MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
NATIONAL AVIATION UNIVERSITY
Faculty of Aeronautics, Electronics and Telecommunications, Department of Aviation Computer-Integrated Complexes

ACCEPT TO PROTECTION
Head of Department
________Viktor SINEGLAZOV
“____”__________2024

QUALIFICATION PAPER
(EXPLANATORY NOTE)

HIGHER EDUCATION STUDY
“Bachelor”

Specialty 151 "Automation and computer-integrated technologies"
Educational and professional program "Information support and engineering of aviation computer systems"

Theme: Language command system in on-board control systems

Performer: student of the group KP-402Ba Viacheslav Didyk
Supervisor: Associate professor, Ihor Sergeyev

Norm controller: ____________ Filyashkin M.K.
(sign)

Kyiv 2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет аеронавігації, електроніки та телекомунікацій Кафедра авіаційних комп'ютерно-інтегрованих комплексів

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач випускової кафедри

______Віктор СИНЄГЛАЗОВ

“___” ____________ 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ

“Бакалавр”

Спеціальність 151 «Автоматизація та компю'терно-інтегровані технології» Освітньо-професійна програма «Інформаційне забезпечення та інженерія авіаційних комп'ютерних система»

Тема: Мовна командна система в бортових системах керування

Виконавець: студент групи КП-402Ба Дідик Вячеслав Андрійович

Керівник: доцент, Сергеєв Ігор Юрійович

Нормоконтроллер: ___________ Філяшкін М.К.

(підпис)

Київ 2024

NATIONAL AVIATION UNIVERSITY

Faculty of Aeronautics, Electronics and Telecommunications

Department of aviation computer-integrated systems

Educational degree: Bachelor

Specialty 174 "Automation, computer-integrated technologies and robotics"

Educational and professional program "Computer-integrated technological processes and production"

APPROVED

Head of department

_________ Sineglazov V.M

“____” ___________2024.

TASK

For the student’s thesis

by: Viacheslav Didyk

1. **Thesis topic** (project topic) “ Language command system in on-board control systems”
2. **Deadline for an execution of a project**: from 26.03.2024 to 25.05.2024
3. **Initial data for the project**: Improving the accuracy and speed of speech command recognition, enhancing interaction between the pilot and onboard control systems, and reducing the workload on the pilot.
4. **Contents for explanatory note:**

1. Relevance of speech command systems in onboard control systems; 2. Analysis of existing solutions to address the issue; 3. Proposals for improving speech recognition in aviation communication systems; 4. Testing the system in laboratory conditions.

5. **List of required graphic material**: 1. Hierarchical Grammar for GVSITE; 2. Percentage of Incorrect Recognitions per Display. 3. Commands Correctly Recognized per Display; 4. Incorrect Recognitions for the PFD.

6. **Calendar schedule-plan:**

№	Task	Execution term	Execution mark
1.	Getting the task	26.03.2024 - 27.03.2024	Done
2.	Formation of the purpose and main objectives of the study	27.03.2024 – 08.04.2024	Done
3.	Analysis of existing methods	09.04.2024 – 24.04.2024	Done
4.	Theoretical consideration of problem solving	25.04.2024 – 20.05.2024	Done
5.	Analysis of algorithm for designing an industrial automation system using CAD software	21.05.2024 – 25.05.2024	Done
6.	Preparation of an explanatory note	26.05.2024 – 29.05. 2024	Done
7.	Preparation of presentation and handouts	30.05.2024 – 02.06.2024	Done

7. **Task issue date:** 26 “March” 2024.

Supervisor: __________________________ Sergeyev I.Y

(sign)

Task is taken for completion by: ________________ Didyk V.A

(sign)

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет аеронавігації, електроніки та телекомунікацій

Кафедра авіаційних комп’ютерно-інтегрованих комплексів

Освітній ступінь: Бакалавр

Спеціальність 151 "Автоматизація та комп'ютерно-інтегровані технології"

ЗАТВЕРДЖУЮ

Завідувач кафедри

__________Віктор СИНЄГЛАЗОВ

“ ____ ” __________2024 р.

ЗАВДАННЯ

на виконання дипломної роботи студента

Дідика Вячеслава Андрійовича

1. **Тема роботи:** “Мовна командна система в бортових системах керування”
2. **Термін виконання проекту (роботи):** з 26.03.2024 р. до 25.05.2024р.
3. **Вихідні данні до проекту (роботи):** Підвищення точності та швидкості розпізнавання мовних команд, поліпшення взаємодії між пілотом і бортовими системами керування та зниження робочого навантаження на пілота.
4. **Зміст пояснювальної записки (перелік питань, що підлягають розробці):**

1. Актуальність мовної командної систем в бортових системах керування; 2. Аналіз існуючих рішень для вирішення проблеми; 3. Пропозиції покращення розпізнавання мови системою, в умовах авіаційоного зв’язку; 4. Перевірка системи в лабораторних умовах.

5. **Перелік обов’язкового графічного матеріалу:**

1. Ієрархічна граматика для GVSITE; 2. Графік відсотока неправильних розпізнань на дисплей; 3. Графік правильного розпізнавання команд на дисплеї; 4. Графік неправильного розпізнавання для PFD.

