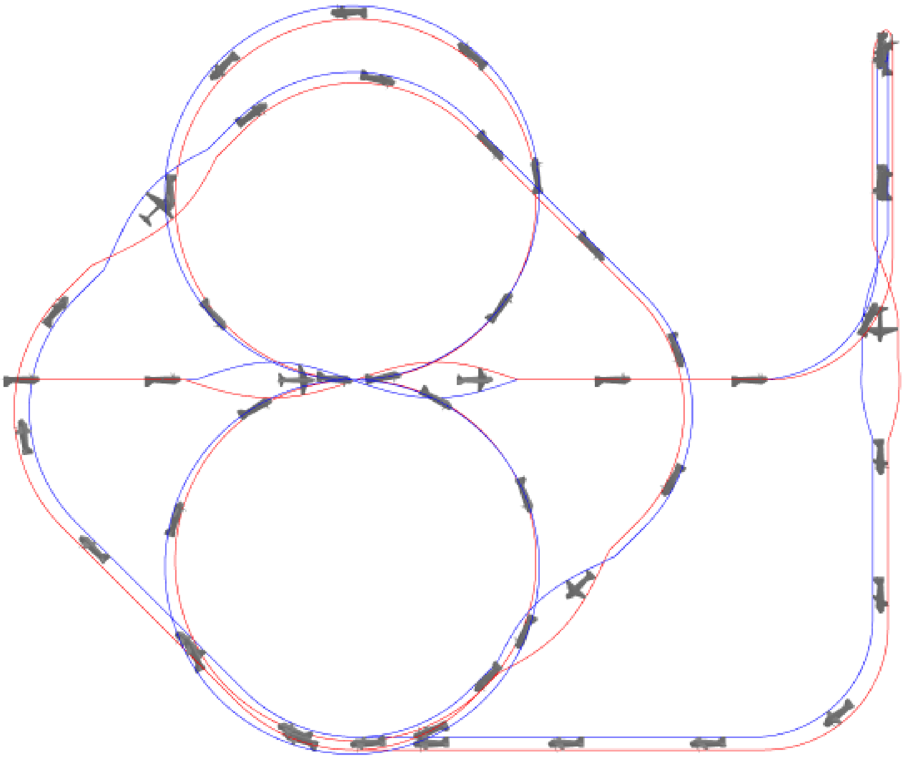


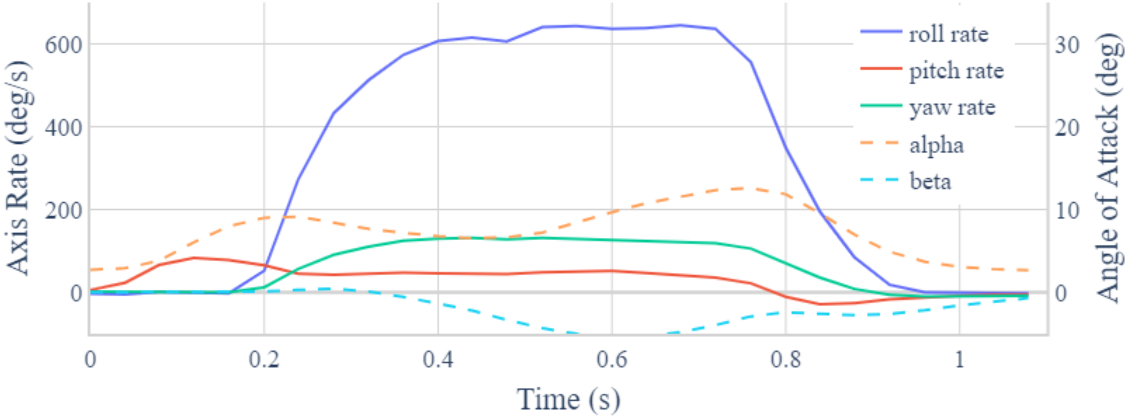
Fig. 5 Visualisation of the template data set constructed from the sequence definition

- Line, for a straight line of a given length
- Loop, for a constant arc initiating with body frame pitch or yaw rotation of the prescribed radius and angle
- Spin, for upright or inverted spins, these begin on a horizontal line and end on a vertical down-line
- Stallturn, for stall turns or wing-overs, These begin on a vertical up-line and end on a vertical down-line
- Snap, for positive and negative snap rolls

