Modeling Pilot Flight Performance on Pre-flight and Take-off Tasks with A Cognitive Architecture

by

Rongbing Xu

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2022

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

Rongbing Xu was the sole author for Chapters 1, 2 and 5 which were written under the supervision of Dr. Shi Cao and were not written for publication.

This thesis consists in part of one manuscript written for publication. Exceptions to sole authorship of material are as follows:

Research presented in Chapters 3 and 4:

The following research conducted by Rongbing Xu under the supervision of Dr. Shi Cao is included in Chapters 3 and 4 of this thesis.

Rongbing Xu and Shi Cao. Modeling pilot flight performance in a cognitive architecture: Model demonstration. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 65, pages 1254–1258. SAGE Publications Sage CA: Los Angeles, CA, 2021.

Contributor	Contribution
Xu, R. (Self)	Conceptual design (80%) Writing and editing (80%)
Cao, S. (Supervisor)	Conceptual design (20%) Writing and editing (20%)

Abstract

Models of cognitive architecture can be used to simulate and forecast human performance in complicated human-machine systems. The current work demonstrates a pilot model capable of performing and simulating pre-flight preparation and take-off duties. The model was developed using the Queueing Network-Adaptive Control of Thought-Rational (QN-ACTR) cognitive architecture and can be connected to flight simulators like X-Plane to create various data types such as performance and mental workload. Declarative knowledge chunks, production rules, and a collection of parameters all contribute to the model's output. A human experiment involving pre-flight and take-off tasks was conducted to acquire the data required for the model's development. At the moment, the model can generate flight operation behaviors that are comparable to that of human pilots. By comparing model data to human data, it was demonstrated that QN-ACTR may be utilized to develop a multi-task model that accurately simulates pilot behavior. With further refinement, including support for situational awareness simulations, such models may help to evaluate interfaces and competency-based pilot training, complementing human-in-the-loop (HITL) experiments in aviation research and development.

Acknowledgements

I'd want to begin by expressing my heartfelt gratitude to Dr. Shi Cao. COVID-19 had a significant impact on my research. Throughout this time period, Dr. Cao gave invaluable assistance and guided me through a particularly trying moment. Without his supervision and assistance during the last two years, I would not have arrived here.

At the same time, I'd like to express my gratitude to my roommates and friends. They assisted me in a variety of ways at this time.

Finally, I'd like to express my gratitude to my entire family, particularly my parents and grandparents, for serving as a solid backup.

This work was supported by an NSERC Discovery Grant (RGPIN-2015-04134) to S.C.

Dedication

This thesis is dedicated to my wonderful parents, Dr. Dongyan Xu and Dr. Ping Dai, who not only raised me and taught me how to be a human being, but also made unwavering contributions to the education I got. I'm hoping to deliver this thesis to you as a special gift to demonstrate my indescribable thanks.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

IMPRINT Improved Performance Research Integration Tool 2, 4, 6, 7, 11, 12

LC Learned Carelessness 11

MAPE Mean Absolute Percentage Error 44, 46, 50

MDV Motion Detection Vision System 14

MIDAS Man-machine Integrated Design and Analysis System 4, 5, 7, 8, 10, 15

NASA National Aeronautics and Space Administration 5, 7, 11, 12, 31

NASA-TLX NASA Task Load Index x, 21, 27, 46

NextGen Next Generation Air Transportation System 12

QN-ACTR Queueing Network-Adaptive Control of Thought-Rational iv, x, 3, 28–31, 35, 36, 44, 46, 50–52

QN-ACTR-XP Queueing Network-Adaptive Control of Thought-Rational-X-Plane x, xi, 28, 32, 33, 36

RMSE Root Mean Squared Error 44, 46, 50

SA Selective Attention 11

SD Standard Deviation 21, 23, 46

SGOMS Sociotechnical GOMS 2, 5, 7, 8, 10, 14, 15

SOS Simple Operating System 14

SVS Synthetic Vision System 9, 12

T-NASA Taxiway Navigation And Situation Awareness 9, 10, 12

TOPAZ Traffic Organization and Perturbation Analyzer 2, 4, 8, 10

UDP User Datagram Protocol 29, 31, 52

XPC X-Plane Connect 31, 32

Chapter 1

Introduction

1.1 Background and Motivation

With the rapid growth of the economy and rising living standards, civil aviation passenger planes play an increasingly essential role in our everyday lives as a modern mode of transportation. Simultaneously, civil aviation safety has taken precedence. Even though passenger aircraft service is the safest kind of public transportation, it has the highest average single accident mortality of any mode of public transportation [34].

Weather, human error, Air Traffic Control (ATC), and mechanical or electrical system failures are the most common causes of civil aviation passenger aircraft accidents [51].The majority of aviation accidents, particularly fatal ones, are caused by human causes. According to the data, human causes were responsible for 80 percent of all aviation accidents in 2012. This figure increased to 83 percent in 2013 and 94 percent in 2016 [30]. Numerous studies demonstrate that, as the aircraft operator, the pilot's demeanor and professionalism have a significant impact on the plane's flight safety.

With a growing awareness of the impact of human factors on flight safety, an increasing number of scientists and psychologists are analyzing pilots' cognitive processing and behavioral operations and developing theoretical and practical cognitive and behavioral models. Simultaneously, an increasing number of national governments and organizations recognize the critical nature of aviation safety, and aircraft manufacturers are constantly enhancing existing control and warning systems and incorporating new technologies. The Federal Aviation Administration (FAA) and the European Aviation Safety Agency (EASA) have begun developing a variety of new civil aviation management systems aimed at adapting to the rapidly growing number of aircraft and passengers in order to improve aviation safety, including NextGen [22] and Air Traffic Management (ATM) Master systems [20]. As a result, validating and ensuring the safety of these technologies has become critical.

Pilot training is another critical human factor in aviation. Pilot training, as it is now practiced, is highly reliant on human instructors and the usage of real aircraft. Human evaluators assess pilot skill and training progress, which is quantified in terms of flight time [38]. A dearth of computational models capable of simulating pilot performance and skill development exists. These models can aid in the construction of intelligent tutors for pilot training by providing skill tracing and scheduling optimization.

Additionally, if validated, simulation models can be used to aid in aviation research and interface design evaluation by generating simulated pilot performance data without HITL trials. Numerous what-if scenarios and potentially dangerous circumstances can be analyzed and anticipated using model simulation [46].

In summary, by simulating and analyzing pilot cognition and behavior, these models contribute significantly to the prediction and reduction of flight accidents, the enhancement of flight safety, the development of autopilot systems, the assistance of pilot training, and the post-analysis of flight accidents.

1.2 Research Contribution

Over the last few decades, some research has been undertaken on pilot behavior and performance. This includes the Human Performance Model (HPM) developed on the basis of Air Man-Machine Integrated Design and Analysis System (Air-MIDAS) [49], which is used to analyze and predict human pilot error; the cognitive model developed on the basis of the Adaptive Character of Thought-Rational (ACT-R) structure [12], which is used to analyze possible pilot error when controlling an aircraft during taxiing; a model developed on the basis of Traffic Organization and Perturbation Analyzer (TOPAZ) and Air-MIDAS to investigate the relationship between pilot behavior and aircraft accidents [15]; an Sociotechnical GOMS (SGOMS) model for analyzing pilot behavior [19]; and a Cognitive Architecture for Safety Critical Task Simulation (CASCaS) model for analyzing the interaction between pilots and the cockpit and how that interaction can effect pilot performance [29]. The relevance of these models is primarily to reduce pilots' human error through behavior analysis, hence increasing flying safety. Additionally, similar models such as the Improved Performance Research Integration Tool (IMPRINT) model [26], the Airport Surface Detection Equipment, Model X (ASDE-X) model [53] and the SimPilot model are used to evaluate pilot behavior [43]. These models create a variety of data kinds by simulating pilot behavior and comparing it to experimental data collected from human pilots in order to determine their performance.

Although several prior research reported on the construction of pilot behavior simulation models, the majority of these studies were funded by the military, and the models are not publicly available. There are insufficient recent data and reports to assess the present state of earlier models, and no open sources of computer simulation models exist to facilitate public research and analysis in the field of civil aviation. According to the information that can currently be accessed, the majority of models are incapable of being compared to and analyzed against human experimental data. They do not perform in-depth study on situation awareness and cognitive structure, such as the links between visual attention, memory, and motion in pilots. All models fall short of proposing a demonstrable mechanism of production. Additionally, only a few can be connected to a flight simulator. As a result, I intend to address this research gap by designing a model to aid in public civil aviation research, with a particular emphasis on the pilot's cognitive process and situation awareness.

To summarize, I intend to employ QN-ACTR to develop a pilot behavior model that will fill a vacuum in this field of research. QN-ACTR has been shown in earlier studies to construct human-like models and replicate situational awareness as a mature cognitive architecture. Simultaneously, the model will be enhanced to handle a variety of missions and aircraft types.

1.3 Outline

Chapter 2 analyzes and summarizes many existing pilot behavior models, which serve as a guide for our research.

Chapter 3 introduces a human experiment for pre-flight and take-off operations. Multiple human data sets are acquired in order to create the model. Additionally, situational awareness and mental workload are measured briefly.

Chapter 4 discusses the model's details for the pre-flight and take-off procedures. It describes the cognitive structure and production principles upon which the model is based, as well as its performance in comparison to acquired human data.

Chapter 5 discusses the model's performance and limitations, as well as future work that could be done to improve it.

Chapter 2

Previous Human Performance Models in Aviation

This chapter covers and reviews recent research on pilot behavior and human performance models that are aimed at enhancing flight safety and minimizing human error. More precisely, this review introduces several models of pilot behavior and performance. It presents their findings regarding the prediction and prevention of human errors at various stages of flight using a variety of model architectures and frameworks, including ACT-R, Man-machine Integrated Design and Analysis System (MIDAS), TOPAZ, and IMPRINT. The majority of these systems are aimed at simulating all human tasks and have been utilized to imitate pilot performance in aviation. Additionally, some models are connected to a flight simulator and are used to operate the aircraft by modeling the pilot's actions, thereby providing theories and methods for minimizing human error. To begin, several well-known and widely-used model architecture will be discussed.

2.1 The Nature of Human Performance Model

The HPM is a technique for assessing human behavioral and cognitive characteristics. It is typically used to replicate human behavior in a specific environment in order to conduct human factors research [45]. Today, HPM is widely employed in a variety of disciplines due to its effectiveness when sufficient knowledge and data are available [18]. HPM has played a critical role in civil aviation, particularly in the research of the impact of human factors on flight safety. Due to the exceedingly low likelihood of flying accidents, researchers often struggle to collect significant data when analyzing pilot errors in the field and laboratory. Nonetheless, with a sufficiently accurate HPM simulation, findings and data can be gathered that are congruent with reality. Additionally, while HITL simulations can provide more realistic data, they can be costly and time consuming, and they cannot test large-scale behaviors [53]. However, HPM has a number of important drawbacks. For instance, airplane operation is quite complicated, since numerous elements must be addressed, which is not easy to perform. Thus, the majority of HPMs can model only a portion of the flight process, but not all. National Aeronautics and Space Administration (NASA) selected five modeling teams in 2005 to build human performance models for taxi operations and runway instrument approaches with and without enhanced displays [18]. Each team made use of a unique modeling architecture. NASA expects that these models will be able to forecast and prevent human error, hence increasing flight safety. The majority of models are built on top of cognitive architectures such as ACT-R and CASCaS. However, models are also constructed using predefined predictive methods that incorporate perceptual and cognitive features, such as Human Information Processor Model with Goals, Operators, Methods, and Selection Rules (GOMS), SGOMS, and MIDAS.

2.2 Previously used cognitive architectures and frameworks

2.2.1 ACT-R

J. R. Anderson pioneered the ACT-R concept in 1993 [2]. It is currently widely used as a high-level cognition model in a variety of modeling fields, including memory and attention control, cognitive neuroscience, and other difficult activities [5]. It has since established itself as one of the most well-known cognitive architectures. As a computational cognitive architecture, it has been demonstrated that it is capable of modeling human cognition through the interaction of lower-level psychological processes when applied to pilot and driver behavior models [10], [40]. ACT-R excels at interacting with the external world in real time to evaluate performance models, such as those generated by other HITL simulations or field testing. ACT-R is often composed of declarative knowledge and particular procedural knowledge acquired through external input [5]. For pilot performance models, these data are typically provided by pilots, aviation specialists, aircraft manufacturers, or airline firms. Chunks are utilized in the ACT-R paradigm to store declarative knowledge and objects in the environment. Procedural memory, on the other hand, is expressed through production rules. Changes in mental state, visual movement, and motor movement are all examples of common production laws.

2.2.2 CASCaS

The CASCaS, developed as part of the Model-Based Analysis of Human Errors During Aircraft Cockpit System Design Project (HUMAN Project), is a novel cognitive structure for simulating human (or, more precisely, pilot) cognitive processes [31]. As with prior cognitive architectures such as ACT-R, this design is built of two components: a general cognitive architecture unrelated to the current activity and a formal architecture connected to the current activity (such as flight procedures and traffic laws). I will not reiterate the model's other components, which are comparable to those of other widely used cognitive models. This slightly modified version of the ACT-R model is capable of predicting probable errors caused by routine learning (Learned Carelessness) and attention allocation (Selective Attention) [32]. I shall return to these two sections in greater detail later. CASCaS classifies human behavior into three categories: skill-based, rule-based, and knowledge-based.

Similarly, these three behavioral categories are induced by three distinct levels of cognitive processing: autonomous, associative, and cognitive [32]. Independent level processes, on the other hand, are often thought to create the most particular behavior, but any complicated behavior or novel task requires cognitive level processes. According to this hypothesis, CASCaS first evaluates the job objective and determines the processing level based on parameters such as knowledge memory and perception of the external environment, and then processes and triggers various motors based on different rules.

2.2.3 IMPRINT

The US Army created the IMPRINT for modeling human performance. IMPRINT is typically used to model the cognitive effort of persons doing a complex task or event and to assess their performance [39]. An IMPRINT model is constructed from a modest amount of data that can be accessed publicly on the US Army's official website. It incorporates tasks, nodes, entities, control flow, and other variables. Although the vast majority of published research projects utilizing the IMPRINT paradigm are applied to military applications, there are a few notable exceptions, such as civil aviation. In the civil aviation industry, an IMPRINT model is frequently coupled with an ACT-R model. The IMPRINT model can be used to represent the external environment, the state of the aircraft, and other aircraft components. The ACT-R model is used to replicate a pilot's cognitive state [18]. Due to the fact that IMPRINT is built of various modules and parameters, the state of the aircraft can be easily altered by simply adding and deleting modules and parameters in order to determine the effect of these variables on human cognitive state and performance [18].

2.2.4 MIDAS

Since 1986, NASA Ames Research Center has used the MIDAS to forecast human-system performance in a variety of research disciplines [17]. MIDAS is a dynamic, integrated modeling and simulation environment for human performance that enables the design, visualization, and computational evaluation of complex human-machine system concepts in simulated operating contexts [21]. MIDAS simulates human behavior by receiving and comprehending data from the external environment, storing it in memory, matching it to a known procedural representation of the relationship between human behavior and external data, and finally executing it [22]. A MIDAS is typically comprised of three components: input, output, and processing. The external environment, tasks, and behavior models comprise the input portion. The processing section consists of a task management model and a procedural model of basic human behavior. The output section contains information about the model's current state of operation and quantifiable performance metrics. By modeling the operator, operating system, and external environment, the MIDAS model can perform a quantitative and visual analysis of a series of complicated interaction events. MIDAS has been demonstrated in the aviation industry to be capable of designing systems, simulating hypothetical occurrences, evaluating human performance, forecasting pilot behavior, simulating pilot cognitive processes, and conducting other sophisticated experiments [45].