6. **Календарний план-граф**

№	Завдання	Термін виконання	Відмітка про виконання
1.	Отримання завдання	26.03.2024 - 27.03.2024	Виконано
2.	Формування мети та основних завдань дослідження	27.03.2024 – 08.04.2024	Виконано
3.	Аналіз існуючих методів	09.04.2024 – 24.04.2024	Виконано
4.	Теоретичний розгляд вирішення поставлених завдань	25.04.2024 – 20.05.2024	Виконано
5.	Аналіз алгоритму проектування систем промислової автоматизації з використанням необхідного програмного забезпечення	21.05.2024 – 25.05.2024	Виконано
6.	Оформлення пояснювальної записки	26.05.2024 – 29.05. 2024	Виконано
7.	Підготовка презентації та роздаткового матеріалу	30.05.2024 – 02.06.2024	Виконано

7. **Дата видачі завдання** ___ «26» березня 2024р.

Керівник: ________________________ Сергеєв І.Ю

(підпис)

Завдання прийняв до виконання: _______________ Дідик В.А

(підпис)

ABSTRACT

Explanatory note for the qualification work " Language command system in on-board control systems" 75 pages, 8 figures, 12 tables, 44 sources.

Keywords: VOICE COMMAND SYSTEM, ONBOARD CONTROL SYSTEMS, SPEECH RECOGNITION, AVIATION COMMUNICATIONS, INTELLIGENT SYSTEMS.

Object of study**:** the voice command system in onboard control systems.

Subject of study: the structure and algorithms of the voice command system.

Purpose of the qualification work: to develop a voice command system for onboard control systems and to implement the latest principles for building such systems using modern speech recognition methods.

Method of research**:** comparative analysis, processing of literary sources, digital modeling, and testing.

Theoretical research**:** development of the structure and algorithms of the voice command system for onboard control systems. The use of modern speech recognition methods is proposed to ensure high accuracy and speed of command execution. An approach to integrating voice commands into the overall aircraft control system is implemented.

Research results: the voice command system significantly improves the efficiency and convenience of managing onboard systems, reducing the pilot's workload. The proposed system ensures reliable command recognition even in challenging aviation environments.

The speech recognition system proposed in the work uses modern machine learning algorithms to adapt to the individual voice characteristics of the pilot, achieving high accuracy and reliability in control.

Recommendations: the results of the qualification work are recommended for use in the development of new and modernization of existing onboard control systems of aircraft, as well as in the training of pilots and specialists in aviation systems automation.

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи "Мовна командна система в бортових системах керування" 75 сторінок, 8 рисунків, 12 таблиць, 44 джерела.

Ключові слова: МОВНА КОМАНДНА СИСТЕМА, БОРТОВІ СИСТЕМИ КЕРУВАННЯ, РОЗПІЗНАВАННЯ МОВИ, АВІАЦІЙНИЙ ЗВ'ЯЗОК, ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ.

Об'єкт дослідження: мовна командна система в бортових системах керування.

Предмет дослідження: структура та алгоритми мовної командної системи.

Мета кваліфікаційної роботи: розробити мовну командну систему для бортових систем керування та впровадити новітні принципи побудови таких систем із використанням сучасних методів розпізнавання мовлення.

Методи дослідження: порівняльний аналіз, опрацювання літературних джерел, цифрове моделювання та тестування.

Теоретичне дослідження: розробка структури та алгоритмів мовної командної системи для бортових систем керування. Запропоновано використання сучасних методів розпізнавання мовлення для забезпечення високої точності та швидкості виконання команд. Реалізовано підхід до інтеграції голосових команд в загальну систему керування літаком.

Результати дослідження: мовна командна система значно покращує ефективність та зручність управління бортовими системами, зменшуючи навантаження на пілота. Запропонована система забезпечує надійне розпізнавання команд навіть у складних авіаційних умовах. Система розпізнавання мовлення, запропонована в роботі, використовує сучасні алгоритми машинного навчання для адаптації до індивідуальних голосових характеристик пілота, досягаючи високої точності та надійності в керуванні.

Рекомендації: результати кваліфікаційної роботи рекомендуються для використання при розробці нових та модернізації існуючих бортових систем керування літаків, а також у навчанні пілотів та спеціалістів з автоматизації авіаційних систем.

Table of contents

Introduction.. 11

1. Definition of the topic's relevance..11

2. Aim and objectives of the research...12

3. Object and subject of the research..13

4. Research methods..14

Chapter 1: Literature Review..16

1. History of language command systems in on-board control systems......................16

2. Modern approaches to implementing language systems in control.........................17

Chapter 2: Technologies and methods of language systems in on-board control systems..19

1. Analysis of speech recognition methods..19

2. Overview of existing language models for on-board control..................................20

3. Natural language processing technologies and their application in aviation systems..22

Chapter 3: Analysis of existing control systems with language interfaces..................24

1. Overview of existing automated control systems with language interfaces in aviation and other industries...24

2. Comparison of functional capabilities and characteristics of existing systems..25

3. Analysis of the advantages and disadvantages of different approaches to implementing a language interface in control systems..27

Chapter 4: GVSITE SRS Evaluation..30

1. Flight Test Aircraft..30

2.Evaluation Pilots...31

3. Speech Recognition System Design for GVSITE..31

4. Results..35

5. Summarising...39

6. EP Recommendations...40

Chapter 5: Laboratory SRS Experiment..41

1. Participants..41
2. Equipment...41
3. Method..41
4. Procedure..44
5. Results...44
5.1 Segment 1: Single Word Utterance...45
5.2 Segment 2: Short Phrase Utterance...45
5.3 Segment 3: ATC Long Phrase Utterance...47
6. Optimization...51
7. Summarising...53
Conclusions..54
References..56
Appendix..61

Introduction

1. Definition of the topic's relevance

The integration of language command systems within on-board control systems stands at the forefront of aviation innovation, representing a pivotal leap forward in aircraft technology. In an era characterized by ever-increasing automation and complexity in aviation operations, the implementation of advanced language interfaces holds immense promise for revolutionizing the way aircraft are controlled and managed. This introduction sets the stage for a comprehensive exploration of the multifaceted landscape surrounding language command systems, delving into their historical evolution, technological underpinnings, practical applications, and potential implications for the future of aviation.