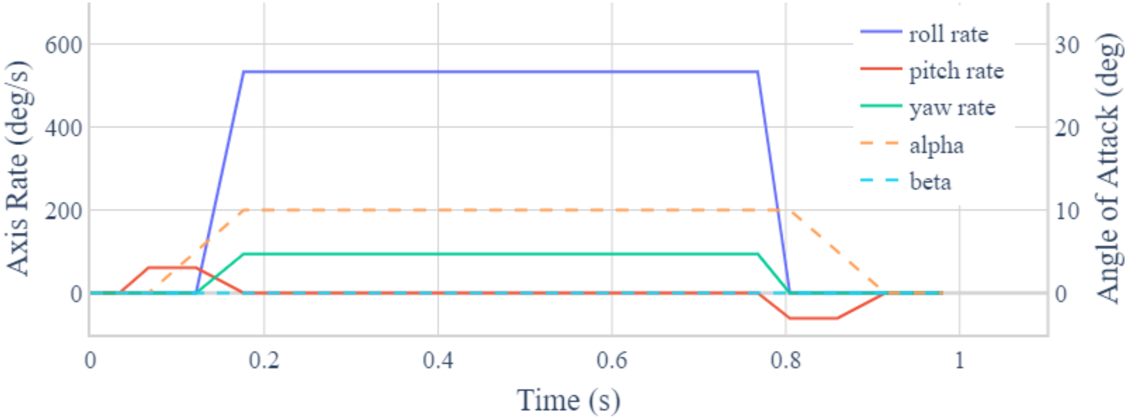
Some primitives have associated data; for lines the length is specified, for loops the radius and proportion of a full loop to perform are specified. The Line, Loop, Snap and Spin primitives may also have a number of rolls that can be integrated through the element. The Loop primitive will not maintain its Knife-Edge or conventional state for the duration of the element if the roll parameter is non-zero.

With a series of primitives generated from the OLAN string, the next step is to create data points that represent the ideal aircraft state throughout the sequence. This state includes the aircraft position, attitude, velocities, axis rates and accelerations. The data is constructed in an idealised judging coordinate frame, for most elements this is similar to the wind axis, but with the x-axis aligned with the direction of flight rather than the oncoming wind vector. Flight data generation methods cover the Loop and Line primitives, creating data points based on the elements associated data, initial aircraft position, orientation and airspeed, which is interpreted as a constant velocity in the x-axis. The number of rolls specified for the primitive element are superimposed on the constructed data with new attitudes, axis rates and accelerations being calculated for each point. The stall turn primitive is constructed through pure body frame yaw rotation with no forward velocity. The snap roll and spin primitive elements are more complicated than lines, loops and stallturns as the judging criteria put additional constraints on the attitude of the aircraft. The assumption that the wind axis is aligned with the body axis and that there is no wind must be broken in order to represent them satisfactorily according to the judging criteria.

The rules for all precision aerobatic disciplines require snap rolls to display a visible pitch departure from the direction of flight prior to, or coinciding with the start of an auto-rotation, where the aircraft rotates around an axis that remains aligned with the velocity vector. During a snap roll one wing should be stalled, whilst the other remains lifting, so the the motion is dominated by a very fast roll rotation. Figure 6 shows a comparison of the axis rates and the calculated angle of attack for a recorded snap roll and the generated template data. The initial pitch rate reflects the pitch departure, which is followed by a reduction in the pitch rate as the aileron and rudder are applied and the roll and yaw rotation begins. Finally the pitch rate reverses at the end of the snap to re-align the body axis with the velocity vector. Some Differences can be seen in the roll rate, where the inertia of the real aircraft causes some lag in initiating and stopping the roll. The recorded snap also shows that some pitch rate is maintained throughout the auto-rotation, where it reduces to zero in the template snap. This is because some deviation from the initial flight path is seen in the real snap which is not represented in the template. These differences are sufficiently small for the template to work both as an input to the temporal alignment process and as a training tool for pilots and judges.



(a)

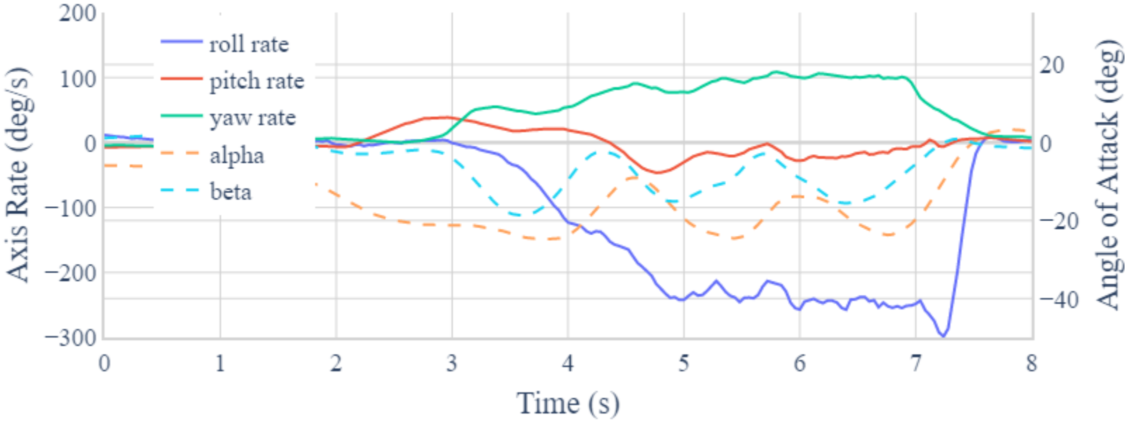


(b)

