

Analysis of Visualization Systems in Flight Simulators

Luis Del Barrio¹, Wojciech Korek²³, Murat Millidere⁴,
Robert Harrison⁵, and James Whidborne⁶

Cranfield University, Cranfield, England, MK43 0AL, United Kingdom
Silesian University of Technology, Gliwice, Poland

This paper details an analysis of different visualization systems for use in an academic flight simulator, Future Systems Simulator (FSS). First, an overview of off-the-shelf flight simulators is done, detailing the primary features of flight simulators such as Flight Gear, Prepar3D, X-Plane, and Microsoft Flight Simulator (2020). Then, the current setup of the FSS is presented (which uses *FlightGear*), followed by the process of introducing *X-Plane* as a scenery-generation tool. To conduct a comparative analysis between *FlightGear* and *X-Plane* visual systems, a total of twelve participants with varying levels of experience were invited to participate in the study. The participants performed flight trials in a simple landing scenario at Heathrow Airport. Additionally, the more complex approach at London City Airport was performed with a group of only four highly experienced participants. Participants then gave their feedback and completed a questionnaire. The data from their attempts were recorded for qualitative and quantitative comparison. The results were analyzed to determine which of the two visual systems could be used in the FSS moving forward.

Nomenclature

<i>AA</i>	=	<i>Anti-Aliasing</i>
<i>AVG</i>	=	<i>Average</i>
<i>CAS</i>	=	<i>Calibrated Airspeed</i>
<i>CPU</i>	=	<i>Central Processing Unit (Processor)</i>
<i>FAA</i>	=	<i>Federal Aviation Administration</i>
<i>FBW</i>	=	<i>Fly-By-Wire</i>
<i>FG</i>	=	<i>FlightGear</i>
<i>FPM</i>	=	<i>Feet Per Minute</i>
<i>FPS</i>	=	<i>Frames Per Second</i>
<i>FSS</i>	=	<i>Future Systems Simulator</i>
<i>FSX</i>	=	<i>Flight Simulator X</i>
<i>GA</i>	=	<i>General Aviation</i>
<i>GPU</i>	=	<i>Graphics Processing Unit (Graphics Card)</i>
<i>HMI</i>	=	<i>Human-Machine Interface</i>
<i>KTS</i>	=	<i>Knots</i>
<i>MPS</i>	=	<i>Metres Per Second</i>
<i>MSFS</i>	=	<i>Microsoft Flight Simulator</i>
<i>P3D</i>	=	<i>Prepar3D</i>
<i>PAPI</i>	=	<i>Precision Approach Path Indicator</i>

¹ Postgraduate Student, Centre of Aeronautics, SATM, Cranfield University, MK43 0AL, United Kingdom.

² Research Fellow, Centre of Aeronautics, SATM, Cranfield University, MK43 0AL, United Kingdom.

³ PhD Student, Faculty of Automatic Control, Electronics and Computer Science, Silesian University of Technology, Gliwice, Poland

⁴ Research Fellow, Centre of Aeronautics, SATM, Cranfield University, MK43 0AL, United Kingdom

⁵ Head of National Flying Laboratory Centre, SATM, Cranfield University, MK43 0AL, United Kingdom

⁶ Prof., Centre of Aeronautics, SATM, Cranfield University, MK43 0AL, United Kingdom.

SDK = *Software Development Kit*
UDP = *User Datagram Protocol*
UI = *User Interface*
XP = *X-Plane*

I.Introduction

Flight simulators have been around almost since the invention of flying in various forms for pilot training. This is a common application but has probably been overtaken in popularity by hobbyists and video game players using flight simulators for recreation [1]. Flight simulators also play a huge role in aircraft design and research [2]. Up until the 1960s, testing and validation of new systems were usually done by flight trials in aircraft prototypes. However, by using simulation, known as engineering flight simulators [3], problems, flaws, and errors in the design and various sub-systems can be identified earlier, reducing cost and risk.

Compared to civil/commercial flight simulators, engineering flight simulators do not necessarily need as many components to create an “immersive” environment for pilot training, and the simulator does not need to meet any specific approvals. What is particularly important is data acquisition. When the systems are tested in the simulator, they can then be tested in flight trials to corroborate the data acquired from the simulator. To achieve accurate results, it is critical for the flight dynamics model of the aircraft to accurately replicate the behaviour of the real aircraft.

The Future Systems Simulator at Cranfield University [2] is a Rolls-Royce-funded engineering flight simulator. It was designed to research and test the implementation of various aircraft systems, particularly the propulsion system and its interaction with the pilots. The FSS has a modular and reconfigurable cockpit with the idea that these systems can be tested for different aircraft types and configurations.

In the FSS, like most other flight simulators, the visual system is vitally important because it provides the visual cues to the pilot; it gives them a reference to a “world” outside the cockpit. One objective of the work reported in this paper was to compare the visualization system that is currently in use at the FSS, with an alternative that could potentially replace it. To evaluate the two systems, piloted simulation flight trials were conducted. The authors are not aware of any other similar studies. This paper reports on this experience and provides some guidelines on how to perform such evaluations.



Figure 1 The Future Systems Simulator at Cranfield University

The following section provides a review of the technology and some of the alternative visualization systems currently available. Section III describes the FSS and its architecture and relevant features. The conduct of experiment is described in Section IV, and the experiment results are given and discussed in Section V. Conclusions and suggestions for further work are presented in the final section.

II. Literature Review

Flight simulators can be a powerful tool for pilot training and research, allowing for the safe and controlled evaluation of existing and alternative designs. Due to the high-stakes environment of aviation, any new concepts must undergo a series of rigorous tests and adhere to strict standards before being deemed airworthy. The consequences of an airborne failure in an aircraft are severe, making it crucial to ensure the safety of the passengers and crew. However, modern technology and research facilities, such as engineering flight simulators, allow for the safe testing of new ideas in controlled environments [4]. There were numerous studies in this area [5] [6]. None of them, however, focused strictly on the scenery generation aspect and the differences between various tools. The visual system of a simulator provides a real-time image seen outside the cockpit from the pilot's point of view. This includes areas important for aviation such as airfields, landmarks, and terrain features such as rivers, roads, trees, mountains, and other aircraft to create an experience for the pilot that is realistic enough to accurately perform the required task. The visual cues may vary on the task; for example, in a helicopter, seeing the grass on the ground realistically react with the wind is important, but for a fixed-wing landing, this is not as important.

To maintain a feeling of immersion for the pilot, ideally, the visuals will update at the same rate or faster than the aircraft's equations of motion are being calculated. The equations of motion in a real-time computer are usually rendered at 50Hz, so to reduce the delay between the aircraft's equations of motion and the visual system, the visual system should also run at a minimum of 50 Hz. Any lower frequency will cause the delay to feel very noticeable, thus making the pilots flying the simulator lose a sense of immersion.