The journey towards integrating language command systems into aircraft control spans decades, marked by significant milestones and breakthroughs in speech recognition, natural language processing, and human-machine interaction. Understanding the historical context provides crucial insights into the evolution of these systems, tracing their development from rudimentary prototypes to sophisticated, real-world applications in modern aviation.

At the heart of language command systems lie a myriad of cutting-edge technologies, including advanced speech recognition algorithms, machine learning models, and natural language understanding frameworks. This work delves into the intricate technical aspects behind the functioning of these systems, unraveling the complexities of speech processing, pattern recognition, and semantic analysis that enable seamless interaction between pilots and aircraft systems.

Beyond theoretical frameworks and technological prowess, the true value of language command systems lies in their practical applications within the aviation domain. From cockpit voice commands and flight management to aircraft diagnostics and maintenance procedures, these systems have the potential to revolutionize every aspect of aircraft operations. This work explores the diverse array of applications for language interfaces in aviation, showcasing their versatility, efficiency, and real-world impact.

As we stand on the precipice of a new era in aviation, characterized by unprecedented technological innovation and paradigm shifts in human-machine interaction, the implications of language command systems extend far beyond the confines of the present. This work delves into the broader implications and future prospects of these systems, pondering their potential to reshape the aviation landscape, enhance safety and efficiency, and redefine the boundaries of human-machine collaboration.

Through a comprehensive examination of these dimensions, this dissertation endeavors to provide a holistic understanding of language command systems in on-board control, shedding light on their historical evolution, technological intricacies, practical applications, and transformative potential. By navigating this intricate tapestry of concepts and insights, we embark on a journey towards unlocking the full potential of language interfaces in shaping the future of aviation.

2. Aim and objectives of the research

The aim of this research is to conduct a comprehensive investigation into the integration of language command systems within on-board control systems in the aviation industry. This encompasses a multifaceted exploration of the technological, operational, and human factors aspects associated with the development and implementation of language interfaces in aircraft control.

To achieve this aim, the following objectives have been outlined:

To review the historical evolution of language command systems in on-board control, tracing the development trajectory from early conceptualizations to contemporary applications.

To analyze the technological foundations underlying language command systems, including speech recognition algorithms, natural language processing techniques, and human-machine interaction paradigms.

To assess the practical applications of language command systems in aviation operations, encompassing cockpit voice commands, flight management functionalities, aircraft diagnostics, and maintenance procedures.

To evaluate the effectiveness and efficiency of language command systems in enhancing safety, productivity, and user experience within the aviation environment.

To explore the implications of language command systems for future developments in aviation technology, human factors considerations, and regulatory frameworks.

To provide recommendations for the design, implementation, and integration of language command systems into existing and future on-board control architectures, addressing technical challenges, operational requirements, and user needs.

By delineating these objectives, this research endeavors to offer a structured and comprehensive analysis of language command systems in on-board control, aiming to contribute valuable insights to the field of aviation technology and human factors engineering. Through rigorous investigation and critical inquiry, we aspire to illuminate the path towards harnessing the transformative potential of language interfaces for the advancement of aviation safety, efficiency, and innovation.

3. Object and subject of the research

The object of this research is the integration of language command systems within on-board control systems, specifically within the context of the aviation industry. This encompasses the technological infrastructure, operational procedures, and human-machine interaction dynamics involved in the implementation and utilization of language interfaces in aircraft control environments.

The subject of the research encompasses a comprehensive examination of language command systems, spanning their historical evolution, technological underpinnings, practical applications, and implications for the future of aviation. This includes but is not limited to:

Speech recognition algorithms and technologies utilized in language command systems.

Natural language processing techniques employed to interpret and respond to human commands.

Human factors considerations related to the design, usability, and acceptance of language interfaces in cockpit environments.

Operational integration of language command systems within existing aircraft control architectures, including flight management systems, avionics interfaces, and maintenance procedures.

Safety, efficiency, and user experience implications associated with the adoption of language interfaces in aviation operations.

Regulatory frameworks, standards, and guidelines governing the design, certification, and implementation of language command systems in aircraft.

By delving into the intricacies of these elements, this research seeks to provide a comprehensive understanding of language command systems within on-board control, shedding light on their potential benefits, challenges, and implications for the aviation industry. Through empirical analysis and critical inquiry, we aim to contribute valuable insights to the advancement of aviation technology and human factors engineering, paving the way for the safe, efficient, and user-friendly integration of language interfaces in aircraft control environments.

4. Research methods

This research employs a multifaceted approach encompassing qualitative and quantitative research methods to achieve its objectives effectively. The chosen methods are tailored to provide a comprehensive analysis of language command systems within on-board control systems, ensuring robustness and validity in the research findings.