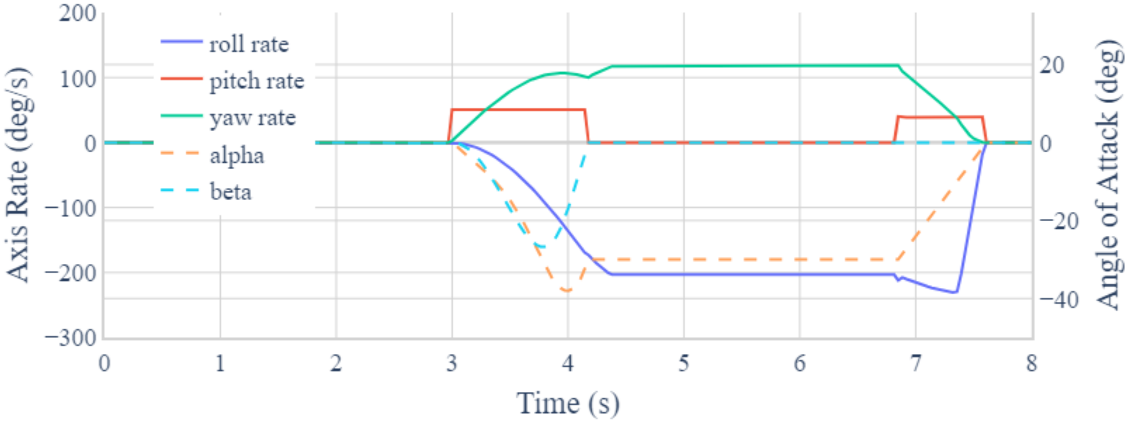
Fig. 6 Comparison of axis rates and angle of attack for the a) flown b) template snap roll primitive elements

Figure 7 compares the axis rates and computed angle of attack values for a recorded spin element and the corresponding template. The spin is initiated with a nose down pitch rotation as the aircraft stalls. This is followed by a yaw rotation, then an increasing roll rotation as the spin becomes more axial. As with the template snap element the template spin reaches the maximum rotation rate slightly earlier than the recorded version due to the simplified representation. This is acceptable as the relative magnitudes of the axis rates reflect the flight data and so have little impact on temporal alignment process.

The processes developed to generate the templates include a number of scaling parameters that can be set individually, according to the sequence or based on measures of the recorded flight data. This process will be discussed in more detail in Section IV. At this stage they have been set so that plots of the template stay roughly within the aerobatic manoeuvring zone and are of a similar scale and speed to those recorded in Section II.



(a)



(b)

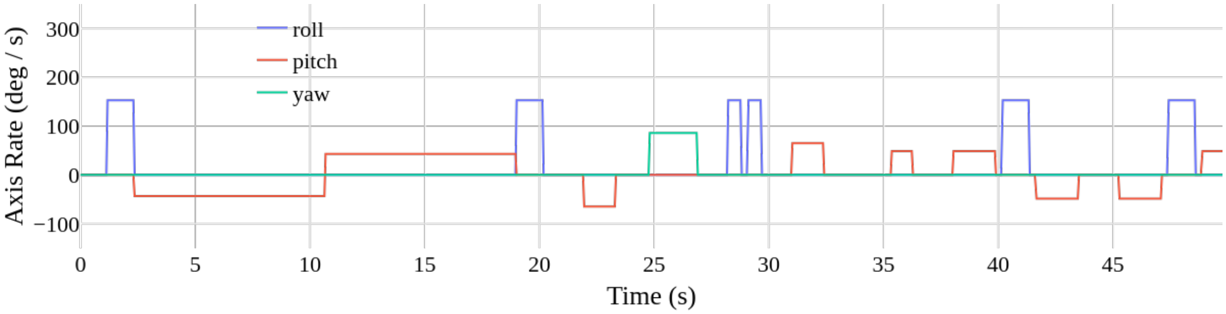
Fig. 7 Comparison of axis rates and angle of attack for the a) flown b) template spin primitive elements