The quality of a visual system can depend on various things, one of the main ones being the graphics engine. This includes details such as the quality of the textures, so buildings, terrain and aircraft ideally do not look blurry; how detailed the objects are, i.e., how many polygons they have, with more being better as that can create more defined shapes; how the aircraft interacts with the environment, like how the grass or leaves move with the wind generated from the aircraft's engines; and how the weather is modelled into the software [3].

Of course, higher detail and fidelity require more powerful hardware to run the visual system, but developers can also find ways to optimize the software while maintaining a good/high level of detail.

The visual systems investigated for the FSS are confined to those available off-the-shelf. These are actually all fully functional flight simulators; however, since the FSS has its flight dynamic models, only the graphical fidelity of the available systems is considered.

A. Flight Gear

FlightGear is an open-source flight simulator that was released in 1997. This is the visualization system that has been used on the FSS since its inception in 2020. Being open-source means that it can be modified in many ways and is compatible with many applications. Since it is open source, users from the community have helped to keep this simulator updated for the past 25 years. This has helped *FlightGear* stay relevant as a commercial flight simulator that is highly compatible with other software [7]. The version in use in the FSS is 2018.3.5, released in 2020.

B. Prepar3D

An option for an alternative visual system in the FSS is *Prepar3D*. It is a flight simulator developed by Lockheed Martin as a commercial version of their professional flight simulators. The first version was released in 2010 and has since been receiving updates every 2-3 years, focused on adding support for more modern technologies, as well as improving various aspects of the simulator, such as terrain data. It has an SDK available to develop various modifications and plugins, which is used by its community to create a lot of content for the simulator.

However, at its base, *P3D* is built using Microsoft's *Flight Simulator X (FSX)* [8] code. *FSX* was released in 2006, and whilst *P3D* has managed to stay up to date with its flight models and dynamic scenarios, the visuals have not improved much [9].

C. Digital Combat Simulator

Digital Combat Simulator World (DCS) is a flight simulator primarily focused on recreating combat aircraft and various scenarios for these. Developed by 'Eagle Dynamics', it was first released in 2008 and has continuously been receiving updates ever since, with new updates coming out every couple of weeks in their open beta branch. Every few GPU and CPU generations, the developer will update the engine the simulator is coded in, to try to optimize its many systems, improve the compatibility with modern technologies, add or update features, and update the graphical fidelity of the simulator. It is safe to say that *DCS* is one of the best-looking flight simulators out of the box, with highly detailed terrain and aircraft models.

Most of the content in *DCS* comes from 'official' 3rd party developers as paid add-ons to the sim. This means there is no public SDK, so modifying aspects of the simulator to work with applications such as Simulink is not possible without official support from the developers. And unlike most simulators, which offer the ability to fly

in any part of the world, albeit with low detail, DCS only offers the ability to fly in the few areas of the world that Eagle Dynamics, or one of the 3rd party developers, have released as an add-on. So, this simulator is limited in where the user wants to fly as well as its support for 3rd party modifications [10].

D. Microsoft Flight Simulator 2020

Microsoft Flight Simulator (2020) (MSFS) is probably one of the most ambitious flight simulators developed to this date. When announced, most of the flight simulation community was cautiously optimistic about the promises made by the developers, Asobo. However, when it was released at the end of 2020, most of the promises seemed to have been fulfilled, and this is in great part thanks to strong support from Microsoft [citation?]. Instead of having to rely on completely automatic generic scenery or having to download large amounts of custom-made scenery by other developers, *MSFS* uses Microsoft's Bing Maps satellite data, as well as their cloud computing service, Azure, to generate and download scenery to the user's local temporal storage on the go.

Asobo provides regular updates, making improvements to various aspects of the simulator, improving performance, flight models and overall user experience, as well as adding more detailed scenery for various parts of the world. This also includes adding photogrammetry data to various cities, further increasing the level of detail. All this new terrain is added to the simulator for free, no extra purchases must be made to access this. However, some of the more detailed aircraft that Asobo adds are paid for.

MSFS also has extra content created by third-party developers, or other users of the simulator, both free and paid. This includes aircraft, detailed scenery, and add-ons such as improved airport AI or ATC, for example. Like other simulators, *MSFS* has an SDK, allowing users to create plugins and addons to modify various parts of the simulator, allowing the possibility for communication with *Simulink*.

The consensus seems to be that *MSFS* provides the best experience out of the box. However, due to it having been released just over 2 years, it lacks content that other simulators may have had more time to implement, such as new aircraft, detailed scenery or extra features. For example, at the time of performing this study, *MSFS* lacked multi-monitor support which is needed for a simulator like the FSS [11].

E. Comparison between *FlightGear* and *X-Plane 11* and *X-Plane 12*

At first glance, the main difference between *FlightGear* (*FG*) and *X-Plane 11* (*XP 11*) is probably the texture quality. Even though *FG* has been getting constant updates during its life cycle, the difference in graphical fidelity of 25 years is quite noticeable. *XP 11* also has more accurate weather simulation, with more realistic-looking wind effects and clouds. *FG*, whilst it may not look as realistic as *XP 11*, has managed to keep up in graphical fidelity to a certain degree, but the weather simulation, and especially the clouds, are much less impressive than *XP 11*.

FG has an advantage over *XP 11* in its loading time: having a lower level of detail and being optimized for use with slower hard-disk drives, *FG*'s loading times are considerably faster than *XP 11*. However, this difference could be made negligible if *XP 11* is installed on a much faster solid-state drive.

FG has a complicated setup: instead of using a graphical interface for the settings, most of the settings need to be programmed using commands in an XML file. In comparison, *XP 11* uses a graphical UI, making it much easier to change and view any setting necessary.

FG, as mentioned earlier, is an open-source program, which means it can be modified easily by other users from the community. *XP*, on the other hand, is not open source, so its code is not accessible for users to modify. However, it does have an in-depth SDK, allowing users to create add-ons that can utilize the simulator's data in many ways.

Both simulators have built-in UDP communication features to transfer and receive data from third-party programs, such as *Simulink*.

XP 12 is largely the same simulator as *XP 11*; it is essentially an upgrade from the previous version. What has changed in the new release is mainly visual, new lighting effects, reflections, higher resolution textures and slightly more detailed default scenery. It is worth noting that the flight models used in the FSS are separate from those running in *X-Plane*. Any improvements in flight dynamics models, therefore, do not affect the visual systems discussed in this paper. Hence, this paper is mainly focused on the visual fidelity of the simulator.

III. Simulator Structure

The FSS setup consists of three computers; a real-time computer, *dSpace*, 'SCALEXIO' [12] where the aircraft model code is executed; a PC that interfaces the HMI displays and flight controls; and a second PC that displays the visualization software. The computers are connected to the same local network (LAN) through a HUB. They communicate using UDP (User Datagram Protocol), which is advantageous as UDP works best when sending small packets of data in a smaller-scale network, as well as most of the applications being used have built-in UDP communication. There is one further computer connected to the network for data recording.