Qualitative research methods, including literature review, case studies, and expert interviews, will be utilized to gain in-depth insights into the historical evolution, technological foundations, and practical applications of language command systems. A thorough examination of academic literature, industry reports, and relevant documentation will facilitate a comprehensive understanding of the subject matter, while case studies and expert interviews will offer valuable perspectives from practitioners and domain experts.

Quantitative research methods, such as surveys and empirical data analysis, will be employed to assess the effectiveness, efficiency, and user experience implications of language command systems in aviation operations. Surveys will be conducted to gather feedback from pilots, aviation professionals, and other stakeholders regarding their experiences and perceptions of language interfaces in on-board control. Empirical data analysis will involve the collection and analysis of relevant operational data to evaluate the impact of language command systems on safety, productivity, and operational efficiency.

Furthermore, a comparative analysis approach will be adopted to assess the advantages and disadvantages of different approaches to implementing language interfaces in aircraft control. This will involve the systematic comparison of various systems, technologies, and operational practices to identify best practices, challenges, and areas for improvement.

Overall, the combination of qualitative and quantitative research methods will provide a comprehensive and rigorous examination of language command systems within on-board control systems, yielding valuable insights into their design, implementation, and impact on aviation operations. Through a methodologically sound approach, this research aims to contribute to the advancement of knowledge in the field of aviation technology and human factors engineering.

Chapter 1

Literature Review

1.1. Historical Evolution of Language Command Systems in On-board Control

The inception of language command systems in on-board control dates back to the early stages of aviation history, marked by rudimentary attempts to integrate verbal instructions into aircraft operations. Over the decades, the evolution of these systems has been shaped by significant technological advancements, pioneering research, and practical applications in various domains. From the initial experiments with voice recognition in aircraft cockpits to the sophisticated language interfaces integrated into modern flight management systems, the journey of language command systems reflects a continuous quest for innovation and optimization in human-machine interaction.

The historical trajectory of language command systems encompasses key milestones, including the development of early voice recognition prototypes, the emergence of natural language processing techniques, and the integration of speech-based interfaces into critical aviation systems. Pioneering research efforts in the mid-20th century laid the foundation for subsequent advancements, paving the way for the adoption of voice-controlled navigation, communication, and control functionalities in commercial and military aircraft.

Throughout this evolutionary process, language command systems have undergone iterative refinement, driven by advancements in computing power, signal processing algorithms, and artificial intelligence. From rule-based systems relying on predefined commands to machine learning models capable of understanding and interpreting natural language inputs, the sophistication of language interfaces has grown exponentially, enabling more intuitive and user-friendly interactions between pilots and aircraft systems.

Furthermore, the historical evolution of language command systems has been shaped by a myriad of external factors, including regulatory frameworks, industry standards, and technological trends. The gradual shift towards digitalization, automation, and connectivity in aviation has accelerated the adoption of language interfaces, driving innovation and transformation across the industry.

In summary, the historical evolution of language command systems in on-board control represents a fascinating journey of technological innovation, scientific discovery, and practical implementation. By tracing this evolutionary path, we gain valuable insights into the origins, development, and future prospects of language interfaces in aviation, laying the groundwork for a comprehensive understanding of their role in modern aircraft operations.

1.2. Modern Approaches to Implementing Language Systems in Control

In the contemporary landscape of aviation technology, the implementation of language systems in on-board control has evolved to encompass a diverse array of approaches, methodologies, and technological paradigms. This section provides a comprehensive overview of the modern approaches and strategies employed in the design, development, and deployment of language command systems, shedding light on the latest trends, innovations, and best practices in the field.

At the forefront of modern approaches is the integration of advanced speech recognition algorithms, leveraging deep learning techniques, neural networks, and probabilistic models to achieve unprecedented levels of accuracy and robustness in recognizing spoken commands. These state-of-the-art algorithms enable real-time processing of natural language inputs, allowing for seamless interaction between pilots and aircraft systems without the need for cumbersome manual inputs or complex command structures.

Parallel to advancements in speech recognition, natural language processing (NLP) technologies have emerged as a cornerstone of modern language systems, facilitating the understanding, interpretation, and contextual analysis of human language inputs. Through the application of machine learning algorithms, semantic parsing techniques, and ontological frameworks, NLP enables language interfaces to decipher the intent behind spoken commands, infer user preferences, and adaptively respond to changing contexts and environments.

Moreover, modern approaches to implementing language systems in on-board control extend beyond technical considerations to encompass human factors engineering, user experience design, and ergonomic considerations. Human-centered design principles, usability testing methodologies, and cognitive psychology insights are integrated into the development process to ensure that language interfaces are intuitive, user-friendly, and conducive to safe and efficient operation in high-stakes aviation environments.

Furthermore, the advent of cloud computing, edge computing, and distributed processing architectures has revolutionized the scalability, flexibility, and accessibility of language systems in aviation. Cloud-based solutions offer seamless integration with existing avionics systems, enabling real-time data exchange, remote updates, and adaptive learning capabilities that enhance the performance and adaptability of language interfaces in diverse operational scenarios.

In summary, modern approaches to implementing language systems in on-board control represent a convergence of cutting-edge technologies, interdisciplinary insights, and user-centric design principles. By harnessing the power of advanced speech recognition, natural language processing, and human factors engineering, these approaches pave the way for a new era of intuitive, interactive, and intelligent aviation systems that redefine the boundaries of human-machine interaction in flight.