IV. Temporal Alignment

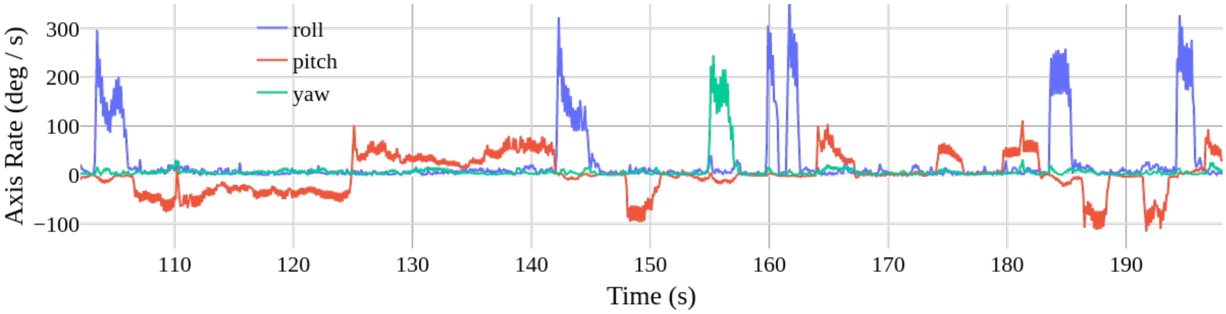
The primitive elements are identified in the recorded flight data using Dynamic Time Warping (DTW). A similar approach is used in [3] and [4] to perform the trajectory alignment. The DTW algorithm is designed for measuring the distance between two time series, which may vary in the time axis. As well as producing a similarity measure DTW calculates the best match between two time series and produces a warping path between the two such that every index of the first sequence is joined to every index of the second. The algorithm works based on the Euclidean distance between the points, so constant offsets and differences in scale of the time varying parameters are not accounted for. Other time series alignment algorithms such as [13] offer better results in those cases.

In Precision Aerobatics a pilot must fly manoeuvres as accurately as possible, so as to receive the minimum downgrade according to predefined grading criteria for each manoeuvre. Whilst meeting these criteria is challenging, they are not sufficient to entirely constrain the manoeuvre geometry. As such each manoeuvre has a set of parameters that may be varied without producing a downgrade. These parameters can differ between competition disciplines. As a result of these variations it is not possible to generate a single template data-set that reflects the error-free flown geometry of every recorded version of a given sequence. The template sequence must therefore be scaled based on measurable values from the flight data prior to running the DTW algorithm, or parameters must be selected that do not vary significantly between flights and do not have zero offsets.

A number of sets of parameters have been investigated for input to the DTW algorithm. Initial results using position data transformed and scaled so that the template roughly aligned with the recording showed some success. The process was refined to the point of performing an optimisation with the scaling parameters as independent variables in order to minimize the DTW distance measure. Whilst this approach showed some success when matching subsets of a sequence it proved inefficient at aligning an entire sequence due to the large number of independent variables and relatively slow evaluation required for each step of the optimization. Faster and more reliable alignments have been achieved by comparing the body frame roll, pitch and yaw rates of the recorded data to those of the template. These variables work better because a constant offset is not required and they can easily be scaled globally based on measurements of the flight data. Figure 8 shows the roll, pitch and yaw axis rates for the template and flight data. As the majority of roll directions in the sequence are at the pilots discretion the matching is performed based on the magnitude of the roll and yaw rate.



(a)



(b)

Fig. 8 Axis rate parameters used for temporal alignment for the three example manoeuvres extracted from a) the constructed template b) actual flight data

Figure 9 shows a visualisation of the output from the temporal alignment process. The lines represent the recorded Centre of Gravity (CG) trace of the aircraft through the three manoeuvres used in the example. The labelling of manoeuvres and elements is illustrated with discrete colouring of the lines. Each manoeuvre is separated by a short section of horizontal line, this is included at the start of the following manoeuvre as a Line primitive element.

The DTW algorithm works on the Euclidean distance between the two input time series. As a result the template dataset needs to be roughly scaled to match the flight data. The scaling of the baseline dataset for each sequence has been manually generated to roughly reflect recorded data. This scaling does not necessarily best reflect the axis rates for all flown sequences, which may be affected by a pilots flying style, aircraft characteristics or the environmental conditions. The FPA proposes taking preliminary measurements of the recorded flight data to use as parameters for scaling the templates prior to running the temporal alignment. Two simple approaches for scaling the templates are assessed in this paper and significant further work is proposed.

The first approach proposed is to measure the recorded axis rate data directly and to set the rate for each type of primitive element based on a percentile rank of a given axis. Values used for this process are shown in Table 1. These values have been selected as when applied they produce template data sets that reflect the flown axis rates reasonably well.