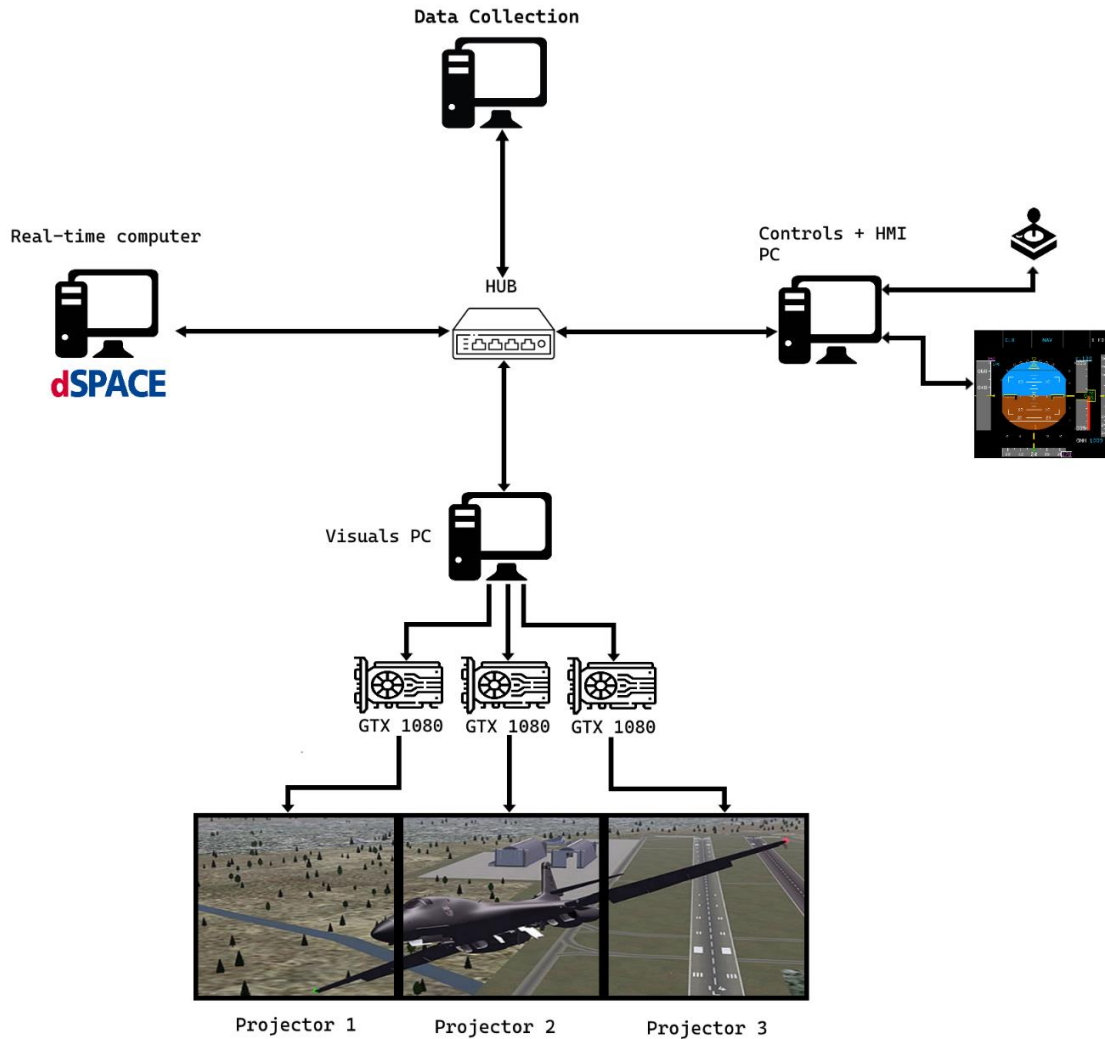


Figure 2 Future System Simulator Setup

The aircraft models are developed using *MATLAB/Simulink*. The *dSpace* computer calculates the states and rotation of the aircraft and sends the position information to the visualization software.

The PC running the visuals is equipped with three graphics cards, a GTX 1080, each connected to one projector that displays a different image. This allows there to be three different instances of *FlightGear*, each running on one GPU. The screens are arranged around the cockpit in a 200° field-of-view to ensure the most immersive experience possible, with the images being displayed from three projectors above the cockpit. The FSS is an academic simulator with a focus on research; this is useful as it allows data from flight trials and any tests to be analyzed objectively. It can be used, for example, to measure the pilot's performance in scenarios with different variables such as weather and time of day.

Because the second visualization system, *X-Plane*, is more computationally demanding than *FlightGear*, it was necessary to update the current GPU setup to allow it to run in the FSS. The previous three GTX 1080s were replaced by just one, but more powerful graphics card (RTX 3090Ti).

IV. Conduct of Experiment

This study includes two distinct scenarios: the first involved a comparison between *XP 11* and *FlightGear*, wherein a total of twelve participants with varying levels of experience were invited to perform flight trials in a simple approach and landing scenario at Heathrow Airport. In the second scenario, a comparison between *XP 12* and *FlightGear* was conducted, with a group of only four highly experienced participants performing the more complex approach at London City Airport. In both scenarios, there were no abnormal conditions applied to the flight, there was no wind, and the time of day was noon to provide the most visibility. Participants had five attempts to perform a landing in both *FG* and *XP 11*.

Since visuals can be a subjective area, it was decided that participants would perform a certain scenario, repeated for *FG* and *XP*, and then would be given a questionnaire where they gave their feedback to gather relevant results on each visual system.

To corroborate these qualitative results, flight data was recorded using the FSS's built-in data recording module. This allowed for the comparison of both subjective and objective data.

The aircraft used in this trial was a model of the Gulfstream G500. The choice was based on the fact that the model of this aircraft was available in working condition in the FSS and had already been interfaced with *XP*. Also, this aircraft has FBW and auto trim, which could help participants with less experience in flying. The gear and flaps were already deployed to decrease the workload on the participant and lower the head-down time at the instruments during the trial.

Participants were given a set of questions that were used to determine their level of relevant experience, such as if they had a pilot's license, how many flying hours they had, and if they had any experience with simulators.

After conducting flight test trials in the first scenario at Heathrow Airport, an upgrade to the scenery generation computer (Ryzen 9 7900X, RTX 3090Ti and 64GB DDR5 RAM) made it possible to create a new scenario running *XP 12*. In this scenario, there were 4 new participants, of which all were pilots who have a lot of experience, mainly with commercial aircraft. In the scenario, participants were tasked with approaching and landing at London City Airport, three times in each visual system.