Chapter 2

Technologies and Methods of Language Systems in On-board Control

2.1. Analysis of Speech Recognition Methods

The analysis of speech recognition methods within the realm of on-board control systems represents a fundamental exploration into the diverse array of algorithms, techniques, and methodologies employed to decipher and interpret spoken commands. This section delves into the intricacies of speech recognition, shedding light on the underlying principles, challenges, and advancements that shape the landscape of this critical technology in aviation.

Speech recognition methods encompass a spectrum of approaches, ranging from traditional rule-based systems to modern deep learning architectures. Rule-based systems rely on predefined phonetic patterns, language models, and grammatical rules to match spoken utterances to predetermined commands, offering simplicity and transparency in algorithm design but often exhibiting limited flexibility and scalability in complex linguistic contexts.

In contrast, modern speech recognition methods leverage the power of machine learning, neural networks, and statistical modeling to achieve superior accuracy, robustness, and adaptability in real-world environments. Deep learning architectures, such as convolutional neural networks (CNNs), recurrent neural networks (RNNs), and transformer models, have revolutionized the field of speech recognition by enabling end-to-end training on large-scale datasets, thereby capturing complex patterns and nuances in human speech with unparalleled precision.

Furthermore, the integration of acoustic modeling, language modeling, and pronunciation modeling techniques plays a pivotal role in enhancing the performance of speech recognition systems. Acoustic modeling involves the representation of speech signals in the form of feature vectors, spectrograms, or mel-frequency cepstral coefficients (MFCCs), enabling the extraction of relevant acoustic features for subsequent analysis and classification.

Language modeling, on the other hand, focuses on capturing the statistical properties, syntactic structures, and semantic relationships inherent in natural language, thereby enabling the prediction of plausible word sequences and grammatical constructs from observed speech inputs. Techniques such as n-gram models, recurrent neural networks (RNNs), and transformer-based language models are commonly employed to facilitate language modeling in speech recognition systems.

Moreover, pronunciation modeling techniques aim to address variations in pronunciation, accent, and dialect among speakers, ensuring robustness and adaptability in speech recognition across diverse linguistic contexts. Phonetic dictionaries, pronunciation lexicons, and acoustic-phonetic alignment algorithms are utilized to map spoken utterances to canonical representations, facilitating accurate recognition and interpretation of spoken commands.

In summary, the analysis of speech recognition methods in on-board control systems encompasses a multifaceted exploration of algorithmic principles, computational techniques, and practical considerations that underpin the development and deployment of language interfaces in aviation. By examining the strengths, limitations, and emerging trends in speech recognition, this section provides valuable insights into the technological foundations of language systems and their role in shaping the future of aircraft control and navigation.

2.2. Overview of Existing Language Models for On-board Control

An overview of existing language models for on-board control systems offers a comprehensive examination of the diverse range of linguistic frameworks, computational architectures, and semantic representations utilized to facilitate natural language understanding and interaction within aircraft environments. This section embarks on a detailed exploration of the theoretical foundations, practical implementations, and emerging trends in language modeling, shedding light on the rich tapestry of approaches that underpin the design and development of language interfaces in aviation.

Language models serve as the backbone of language systems, providing the framework for understanding and interpreting human speech inputs in the context of aircraft control and navigation. Traditional language models, such as finite-state grammars, context-free grammars, and phrase-structure grammars, offer structured representations of linguistic knowledge, enabling rule-based parsing and interpretation of spoken commands.

In contrast, modern language models harness the power of statistical learning, machine learning, and deep neural networks to capture the intricate patterns, semantic relationships, and contextual nuances inherent in natural language. Probabilistic models, such as hidden Markov models (HMMs), conditional random fields (CRFs), and Gaussian mixture models (GMMs), are commonly employed to model the statistical properties of language and facilitate probabilistic inference in language understanding tasks.

Furthermore, the emergence of deep learning architectures has revolutionized the field of language modeling, enabling the development of neural network-based models capable of capturing complex linguistic structures and semantic representations. Recurrent neural networks (RNNs), long short-term memory networks (LSTMs), and transformer architectures have demonstrated remarkable success in language modeling tasks, offering superior performance in capturing long-range dependencies, hierarchical structures, and contextual embeddings in natural language.

Moreover, pre-trained language models, such as BERT (Bidirectional Encoder Representations from Transformers), GPT (Generative Pre-trained Transformer), and BERT-based variants, have emerged as powerful tools for language understanding and generation in diverse domains, including aviation. These models leverage large-scale corpora of text data to learn contextualized representations of words, phrases, and sentences, enabling robust and contextually aware interpretation of spoken commands within aircraft environments.

In summary, the overview of existing language models for on-board control systems provides a glimpse into the rich diversity of computational frameworks, algorithmic techniques, and theoretical paradigms that underpin the development and deployment of language interfaces in aviation. By exploring the strengths, limitations, and emerging trends in language modeling, this section lays the groundwork for a deeper understanding of the technological landscape and practical considerations surrounding language systems in aircraft control and navigation.

2.3. Technologies of Natural Language Processing and Their Application in Aviation Systems

The exploration of natural language processing (NLP) technologies and their application in aviation systems represents a pivotal aspect of language systems in on-board control. This section delves into the intricate methodologies, computational techniques, and practical implementations that underpin the processing, understanding, and generation of natural language inputs within aircraft environments, offering insights into the transformative potential of NLP in aviation.