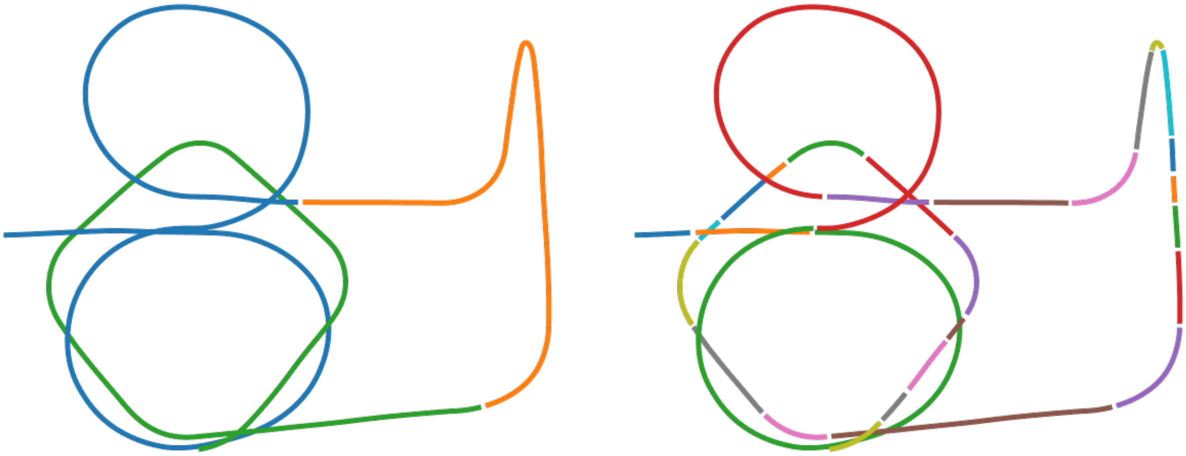

(a) **(b)**

Fig. 9 CG trace of the recorded flight data. Coloured segments of line represent automatically labelled a) manoeuvres and b) primitive elements

Table 1 Axis Rate Scaling

Primitive Element	Axis	Proportion of Recorded Range
Loop	Pitch	90 %
Line	Roll	95 %
Snap	Roll	100 %
Stall Turn	Yaw	99.5 %
Spin	Roll	99.5 %

The second approach proposed is to take measurements of the average distance from the maneuvering plane to the pilot and of the average speed of the aircraft during the flight. All the datasets tested were from F3A sequences, where the manoeuvres must be performed in a box, defined by the intersect of vertical and horizontal 60 degree passing through the pilot position and the manoeuvring plane. It is therefore possible for an F3A flight to construct a bounding box for the entire flown sequence from this distance measure alone, and this method could easily be adapted to other disciplines by direct bounding box measures. The baseline template for sequence was manually constructed so that it fits within the manoeuvring zone and methods have been written to allow global scaling of the entire sequence. When the geometric features are scaled the constructed template will have the correct axis rates for a given speed.

To assess the quality of the automated alignments an evaluation process has been developed. The Flight Coach project [1] provides a convenient browser based plotting tool which allows flight logs to be manually split into individual manoeuvres. The tool also allows files containing the split information and the flight data to be output. In order to use this manually split data a number of hurdles were faced. Manually split logs are split on the connecting line between manoeuvres, where the FPA process uses the convention that this connecting line is part of the following manoeuvre. The manually split data only goes to the level of individual manoeuvres, whereas the automated temporal alignment process identifies individual primitive elements. In order to identify the individual primitive elements within the manually split data an additional temporal alignment process was performed, comparing a template to each manoeuvre individually. As with the full alignment process, this secondary temporal alignment process is not perfect, but the alignment algorithm has a significantly easier job and therefore if the differences between the two approaches are small for a given log then the full approach can be deemed successful.

Forty flights were used for the assessment, featuring four different pilots flying a number of F3A aerobatic sequences (P21, F21, P23 and F23). The pilots exhibited a range of flying styles and the flights were performed in a range of wind conditions with different aircraft. The alignment process was considered successful for a given flight if the largest difference in the calculated split location between two elements was less than two seconds. In addition to the two template scaling options the tests were also run with signal whitening applied to the axis rates in an attempt to make the magnitudes of the features being compared more similar. The results of this assessment are presented in Table 2.

Table 2 Temporal Alignment Assessment Results

Scaling Method	Signal Whitening	Success Rate
Axis Rates	True	15 %
Axis Rates	False	15 %
Distance and Speed	True	65 %
Distance and Speed	False	72.5 %

Table 2 shows that globally scaling the template sequence based on the distance and speed at which the original flight was flown gives significantly better results. This difference is due to the assumption in the axis rate scaling approach that the axis rate is constant for a given type of primitive element. In order to fit the sequence within the manoeuvring zone a pilot must break this assumption, making some loops tighter than others, or adjusting the roll rate to suit a manoeuvre. The template sequence was constructed to fit within the manoeuvring zone and so takes account of the decisions on axis rate that a pilot is likely to make. The results also show that signal whitening has a modest detrimental effect on the alignment success rate. Signal whitening makes the relative magnitudes of the roll, pitch and yaw rates more similar. In most manoeuvres the yaw rates are very low, and the pitch rates are reasonably low compared to the roll rates. Whitening the signal increases the contribution of pitch and yaw rates in the alignment, but it also increases the contribution of the noise in those signals. In addition, any errors the pilot makes and the effect of wind and flight physics, which have not been accounted for in the FPA process, have the biggest impact on the pitch and yaw rates.

V. Path Assessment

The next stage in the development of the FPA process is to build a metric with which to assess the deviation from the flown path. All the major aerobatic competition disciplines define rules describing how this should be performed by a judge watching the flight in real time. In all cases judges start with a score of ten for each manoeuvre and subtract in half point intervals to a minimum of zero. Downgrades may be applied purely for mistakes within an element or at the manoeuvre level, by comparing the relative sizes of flown elements or overall positioning. The exact downgrades applicable to specific deviations may differ between disciplines.