Landing at London City Airport was selected as the second scenario. Although it is a challenging airport to land at, participants had considerable flight experience, so they quickly became comfortable after a few practice approaches. Also, since London City airport is located right in the heart of London, participants could use visual cues, such as the river Thames, or the high-rise buildings of Canary Wharf, which the pilots had to use as a reference to start the descent into the runway.

Participants would start their approach 2000 feet over the airport, flying at 200kts, heading 304°. Following the instrument approach procedure chart of London City airport, participants had to continue level flight until they were 5nm away from the airport, and then start a left-hand turn to a heading of 96° to line up with the runway. They then had to continue level until 3.4nm away from the airport, just above Canary Wharf, and then start their descent to the runway. For more information about the landing flight path chart at London City Airport, the reader is referred to Figure A.1 in Appendix.

Following the checkpoints on the approach charts provided to the participants, when starting the left-hand turn, they would set the flaps setting to 1 (out of 3) to maintain a speed of 180kts through the turn. When leaving the turn, they would set the flaps to 2, and lower the landing gear to maintain a speed of 150kts. When they were 3.4nm from the runway, at the start of the descent, they would lower the flaps to 3 and extend the airbrake, to maintain a steady speed of around 120kts to touchdown.

A. Subjective Ratings

After each flight trial, participants were given a questionnaire to fill in, asking about their experience in the scenario.

The questionnaire consisted of four main questions, where participants could mark each answer with a score between 1 and 7 (1 being excellent/easy, and 7 being inadequate/hard). These were:

1. How easy was it to perceive your altitude above the ground in relation to visual cues?
2. How easy was it to perceive the distance of objects on the ground?
3. How easy was it to perceive the airspeed?
4. Did you feel any transport delay (latency) between your inputs and the aircraft's response?

Since these ratings are subjective, it must be noted that the rating scale for each participant might be different. Whilst these biases could affect the results, the higher number of participants would ideally average the results out. Additionally, participants were asked an open-ended question, where they had to state which visualization system, they felt more comfortable with and write the reasons for their choice.

B. Objective Ratings

To support the qualitative data, as mentioned before, data from the simulator is recorded to analyze the participant's performance. Ideally, this data would corroborate the answers given by the participants in the questionnaire; however, even if it did not, it would be objective data that could be compared to the subjective data.

As mentioned earlier, the FSS can record any information that is needed for this study. However, due to the limited scope of this study, the information chosen to be used to analyze the participant's performance is; altitude, to determine how smooth the approach was; airspeed, to see if the participants were able to have a touchdown speed within the recommended limits for the aircraft; and vertical speed, to determine the touchdown speed. All this data together can also be used to determine how smooth, or how controllable the approach was in a specific attempt.

Participants were asked which attempt they thought was the best for each visualization system. This was done to reduce the amount of data that needed to be analyzed. However, this could introduce some bias into the experiment, as participants could misjudge which attempt was their best one, and if a participant had 4 attempts where they were unable to land on the runway, but had one stable attempt, the 4 other attempts would not be accounted for. This could be remedied in further study.

V. Results and Discussion

A. Subjective Ratings

After the flight trials were completed in the first scenario, the answers from the participants to compare both visualization systems were analyzed.

Table 1 Scenario 1: *FlightGear* Flight Trial Answers

Question	Participant												Average
	1	2	3	4	5	6	7	8	9	10	11	12	
1	3	4	5	4	4	4	2	1	4	3	X	2	3.27
2	4	4	4	4	4	4	2	1	5	4	X	2	3.45
3	2	3	6	4	4	5	2	2	3	3	X	2	3.27
4	4	4	4	5	3	3	5	1	3	6	X	2	3.45

Table 2 Scenario 1: *X-Plane 11* Flight Trial Answers

Question	Participant												Average
	1	2	3	4	5	6	7	8	9	10	11	12	
1	3	3	3	3	3	3	4	1	2	4	X	3	2.9
2	3	3	2	3	3	3	4	1	3	3	X	3	2.81
3	2	3	3	4	3	5	3	2	3	3	X	3	3.09
4	3	4	3	3	3	3	3	1	3	7	X	2	3.18

Note: Participant 11 did not complete the score-based questions but did answer the open-ended question.

As is evident from Table 1 and Table 2, *X-Plane 11* manages to achieve an average lower score (better) compared to *FlightGear* in the four questions. The biggest difference is in the second question, which is related to being able to perceive the distance from objects (buildings, trees, roads, etc).

On *FlightGear*, this question got the highest score, whilst, on *X-Plane 11*, it achieved the lowest score. This is likely because, in *FlightGear*, there are very few objects rendered in the world. There are a few low-detail buildings that can be seen on the ground, but these are very sparse with the autogen (automatically generated scenery) the simulator uses for the scenery.

X-Plane 11, on the other hand, renders many more objects in the world around the aircraft. And whilst they might not render as far in the distance compared to *FlightGear*, there are many more, and when the aircraft gets closer to the ground, and more objects are in view, it is much easier to determine their distance.

Questions 1 and 3 which concern altitude and airspeed perception, respectively, also scored better in *X-Plane 11*. This is probably due to the same reasons as question 2; more objects in the world allow the participants to get a better perception of where they were located using the visuals.

For question 4, which is about the latency between the two visual systems, participants gave *X-Plane 11* a slightly better score. This is somewhat surprising as *XP 11*'s performance was worse compared to *FG*'s (more on this later).

Dividing the average scores between the experience level of the participants, by adding all the scores together and then dividing them by the number of participants in each category, gives a better insight. The results are shown in Table 3.

Table 3 Scenario 1: Ratings with respect to Experience

Experience	Participants	Average (<i>FlightGear</i>)	Average (<i>X-Plane 11</i>)
1	2,4	4	3.25
2	3,5,7,	3.75	3.08
3	6,8,9	3.25	2.5
4	1,10,11,12	3.08	3.25

Similar outcomes were observed as in the previous analysis. However, a significant contrast was observed in the advanced experience level (3) participants, who reported a higher preference for *X-Plane 11*, with a mean score of 0.75 compared to *FlightGear*. This could be attributed to that these participants had much more experience with desktop flight simulators (even compared to the pilots), and they were more aware of the lower fidelity visuals of FG compared to *XP 11*.

The pilots who participated in the study gave *FlightGear* slightly better scores than *X-Plane 11*. They were the only set of participants to do this. Most of the pilots made note that, whilst *XP 11* looked more “realistic” up closely, with the higher amount of detailed objects, in the distance, textures became blurry, and it was harder to distinguish the runway from the scenery around it. Whilst *FG*, despite having less detail when close to the ground, the terrain does not become blurry in the distance, making it much easier to distinguish features in the terrain that are further away, such as distinguishing the runway from the rest of the scenery.

Table 4 lists the results of which visualization system participants felt more comfortable with. Most said they felt more comfortable with *XP 11*, whilst only two said they felt more comfortable with *FG*.