2.2.5 SGOMS

SGOMS, presented by West and Roberts in 2011, is a methodology for establishing a hierarchical control structure within ACT-R to address agent performance in macro-cognitive tasks [50]. As an expansion of GOMS, SGOMS employs a hierarchical control framework, which includes modules for production, motor, visual, and auditory processing. Control is implemented hierarchically via planning unit, unit task, and operator buffers [50]. Operators represent low-level cognitive processes and motor commands, and motor operators can be triggered when the output of the motor module matches the output of the operator buffer [50]. On the other hand, the unit task exerts the most influence over the scheduling of sub-methods and sub-operators; a unit task can be a collection of actions that are either not interruptible or are intended to be interrupted [50]. A planning unit is a collection of multiple-unit tasks with their own dedicated buffer, which will be executed in the sequence determined by the buffer [50]. Due to the properties of SGOMS, it is easier to construct an expert or macro threaded cognition model based on SGOMS compared to the threaded cognition model introduced by Salvucci and Taatgen [50], [42].

2.2.6 TOPAZ

The TOPAZ is a stochastic analysis framework for assessing the safety of ATM. It supports stochastic models with discrete and continuous variables that change in real time [8]. This model can be used to describe the effect of variables that may cause deviations or uncertainty on the overall system's performance and to forecast potential mishaps. A TOPAZ model may include several critical agents, such as the pilot's perception, flight status, and current mission; the relationship between different agents, such as the effect on other agents when one agent's status changes; and the mode and other characteristics of agents, such as the status and duration of a task. By running the model with the Monte Carlo approach and providing a large number of random agent-agent interactions, the chance of accidents can be determined under various situations based on the result [8]. Due to this property, the TOPAZ model is capable of simulating highly rare events. TOPAZ is primarily used to determine the likelihood of an aircraft colliding in specific conditions (such as runway incursions). Numerous investigations have established that some variables have a significant role in collision accidents when a TOPAZ model is used.

2.3 Models focused on reducing human error

The majority of currently funded research focuses on the impact of human factors on the aircraft taxiing and approach/landing phases. Verma et al. proposed an HPM based on Air-MIDAS in 2003 that emphasized human error analysis and prediction [49]. Due to the model's foundation in the Air-MIDAS, it is capable of assessing pilot performance in terms of human memory, attention, and conduct. However, because the original MIDAS model is incapable of analyzing the current state of affairs in real time, the authors made adjustments. Simply put, they assign a value to each task and prioritize them depending on the sum of the values assigned to current tasks. Assume that the resources necessary to complete the current jobs are unavailable, or that the overall number of tasks exceeds the operator's processing capacity, the work will be postponed or suspended in this scenario. Each time unit has a predefined amount of executable tasks (usually one second). The authors utilized this model in the experiment to simulate two planes attempting to take off and invading the runway. They used GOMS to dissect the captain's and co-pilot's behavior during the two planes' takeoffs and landings. Simultaneously, the pilot's present visibility and any variables affecting his or her eyesight are established. When neither aircraft detects the other, it is assumed that an error has occurred. Experiments indicate that in the majority of circumstances, the taxiing aircraft can detect an airplane taking off or responding to ATC directions in a timely manner. Only 10 percent of the time does the aircraft abort take-off due to reaching take-off speed in other conditions [49]. Due to a dearth of HITL simulations, the model's accuracy relative to the fundamental human reaction is unclear, but the author remains optimistic [49].

Bryne et al. published a computational model of a closed-loop, pilot-display-plane system in 2004 to assess the influence of a Synthetic Vision System (SVS) on approach and landing stages in a commercial aircraft, which had a significant impact on a number of later studies [12]. The experiment as a whole was separated into two sections. To begin, they instructed the pilots to perform a flight simulator simulation of the plane's landing. They recorded their eye movements both with and without the SVS system to determine their attention distribution during the landing procedure. Then, in the second half, they replaced the human pilot with the created ACT-R model and repeated the experiment on the flight simulator. The experimental results indicate that when the SVS is turned on and off, the pilot's attention distribution changes dramatically in both the HITL simulation and the ACT-R model simulation experiment. Regardless of whether the pilot's attention is focused on the SVS or on the world outside the window, professional pilots' performance does not vary significantly [49]. In other words, the SVS system has no impact on the pilot's performance during the landing procedure or on the aircraft's ability to land safely.

Brune and Kirlik developed another closed-loop pilot model in 2005. It is a pilot-aircraft-visual-scene-taxi-way system that is focused on various sources of taxi error and implemented using ACT-R [11]. Once the pilot-related knowledge is imported into the ACT-R model, the model can make real-time decisions depending on the external environment and operate the simulated aircraft in the flight simulator under the Taxiway Navigation And Situation Awareness (T-NASA) scenario. This experiment follows the same procedure as the SVS experiment stated previously and is similarly separated into two parts. To begin, each of the 18 crews was evaluated in this scenario through three distinct routes, totaling 54 trails. At Chicago O'Hare Airport, pilots conducted an approach and landing procedure before taxiing to the gate. The model then replicated the process under the identical scenario. Due to ACT-R's lack of multitasking capabilities, this model emphasizes decision making, with control performance being a secondary objective [11]. In this situation, twelve decision mistakes were identified. The research discovered that the choice error made by pilots in the T-NASA scenario while taxiing after landing is primarily due to a complicated external environment, including as weather, timing, and ATC communication [11].

Corker et al. (2006) proposed a method for evaluating the probability of aircraft accidents in 2006 by merging the Air-MIDAS and TOPAZ models [15]. As I previously stated, the MIDAS model is an HPM that, by modeling human behavior, can quantify and visually assess a complicated interaction event. In comparison, the TOPAZ model can mimic a difficult interaction event by randomly modifying the event variables, and the resulting simulation results can be utilized to determine their influence. By combining these two models, the impact of the pilot's individual operations on the aviation accident may be evaluated in greater detail, in this case, the aircraft collision caused by the runway invasion. The experiment detailed in this article used two planes on two parallel runways, one attempting to take off and the other attempting to taxi. Due to the properties of Air-MIDAS and TOPAZ, they were utilized in the experiment to replicate pilot behavior and cognitive processes, as well as the effect of numerous external environment variables on the runway incursion accident. By recording and analyzing whether the pilot can abort take-off or stop taxiing in time, as well as the corresponding reaction time, the results demonstrate that the TOPAZ model based on Air-MIDAS can reduce collision risk by twofold compared to the original TOPAZ model, demonstrating that coupling these two models is effective and feasible [15]. Additionally, because the presence of Air-MIDAS lessens the likelihood of collision and Air-MIDAS more precisely simulates the pilot's conduct and ATC orders, this experimental result can be used to highlight the crew's and ATC's impact on the crash from the side [15].

Frische et al. introduced a novel methodology in 2009 for detecting possible pilot errors during flight [19]. By comparing the pilot's actual performance to the projected performance, this model can evaluate if the pilot's operation can be regarded a mistake. To analyze the expected performance, the current task is divided into many subtasks depending on time and hierarchical linkages. This model is structurally similar to the GOMS model (refer to SGOMS section). To determine whether the pilot accurately detects his or her own errors, a considerable amount of human pilot data must be provided. Based on real-time human data and synchronously simulated flight state, an error type checking unit will determine whether the pilot's actions were missed or delayed. If this activity fits specific criteria, such as being required to complete the current task, it may be considered an error. However, some low-level faults are impossible to identify correctly, and it is insufficient to cover all parts of standard flight operations [19].

In the same year, Lüdtke et al. developed a novel cognitive model for reducing human error in safety-critical assistance systems, which was ultimately dubbed CASCaS [29]. The authors used CASCaS to develop a pilot performance model in order to assess its suitability for modeling pilot cognitive processes. I will only discuss two relatively new notions in this section, which I mentioned previously: Learned Carelessness (LC) and Selective Attention (SA) [29]. During the flight, the pilot may purposefully disregard certain acts on the checklist to boost his self-esteem, such as neglecting hitting a button at a specified time, despite the fact that it should be encouraged. This type of surgery is referred to as LC. While many LCs may not pose a threat to aviation safety, they should nevertheless be deemed mistakes. The notion of SA was established to avoid the occurrence of LC. The authors previously discovered that some visual stimuli (such as light flashing) can impair operators' cognitive functioning. They discovered that certain types of light effects can influence the pilot's cognitive process and lower the amount of LCs [29]. One such effect is the flashing pattern of the autopilot switch when it is disabled throughout the flight. I feel that both the LC and SA approaches will be critical in future research on pilot performance and cognitive models.

Later, Lüdtke et al. created an HPM that focuses on simulating the interaction between the flight crew and the cockpit interface and evaluating the new system's effect on pilot performance [28]. NASA's paper demonstrates that models based on ACT-R, IMPRINT, Air-MIDAS, and other structures can be used to forecast and evaluate certain pilot performance characteristics. They are, nevertheless, insufficient to cover all facets. The scientists hoped that by enhancing the present CASCaS model, they could more accurately and widely forecast human errors caused by LC and SA. Two distinct scenarios are developed; each scenario contains a variety of difficult flight duties, such as cruise, approach, and final approach. The flying time for each scenario is roughly 30 minutes. The researchers ran hundreds of simulations and collected three types of data from the model: gaze behavior, such as gaze distribution and duration; temporal behavior, such as task execution time and reaction time; and human error analysis, such as detection of omission and commission errors [28]. Similar to most other models, this model cannot be confirmed to imitate human behavior and cognitive patterns effectively due to the lack of human data. The authors thoroughly assessed the model's performance using the recorded data and concluded that additional HITL simulations should be done in the future to verify the model.

2.4 Models focused on simulating or evaluating human performance

In this section, I will discuss some models that are used to forecast pilot behavior and simulate human pilot performance.

As another critical component of the NASA HPM effort, Lebiere et al. proposed another HPM technique based on the ACT-R and IMPRINT models in 2003 [26]. The ACT-R model is primarily responsible for replicating pilot cognition and behavior during the aircraft's landing phase. The IMPRINT model mimics the aircraft's status throughout the landing phase, as well as the external environment, which includes weather, ATC instructions, and airport layout. To make this model as universal as feasible, the authors modularized it extensively, allowing it to be used to various stages of flight [26]. The authors utilized this model to recreate the complete landing procedure of a Boeing 757 airplane with or without the SVS system. They evaluated the model's validity using five parameters: procedural (rule execution) speed, visual (eye movement) speed, aural (hearing and speaking) speed, manual (hand movement) speed, and activation noise (decision-making stochasticity) [26]. Due to a lack of pilot data and individual variances in available pilot data, the experimental outcome can only demonstrate that this HPM can do tasks in the same manner as a human pilot. Nonetheless, it is unable to establish that this HPM is equivalent to human pilots. Lebiere et al. conducted more large-scale testing on this model in a subsequent paper released in 2005 [27]. They asked numerous pilots to engage in the HITL simulation to practice some approach and landing procedures in the T-NASA scenario, which was discussed before in this review, and used the HPM to mimic the same state of affairs. They discovered that: the HPM simulation results are consistent with the HITL simulation in all scenarios; even though the HPM simulation did not complete (or did so with a delay) some necessary actions during certain stages of the approach and landing (such as descending), HPM still completes all actions in the majority of scenes; and there is a significant difference between the eye movement data from the HPM simulation and the HITL simulation [7]. I concur with the author's conclusion that an adequate model may be used to mimic a pilot flying an aircraft and to forecast the pilot's behavior and cognition in a particular environment, as well as the aircraft's flight condition.

Zemla et al. (2011) created an ACT-R model of commercial airplane taxiing in 2011 [53]. This model was originally created to evaluate ASDE-X, a novel integrated positioning technology for the Next Generation Air Transportation System (NextGen) developed by the FAA by simulating pilots commanding aircraft in a flight simulator. The purpose of this model is to demonstrate how NextGen can assist enhance the efficiency of ground transportation management while also reducing congestion and airport ground safety [1]. However, following the trial, the author argued that ATC commands may be delivered as digital data in the future, reducing the pilot's reliance on working memory and making the taxiing process safer. The experimental results demonstrate that this ACT-R model can be employed in place of humans in typical HITL simulations and provide same or similar experimental outcomes [53].

A highly simulation-based model is required to forecast the performance of pilots taxiing on the ground from the gate to the runway under a variety of conditions and external settings [44]. Schoelles and Gray (2011) described a model based on ACT-R 6.0 called SimPilot [44]. This model is intended to predict and avoid runway incursions, a common cognitive and situational awareness-based event. This model, which incorporates ACT-R and multi-threaded cognition, is connected to a flight simulator and replicates multi-task processing while taxiing on the ground, such as interacting with ATC, managing the aircraft's rudder and throttle, and detecting the external surroundings. Following that, the author discovered that, due to the complexity of the flight simulator, the model was unable to establish a reliable link with it, rendering it impossible to completely imitate the behavior and cognition of a human pilot [43].

Klaproth et al. presented an ACT-R model in 2019 to determine if a pilot followed the proper actions when confronted with aircraft alerts or alarms [25]. Pilots are not always able to respond to aircraft alerts due to the complexity of aircraft control systems. Even said, pilots will occasionally purposefully disregard notifications they deem unimportant. As a result, it is vital to examine whether pilots responded appropriately to signals and to enhance response effectiveness. In the experiment, the model was connected to the pilot's Electroencephalogram (EEG) and a flight simulator. It determines whether an alert exists by evaluating the pilot's EEG signals, then retrieves aircraft status data from the flight simulator (such as airspeed and altitude), and finally replicates a human pilot's response to the alert. This trial had a total of 21 senior pilots. Each participant was exposed to an 18-minute scene comprised of nine to fourteen incidents [25]. Each occurrence is preceded by an alert or an ATC directive. All human pilots and the cognitive model have their reactions recorded, including their reaction time and behavior. The authors assessed the pilot's and model's response actions by comparing them to ATC communications and standard checklists. Experimental results indicate that the cognitive aid model responds more accurately to ATC directives and alerts than human pilots do [25]. Given that how well alerts and ATC instructions are implemented has a significant impact on flying performance and safety, I feel that this research has a profound impact on aviation safety.

2.5 Other related models

This section examined numerous additional human performances and cognitive models that were not specifically related to flying. I believe that with modest tweaks, these models can be used to improve aircraft safety while also serving as a starting point for future study.

In 2004, Ball, Rodgers, and Gluck proposed this paradigm, which is centered on the development of a language-enabled intelligent agent [6]. On the ACT-R architecture, a language comprehension system is created, and a Cyc knowledge base offers basic language knowledge. Although this model is not specifically designed to improve aviation safety, I believe that by expanding the Cyc knowledge base and adding more flight-related terms, it can increase the efficiency of non-native English-speaking pilots communicating with ATC and reduce the possibility of miscommunication, thereby improving aviation safety.

When West and Roberts launched SGOMS in 2011, they combined an SGOMS model with a commercial airliner's landing technique [50]. Following the simulation, the author discovered that, despite the complexity of the landing technique, this model can accurately imitate the complete process of human pilots managing the plane. They anticipated that by incorporating additional pilot rules and decision-making processes, the model might be greatly enhanced.