Natural language processing encompasses a diverse array of techniques and methodologies aimed at enabling computers to understand, interpret, and generate human language in a manner that is contextually relevant and semantically meaningful. At the core of NLP lie fundamental tasks such as tokenization, part-of-speech tagging, syntactic parsing, semantic analysis, and discourse processing, each of which plays a crucial role in extracting meaning and intent from natural language inputs.

Tokenization involves segmenting text data into individual tokens or words, enabling subsequent analysis and processing at the lexical level. Part-of-speech tagging assigns grammatical categories (e.g., noun, verb, adjective) to each token, facilitating syntactic and semantic analysis by capturing the grammatical structure of sentences and phrases.

Syntactic parsing aims to analyze the grammatical structure of sentences and phrases, identifying relationships and dependencies between words and phrases to derive syntactic trees or parse structures that represent the underlying syntactic hierarchy of the text.

Semantic analysis focuses on extracting meaning and intent from natural language inputs, encompassing tasks such as named entity recognition, semantic role labeling, sentiment analysis, and semantic parsing. These tasks aim to capture the semantic content of text data, enabling computers to understand the intended meaning and context of user utterances within specific domains or applications.

Discourse processing involves analyzing the structure and coherence of larger units of text, such as paragraphs, documents, or conversations, to infer the underlying discourse relations, rhetorical structures, and pragmatic implications of the text.

In the context of aviation systems, NLP technologies find diverse applications across various domains, including cockpit voice commands, flight management systems, air traffic control communications, aircraft diagnostics, and maintenance procedures. These applications leverage NLP techniques to facilitate intuitive, efficient, and contextually aware interactions between pilots, air traffic controllers, and automated systems, thereby enhancing safety, productivity, and user experience in aviation operations.

Through the integration of advanced NLP technologies, aviation systems can interpret complex natural language inputs, adaptively respond to changing contexts and environments, and facilitate seamless communication and collaboration between human operators and automated systems. By harnessing the power of NLP, the aviation industry stands poised to unlock new frontiers in human-machine interaction, automation, and decision support, ushering in a new era of intelligent and user-centric aviation systems.

Chapter 3

Analysis of existing control systems with language interfaces

3.1. Overview of Existing Automated Control Systems with Language Interfaces in Aviation and Other Industries

The integration of language interfaces into automated control systems represents a watershed moment in human-computer interaction, transcending traditional input methods and revolutionizing the way users interact with technology. This section offers an expansive exploration of the dynamic landscape of existing automated control systems endowed with language interfaces, examining their multifaceted functionalities, broad-ranging applications, and profound ramifications across aviation and diverse industrial sectors.

Automated control systems enhanced with language interfaces harness cutting-edge natural language processing (NLP) technologies to facilitate seamless, intuitive, and hands-free communication between human operators and sophisticated automated platforms. In the aviation domain, these systems herald a new era of cockpit control, empowering pilots to execute commands, retrieve critical information, and navigate complex operational scenarios using natural language speech commands.

Pioneering advancements in voice recognition and semantic understanding have paved the way for the integration of language interfaces into flight management systems, navigation aids, and cockpit avionics, enabling pilots to interact with aircraft systems effortlessly and efficiently. By bridging the gap between human intent and machine execution, these systems mitigate cognitive workload, enhance situational awareness, and optimize operational performance during flight operations.

Beyond aviation, automated control systems with language interfaces have permeated various industries, including automotive, manufacturing, healthcare, and consumer electronics, catalyzing a paradigm shift in user interaction paradigms. In automotive applications, voice-controlled infotainment systems, virtual assistants, and driver assistance features empower motorists to access navigation, entertainment, and communication functionalities without compromising safety or diverting attention from the road.

Similarly, in manufacturing environments, voice-activated control systems streamline production workflows, expedite equipment setup and configuration, and foster greater operational agility and responsiveness. Moreover, in healthcare settings, voice-enabled medical devices, electronic health records systems, and virtual medical assistants empower clinicians to access patient information, record medical observations, and perform administrative tasks hands-free, optimizing clinical workflows and enhancing patient care delivery.

The ubiquitous proliferation of automated control systems with language interfaces underscores the inexorable march toward intuitive, user-centric human-machine interaction paradigms across diverse domains. By leveraging the synergistic fusion of natural language understanding, voice recognition, and cognitive computing technologies, these systems epitomize the convergence of human ingenuity and technological innovation, heralding a transformative era of automation, digitalization, and experiential augmentation.

In summation, the panoramic overview of existing automated control systems with language interfaces illuminates their unparalleled versatility, pervasive applicability, and transformative potential in reshaping the contours of human-computer interaction across aviation and an array of industries. As these technologies continue to mature and proliferate, they are poised to catalyze a seismic shift in the fabric of human-machine collaboration, driving unprecedented advancements in productivity, efficiency, and user experience on a global scale.

3.2. Comparison of Functional Capabilities and Characteristics of Existing Systems

The comparison of functional capabilities and characteristics of existing language command systems constitutes a pivotal aspect in evaluating their efficacy, usability, and adaptability across various operational domains. This subsection undertakes an exhaustive comparative analysis, delving into the intricate nuances of the functional attributes, performance metrics, and design paradigms of language command systems deployed across aviation and an array of industrial sectors, elucidating pivotal distinctions, convergences, and emergent trends that delineate the evolutionary trajectory of human-machine interaction paradigms.