Table 4 Scenario 1: Participant’s Preferences

Participants	1	2	3	4	5	6	7	8	9	10	11	12
<i>FlightGear (FG), X-Plane 11 (XP) or Neither (N)?</i>	N	XP	XP	XP	N	XP	XP	FG	XP	FG	XP	FG

Most participants gave the reason that *XP* looked more realistic, especially with the higher number of buildings and detailed airport once they landed. However, they also mentioned that despite this, *FG*’s colours looked better. This is likely because the default scenery in *XP* appeared to be from the middle of summer, so most of the ground had more muted and yellow colours, whilst, in *FG*, the ground had a more vibrant, green colour.

Two of the pilots and one “advanced” participant, mentioned that the PAPI lights (the set of 4 lights next to the runway to assist pilots in maintaining the recommended glideslope) in *XP 11* were harder to notice compared to *FG*. This is because in *FG* the lights are exaggerated, making it much easier to see even in harsh conditions. Whilst this might not be realistic, it does help in a simulator setup like this. In *XP 11*, however, the opposite seems to be true. The lights are more muted than they would be in real life, making them much harder to spot compared to *FG* and real life.

Three participants mentioned a noticeable graphical performance difference between *FG* and *XP 11* that affected their experience. This is likely because *FG*, ran at between 30 to 40 fps, whilst *XP 11* ran between 20 and 30 fps with noticeable stutters every few seconds.

Table 5 lists the results of which visualization system participants felt more comfortable with when they conduct flight tests in second scenario. Most participants reported feeling more comfortable with *FlightGear (FG)*, whereas only one participant stated feeling more comfortable with *X-Plane 12 (XP 12)*. The preference for *FG*’s older and less detailed scenery over the more recent *XP 12* is attributed to the higher framerate of *FG*. With lower graphics demand, *FG* was able to achieve a stable framerate of 50-60fps, while *XP 12* struggled to maintain anything above 30fps and was unstable. Participants noticed the significant difference in framerate and noted that despite the much more detailed and realistic scenery of *XP 12*, the low and unstable framerate resulted in added lag between their inputs and the screen, causing pilot-induced oscillation and leading to overcorrection while flying.

Table 5 Scenario 2: Participant’s Preferences

Participants	1	2	3	4
<i>FlightGear (FG), X-Plane 12 (XP) or Neither (N)?</i>	XP	FG	FG	FG

B. Objective Ratings

In addition to the qualitative data analyzed, the objective data recorded from the flight trials can be used to see if the user’s experience matches their actual performance in the scenario.

Starting by comparing all the attempts in *FG* with all the attempts in *XP 11*. The aircraft used in the trials, the G500, was not properly trimmed at the beginning of the scenario, giving it a tendency to pitch down, right at the start. Therefore, in all the graphs, especially the height graph, there is a slight sudden descent right at the beginning, where users had to pitch up to stabilize the aircraft. After a couple of seconds, the auto-trim in the aircraft adjusted properly and was able to help maintain a steady approach.

Looking at the results in Figure 3, which shows the landing trajectories for different participants, there were a few participants who had more trouble maintaining a stable approach in *FG* compared to *XP 11*. However, if we disregard these attempts, the rest of the participants seemed to perform slightly better in *FG* than in *XP 11*. This can be seen by the tighter spread from the different attempts.

Dividing the graphs between experience levels and grouping both visualization systems can help visualize this slightly better.

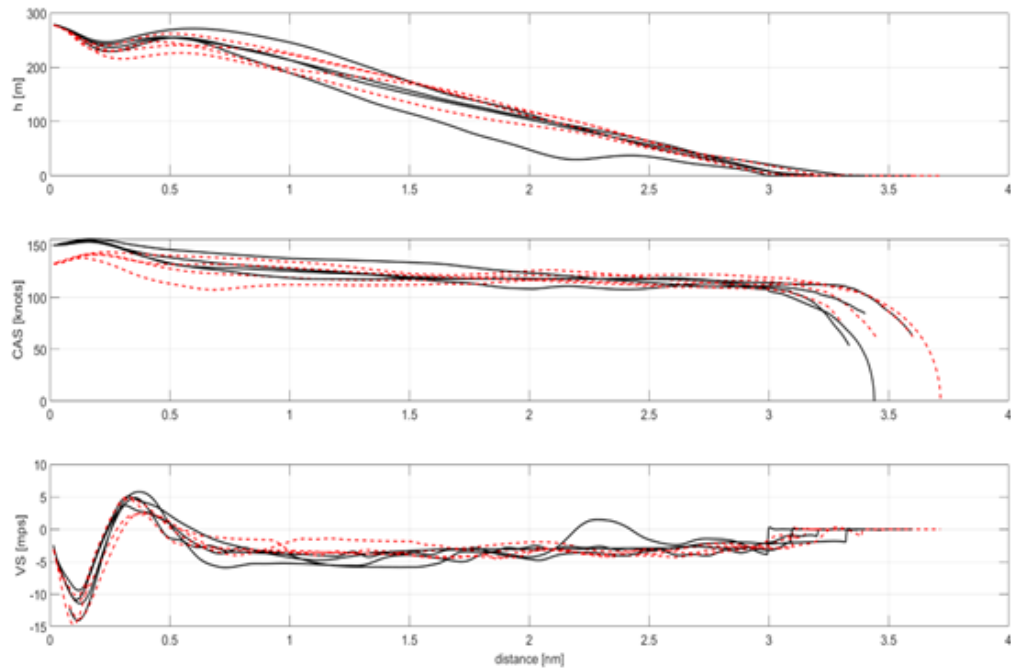


Figure 3 Scenario 1: Evaluation of *FlightGear* and *X-Plane 11* for Pilots

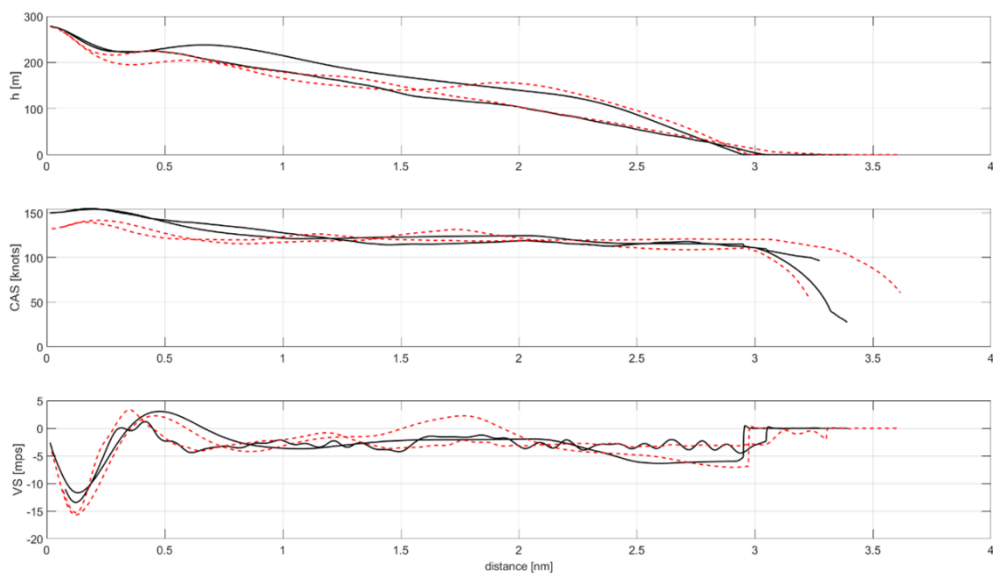


Figure 4 Scenario 1: Evaluation of *FlightGear* and *X-Plane 11* for Advanced Participants