Somers and West created a new ACT-R model of a single-engine propeller aircraft take-off technique in 2013 [47]. In comparison to some of the other models discussed in this article, this model utilizes a Simple Operating System (SOS) vision module and a Motion Detection Vision System (MDV) to imitate the pilot's vision system. SOS functions identically to ACT-R's declarative memory in that it places the object into the visual buffer in response to a vision request. Due to the simplicity of SOS, after connecting to the flight simulator, this model can easily observe the instruments and buttons in the cockpit [47]. Regarding the MDV system, it is based on research on retinal flow and intercepts the flight simulator's interface in real time using Python and OpenCV to recreate the pilot's observation of the aircraft's exterior environment [47]. I believe that combining these two technologies will significantly improve pilot behavior and human performance models' abilities to gather visual information in the future.

2.6 Discussion

After removing models that were not directly related to human error and aviation safety from the HPM I studied, I discovered that the majority of models did not properly assess their validity. The experimental results demonstrate simply that they are capable of simulating pilot behavior in specified conditions. There are, in my opinion, various viable explanations for this circumstance. To begin, numerous articles have been written to demonstrate that it is possible to model human behavior and utilize it to anticipate and analyze pilot performance. At the moment, there is little research aimed at validating or enhancing the model's precision. Second, because to a lack of human data, many models used to examine accidents caused by human error cannot be compared to the real world. For instance, an aviation collision mishap caused by runway incursion is extremely unlikely to occur. Obviously, it is impossible to conduct a large number of simulations in the field. While this is possible, it is also possible that the HITL simulation will be incomplete due to limited financial support and time. Additionally, several data points about pilot performance are difficult to quantify. Until now, there has been no widely accepted structural system capable of quantifying and evaluating pilot performance. Additionally, certain data on pilot performance are difficult to quantify. Until now, there has been no widely accepted structural system capable of quantifying and evaluating pilot performance. Human error is sometimes trivial but can be fatal in some circumstances; pilots whose native language is English may understand ATC instructions more quickly and accurately, but this may also be the case due to ATC's accent. Finally, while some models replicate the pilots' total performance and compare it to human data gathered by HITL simulation or other sources, they evaluated only a subset of the data, not all of the experimental data. I have cause to believe that the non-public data may not be adequate.

To mitigate the influence of the aforementioned issues on HPM, I expect that future related research will employ many model structures rather than simply one. For example, SGOMS and MIDAS can more precisely separate and replicate the flight mission and the pilot's motions; ACT-R can be used to represent the pilot's cognitive process, as it has been extensively validated in the field of cognitive models. In my research, I intend to model pilot behavior using a large quantity of experimental data in order to assure the model's accuracy and applicability, rather than simply remaining at the level of theoretical analysis. Simultaneously, I will post all data collected on pilot behavior and data generated by the model, something that has never been done before.

In general, due to the complexity of modern passenger aircraft structures and pilot cognitive models, it is difficult to completely imitate a human pilot operating a modern passenger aircraft using HPM. However, it is gratifying to know that by continuously improving existing models and conducting additional research in this field, it is entirely possible to develop a useful model for analyzing pilots' behavior and cognitive patterns and thereby increasing aviation safety by eliminating any potential human error.

Chapter 3

Human Experiment of Pre-flight and Take-off Procedures for Model Development

Due to a dearth of published research on pilot behavior analysis and a scarcity of relevant data for the model's construction, I devised an experiment to collect human data for critical model parameters. This includes, but is not limited to, the human pilot's response to various situations, the amount of time available for thought, observation, and reaction during various operations, and the status of the aircraft in the flight simulator at various times. The University of Waterloo's Office of Research Ethics approved this study.

The Tustin-McRuer model was introduced in this section of the investigation. This is the most generally used model of human pilot behavior, which Tustin proposed in the 1940s [48]. The model is now described more straightforwardly by the equation 3.1

$$F(s) = K \cdot \frac{(T_Z s + 1)}{(T_1 s + 1)(T_2 s + 1)} \cdot exp(-T_d s), \tag{3.1}$$

where K is the pilot's gain defined by the type of control, T_z represents the time constantly lead regarding the pilot's experience, T_1 means the time lag constant related to the pilot's knowledge and routines, T_2 denotes the time constant of the neuromuscular system, T_d is the pilot response delay caused by brain and muscle, and s means the Laplace operator [35]. It is modified from another well-known model, McRuer and Krendel's accuracy model based on the Crossover law [33]. Typically, T_d has a value from 0.1s to 0.5s, T_2 has a value of 0.1s, T_z has a value of 0.2s to 6.1s, and T_1 has a value of 0.1s to 3.7s [24].

This feature is beneficial for feedback on single-loop pilot-aircraft interaction in the majority of studies, as illustrated in Figure 3.1 [33], [9], [23]. It concisely explains the pilot's relationship with the aircraft. The aircraft's state will be displayed on the display in real time, alerting the pilot via visual cues. After the pilot collects the information, it passes it through the cognitive system's processing and then executes the operation provided to the system. The system output indicates the aircraft's state.

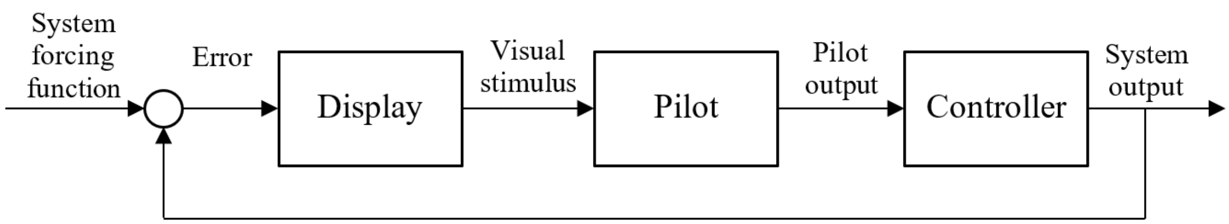

Figure 3.1: Single-loop pilot-Aircraft interaction.

3.1 Participants

We recruited 18 undergraduate students from the University of Waterloo between the ages of 18 and 24. They are currently enrolled students with extensive knowledge of the Region of Waterloo International Airport and the Cessna 172SP aircraft. Participants have an average of 68 hours of flight experience. Their physical differences were overlooked because their education and training backgrounds were comparable, and they all passed the civil aviation medical examination. They already have experience flying small propeller aircraft, such as the Cessna 172SP, as student pilots. Not only that, due to their inexperience, student pilots usually exercise extra caution and meticulously adhere to the checklist and mandatory procedures when flying. It can assist in ensuring the consistency of the outcomes I acquire. I offered $20 as an incentive for participation in this experiment, which will last between one and two hours and will consist of two tasks in several circumstances, as well as post-experiment questionnaires.

3.2 Hardware Devices and Software

The complete simulator system is composed of numerous displays and controllers (see Figure 3.2), including a Logitech yoke and throttle quadrant, a Logitech rudder paddle, a Logitech flight multi-panel, a Logitech radio panel, a Logitech flight switch panel, and six Logitech flight instrument panels. Participants replicate real-world aircraft control by interacting with the virtual cockpit's components. Three 24-inch monitors are positioned side by side to display the external surroundings. Volair Sim created the base, as well as the chair. I attempted to recreate the Cessna 172SP environment as closely as possible, but due to the compatibility of various devices, this is already the best presentation.

In terms of simulator software, I utilized X-Plane 11, as previously explained. X-Plane 11 works well with the equipment I use and requires less computer components (compared to other similar software). I used built-in X-Plane's Dataref function to keep track of the aircraft's status. Each 50 milliseconds, X-Plane uploads hundreds of pieces of information about the aircraft to the record file. I conducted this investigation using a Cessna 172SP. I utilized the replicated Cessna 172SP plugin produced by Airfoillabs in the flight simulator. According to our consultations with pilot trainees, this plug-in provides an extremely high level of simulation for the Cessna 172SP. It is capable of performing the majority of the functions and capabilities of this model in reality. Additionally, I included a checklist outlining pre-flight and take-off protocols (Appendix A). Each item is taken directly from the pilot's real-world checklist.

Apart from the simulator, I used ForeFlight, the most useful Electronic Flight Bag (EFB) application in the real world, which runs on the iPad, and distributed it to the participants. It includes Jeppesen's most recent charts and navigation data, a source of real-time flight data. Additionally, I will install the most recent navigation data offered by Navigraph, a source of flight simulation navigation data based on Jeppesen's data. Jeppesen and ForeFlight are both owned by Boeing.

3.3 Experiment Design and Procedures

The experiment is divided into two phases: pre-flight and take-off. These two distinct scenarios reflect the various stages of aircraft piloting. Before the trial began, participants were emailed an information letter, consent form, and a pre-experiment questionnaire containing demographic questions. Due to the COVID-19 problem, an additional COVID-19 information letter was provided, and all participants underwent another COVID-19 screening to ensure they fit the study's eligibility criteria.

Participants would be permitted to ask questions and the experiment would begin only after they signed the consent form properly. Following their agreement to begin the experiment, they were able to rehearse freely for ten minutes in a sample scenario that differed

Figure 3.2: Devices used in the human experiment.

from the actual scenario utilized in the experimental task. It assisted participants in becoming acquainted with the flight controllers and simulator. Following this preliminary practice, I gave participants some sample instructions to check they could follow them and do the task safely and accurately, just as they would if they were flying an actual aircraft. Following the practice session, participants will complete two tasks sequentially: pre-flight preparation and take-off procedures. Between those tasks, participants were given a 5-minute rest. However, the gap may be prolonged or canceled without consequence based on participant requirements.

All scenarios are based on a 9 a.m. arrival at Region of Region of Waterloo International Airport (CYKF). Two runways intersect approximately in the center of this airport. Runway 08/26 measures 7003 feet in length, whereas Runway 14/32 measures 4103 feet in length. Additionally, the airport features seven aprons. The weather is sunny and calm, with a temperature of 24°C. The runways are completely dry, and visibility is excellent. The pressure in the air is 29.92 inHg (QNE).

Task 1: Pre-flight preparation. The most crucial element of the trip is the pre-flight preparation. Pilots must evaluate meteorological and aeronautical information that may affect the flight, check the flight route, prepare the chart and flight plan, and ensure the aircraft is ready to taxi and take-off during pre-flight preparation. Participants would mostly adhere to checklists and take the needed activity. They will be responsible for operating buttons and switches. The aircraft was initially parked at CYKF's Apron 3. Participants were asked to prepare a Cessna 172SP aircraft using the provided pre-flight checklist (Appendix A), just as they would in real life. Participants were required to monitor the aircraft's state in real time and decide when to take the next action based on their experience.

Task 2: Take-off procedures. The second portion required participants to fly the aircraft to CYKF. Take-off is another critical component of the flight. Numerous flight disasters occurred during take-off. Instead of buttons and switches, participants should concentrate on managing the yoke, rudder, and throttle during the take-off operation. The aircraft was scheduled to depart from CYKF on runway 08, rise to 4000', and maintain a heading of 74°. To ensure that the experimental results are statistically significant, no random events or unknown variables such as aircraft mechanical or electronic malfunctions occurred during the experiment. This indicates that individuals were unaffected by irrelevant occurrences and provided more responses.

Throughout the test, I logged the flight simulator's different data and condition, including the aircraft's location, airspeed, flap rate, and Euler angles, etc. Simultaneously, the entire experiment is videotaped to document the start and end times, as well as the time interval between each action. Two cameras were used: one focused on the flight control panel to record the participant's hand movement, and the other on the overall panel and displays. Throughout the trial, both video and audio recordings were made to examine human performance and reaction time. Individually identifying characteristics (e.g., face, body shape, gender, etc.) will not be intentionally recorded, and no recordings will be disseminated or broadcast to the public.

Following each activity, I distributed two post-experiment questionnaires to the subjects. One is a post-experiment situational awareness questionnaire (Appendix B), which is used to assess participants' situational awareness during the experiments; the other is the NASA-TLX questionnaire (Appendix C), which is widely used to assess the level of mental workload experienced by humans. Following the questionnaire, I asked participants to provide any comments they had on the experiment, and they would receive a feedback letter after the trial.

3.4 Results

3.4.1 Pre-flight Task

Given that the primary objective of our empirical research is to observe human behavior and collect relevant data to aid in the model's development, I recorded the detailed time intervals during which participants performed checklist items and calculated multiple mean values for 18 participants, as shown in Table 3.1.

When examining the time required for participants to complete the checklist's many parts, I separated each item into four sections: Reading, Checking, Action, and Others. Reading denotes the time required for participants to read and comprehend the checklist; Checking denotes the time required for participants to verify the current status of aircraft components; Acting denotes the time required for participants to actually control aircraft components; Other denotes the time required for participants to perform certain actions which are not directly controlling the aircraft (e.g., wait for several seconds).

The results, particularly the mean and Standard Deviation (SD) for each segment, indicate that participants took the longest time checking the status of aircraft components. The SD of checking is greater than that of other operations. I believe there are two scenarios in which this could occur. One is that because checking the aircraft components' status is the most critical portion of each checklist item, it takes longer for participants. Second, due to the interface differences between the simulator and the real Cessna 172SP, and despite

Table 3.1: Mean of human movement time interval for pre-flight task.

Checklist Item	Mean of Time Interval (s)			
	Reading	**Checking**	**Acting**	**Other**
Ignition Switch	1.88	1.88	*N/A*	*N/A*
Avionics	0.6	1.06	*N/A*	*N/A*
Master Switch	1.07	1	1.04	*N/A*
Fuel Level	1.31	2.62	*N/A*	*N/A*
Flaps	1.32	1.49	1.25	*N/A*
Throttle	1.78	1.34	1.91	*N/A*
Mixture	1.26	0.94	*N/A*	*N/A*
Beacon Light	1.57	1.37	1.14	*N/A*
Auxiliary Fuel Pump	1.76	1.59	1.16	*N/A*
Mixture	1.75	1.19	0.63	*N/A*
(Wait 5 Sec)	*N/A*	*N/A*	0.56	5.28
Auxiliary Fuel Pump	0.86	0.84	0.74	*N/A*
Ignition Switch	1.74	1.44	1.58	*N/A*
Mixture	*N/A*	0.67	1.09	*N/A*
Ignition Switch Check	*N/A*	3.27	0.67	*N/A*
Oil Pressure	1.85	3.87	*N/A*	*N/A*
Avionics	1.43	1.23	1.1	*N/A*
Mean	1.43	1.61	1.07	5.28
SD	0.39	0.9	0.4	*N/A*

the fact that participants are allowed to rehearse for a specified length of time prior to the task beginning, it may take some time for participants to locate aircraft components. I believe that participants would not spend that much time on a real plane.

In comparison, the participants spent the least amount of time dealing with the airplane components that had the lowest SD. The primary reason for this is that the overwhelming majority of interactive items in this study phase are straightforward buttons or switches. Interactions with non-button or switch-based controls (such as the mixture and throttle) will take significantly longer.

3.4.2 Take-off Task

The take-off mission is designed to assess the participant's continuous control of the aircraft, which includes the yoke, pedal, throttle, and mixture. Because the time required to regulate the aircraft's attitude cannot be measured, I study the simulator's flight data and controller feedback and extrapolate human behavior from these facts. The most crucial data during the take-off phase are the aircraft's attitude, airspeed, and engine RPM. This section is devoid of buttons and switches.

For task 2, I collected a variety of data from X-Plane, including airspeed, vertical speed, altitude, controller input, engine RPM, pitch/roll degree, heading, and G load. The simulator began recording automatically when participants pulled the yoke and stopped recording when the aircraft reached 4000ft for the first time. To facilitate analysis and interpretation, I separated the pilot data into four categories: rudder input and heading, elevator input and pitch degree, aileron input and roll degree, and throttle and engine RPM. Due to the volume of data collected, not all graphs can be included here. However, Figure 3.3 illustrates four data sets from participant #1 using example graphs.