In addition to the distinction between manoeuvre and element level downgrades, a categorisation can be made at the element level between those downgrades that apply to the flight path of the aircraft and those that apply to the orientation. For the loop and line primitive elements the downgrades applicable to the orientation consider only the error in roll angle. For the snap and spin primitive elements the pitch angle is also used as an aid to assessing whether the aircraft has stalled and some aspects of the flight path grading criteria are relaxed. Aerobatic sequences are constructed primarily from combinations of loop and line primitive elements, so the following discussion will focus on those. The enumeration below summarises the grading process for the loop and line primitive elements. This is not an exhaustive list, but is sufficient to demonstrate the judges workload and the kind of analyses that need to be catered for within the FPA process.

1) Flight Path
    1) Angle
       For line elements, and in the axial direction of loop elements, downgrades are applicable for angular deviations from the desired flight path. These vary between disciplines, but are generally between 0.5 and 9 marks for up to 90 degrees in heading error.
    2) Loop Diameter
       For loop elements flight path downgrades are applicable for perceived variations in the diameter of the loop. These vary from 0.5 or 1 mark for a visible deviations, to a maximum of 4 marks for severe (greater than 2 to 1) deviations. Some disciplines provide additional tools to judges to simplify radius assessment such as fixed downgrades for small visible straight regions in a looping element.
2) Orientation
    1) Roll Angle

    For all disciplines downgrades between 0.5 and 9 marks are applicable for angular deviations in roll angle up to 90 degrees.
  2) Roll Rate
    Variations in roll rate, when a roll is specified in a roll or loop element, are down-gradable by 0.5 or more marks per occurrence, depending on the severity.
3) Manoeuvre
  1) Roll Positioning
    Where a roll is integrated on a line, between two other non-rolling line elements the roll should be centred. Errors in roll positioning are down-gradable by 0.5 or 1 marks, up a maximum of 3 or 4 marks where no line is present before or after the line.
  2) Loop Diameter
    In some cases looping elements within a manoeuvre must share the same radius, downgrades of 0.5 or 1 marks for visible differences, with more marks being taken for major deviations.
  3) Positioning
    In some cases the overall position of a manoeuvre may be applicable, either by centering in front of the pilot or fitting within the edges of the aerobatic box.

The main challenge associated with assessing the quality of a flight path is in constructing a comparison to the ideal geometry, rather than in applying the downgrades listed above to that comparison. The most important output of the FPA process is therefore the exposure of an environment within which the grading criteria can be applied.

The temporal alignment process has identified the section of data corresponding to each primitive element in the sequence definition. The elements can then be accurately scaled based on measurements of the data that has been attributed to them. For line elements, this is the length of the line in the direction parallel to the corresponding template geometry. For loop elements the plane and proportion of loop are corrected and the radius is measured by fitting a circle to the projection of the recorded data on the corrected plane. Where rolls are integrated within a line or loop element the direction is identified by taking the sign of the mean body frame x axis rotational velocity.

Figure 10 shows a 3D visualisation of a comparison between a recorded loop element and template scaled to match the pilots intent. The starting position of each template primitive element is shared with the recorded data, but the end positions may not match if there are errors in the flown geometry. This results in the discontinuities in the template track between primitive elements.

Figure 11 shows the key results of the primitive element level analysis of the FPA process for the 7/8 loop primitive element from Figure 10. The loop radius, heading error and roll angle error are plotted against radial position around the loop. The loop radius is calculated by taking the distance to the centre of the loop, rather than by calculating the actual instantaneous radius at each point. This is the simplest approach, but future work is required to assess whether this best reflects the approach a human judge follows. In the loop element analysed here the radius appears to reduce towards the end, which is in fact caused by a tightening of the loop slightly earlier on. Heading error is calculated by taking the angle between the body velocity vector and the template loop plane. This reflects the definition in the judging criteria for all aerobatic disciplines for model aircraft and could be interpreted as three individual deviations of between 7 and 10 degrees. The roll error is calculated by calculating the angle between the flown body frame y axis and the corresponding template body frame xy plane. Here two discrete roll angle deviations are seen of 15 degrees and 7.5 degrees.