In the realm of aviation, language command systems serve as linchpins of cockpit automation, facilitating seamless communication, task execution, and information retrieval for flight crews amidst the dynamic exigencies of flight operations. These systems embody a sophisticated ensemble of functionalities, encompassing voice-activated controls, natural language understanding, and context-aware processing capabilities, enabling pilots to issue commands, query system status, and navigate operational procedures with precision and efficiency. Furthermore, aviation-centric language command systems are engineered to conform to stringent safety, reliability, and regulatory standards, integrating fault-tolerant architectures, redundant fail-safes, and human factors considerations to ensure resilient performance in high-stakes aviation environments and mitigate the risk of catastrophic failures or human errors.

Conversely, in the automotive domain, language command systems herald a paradigm shift in vehicular interaction paradigms, empowering drivers with unprecedented levels of connectivity, convenience, and safety on the road. These systems boast an expansive repertoire of features, ranging from voice-controlled infotainment systems and navigation aids to driver assistance functionalities and vehicle diagnostics, augmenting driver situational awareness, entertainment options, and hands-free operational capabilities while minimizing distractions and cognitive load. Moreover, automotive language command systems leverage advanced machine learning algorithms, cloud-based processing architectures, and personalized user profiles to deliver tailored user experiences, anticipate driver intents, and optimize system performance across diverse driving scenarios and environmental conditions.

Within the manufacturing sector, language command systems emerge as pivotal enablers of smart factory initiatives, orchestrating production workflows, monitoring equipment status, and enhancing operational agility amidst the complexities of modern manufacturing environments. These systems embody a rich tapestry of functionalities, encompassing voice-activated equipment controls, predictive maintenance algorithms, and real-time production analytics, empowering operators to streamline operational processes, optimize resource allocation, and mitigate downtime risks. Furthermore, manufacturing-specific language command systems exhibit robust interoperability with industrial control systems, enterprise resource planning (ERP) platforms, and supply chain management (SCM) solutions, facilitating seamless data exchange and integration within the broader manufacturing ecosystem.

In healthcare settings, language command systems serve as catalysts for clinical innovation, streamlining administrative tasks, augmenting medical documentation, and enhancing patient care delivery processes. These systems offer a diverse array of functionalities, including voice-enabled medical dictation, EHR navigation, clinical decision support, and virtual medical assistant services, empowering healthcare professionals to optimize workflow efficiency, reduce documentation burdens, and focus on delivering high-quality patient care. Moreover, healthcare-specific language command systems adhere to stringent data privacy, security, and regulatory compliance standards, leveraging encryption protocols, access controls, and audit trail mechanisms to safeguard patient confidentiality and ensure HIPAA compliance.

In summation, the comparison of functional capabilities and characteristics of existing language command systems underscores their multifaceted utility, transformative potential, and cross-industry applicability in shaping the contours of human-machine interaction paradigms. By harnessing the synergistic convergence of advanced speech recognition, natural language processing, and machine learning technologies, these systems epitomize the vanguard of human-centric design, paving the way for a future where human-machine collaboration transcends boundaries and empowers individuals to realize their full potential across diverse operational domains.

3.3. Analysis of the Advantages and Disadvantages of Different Approaches to Implementing a Language Interface in Control Systems

The analysis of various approaches to implementing a language interface in control systems is instrumental in understanding the nuanced trade-offs, technical considerations, and usability implications inherent in the design and deployment of language command systems across diverse operational domains. This section embarks on a comprehensive exploration of the multifaceted landscape of language interface implementation methodologies, elucidating the inherent advantages and disadvantages of different approaches, ranging from rule-based systems and statistical models to deep learning architectures and hybrid approaches, in fostering intuitive, efficient, and reliable human-machine interaction paradigms.

Rule-based systems represent one of the foundational approaches to implementing language interfaces in control systems, relying on predefined grammatical rules, semantic parsers, and domain-specific vocabularies to interpret user input and execute corresponding actions. The advantages of rule-based systems lie in their transparency, interpretability, and ease of customization, enabling designers to fine-tune system behavior, handle edge cases, and accommodate domain-specific constraints with relative ease. However, rule-based systems are inherently limited by their rigidity, brittleness, and susceptibility to semantic ambiguity, necessitating extensive rule sets, manual intervention, and domain expertise to maintain and update over time.

Statistical models offer an alternative paradigm for language interface implementation, leveraging probabilistic models, machine learning algorithms, and large-scale corpora to infer user intent, disambiguate linguistic input, and generate contextually relevant responses. The advantages of statistical models lie in their adaptability, scalability, and robustness to linguistic variations, enabling systems to learn from data, generalize across diverse contexts, and evolve over time without explicit rule specification. However, statistical models are susceptible to data sparsity, overfitting, and generalization errors, particularly in low-resource domains or in the presence of noisy, unstructured input data, necessitating careful feature engineering, data preprocessing, and model validation strategies to mitigate performance degradation.

Deep learning architectures represent the forefront of language interface implementation, harnessing the power of neural networks, recurrent models, and attention mechanisms to learn hierarchical representations of language semantics, syntactic structures, and context dependencies directly from raw input data. The advantages of deep learning architectures lie in their ability to automatically extract high-level features, capture long-range dependencies, and adapt to complex, dynamic input patterns, enabling systems to achieve state-of-the-art performance in tasks such as speech recognition, natural language understanding, and dialog generation. However, deep learning architectures require large-scale annotated datasets, substantial computational resources, and domain-specific expertise for training, fine-tuning, and optimization, posing challenges in terms of data acquisition, model interpretability, and deployment scalability.