As with Task 1, I calculated the mean value of the data collected from X-Plane using the 95 percent confidence interval for human input, such as elevator, aileron, and redder (Table 3.2 and Figure 3.4). These data will be used to construct and evaluate subsequent models.

3.4.3 Situational Awareness and Workload

As previously stated, all participants must complete a post situational awareness and a mental workload assessment. To compare pre-flight and take-off tasks, a paired-samples t-test was used (Figure 3.5). Even though there is no planning or design for situational

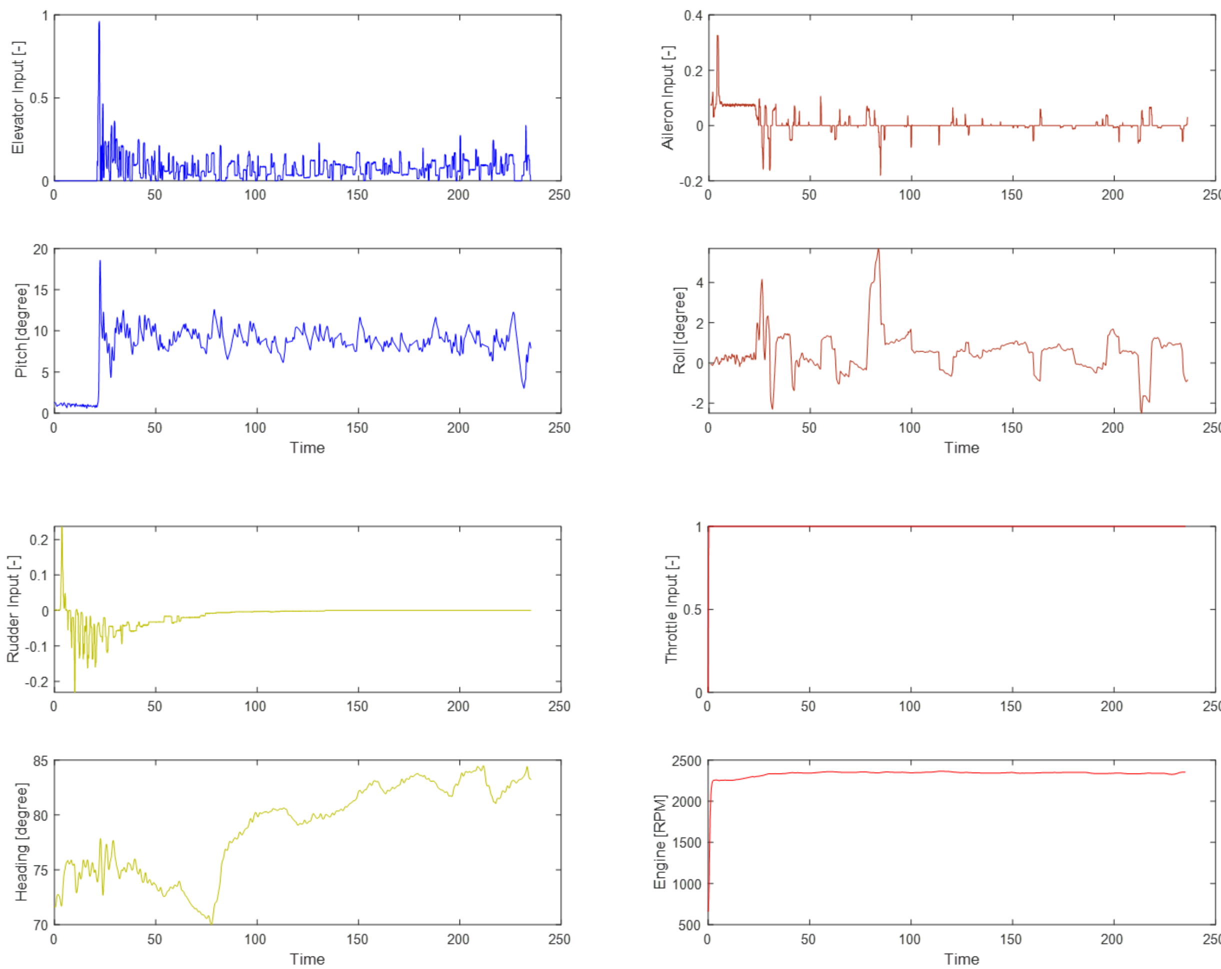

Figure 3.3: Participant #1's data measured and collected by X-Plane 11.

Table 3.2: Mean of human airspeed, pitch/roll degree, heading, and total time for take-off task.

Participants #	**Mean of Data**				
	Airspeed	**Pitch**	**Roll**	**Heading**	**Time**
1	73.66	8.95	0.57	79.23	230.41
2	71.51	5.85	0.13	73.88	308.73
3	75.62	7.92	-0.16	75.15	233.05
4	63.73	7.89	0.47	84.47	286.62
5	77.04	8.29	0.3	72.43	226.31
6	65.84	7.36	0.08	72.06	309.77
7	74.84	8.66	0.63	72.93	230.12
8	68.01	10.38	0.19	73.46	235.07
9	74.11	8.89	0.02	78.54	229.57
10	69.18	10.08	0.23	73.72	231.21
11	73.97	8.89	0.3	77.23	230.48
12	72.44	5.65	-0.01	74.19	310.67
13	70.36	9.88	0.63	68.62	227.8
14	70.51	6.21	0.2	81.89	302.42
15	71.01	9.56	1.81	71.86	234.41
16	76.17	8.41	-0.05	69.73	223.95
17	70.12	9.92	-0.09	82.7	225.67
18	67.51	9	0.06	72.61	256.74
Mean	71.42	8.43	0.3	75.26	251.83
SD	3.67	1.42	0.45	4.47	34.11

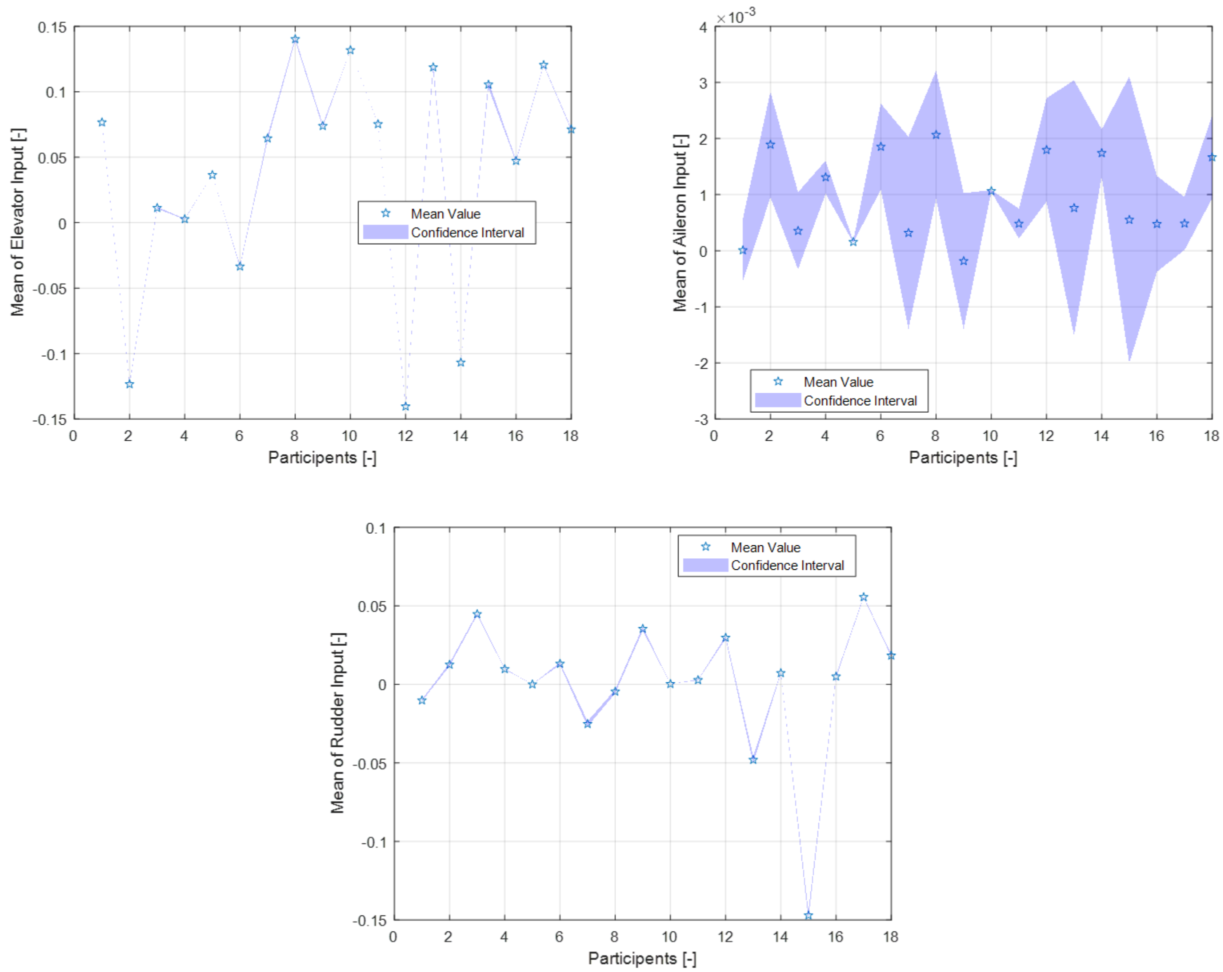

Figure 3.4: Mean value of human input with 95 percent confidence interval measured by X-Plane.

awareness at this level of the model's development, I intend to learn about the pilots' situational awareness and workload during the pre-flight and take-off processes in our study.

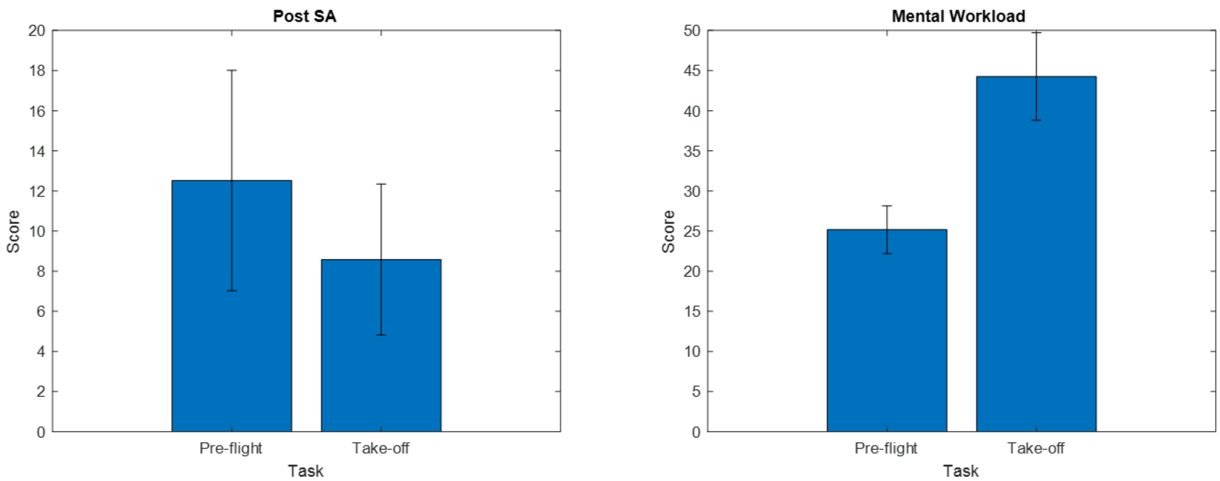

Figure 3.5: Human post-experiment situational awareness and NASA-TLX score for pre-flight and take-off tasks. Error bars represent 95 percent confidence intervals.

The difference in situational awareness scores between pre-flight ($M = 51, SD = 12.52$) and take-off ($M = 80, SD = 8.58$) tasks was statistically significant; $t(19) = -9.72, p < 0.001$. There was also a significant difference in workload scores between pre-flight ($M = 25.17, SD = 6.75$) and take-off ($M = 44.25, SD = 12.45$) tasks; $t(19) = -8.83, p < 0.001$. These findings imply that the take-off procedure necessitated increased situational awareness and a greater workload. This could be because pilots devote more time and effort to take-off than to pre-flight. This, however, requires additional investigation to establish.

Chapter 4

Pilot Behavior Performance Model Design and Development

Piloting an airplane is a difficult endeavor that requires a range of mental abilities, including perceptual, cognitive, and motor responses. A complete flight consists of several stages with varying degrees of complexity, including preparation, taxiing, take-off, cruising, approaching, and landing. For instance, pilots must pay special attention to current weather conditions and heed air traffic control directions during take-off. During the take-off phase, the pilot must manually manipulate the throttle, rudder, and yoke to maintain the aircraft's attitude and direction. Cognitive knowledge and abilities are critical components of task accomplishment for pilots, and they are also the focus of pilot training. Because cognitive architecture models are composed of declarative knowledge chunks and procedural production procedures, they are well-suited for simulating pilot competency and performance [3].

I created pilot models for the QN-ACTR cognitive architecture in the current study [14], I called it QN-ACTR-XP. The models may be used in conjunction with flight simulators (X-Plane 11) to provide a variety of human factors data, including as performance, mental workload, and situation awareness.

4.1 Tools and Cognitive Structure

4.1.1 X-Plane 11

The current study's flight simulation was conducted using X-Plane 11 and a Cessna 172SP aircraft plug-in, as seen in Figure 4.1. X-Plane is a flight simulator developed by Laminar Research that runs on Mac OS and Windows computers. X-Plane has traditionally been regarded as a video game by the general audience. However, as the sole FAA-certified simulator for pilot training, it is now widely employed in scientific study and commercial development. X-Plane supports data communication via User Datagram Protocol (UDP) with external programs at a rate of 99 data packets per second. Through the UDP interface, a pilot model in QN-ACTR can receive and send data to X-Plane in real time. The data link enables the model to assess the aircraft's current status and its external environment and to provide control directives to the aircraft.

Figure 4.1: Screenshot of Cessna 172SP in X-Plane 11.

4.1.2 ACT-R and QN-ACTR

This pilot performance model was developed using QN-ACTR as the cognitive architecture. QN-ACTR has been demonstrated to be relevant to multi-task scenarios and data-link connections in earlier work on driving simulation tasks [36]. Because simulated driving and flight activities are comparable, I believe that QN-ACTR can be perfectly tailored to the needs.

QN-ACTR blends ACT-R with a queuing network (QN) to manage and process multiple tasks at the cognitive module level [14]. The QN-ACTR framework is illustrated below (Figure 4.2). The queuing network enables simulation of multi-task performance, while the ACT-R component continues to simulate cognitive processes. The queuing network significantly aided cognitive processes in multitasking. For instance, the pilot may be required to simultaneously operate the flying yoke and the throttle during the entire mission. While the pilot model's ACT-R architecture is incapable of handling such concurrent activities, the QN-ACTR is. As a result, I chose QN-ACTR as the foundation for our approach. Recent advancements in QN-ACTR have increased its capability for modeling and simulating mental workload under time constraints [14], driver take-over performance in partially automated cars [16], and driver situation awareness [36]. These studies have broadened the domains and task settings in which cognitive architecture models can be applied to human factors engineering. The source code and models for QN-ACTR are available on its website (https://github.com/HOMlab).