In Figure 3 and Figure 4, black solid lines correspond to *X-Plane 11* and red lines correspond to *FlightGear*. In Figure 3, there is a clear outlier compared to the rest of the attempts. One of the pilots seems to have misjudged his performance in his attempt in *FG*. However, the rest of the attempts show that in both visualization systems, pilots were more aware of how their actions affected the aircraft and were able to predict what the aircraft was going to do without having to rely as much on the visuals, which explains why in both *FG* and *XP 11* the performance in the landing attempts does not seem to differ much. In Figure 4, for the advanced participants, this seems to be largely the same. In both visualization systems, the performance of both participants was similar. There was one participant who had some trouble controlling the aircraft as they were not used to it, and their performance was reflected equally in both visualization systems, whilst the other participant did not have much

trouble controlling the aircraft in both visualization systems and their performance was much the same, almost following the same trajectory as can be seen from the height graph.

The biggest noticeable difference is that in *X-Plane 11*, participants were able to control the initial pitch-down tendencies the aircraft had.

In the next figures, the touchdown speeds and calculate the general performance for each visual system are focused. For a better comparison, the graphs are divided between experience levels.

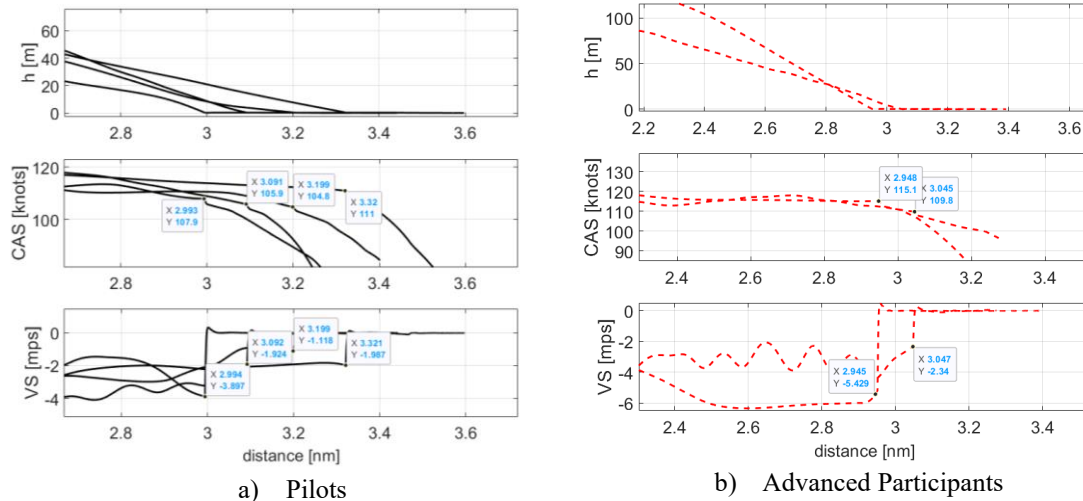


Figure 5 Scenario 1: Touchdown Speeds in *FlightGear*

In Figure 5 on the left, we have the trials from the pilot participants with both their vertical speed and airspeed when the wheels touched the ground. On the right graph, we have the same but for the participants with “advanced” experience. The average touchdown speed for the pilots was 107.4 knots, and the vertical speed was 2.2315 mps or 439.3fpm. For the advanced participants, this was 112.45 knots airspeed on touchdown, and 3.89 mps or 764 fpm descent rate.

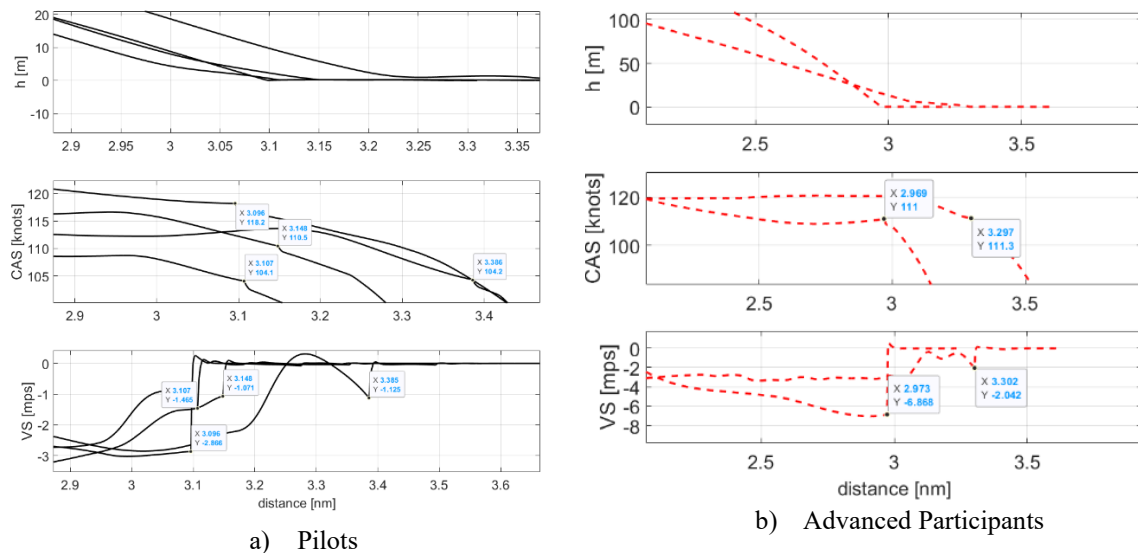


Figure 6 Scenario 1: Touchdown Speeds in *X-Plane 11*

Figure 6 display the same data as the Figure 5 but with data from *XP 11*. The average speed for pilots in *XP 11* was 109.25Kts airspeed on touchdown, and 1.63mps or 321.2fpm descent rate. For the advanced participants, this was 111.15Kts airspeed, and 4.46mps or 877fpm descent rate.