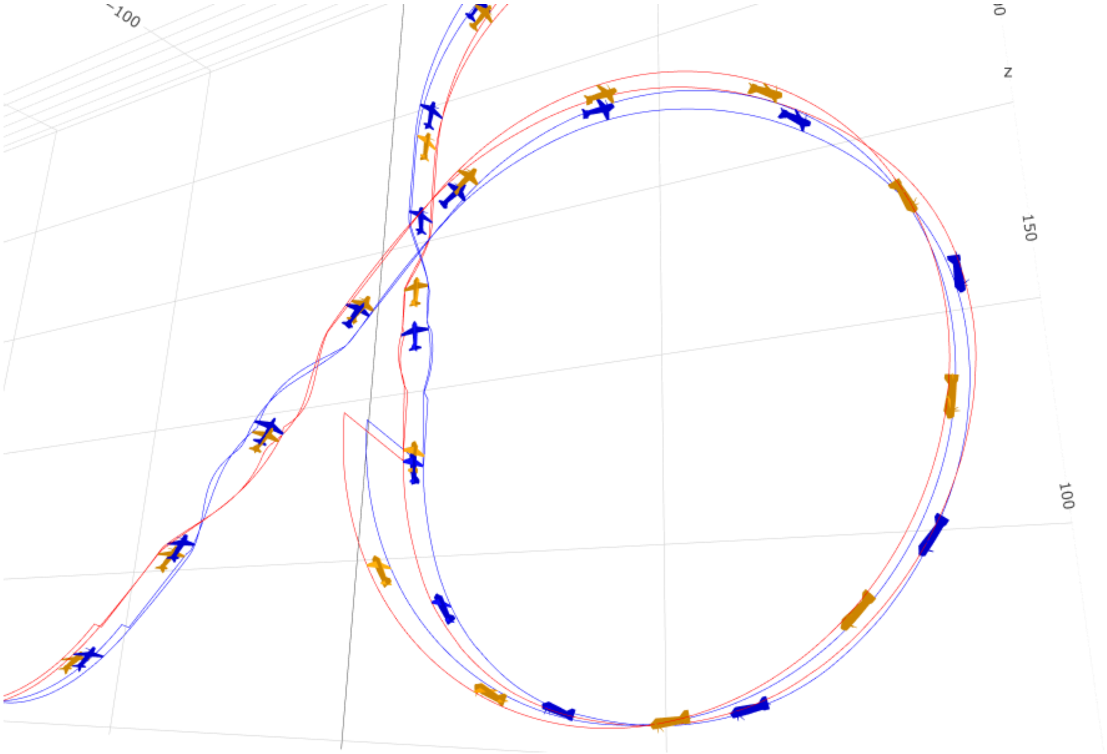


Fig. 10 **Comparison of recorded flight data (blue) to generated template data scaled to match intended flown geometry (yellow)**

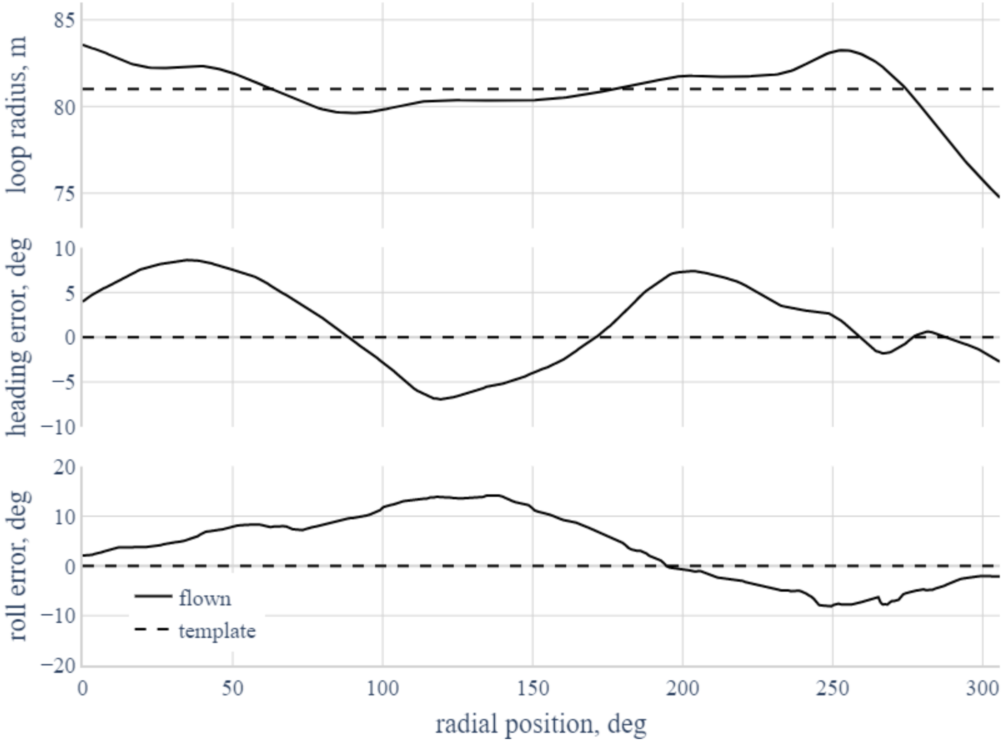


Fig. 11 Analysis of a flown loop element

VI. Conclusions

This paper describes the current state of development of a flight path assessment process for enabling the automated analysis of precision aerobatic manoeuvre geometry. At this stage the work offers a huge step towards the automation of judging for precision aerobatic competitions. The work is part of the Flight Coach project [1], which is already delivering tools that are receiving significant uptake within the aerobatic community. The Flight Coach Plotter is a free web app for plotting Ardupilot flight logs. PyFlightCoach [14] is an open source python package aimed at post processing flight data. PyFlightCoach was used to generate most of the figures and perform all the analyses in this paper.