ACT-R represents declarative memory (in chunks) and procedural memory through the application of production rules. It consists of four major modules and a production system: a visual/aural module, a goal module, a declarative module, and a manual module [37]. Additionally, there will be a central system that will coordinate the connection of each module via production rules and buffers [37]. ACT-R determines whether to take an action or change the model's state by accepting external information (often from visual and aural modules) and declarative knowledge. However, the primary constraint of ACT-R is that each chunk may only be kept in a single buffer, and each production rule can only be matched once [4]. Multiple goals can be saved in a single buffer when queuing is network-integrated. When a busy module receives a large number of requests, QN-ACTR can sort and execute them sequentially to avoid jamming.

Both QN-ACTR and ACT-R have two distinct types of memory: declarative and procedural. Procedural memory is constructed from productions and is used to resolve issues. Declarative memory stores data, and it is typically associated with a certain situation. When the contents of buffers match existing production rules, actions are taken in accordance with the rules.

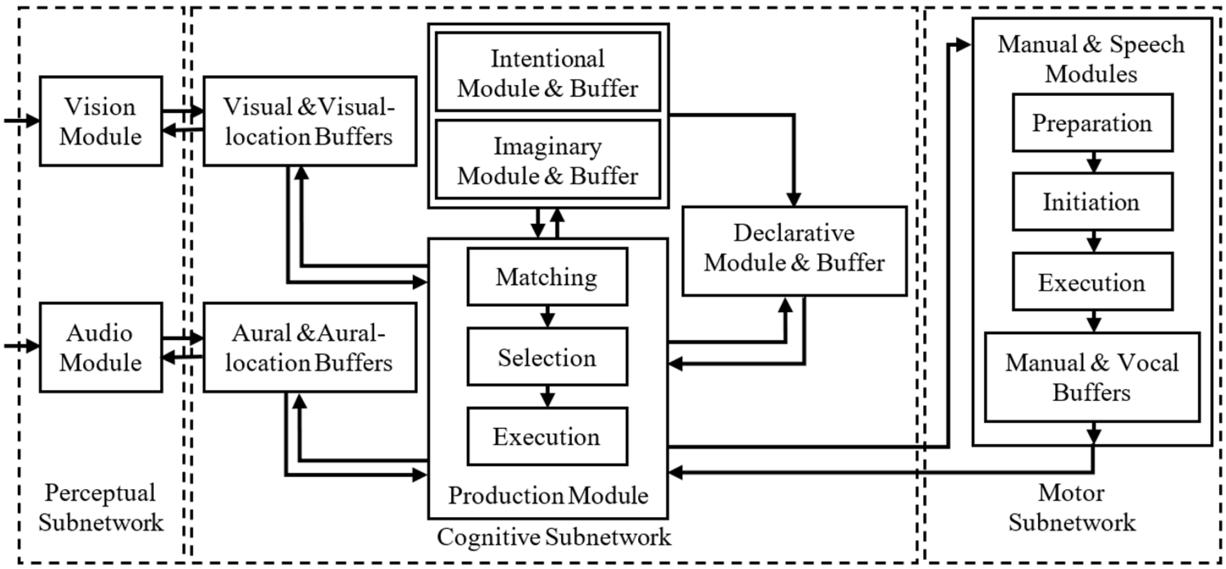

Figure 4.2: Structure of QN-ACTR, adapted from Cao & Liu 2013.

4.1.3 X-Plane Connect

The simulation is run as a control loop between the cognitive model in QN-ACTR (java programming language) and X-Plane 11 via a UDP connection. X-Plane will continuously update the model with aircraft and environment data. The model will adhere to its own production rules, process data, and issue control commands. Commands such as pressing a button or turning the yoke are transformed to X-Plane-compatible data and returned to X-Plane. The simulation of piloting is accomplished by the use of a continuous loop. When a pilot prepares an airplane, for example, he or she must interact with numerous buttons and switches in the cockpit and monitor the state of each component. To emulate this, X-Plane will communicate this data to the model through UDP. The model will then influence operations such as monitoring the state of those components and making decisions based on the data. The model will then replicate the pilot's subsequent actions, transform them to data commands, and send them back to X-Plane.

To connect the model to X-Plane, I utilized NASA's X-Plane Connect (XPC). XPC is an open-source research tool that communicates with the commercial flight simulator software X-Plane. XPC enables users to operate aircraft and receive status information from aircraft simulators in real time over a network using C, C++, Java, MATLAB, or Python functions. This research tool has been used to visualize aircraft pathways, evaluate control algorithms, simulate active airspace, and produce out-of-the-window visuals for in-house flight simulation software. The source code for XPC is available on GitHub (https://github.com/nasa/XplaneConnect).

Figure 4.3: The connection between QN-ACTR-XP and X-Plane.

The link between QN-ACTR-XP and X-Plane is depicted in Figure 4.3. I verified the connectivity of the various components and the presence of any probable delay. I discovered that the delay introduced by each component of connecting is really little, insignificant enough not to impair the model's performance or accuracy, and so the delay introduced by data transmission may be ignored during development and validation.

4.2 Model Development

4.2.1 Basic Design

In general, pilots must monitor the external environment and aircraft status continuously throughout the flight and communicate with ATC. This requires the pilot model to continuously utilize the visual and auditory modules to process flight-related data and forward it to cognitive stages for further processing. The procedural module will execute pattern matching using the model's procedural memory and will consult the model's existing declarative memory in order to comprehend the present situation and perform specified operations. For example, when a small airplane with a single pilot takes off, the pilot must manually manage the airplane's attitude and throttle and determine the aircraft's present state using the attitude indicator on the dashboard. Simultaneously, the pilot must check the aircraft's engine speed as shown by the speedometer and continuously change the throttle. At this level, because ATC rarely communicates with the pilot, the pilot's aural module does not require extensive speech processing. The visual module consumes the majority of the mental power. The pilot's manipulation of the plane to take off reflects the outcome.

I separated the pilot's cognitive processing and behavior into three stages in a continuous cycle, namely monitoring, decision-making, and control, in order to analyze them

(Figure 4.4). The monitoring stage entails attending to external displays, perceiving external information via the visual or aural module, such as reading the aircraft's instruments or hearing the information communicated with ATC during the flight, and cognitively preparing the information for the production module. The decision-making stage refers to the process through which the production module makes decisions and selects replies, before creating actions via production rule pattern matching. The control stage is the stage in which the motor module receives action commands and executes motor control operations. The model determines the amount of time required to finish the mental processing. To be explicit, I anticipated that ATC would not send any instructions to the pilot at this stage of the study. I did not refer to the aural module during the model's development.

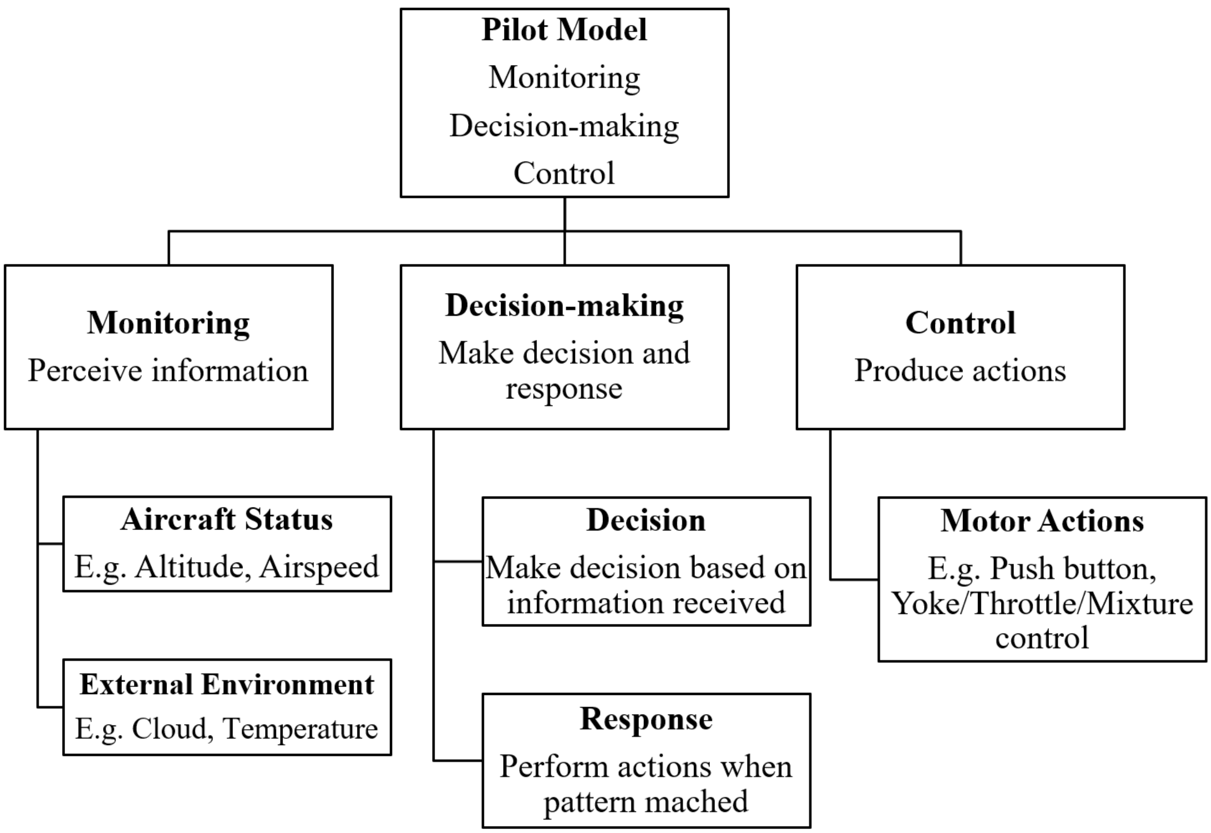

Figure 4.4: The structure of the QN-ACTR-XP.

4.2.2 Control Actions

The model can use several types of control actions to control the aircraft.

Buttons and Switches. The simplest sort of control action is the interaction between the model and the buttons or switches. When a button is to be pressed or a switch adjusted, the model simulates the human pilot moving the sight to the target position, then moving the hand to the target position and operating it. The model will communicate with X-Plane via UDP in order to manipulate the button or switch. Once X-Plane receives the command, the modification is immediately implemented.

Steering. When manipulating the aircraft's steering pedals during taxiing, the pilot should maintain their gaze on the taxi line directly front of the aircraft to assess the aircraft's current location and change the pedal angle accordingly. Additionally, the model's steering control is implemented by sending appropriate commands to X-Plane.

Rotation on Longitude/Vertical/Lateral Axis. When manipulating the aircraft's steering pedals during taxiing, the pilot should maintain their gaze on the taxi line directly front of the aircraft to assess the aircraft's current location and change the pedal angle accordingly. Additionally, the model's steering is controlled by sending appropriate orders to X-Plane.

Throttle. Because throttle control is mostly dictated by the aircraft's speed and current flight condition, the pilot's gaze should be directed onto the speedometer and engine RPM. The mode of communication between the model and X-Plane is identical to the mode of steering and rotation.

In general, the many sorts of operations discussed above can be summarized as the following two activities in the model:

x-plane-button-action. This action is applicable to all pilot button and switch interactions. This is often a one-time action, and the pilot is not required to interface with these components on a constant basis. When the model needs to perform this action, it simply needs to transmit a single instruction to X-Plane.

x-plane-continuous-action. This action is applicable to all pilot-yoke, rudder, throttle, and mixture interactions. This action is often linear and takes time, with the pilot manipulating these controls continuously and altering input values in real time. The model creates N discrete actions and continually transmits them to X-Plane using the specified execution time and number of actions. The time interval between discrete actions is typically 50ms, which is deemed small enough to imitate the actual linear input.

4.2.3 Pilot Performance Model for Pre-flight Task

In comparison to the take-off task, developing the pre-flight task model is rather straightforward. Because the pilot only needs to complete the items on the checklist sequentially when executing pre-flight task, modeling does not require consideration of multitasking. Because QN-ACTR calculates the time required for human hand and eye movement based on the distance between various components, I modified the simulator's control panel to a two-dimensional coordinate system. I defined the position and size of the various components using coordinates and area (Figure 4.5). The panel is 2000 pixels in height and 1100 pixels in breadth. The checklist is supposed to be located at (1600,500).

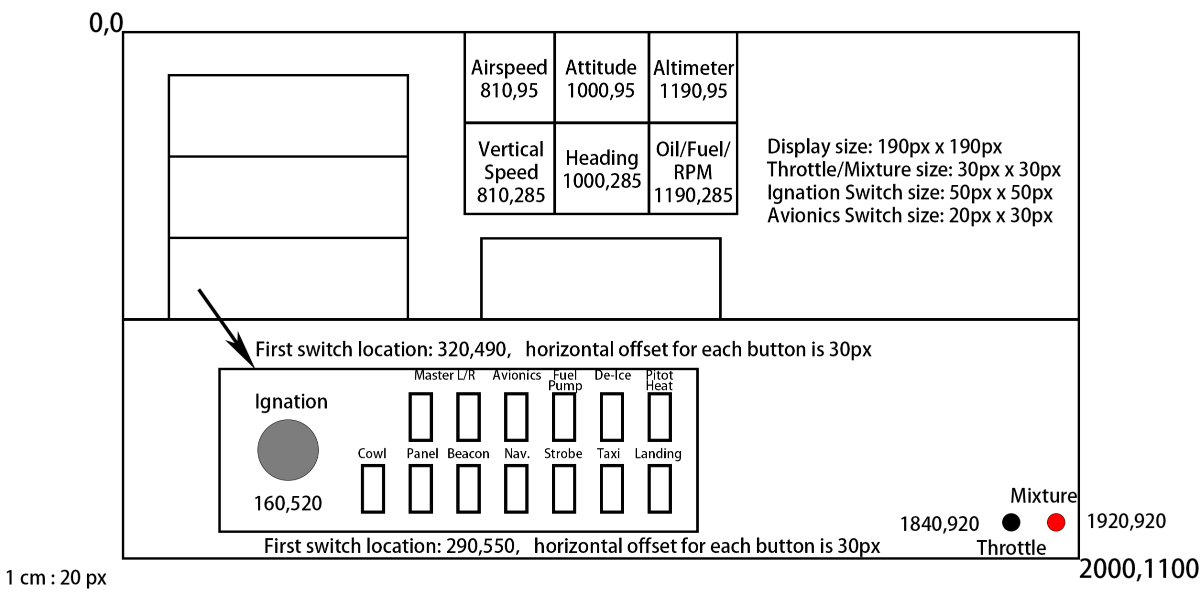

Figure 4.5: The control panel displayed in two-dimensional coordinate system.

I separated each step on the checklist into the following phases for the model that simulates a pilot executing a flying checklist (Appendix A). The model's major parameters are listed in Table 4.1. I did not utilize the default value of QN-ACTR, which is 0.02s, for the delay when setting or altering the imaginal buffer. The default value is derived from straightforward objects, such as a word or a symbol. Pilots are typically confronted with more complex circumstances while studying pilot behavior, such as examining values from a particular instrument panel. This suggests that 0.02 seconds is insufficient time to accurately imitate the pilot's conduct. I used 1.0s for the time being, and this figure may need to be examined further. Table 4.2 contains full descriptions of the model's steps derived from the checklist, whereas Table 4.3 contains the production rules for each step. The default settings of the QN-ACTR parameters are utilized to determine the model parameters. For instance, the duration of time spent by the hand and eyes when pressing a button or manipulating a switch. Additionally, each production rule takes 0.05 seconds to execute by default in QN-ACTR.