Comparing these averages, the pilots were able to stay just under the recommended landing speed for the G500 of 100 knots in *X-Plane 11* at 109.5 knots, whilst in *FG* they went down to 107.4Kts, which is not much of a difference and, likely, if all the attempts were included instead just one from each participant, the difference would not be as much as they relied more on the instruments to gauge their speed.

However, the vertical speed did have a big improvement; 439.3fpm in *FlightGear* compared to 321.2fpm in *X-Plane 11*; this is a big improvement, likely due to what the participants said in their questionnaires about *X-Plane 11* looking more realistic, which likely helped with ground perception to assist the pilots in performing a smoother landing.

For the advanced participants, this is shown to be the opposite, with a higher descent rate in *X-Plane 11* of 877 fpm compared to 764 fpm in *FlightGear*. This is still a relatively high descent rate for both cases, above the recommended descent rate of the aircraft. It is probable, that because there were only two participants with “advanced” experience, the performance of one greatly affected the other. With more participants, this would probably not have been as noticeable.

VI. Conclusion and Future Work

This paper detailed an analysis of two visualization systems, *FlightGear* and *X-Plane*, for use in an academic flight simulator, Future Systems Simulator (FSS). This study worked on two distinct scenarios: 1) the first involved a comparison between *FlightGear* and *X-Plane 11*, wherein a total of twelve participants with varying levels of experience were invited to perform flight trials in a simple approach and landing scenario at Heathrow Airport. 2) a comparison between *FlightGear* and *X-Plane 12* was conducted, with a group of only four highly experienced participants performing the more complex approach at London City Airport. In both scenarios, there were no abnormal conditions applied to the flight, there was no wind, and the time of day was noon to provide the most visibility. Participants had five attempts to perform a landing in both *FlightGear* and *X-Plane* visual systems.

In general, the main factors that affect the visual systems in flight simulators include resolution, field of view, refresh rate, anti-aliasing, and texture quality and level of detail.

- 1) Resolution refers to the number of pixels that make up the visual image, and higher resolutions typically provide clearer and sharper visuals.
- 2) Field of view is the angle and extent of the simulated visual environment that the pilot can see from the cockpit, and a wider field of view can improve situational awareness and depth perception.
- 3) Refresh rate refers to the frequency at which the simulator's visuals are updated, measured in frames per second (fps), and higher refresh rates can provide smoother and more responsive visuals.
- 4) Anti-aliasing is a technique used to smooth out the edges of objects in the visual image and can improve the overall visual quality.
- 5) Texture quality and level of detail refer to the level of detail and realism in the visual representation of the environment, including textures, lighting, shadows, and weather effects. Higher texture quality and level of detail can provide a more immersive and realistic visual experience.

When comparing two visual systems, with the resolution and field of view being the same, feedback on the other three factors can be given. Based on research, it can be concluded that the importance of the three factors, namely refresh rate, anti-aliasing, and texture quality and level of detail, are in increasing order. A higher refresh rate can provide smoother and more responsive visuals, which is crucial in providing an immersive and realistic training experience. Anti-aliasing smooths out the edges of objects in the visual image, resulting in a more realistic and visually pleasing environment. Texture quality and level of detail add depth and realism to the simulated environment, providing a more immersive and engaging experience for the trainee. Therefore, in order to create an optimal visual system for flight simulators, it is essential to prioritize these three factors in the design and development process. The more detailed conclusion about these two scenarios can be found as follows:

A. FG and XP11

To conclude, whilst participants may prefer *X-Plane 11* over *FlightGear*, there is not a noticeable difference in the participant's performance. Only the pilots had a noticeable improvement from one visual system to another. It is likely that, in a small scenario like the one in this study, landing on Heathrow, the visuals did not have a big effect on the participant's performance. And in this scenario, that is understandable.

There may have been some bias in the participant's results, as in most cases, participants had their first landing attempts in *FlightGear* and then in *X-Plane 11*, which could have allowed participants to get more comfortable with the FSS and perform better in later attempts. Participants were also told which visualization system they were using before their trial started, and, if they were familiar with the visual systems, like *X-Plane 11*, they may have had some bias towards that specific visual system.

B. FG and XP12

Following the upgrade of the system's CPU, RAM, and GPU, *FlightGear's* performance increased to 60fps. With this improvement, a comparison between *FlightGear* and the more recent release *X-Plane 12* was conducted. However, the results did not yield the expected outcome.

Whilst the main objective of this paper was to analyze how different visual systems, and their visual fidelity, can affect a pilot's performance and perception of the environment, there was a pattern in the participant's answers; they preferred the older, less detailed scenery of *FlightGear*, as opposed to the more recent, *X-Plane 12*.

Why is this? Well, the answer is quite straightforward, framerate. *FlightGear*, being less graphically demanding, was able to run at a stable 50-60fps, whilst *X-Plane 12*, struggled to maintain anything over 30, and was not stable. Participants noticed the big difference in framerate and mentioned that, whilst *X-Plane 12* had much more detailed and realistic scenery (being a near-exact one-to-one recreation of London), due to the low and unstable framerate, added lag between their inputs and the screen, causing pilot-induced-oscillation, leading to overcorrecting whilst flying.

X-Plane 12 runs quite well on a single screen, even on 4K with the latest hardware available you will get stable framerates. However, and this is a problem that has been present on previous *X-Plane* versions, *X-Plane* does not scale well when adding multiple screens. The graphics card we had available was more than capable of running *X-Plane 12* on three screens, but the way the *X-Plane* engine is modelled, it does not take advantage of multiple cores on the CPU. So, instead of allocating each screen to a different core on the CPU, so they can be rendered simultaneously, only one core does all the work, and it has to render each screen, one after the other, essentially cutting down the framerate in three, since this setup has three screens. In order to achieve better performance in flight simulators with multiple screens, two options are available: either upgrading to the latest and most powerful CPUs and GPUs available, or waiting for Laminar Research, the developer of *X-Plane*, to optimize the simulator for multiple screens.

Another issue with *X-Plane 12* that many users are currently reporting, is the anti-aliasing. It appears that the anti-aliasing is not properly implemented, and some bugs are causing the image to look quite blurry and pixelated, making it quite hard to distinguish details in the scenery. One potential solution is to upgrade the projectors to ones with a resolution higher than 1080p, which may help to alleviate the problem. However, this solution would require significant investment and may not be feasible for all users. Thus, for the time being, waiting for Laminar Research to address these issues may be the best course of action.

In the meantime, it is not possible to make a fair comparison between *FlightGear* and *X-Plane 12* due to the stark frame rate differences. Therefore, other visual systems, such as *Microsoft Flight Simulator 2020*, will be implemented as a future study.