The FPA process takes flight data recorded with a UAV flight controller during an aerobatic flight, extracts sections of it representing the individual manoeuvres and elements of an aerobatic sequence, constructs a template corresponding to the equivalent, perfect, flown sequence and presents the data in a way that allows the judging criteria to easily be applied. The temporal alignment process was shown to work well for 72.5% of a representative sample of flown aerobatic sequences, but it remains dependent on the quality of the flown sequence and on the template being representative. Further work is planned to improve the success rate of the alignment process and to formalise an environment which will be exposed for the encoding of grading criteria.

The methods developed as part of the FPA process have the potential to make wider contributions to work in fields such as Aerial Robotics and aircraft loads estimation. For machine learning based flight controllers the calculation of reward based on deviations from parametric template trajectories has the potential to be less risky than the optimal trajectories used in other work. For aircraft loads estimation the flight data collected in this project is a unique resource in itself, containing hundreds of repeat recordings of known sequences of manoeuvres. This labelled data could be used to train a Hidden Markov Model or Neural Network to recognise manoeuvres.

Acknowledgements

The funding for this study came in part from the CASCADE (Complex Autonomous aircraft Systems Configuration, Analysis and Design Exploratory) programme grant (EP/R009953/1).

References

[1] Uzieblo, A., David, T., Palmer, A., and Edwards, R., "FlightCoach," `https://www.flightcoach.org/`, 2021.

[2] Fletcher, L., Clarke, R., Richardson, T., and Hansen, M., "Reinforcement learning for a perched landing in the presence of wind," *AIAA Scitech 2021 Forum*, , No. January, 2021, pp. 1–14. https://doi.org/10.2514/6.2021-1282.

[3] Abbeel, P., Coates, A., and Ng, A. Y., "Autonomous helicopter aerobatics through apprenticeship learning," *International Journal of Robotics Research*, Vol. 29, No. 13, 2010, pp. 1608–1639. https://doi.org/10.1177/0278364910371999.

[4] Levin, J. M., Paranjape, A. A., and Nahon, M., "Agile maneuvering with a small fixed-wing unmanned aerial vehicle," *Robotics and Autonomous Systems*, Vol. 116, 2019, pp. 148–161. https://doi.org/10.1016/j.robot.2019.03.004, URL https://doi.org/10.1016/j.robot.2019.03.004.

[5] Kim, D., and Péchaud, L., "Maneuver recognition and prediction of empennage flight loads of general aviation aircraft," *1st AIAA, Aircraft, Technology Integration, and Operations Forum*, , No. c, 2001. https://doi.org/10.2514/6.2001-5273.

[6] Berry, J., Vaughan, R., Keller, J., Jacobs, J., Grabill, P., and Johnson, T., "Automatic Regime Recognition Using Neural Networks," *American Helicopter Society 60th Annual Forum*, , No. October, 2006.

[7] Rantanen, E., Talleur, D., Taylor, H., Bradshaw, G. L., Emanuel, T. W., Lendrum, L., and Charles L. Hulin, "Derivation of Pilot Performance Measures from Flight Data Recorder Information," , No. January, 2001, pp. 1–5. URL http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Derivation+of+pilot+performance+measures+from+flight+data+recorder+information#0.

[8] Wróblewska, A., and Linka, A., "Automation of training methods and judging of aerobatic flights," *Journal of KONES*, Vol. 24, No. 4, 2017, pp. 165–171. https://doi.org/10.5604/01.3001.0010.3138.

[9] Tridgell, A., Ferreira, F., Morphett, G., Walser, J., De Marchi, L., du Breuil, M., Barker, P., Mackay, R., Pittenger, T., Geyer, B., Olson, C., Castelnuovo, E., Shamaev, E., Staroselskii, G., José de Sousa, G., Beraud, J., Hall, L., Lawrence, M., Badaire, M., Denecke, M., Riseborough, P., Kancir, P., Mayoral Vilches, V., and Lucas, A., "ArduPilot," https://github.com/ArduPilot/ardupilot, 2021.

[10] Guide, M. E., “FAI Sporting Code Volume F3 Radio Control Aerobatics,” Fédération Aéronautique Internationale, 2013, Chap. 4, 2019th ed.

[11] Aresti Aguirre, J. L., “ArestSystem,” http://www.arestisystem.com/, 2012.

[12] Massa, R., Liefting, W., Guillemard, G., and Falck, C., “OpenAero,” https://openaero.net/, 2021.

[13] Aung, Z., Sim, K., and Ng, W. S., “TrajAlign: A method for precise matching of 3-D trajectories,” *Proceedings - International Conference on Pattern Recognition*, 2010, pp. 3818–3821. https://doi.org/10.1109/ICPR.2010.930.

[14] David, T., “PyFlightCoach,” https://github.com/PyFlightCoach/PyFlightCoach, 11 2021.