Table 4.1: Parameter used in QN-ACTR-XP model.

Parameters	**Value**	**Description**
:visual-attention-latency	0.085	It defines the time it takes when visual attention shifts. The default value is 0.085s.
:imaginal-delay	1.0s	It defines how long it takes to change or request the imaginal buffer. The default value is 0.2s.
preparation-duration	0.05	It defines the time it takes when a motor action is prepared. The default value is 0.05s.
initiation-duration	0.05	It defines the time it takes when a motor action is initialized. The default value is 0.05s.
execution-duration	1.0	It defines the time it takes when a motor action is performed. The default value is 1.0s.
finish-duration	1.0	It defines the time it takes when a motor action is finished. The default value is 1.0s.

Table 4.2: Model steps and detailed descriptions for pre-flight task.

Procedure	Description
ignition-off	The model moves the gaze to the ignition switch on the control panel, checks the status, makes sure it is off. If it's not, turn it off.
avionics-off	The model moves the gaze to the avionics switch on the control panel, checks the status, makes sure it is off. If it's not, turn it off.
master-off	The model moves the gaze to the master switch on the control panel, checks the status, makes sure it is off. If it's not, turn it off.
fuel-level-check	The model moves the gaze to the fuel level display on the control panel and the current fuel level.
flaps-up	The model moves the gaze to the flaps controller on the control panel, checks the status, makes sure the flaps are up. If it's not, retract flaps.
throttle-open	The model moves the gaze to the throttle controller on the control panel, checks the status, opens it to about 10%.
fuel-pump-on	The model moves the gaze to the fuel pump switch on the control panel, checks the status, makes sure it is on. If it's not, turn it on.
mixture-rich	The model moves the gaze to the mixture controller on the control panel, checks the status, makes it rich for 5 seconds, and then cuts it off.
fuel-pump-off	The model moves the gaze to the fuel pump switch on the control panel, checks the status, makes sure it is off. If it's not, turn it off.
ignition-start	The model moves the gaze to the ignition switch on the control panel, checks the status, switches on, and keeps holding on to it until the engine RPM is stable.
mixture-advance	The model moves the gaze to the mixture controller on the control panel, checks the status, makes it rich.
oil-pressure-check	The model moves the gaze to the oil pressure display on the control panel and checks the current oil pressure.

Continued on next page.

Table 4.2 Continued: Model steps and detailed descriptions for pre-flight task.

Procedure	Description
avionics-on	The model moves the gaze to the avionics switch on the control panel, checks the status, makes sure it is on. If it's not, turn it on.

Table 4.3: Production rules and detailed descriptions for pre-flight task.

Production Rule	Description
visually-attend-checklist	**IF** goal phase is 1, the visual module is free, and the imaginal module is free, **THEN** set visual location to (1600,500), and change goal step to x, goal phase to 2. *The model simulates pilots moving their gaze to the checklist. This stage can be shared with all steps.*
visually-encode-checklist-item	**IF** goal phase is 2, visual location is found, and the visual module is free, **THEN** move visual attention to the visual location to encode the information, store the coordinate of the visual location to the goal buffer, and change goal step to x, goal phase to 3. *The model simulates pilots reading the checklist. This stage can be shared with all steps.*
form-checklist-item-representation	**IF** goal step is x, goal phase is 3, the information from the visual module is found, and the imaginal module is free, **THEN** create an imaginal buffer representing the information from the goal buffer, and change goal step to x, goal phase to 4. *The model simulates pilots' understanding from what they read from the checklist. This stage is individually set based on the checklist and can't be shared.*
visually-attend-aircraft-component	**IF** goal step is x, goal phase is 4, and the information is the same as in imaginal buffer, **THEN** set visual location to the component's location saved in imaginal buffer, and change goal step to x, goal phase to 5. *The model simulates pilots moving their gaze to the component they are required. This stage is individually set based on the checklist and can't be shared.*

Continued on next page.

Table 4.3 Continued: Production rules and detailed descriptions for pre-flight task.

Production Rule	Description
visually-encode-checklist-item	**IF** goal phase is 5, visual location is found, and the visual module is free, **THEN** move visual attention to the visual to encode the information, store the coordinate of the visual location to the goal buffer, and change goal step to x, goal phase to 6. *The model simulates pilots reading the current status of the component. This stage can be shared with all steps.*
form-aircraft-component-status-representation	**IF** goal step is x, goal phase is 6, the information from the visual module is found, and the imaginal module is free, **THEN** create an imaginal buffer representing the information from the goal buffer, and change goal step to x, goal phase to 7. *The model simulates pilots' understanding from what they read from the component. This stage is individually set based on the checklist and can't be shared.*
take-control-action	**IF** goal step is x, goal phase is 7, the information in imaginal buffer matches the condition, and the motor module is free, **THEN** acting by calling either x-plane-button-action or x-plane-continuous-action, and change goal step to x+1, goal phase to 1. *The model simulates pilots' acting based on what they got from the previous stage. This stage is individually set based on the checklist and can't be shared.*
no-action-needed	**IF** goal step is x, goal phase is 7, and the information in the imaginal buffer doesn't match the condition, **THEN** do nothing, and change goal step to x+1, goal phase to 1. *The model simulates pilots not acting based on what they got from the previous stage. This stage is individually set based on the checklist and can't be shared.*

4.2.4 Pilot Performance Model for Take-off Task

As previously said, take-off procedures are more difficult than pre-flight. When pilots complete take-off operations, they spend less time checking the aircraft's state and external surroundings and more time controlling the aircraft in real time using the yoke and rudder. Simultaneously, the pilot has little to no interaction with buttons and switches in this activity, preferring to use the yoke and rudder continuously. The model represents the pilot's continual movements in this task by rapidly transmitting commands to the simulator. As a result, simulating the pilot's judgment of the aircraft's current state and performing actions has become critical. To begin, a model replicating the driver's control of the vehicle's steering is presented in order to address this issue. In 2004, Salvucci and Gray created this approach, which enables drivers to center their automobiles around two points: near point and far point [41]. The term "near point" refers to the distance between the vehicle and the road's center. The far point represents the driver's estimated value for the current curvature of the road. The following equation 4.1 can be used to represent the function:

$$\Delta\varphi = k_f \Delta\theta_f + k_n \Delta\theta_n + k_I \theta_n \Delta t, \tag{4.1}$$

where $\Delta\varphi$ means the change of steering angle. k_f, k_n, and k_I represents three constants. $\Delta\theta_f$ and $\Delta\theta_n$ are the change of far point and near point. θ_n means the current near point, and Δt is the control cycle. The typical default value of the control cycle is 0.15s.

Because prior study has demonstrated that the model stimulates the driver's steering and centering of the vehicle [36], I attempted to adapt this formula to the pilot's behavior when operating the plane's pitch/roll/yaw axis. To begin, unlike when driving a vehicle, pilots do not have an external visual reference to assist them in identifying the current flight route. As a result, I believe the distant point does not apply to the pilot's control model. In comparison to the distance between the car and the road's midpoint, the difference between the plane's present state and expected state serves a similar role in the control model as the distance between the automobile and the expected state. For instance, if the expected direction at the moment is 80 degree and the actual heading is 75 degree, the difference is 5 degree. Thus, on the pitch/roll/yaw axis, the pilot's control model can be written as the following equation 4.2:

$$\Delta\varphi = k\Delta\theta + k_I \theta \Delta t, \tag{4.2}$$

where $\Delta\varphi$ means the change of control input on the pitch/roll/yaw axis. k and k_I represents two constants. $\Delta\theta$ is the change of the difference between the current state of the airplane and the expected state. θ means the current difference between the current state of the airplane and the desired state, and Δt is the control cycle.

After identifying the control function, the constants must be determined. Due to the fact that I modified Salvucci & Gray's code, I did not utilize their default values of $k = 13.5, k_I = 35$. I did numerous tests and discovered that, when k is maintained constant, the bigger the value of k_I, the faster the aircraft attitude recovery time. When k_I remains constant, the larger the value of k, the greater the aircraft attitude swing. Technically, we would have three distinct equations, one for each axis operation, which means that I would need to determine three sets of k and k_I. However, after conducting several experiments, I discovered that employing three distinct sets of constants had a negligible effect on the model's performance. As a result, the current study's k and k_I values are identical for the three groups, which are 35 and 15, respectively.

As with the pre-flight work, I breakdown the take-off checklist elements (Table 4.4). The pre-flight task reuses all production rules. Nevertheless, certain new production rules have been added (Table 4.5). All parameters from the pre-flight task model are retained.

Table 4.4: Model steps and detailed descriptions for take-off task.

Procedure	**Description**
throttle-full	The model moves the gaze to the throttle controller on the control panel, checks the status, sets it full.
mixture-rich	The model moves the gaze to the mixture controller on the control panel, checks the status, makes it rich for 5 seconds, and then cuts it off.
pilot-control	The model moves the gaze between the aircraft attitude indicator on the control panel and the outside to check the current aircraft status, then controls the rudder and yoke to take-off.
throttle-full	The model moves the gaze to the throttle controller on the control panel, checks the status, makes sure it's full.
fuel-level-check	The model moves the gaze to the control panel's fuel level display and checks the current fuel level.
instruments-check	The model moves the gaze to all necessary instruments and check their status.
throttle-adjust	The model moves the gaze to the throttle controller on the control panel, checks the status, and adjusts it.

Table 4.5: Production rules and detailed descriptions for take-off task.

Production Rule	Description
pilot-control-attend-outside	**IF** goal step is 1, **THEN** set visual location to (20,20), and change goal step to 2. *The model simulates pilots moving their gaze outside and checking aircraft status.*
pilot-control-perceive-attitude	**IF** goal step is 2, and visual location is found, **THEN** move visual attention to the visual location to encode the information, store the coordinate of the visual location to the goal buffer, and change goal step to 3. *The model simulates pilots getting the aircraft's current pitch/roll/yaw.*
pilot-control-action	**IF** goal step is 3, the information from the visual module is found, and the manual module is free, **THEN** perform an action by x-plane-do-pitch-control, x-plane-do-yaw-control, and x-plane-do-roll-control, and change goal step to 1. *The model simulates pilots controlling the aircraft on the pitch/roll/yaw axis.*

4.3 Results

Along with the dataref included with X-Plane, QN-ACTR additionally records data generated by the model individually. Following the model's execution, QN-ACTR reports the name of each task and the time at which each production rule was executed. Additionally, the model outputs the required workload on a second-by-second basis. After developing the model, I utilized it to replicate the pre-flight and take-off procedures independently. The model carried out pre-flight and take-off activities sequentially and 10 times each. Each piece of data was captured and compared to previously collected human data. For the simulator setup, I used the same settings as in the human tests. Please refer to Chapter 3 for details on the essential stages of scenarios, the aircraft, and the data output mechanism; I am not repeating them here.

4.3.1 Pre-flight Model

The time required to perform each step and stage can be determined using the data output from the model, as shown in Table 4.6, similarly to Table 3.1.

To verify the model's correctness, I calculated the Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE) for the model data and human data. The RMSE indicates the difference in data between model and human experiments, whereas the MAPE indicates the model's prediction accuracy. Both equations 4.3 and 4.4 can be expressed as follows:

$$RMSE = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}(x_{hi} - x_{mi})^2}, \tag{4.3}$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{x_{hi} - x_{mi}}{x_{hi}}\right|, \tag{4.4}$$

where X_h represents the human data, X_m means the model data, i is the index, and n is the sample size.

The model's entire running time is 69.84 seconds. In comparison to the average human total time of 64.13 seconds, this is a difference of 8.18 percent. In general, the model is capable of simulating human behavior in task 1. Figure 4.6 depicts the overall model fitness based on the mean value of each stage in task 1 for both human and model data. The RMSE is 0.65 seconds, the MAPE is 40 percent, and the coefficient of determination

Table 4.6: Mean of model behavior time interval for pre-flight task.

Checklist Item	**Mean of Time Interval (s)**			
	Reading	**Checking**	**Acting**	**Other**
Ignition Switch	1.65	1.70	*N/A*	*N/A*
Avionics	1.65	1.70	*N/A*	*N/A*
Master Switch	1.65	1.65	1.15	*N/A*
Fuel Level	1.75	1.55	*N/A*	*N/A*
Flaps	1.65	1.70	1.20	*N/A*
Throttle	1.65	1.65	1.15	*N/A*
Mixture	1.70	1.65	*N/A*	*N/A*
Beacon Light	1.70	1.60	1.15	*N/A*
Auxiliary Fuel Pump	1.70	1.60	1.15	*N/A*
Mixture	1.65	1.49	1.35	*N/A*
(Wait 5 Sec)	*N/A*	*N/A*	1.20	5.05
Auxiliary Fuel Pump	1.65	1.65	1.15	*N/A*
Ignition Switch	1.70	1.60	2.10	*N/A*
Mixture	*N/A*	1.65	1.60	*N/A*
Ignition Switch Check	*N/A*	1.60	1.15	*N/A*
Oil Pressure	1.65	1.55	*N/A*	*N/A*
Avionics	1.65	1.55	1.15	*N/A*
Mean	1.67	1.62	1.29	5.05
SD	0.032	0.062	0.29	*N/A*

(r-squared) is 0.06. Both MAPE and RMSE indicate that the model cannot accurately imitate human behavior in task 1, and the model also has a significantly smaller SD than the human data.

4.3.2 Take-off Model

As in Chapter 3.4.2, I estimated the mean of several values from the output data of the model. The average airspeed is 75.20 knots, the average pitch angle is 7.95 degree, the average roll angle is 0.03 degree, the average heading is 73.92 degree, and the average running duration is 196.11 seconds. Then I made a comparison between those ideals and human values. The RMSE is 24.93 percent, while the MAPE is 24.13 percent. The mean value of elevator/aileron/rudder input is shown in Figure 4.7, together with the 95 percent confidence interval and human values, to demonstrate the model's fitness. Each graph's final value represents the model's data. The aircraft's height change from the ground to 4000 feet is depicted in Figure 4.8. The solid line reflects the data for the model, while the other lines indicate the data for the participants.

Both the RMSE and MAPE values demonstrate that the model accurately predicts the pilot's behavior during take-off tasks. The model's input values and altitude change also fall within the range of human data. Similar to the model's performance in the pre-flight job, the model's standard deviation is similarly smaller than that of human data.

4.3.3 Workload

QN-ACTR estimates mental workload by examining overall server use [13]. The average utilization per second shows the mental workload each second, ranging from 1.0 (very busy) to 0.0. (idle). I computed the workload of the model executing pre-flight and take-off operations using the workload data collected by QN-ACTR. Pre-flight task workload is 0.22 and take-off task workload is 0.13 for the model. Due to the fact that the units of the data output by QN-ACTR and the human data I obtained are significantly different, and there were only two tasks, the data cannot be compared directly, and I was unable to perform regression. If, however, a score of zero in NASA-TLX indicates that the workload is zero, then the human workload in the take-off task is approximately 1.75 times that of the pre-flight activity (see Chapter 3.4.3). According to the model's output, the model's workload during take-off is approximately 1.7 times that of the pre-flight task, which indicates that the model accurately forecasts human effort.