Acknowledgements

This research was funded by the UK Research and Innovation under the Powerplant Integration of Novel Engine Systems (PINES) project (Rolls-Royce). The research is also co-financed by the European Union through the European Social Fund (grant POWR.03.02.00-00-I029). The authors of this paper are extremely grateful to Peter Beecroft (Rolls-Royce) for his invaluable assistance and feedback. The authors are also thankful to Dr Kacper Grzedzinski (Centre for Autonomous and Cyberphysical Systems) for his collaboration in the Future System Simulator. The authors would like to extend their sincere thanks to all participants including Donald Putnam, Jamini Hirani, Wayne Precious, and Neil Scarborough for performing flight tests. They all greatly assisted the research, although they may not agree with all of the interpretations/conclusions of this paper.

References

- [1] "Flight simulation video game," Wikipedia, 03 March 2023. [Online]. Available: https://en.wikipedia.org/wiki/Flight_simulation_video_game. [Accessed 03 May 2023].
- [2] Rolls-Royce, "Welcome to the future," [Online]. Available: <https://www.rolls-royce.com/media/our-stories/discover/2020/intelligentengine-flight-simulator.aspx>. [Accessed 18 September 2022].
- [3] D. Allerton, Principles of Flight Simulation, 1st Edition ed., John Wiley & Sons, 2009.
- [4] D. J. Allerton, "The Impact of Flight Simulation in Aerospace," *The Aeronautical Journal*, vol. 114, no. 1162, pp. 747-756, 2010.
- [5] L. E. Ross, P. W. Slotten and L. M. Yeazel, "Pilot's Evaluation of the Usefulness of Full Mission IFR Simulator Flights for General Aviation Pilot Training," *Journal of Aviation/Aerospace Education & Research*, vol. 1, no. 2, 1990.
- [6] R. Pausch, T. Crea and M. Conway, "A Literature Survey for Virtual Environments: Military Flight Simulator Visual Systems and Simulator Sickness," *Presence: Teleoperators and Virtual Environments*, vol. 1, no. 3, pp. 344-363, 1992.
- [7] D. Murr, C. Olson, M. Basler and E. Korpela, "Flight Gear," 1997.

- [8] Xbox Game Studios, Dovetail Games, Aces Game Studio, "Microsoft Flight Simulator X," Microsoft Corporation, Xbox Game Studios, Dovetail Games, 2006.
- [9] Lockheed Martin, "Prepar3D," Lockheed Martin, 2010.
- [10] Eagle Dynamics, "Digital Combat Simulator World," The Fight Collection, 2008.
- [11] Asobo Studio, "Microsoft Flight Simulator," Xbox Game Studios, 2020.
- [12] dSpace, "SCALEXIO," dSpace, [Online]. Available: https://www.dspace.com/en/ltd/home/products/hw/simulator_hardware/scalexio.cfm. [Accessed 19 September 2022].

Appendix

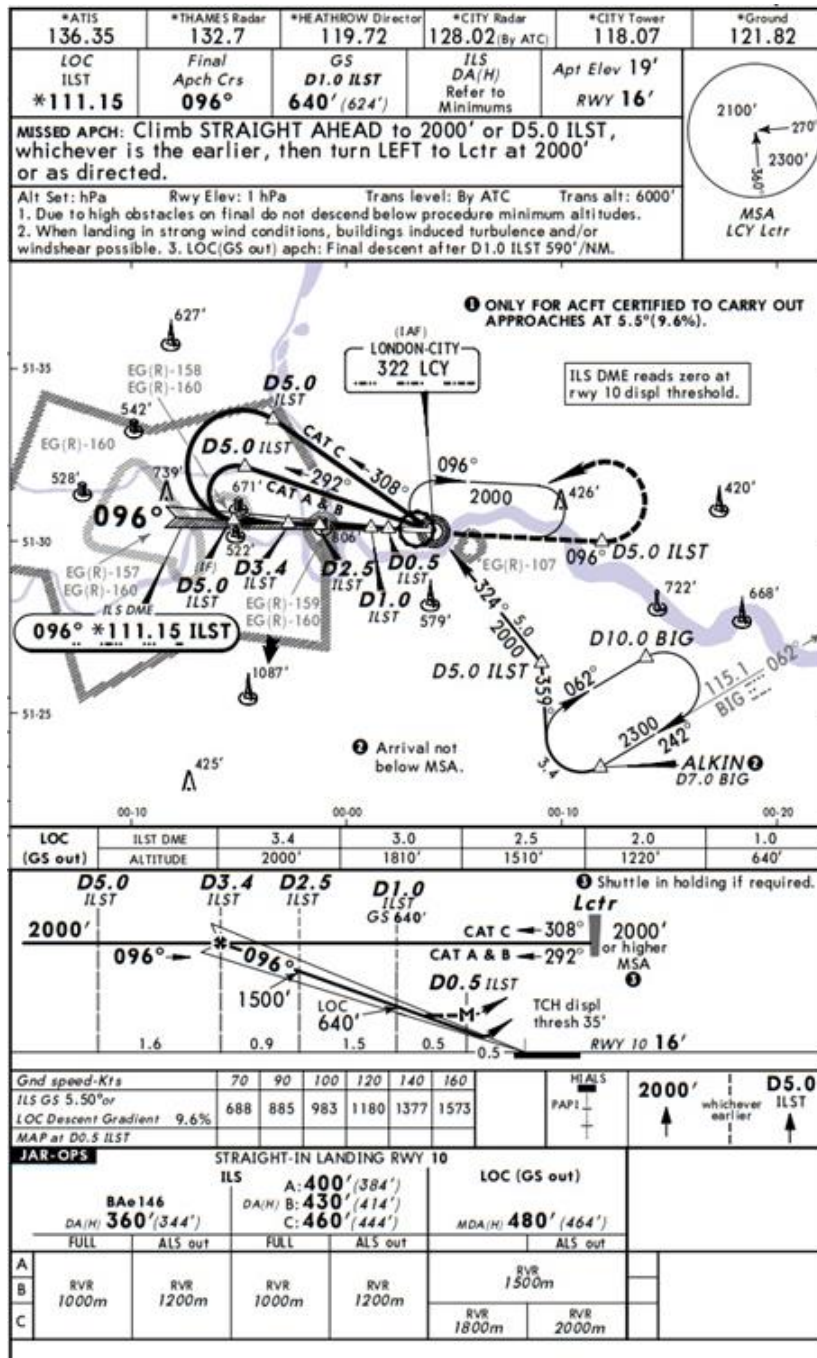


Figure A.1 Landing Flight Path Chart at London City Airport

Analysis of visualization systems in flight simulators

Barrio, Luis D.

2023-06-08

Attribution 4.0 International

Del Barrio L, Korek W, Millidere M, et al., (2023) Analysis of visualization systems in flight simulators. In: 2023 AIAA Aviation and Aeronautics Forum and Exposition (AIAA AVIATION Forum), 12-16 June 2023, San Diego, USA. Paper number AIAA 2023-3476

<https://doi.org/10.2514/6.2023-3476>

Downloaded from CERES Research Repository, Cranfield University