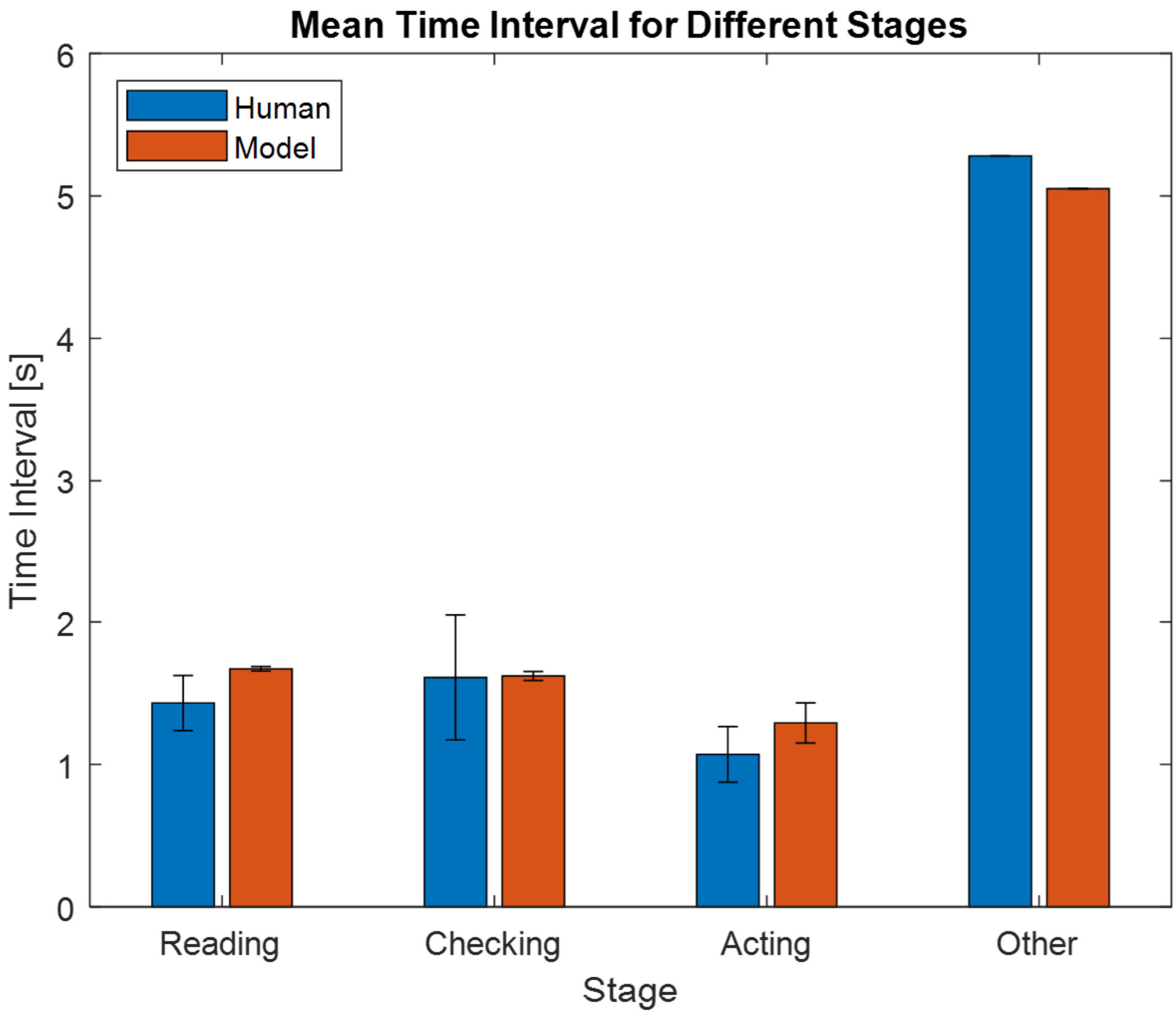

Figure 4.6: Mean time interval for different stage of human and model. Error bars represent 95 percent confidence intervals.

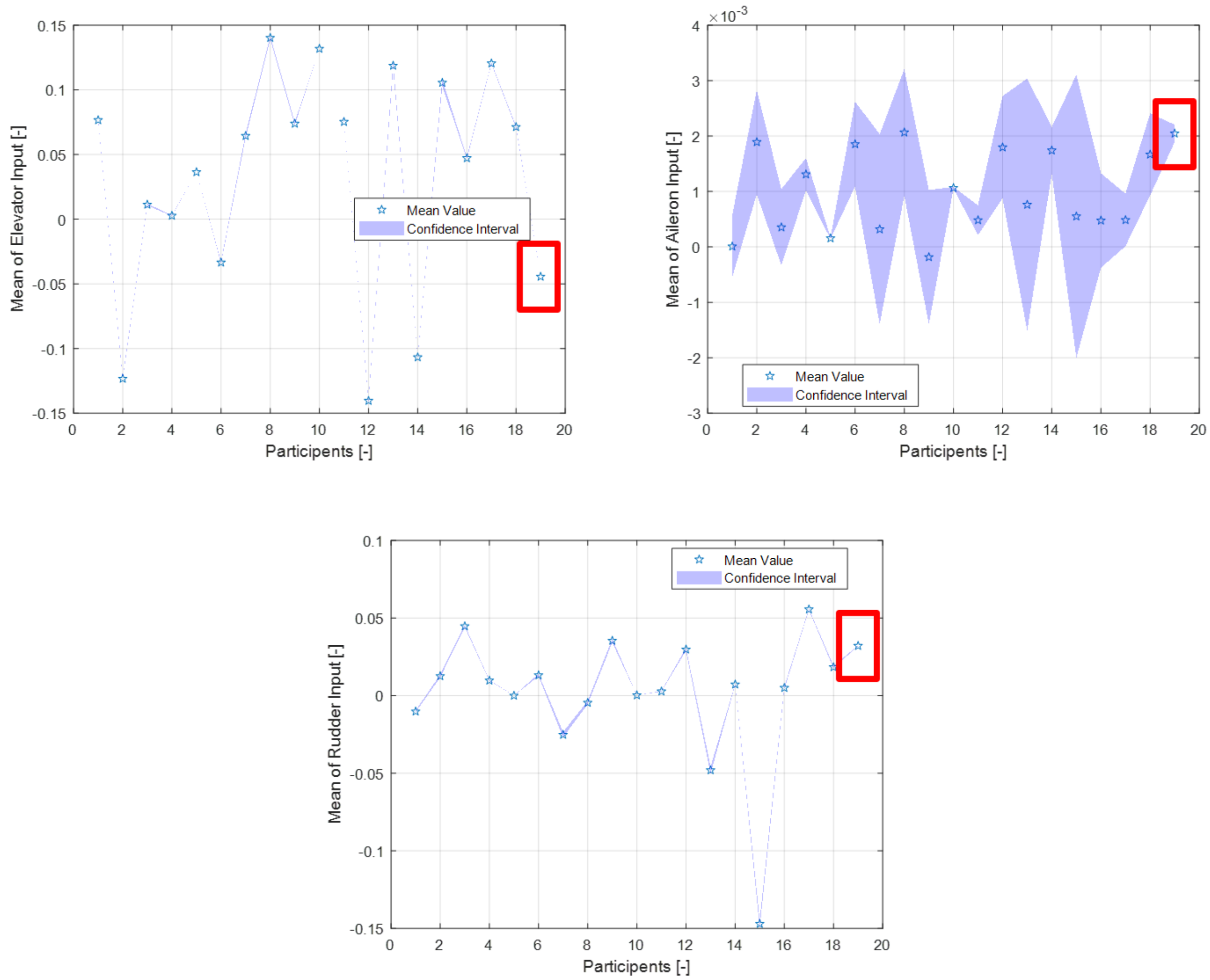

Figure 4.7: Mean value of elevator/aileron/rudder input with 95 percent confidence interval measured by X-Plane, the value in the red box represents the model's value.

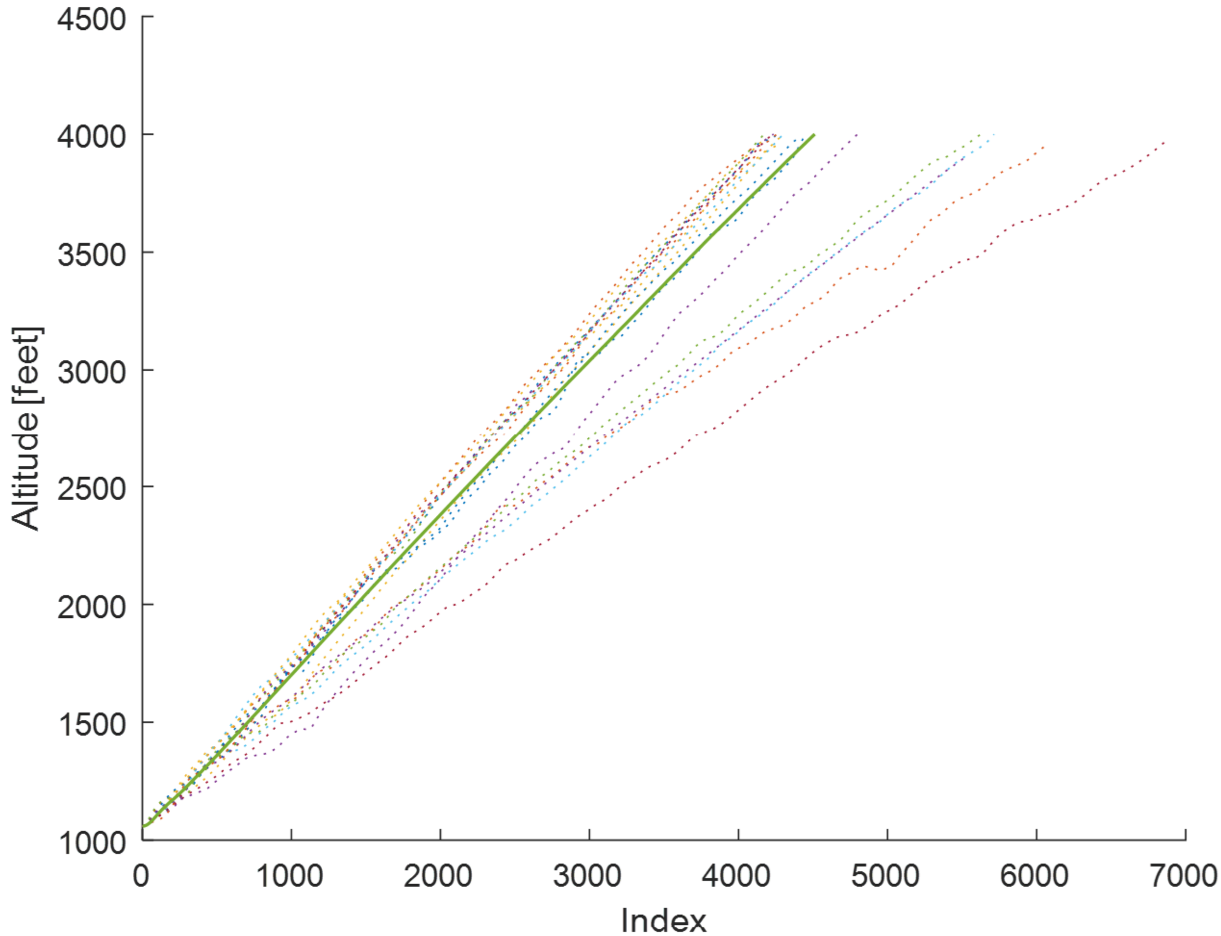

Figure 4.8: Aircraft's altitude changes for all participants and the model. The solid line represents the model's data.

4.4 Discussion

Again, based on the data above, the model is capable of performing well in pre-flight and take-off tasks and is consistent with data from human experiments. More precisely, the model's fitness for take-off task is superior to that for pre-flight task.

In my opinion, the higher RMSE and MAPE results for the pre-flight task are primarily due to the usage of QN-ACTR's default parameters. Additionally, when executing a pre-flight activity, the model will rigorously adhere to the pre-programmed stages. When participants complete tasks, they may take specific actions based on their own experience rather than the checklist, which results in shorter execution times for certain stages. Simultaneously, participants may be unfamiliar with the simulator interface, resulting in a lengthier execution time for some actions. I may conclude that more complex factors influence when humans do task 1 steps, which will be examined further in future studies.

In terms of take-off tasks, the model is capable of completing the task and obtaining human-like data. The reduced RMSE and MAPE values further demonstrate the model's fitness. However, it is worth noting that the model's output is excessively smooth and lacks the large fluctuations observed in human pilots. This demonstrates that the control function I employed is functional but not flawless. Additional variables should be added to the function to better simulate human behavior in future studies—for example, the aircraft's current speed, vertical speed, and wind speed.

The model is far from complete. The primary cause for this could be that I utilized the default QN-ACTR values. The default parameters of QN-ACTR, on the other hand, are suitable for performing some simple human tasks, such as reading a word or typing a letter. In light of the unique characteristics of pilot behavior and the complexity of flight duties, I modified certain parameters, such as imaginary-delay and execution-duration, which remain constants. Modifying these values will merely raise the overall duration of jobs, which will still fall short of accurately simulating pilots performing time discrepancies between separate steps.

Chapter 5

Conclusion

In the current study, QN-ACTR was employed as a cognitive architecture to develop a pilot behavior model that was used to replicate pilot behavior during pre-flight and take-off activities. The model includes human-like thinking patterns and knowledge reservoirs as a result of the incorporation of cognitive architecture. I design two sets of human experiments concurrently to gather relevant data for learning and comprehending the logic of human operation and applying it to the model. QN-ACTR defines the majority of parameters by default during model construction. I utilized the model to replicate two human experiments: pre-flight and take-off. Following data analysis, I believe that the model can accurately imitate human pilots performing the aforementioned duties. Although the generated data is comparable to human data, it still need development.

While I recorded participants' post-situational awareness during human studies, I did not simulate post-situational awareness during the model's development. At the moment, I am unable to conclude whether the model can accurately imitate human situational awareness. Nonetheless, earlier publications on models constructed using QN-ACTR, such as the driver behavior model [36], indicate that QN-ACTR is capable of completely simulating human situational awareness.

I am not considering any external aspects that may have an effect on humans in my present development. Additional human experiments in a variety of circumstances, including varying weather and temporal conditions, are required to further refine and test the model. I had documented only the participants' hand behavior in a previous experiment. In future investigations, researchers could incorporate other monitoring techniques, such as eye movement and EEG examinations. These tests will generate additional data.

Additionally, because the cognitive model and the flight simulator communicate via

UDP, numbers represent perceptual and motor information. This form of abstract simulation has trouble simulating actual display elements such as contrast ratio, glare, and layout. Computer vision modules can be integrated directly into cognitive architectures to recognize various items on the visual display.

Due to budget constraints, the simulator I've developed so far is relatively inexpensive, costing around \$5000. As a result, the interactive interface and operation mode of the simulator are considerably different from those of the genuine Cessna 172SP. This has the ability to have an effect on the participants' performance. In future studies, researchers could use ALSIM AL250 simulator from the Waterloo Institute for Sustainable Aeronautics Sim Lab. ALSIM AL250's innovative features, such as actual tactile force feedback and a dynamic sound system, are easily adaptable to suit different aircraft.

In future study, researchers could incorporate situational awareness simulation into the model and further improve it, especially by adding other factors that will allow it to adapt to human behavior in a variety of settings. Additionally, researchers could continue to construct models for additional missions, including approach, landing, and taxiing. Additionally, if practical, researchers could apply this concept to other aircraft types such as multi-engine piston planes and jets.

In conclusion, this work offers a model based on QN-ACTR that may imitate some of the behavior of pilots managing a Cessna 172SP aircraft, with satisfactory results. It establishes the groundwork for the application of cognitive models in civil aviation research. The distinction between this study and earlier ones is that this is one of the few published human behavior models in the field of civil aviation that includes thorough development methodologies and human trial data. Simultaneously, the model's adaptability is enhanced by the QN-ACTR cognitive architecture, which enables it to be applied to more flight tasks and aircraft types in future study. I expect that this model will be expanded and enhanced further in future study and that it will serve as a reference for other studies examining the use of cognitive models to simulate human behavior and other studies of civil aviation.

References

[1] Federal Aviation Administration. *NextGen implementation plan*. Federal Aviation Administration Washington, DC, 2013.

[2] John R. Anderson. *Rules of the Mind*. Psychology Press, 1 2014.

[3] John R Anderson and Kevin Gluck. What role do cognitive architectures play in intelligent tutoring systems. *Cognition & Instruction: Twenty-five years of progress*, page 227–262, 2001.

[4] John R Anderson, Christian Lebiere, Marsha Lovett, and Lynne Reder. Act-r: A higher-level account of processing capacity. *Behavioral and Brain Sciences*, 21(6):831–832, 1998.

[5] John R. Anderson, Michael Matessa, and Christian Lebiere. Act-r: A theory of higher level cognition and its relation to visual attention. *Human–Computer Interaction*, 12(4):439–462, 12 1997.

[6] Jerry Ball, Stuart Rodgers, and Kevin Gluck. Integrating act-r and cyc in a large-scale model of language comprehension for use in intelligent agents. In *AAAI workshop*, pages 19–25, 2004.

[7] Brad Best, Christian Lebiere, Dan Schunk, Ian Johnson, and Rick Archer. Validating a cognitive model of approach based on the act-r architecture. Technical report, Technical Report). Boulder, CO: Micro Analysis and Design, Inc, 2004.

[8] HAP Blom, GJ Bakker, PJG Blanker, J Daams, MHC Everdij, and MB Klompstra. Accident risk assessment for advanced air traffic management. In *Proceedings of 2nd USA/Europe Air Traffic Management R&D Seminar*. National Aerospace Laboratory NLR, 1998.

[9] Jan Boril, Miroslav Jirgl, and Rudolf Jalovecky. Use of flight simulators in analyzing pilot behavior. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 255–263. Springer, 2016.

[10] Michael D Byrne and Alex Kirlik. Modeling errors in taxiing of commercial aircraft. *Retrieved March*, 15:2002, 2002.

[11] Michael D. Byrne and Alex Kirlik. Using computational cognitive modeling to diagnose possible sources of aviation error. *The International Journal of Aviation Psychology*, 15(2):135–155, 4 2005.

[12] Michael D Byrne, Alex Kirlik, Michael D Fleetwood, David G Huss, Alex Kosorukoff, Ruei-sung Lin, and Chris S Fick. A closed-loop, act-r approach to modeling approach and landing with and without synthetic vision system (svs) technology. In *Proceedings of the Human Factors and Ergonomics Society annual meeting*, volume 48, pages 2111–2115. SAGE Publications Sage CA: Los Angeles, CA, 2004.

[13] Shi Cao and Yili Liu. Mental workload modeling in an integrated cognitive architecture. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 55, pages 2083–2087. Sage Publications Sage CA: Los Angeles, CA, 2011.

[14] Shi Cao and Yili Liu. Queueing network-adaptive control of thought rational (qn-actr): An integrated cognitive architecture for modelling complex cognitive and multi-task performance. *International Journal of Human Factors Modelling and Simulation 55*, 4(1):63–86, 2013. publisher: Inderscience Publishers Ltd.

[15] Kevin Corker. Study on the integration of human performance and accident risk assessment models: Air-midas & topaz. In *2005 International Symposium on Aviation Psychology*, page 147, 2005.

[16] Chao Deng, Shi Cao, Chaozhong Wu, and Nengchao Lyu. Modeling driver take-over reaction time and emergency response time using an integrated cognitive architecture. *Transp. Res. Rec.*, 2673(12):380–390, 12 2019. publisher: SAGE Publications.

[17] Thomas E Ferrin, Conrad C Huang, Laurie E Jarvis, and Robert Langridge. The midas display system. *Journal of molecular graphics*, 6(1):13–27, 1988.

[18] David C Foyle, Becky L Hooey, Michael D Byrne, Kevin M Corker, Stephen Deutsch, Christian Lebiere, Ken Leiden, and Christopher D Wickens. Human performance

models of pilot behavior. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 49, pages 1109–1113. SAGE Publications Sage CA: Los Angeles, CA, 2005.

[19] Florian Frische, Tomasz Mistrzyk, and Andreas Lüdtke. Detection of pilot errors in data by combining task modeling and model checking. In *IFIP Conference on Human-Computer Interaction*, pages 528–531. Springer, 2009.

[20] Brian F Gore. Man–machine integration design and analysis system (midas) v5: Augmentations, motivations, and directions for aeronautics applications. In *Human modelling in assisted transportation*, pages 43–54. Springer, 2011.

[21] Brian F Gore, Becky L Hooey, David C Foyle, and Shelly Scott-Nash. Meeting the challenge of cognitive human performance model interpretability though transparency: Midas v5. x. In *2nd International Conference on Applied Human Factors and Ergonomics*, pages 14–17, 2008.

[22] Brian F Gore and Peter A Jarvis. New integrated modeling capabilities: Midas'recent behavioral enhancements. *SAE transactions*, page 752–759, 2005.

[23] Miroslav Jirgl, Jan Boril, and Rudolf Jalovecky. Statistical evaluation of pilot's behavior models parameters connected to military flight training. *Energies*, 13(17):4452, 2020. publisher: Multidisciplinary Digital Publishing Institute.

[24] Miroslav Jirgl, Marie Havlikova, and Zdenek Bradac. The dynamic pilot behavioral models. *Procedia Engineering*, 100:1192–1197, 2015. publisher: Elsevier.

[25] O Klaproth, Marc Halbrügge, and Nele Russwinkel. Act-r model for cognitive assistance in handling flight deck alerts. In *Proceedings of the 17th International Conference on Cognitive Modeling, Montreal*, pages 83–88, 2019.

[26] Christian Lebiere, Rick Archer, Dan Schunk, Eric Biefeld, Troy Kelly, Laurel Allender, C Lebiere, R Archer, D Schunk, E Biefeld, et al. Using an integrated task network model with a cognitive architecture to assess the impact of technological aids on pilot performance. In *NASA Aviation Safety Program Conference on Human Performance Modeling of Approach and Landing with Augmented Displays*, page 165. Ames Research Center Mountain View, CA, 2003.

[27] Christian Lebiere, Rick Archer, Walter Warwick, and Dan Schunk. Integrating modeling and simulation into a general-purpose tool. In *Proceedings of the 11th International Conference on Human-Computer Interaction*, pages 22–27, 2005.

[28] Andreas Lüdtke, Jan-Patrick Osterloh, and Florian Frische. Multi-criteria evaluation of aircraft cockpit systems by model-based simulation of pilot performance. In *Embedded Real Time Software and Systems (ERTS2012)*, 2012.

[29] Andreas Lüdtke, Lars Weber, Jan-Patrick Osterloh, and Bertram Wortelen. Modeling pilot and driver behavior for human error simulation. In *International Conference on Digital Human Modeling*, pages 403–412. Springer, 2009.

[30] Nikolay Lyssakov and Elena Lyssakova. Human factor as a cause of aircraft accidents. In *II International Scientific-Practical Conference" Psychology of Extreme Professions"(ISPCPEP 2019)*, pages 130–132. Atlantis Press, 2019.

[31] Andreas Lüdtke, Denis Javaux, HUMAN Consortium, et al. The human project: Model-based analysis of human errors during aircraft cockpit system design. In *Human Modelling in Assisted Transportation*, page 79–95. Springer, 2011.

[32] Andreas Lüdtke, Jan-Patrick Osterloh, Tina Mioch, Frank Rister, and Rosemarijn Looije. Cognitive modelling of pilot errors and error recovery in flight management tasks. In *Human Error, Safety and Systems Development*, page 54–67. Springer, 2010.

[33] Duane T McRuer and Ezra S Krendel. Mathematical models of human pilot behavior. Technical report, ADVISORY GROUP FOR AEROSPACE RESEARCH AND DEVELOPMENT NEUILLY-SUR-SEINE (FRANCE), 1974.

[34] Clinton V. Oster, John S. Strong, and C. Kurt Zorn. Analyzing aviation safety: Problems, challenges, opportunities. *Research in Transportation Economics*, 43(1):148–164, 7 2013.

[35] Daan Marinus Pool, Peter MT Zaal, Herman J Damveld, Marinus Maria van Paassen, and Max Mulder. Pilot equalization in manual control of aircraft dynamics. In *2009 IEEE International Conference on Systems, Man and Cybernetics*, pages 2480–2485. IEEE, 2009.

[36] Umair Rehman, Shi Cao, and Carolyn MacGregor. Using an integrated cognitive architecture to model the effect of environmental complexity on drivers' situation awareness. *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, 63(1):812–816, 11 2019. publisher: SAGE Publications.

[37] Frank E Ritter, Farnaz Tehranchi, and Jacob D Oury. Act-r: A cognitive architecture for modeling cognition. *Wiley Interdisciplinary Reviews: Cognitive Science*, 10(3):e1488, 2019. publisher: Wiley Online Library.

[38] Alan H Roscoe and Georges A Ellis. A subjective rating scale for assessing pilot workload in flight: A decade of practical use. Technical report, Royal Aerospace Establishment Farnborough (United Kingdom), 1990.

[39] Christina F Rusnock and Christopher D Geiger. Using discrete-event simulation for cognitive workload modeling and system evaluation. In *IIE Annual Conference. Proceedings*, page 2485. Institute of Industrial and Systems Engineers (IISE), 2013.

[40] Dario D. Salvucci. Modeling driver behavior in a cognitive architecture. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 48(2):362–380, 6 2006.

[41] Dario D Salvucci and Rob Gray. A two-point visual control model of steering. *Perception*, 33(10):1233–1248, 2004. publisher: SAGE Publications Sage UK: London, England.

[42] Dario D Salvucci and Niels A Taatgen. Threaded cognition: An integrated theory of concurrent multitasking. *Psychological review*, 115(1):101, 2008.

[43] Michael Schoelles and Wayne Gray. Simpilot: An exploration of modeling a highly interactive task with delayed feedback in a multitasking environment. *Proceedings of the 11th International Conference on Cognitive Modeling, ICCM 2012*, 1 2012.

[44] Michael J Schoelles and Wayne D Gray. Cognitive modeling as a tool for improving runway safety. In *16th International Symposium on Aviation Psychology*, page 541, 2011.

[45] Angelia Sebok, Christopher Wickens, and Robert Sargent. Using meta-analyses results and data gathering to support human performance model development. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 57(1):783–787, 9 2013.

[46] Jessica Zeitz Self, Radha Krishnan Vinayagam, James Thomas Fry, and Chris North. Bridging the gap between user intention and model parameters for human-in-the-loop data analytics. In *Proceedings of the Workshop on Human-In-the-Loop Data Analytics*, pages 1–6, 2016.

[47] Sterling Somers and R West. Steering control in a flight simulator using act-r. In *Proceedings of the 12th International Conference on Cognitive Modeling, eds RL West and TC Stewart (Ottawa: Carleton University)*, 2013.

[48] Arnold Tustin. The nature of the operator's response in manual control, and its implications for controller design. *Journal of the Institution of Electrical Engineers-Part IIA: Automatic Regulators and Servo Mechanisms*, 94(2):190–206, 1947. publisher: IET.

[49] Savita A Verma, Kevin Corker, and Amit Jadhav. An approach to modeling error in air-midas using contextual control model. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 47, pages 21–25. SAGE Publications Sage CA: Los Angeles, CA, 2003.

[50] Robert West and Sterling Somers. Scaling up from micro cognition to macro cognition: using sgoms to build macro cognitive models of sociotechnical work in act-r. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 33, 2011.

[51] Douglas A Wiegmann and Scott A Shappell. Human error analysis of commercial aviation accidents using the human factors analysis and classification system (hfacs). Technical report, United States. Office of Aviation Medicine, 2001.

[52] Rongbing Xu and Shi Cao. Modeling pilot flight performance in a cognitive architecture: Model demonstration. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 65, pages 1254–1258. SAGE Publications Sage CA: Los Angeles, CA, 2021.

[53] J. C. Zemla, V. Ustun, M. D. Byrne, A. Kirlik, K. Riddle, and A. L. Alexander. An act-r model of commercial jetliner taxiing. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 55, page 831–835, 9 2011.

APPENDICES

Appendix A

Cessna 172SP Checklist

C172SP CHECKLIST

FOR EXPERIMENT PURPOSE ONLY. DO NOT USE IN REAL LIFE!

STAGE 1

PRE-FLIGHT

Ignition Switch…………………………………..OFF

Avionics…………………………………………OFF

Master Switch…………………………….………ON

Fuel Level…………………………..……CHECKED

Flaps………………………………………………UP

BEFORE START/STARTING/AFTER START

Throttle………………………………..….Open ¼ IN

Mixture…………………………………….CUTOFF

Beacon Light……………………………..…….ON

Auxiliary Fuel Pump…………………..……….ON

Mixture…………………….…FULL 5 SEC/IDLE

Auxiliary Fuel Pump…………………………….OFF

Ignition Switch…………………..………..START

Mixture…………………………………ADVANCE

Oil Pressure……………………...………..CHECK

Avionics…………………………..……………ON

STAGE 2

TAKEOFF

Throttle…………………..…………………..FULL

Mixture…………………..…………………..RICH

Lift-off Speed……………..……..……55 KNOT

Initial Climb Speed………..…………70-80 KNOT

CLIMB

Power……………………………………..……FULL

Fuel Level/Temp………………..…….CHECKED

Engine
Instruments………………………………CHECKED

CRUISE

Power………………………………………ADJUST

Recommended: 75%

Appendix B

Post Situational Awareness Questionnaire

Post Situational Awareness Questionnaire (Pre-flight)

1. What was the initial position of fuel selector?

2. What was the fuel quantity (left tank and right tank)?

3. What was the fuel temperature and pressure?

4. What was the initial active NAV1 frequency?

5. What was the initial standby NAV1 frequency?

6. What was the initial active COM1 frequency?

7. What was the initial standby COM1 frequency?

8. What was the initial standby squawk code?

9. What was the initial altitude setting?

10. Is there any aircraft on your left/right side?

Post Situational Awareness Questionnaire (Take-off)

1. What was the fuel quantity (left tank and right tank)?

2. What was the engine RPM?

3. Were there any annunciators on the annunciator panel?

4. What was the active NAV1 frequency?

5. What was the active COM1 frequency?

6. What was the departure time?

7. What was climb speed?

8. When you received the ATC instruction, what was your altitude and heading?

9. What was your cruise speed?

10. What was the engine RPM when you cruised?

Appendix C

NASA Task Load Index Questionnaire

NASA Task Load Index Questionnaire

Mental Demand How mentally demanding was the task?

Very Low Very High

Physical Demand How physically demanding was the task?

Very Low Very High

Temporal Demand How hurried or rushed was the pace of the task?

Very Low Very High

Performance How successful were you in accomplishing what you were asked to do?

Perfect Failure

Effort How hard did you have to work to accomplish your level of performance?

Very Low Very High

Frustration How insecure, discouraged, irritated, stressed, and annoyed were you?

Very Low Very High