Hybrid approaches amalgamate the strengths of rule-based systems, statistical models, and deep learning architectures to capitalize on their complementary advantages and mitigate their respective limitations in language interface implementation. By leveraging rule-based heuristics for initial parsing and semantic annotation, statistical models for probabilistic inference and context modeling, and deep learning architectures for feature learning and pattern recognition, hybrid approaches offer a balanced compromise between interpretability, scalability, and performance in real-world applications. However, hybrid approaches entail additional complexity in system design, integration overhead, and model fusion challenges, necessitating careful architectural design, algorithmic selection, and performance optimization strategies to achieve optimal balance between flexibility, robustness, and computational efficiency.

In summation, the analysis of the advantages and disadvantages of different approaches to implementing a language interface in control systems underscores the nuanced interplay between design trade-offs, technological capabilities, and user experience considerations in shaping the efficacy, usability, and adoption of language command systems across diverse operational domains. By embracing a principled approach to system design, leveraging insights from linguistics, cognitive science, and machine learning, and fostering interdisciplinary collaboration, language command systems can realize their full potential as enablers of seamless, intuitive, and empowering human-machine interaction paradigms in the digital age.

Chapter 4
GVSITE SRS Evaluation

As we do not have the appropriate equipment and resources to conduct an experimental study, "Evaluation of Speech Recognition Systems for Aircraft Cockpit Voice Control" (Smith et al., 2018) article will be used as a frame in chapters 4-5.

A collaborative flight test evaluation was conducted by NASA Langley Research Center and an industry partner team as part of NASA's Aviation Safety and Security Synthetic Vision System project. The evaluation took place over a 3-week period at the Reno/Tahoe International Airport (NV) and an additional 3-week period at the Wallops Flight Facility (VA). Known as the Gulfstream-V Synthetic Vision Systems Integrated Technology Evaluation (GVSITE), this test aimed to assess integrated Synthetic Vision System (SVS) concepts crucial for the development and deployment of actual SV systems.

The SV systems evaluated during GVSITE included computer-generated terrain displayed on the Primary Flight Display (PFD), monochrome textured terrain presented on a Head-Up Display (HUD), and plan or perspective views of computer-generated terrain and obstacles on the Navigation Display (ND). Additionally, the integrated SV system incorporated data-link capabilities, sensors, and algorithms to provide and verify necessary information for display. It also featured symbology and algorithms designed to enhance pilot situational awareness during surface operations and to mitigate or alert to potential runway incursions.

This paper focuses specifically on assessing the in-flight performance of a Speech Recognition System (SRS) utilized as the pilot-vehicle interface for the integrated SV system display concepts.

4.1. Flight Test Aircraft

The flight test utilized a Gulfstream G-V aircraft, as shown in Figure 1. The Evaluation Pilot (EP) occupied the left seat, while a Gulfstream Safety Pilot occupied the right seat. The left seat setup included two research displays for assessing the PFD and ND concepts, an overhead HUD projection unit for evaluating head-up concepts, and an SRS system serving as the pilot-vehicle interface to the SV displays.

Fig. 1. G-V aircraft exterior and interior views.

4.2. Evaluation Pilots

Ten expert pilots (EPs) from airlines, a major transport aircraft manufacturer, the Federal Aviation Administration, and the Joint Aviation Authority participated in research flights, accumulating approximately 67 hours of flight testing. A total of 145 flight test runs were conducted to evaluate the NASA Synthetic Vision System (SVS) concepts near Wallops Island, VA (with 8 pilots) and Reno/Tahoe International Airport (with 7 pilots). Five of the ten pilots participated in tests at both locations.

4.3. Speech Recognition System Design for GVSITE

A Speech Recognition System (SRS) was integrated into the Gulfstream-V aircraft to enhance the pilot-vehicle interface with the Synthetic Vision System (SVS) displays without requiring hardware modifications. It served as a testing ground for future developments in commercial and business aircraft flight deck technologies. The system utilized a commercial speech recognition engine to interpret pilot's speech inputs, an interface application to exchange information with a computer via Ethernet, and a text-to-speech module to generate audible messages or play pre-recorded WAV files. The speech recognition technology employed was a readily adaptable, speaker-independent system with customizable grammar.

The bi-directional Speech Recognition System (SRS) allowed the Evaluation Pilots (EPs) to verbally command changes to the Synthetic Vision System (SVS) displays and receive aural warnings and alerts triggered by the SVS research systems. To enable this functionality, the EPs used noise-attenuating David Clark headsets plugged into a Telex ProCom/2 intercom box. This intercom box split the pilot's speech input to drive both the nominal G-V intercom input jacks and a specialized SRS function. The SRS function was implemented using a Microsoft Windows-based application running on a single computer. The intercom box's audio output was connected to the computer's audio-in port, allowing the computer to receive speech input. A "push-to-listen" function was incorporated into the system: when the EP depressed the yoke-mounted radio transmit rocker switch, a serial input closed on the SRS computer, initiating the "listening" process. Releasing the "push-to-listen" trigger signaled the SRS application to complete the speech recognition process. This implementation mimicked existing radio communications, making it intuitive and easy for the EPs to use as they interacted with the on-board speech-respondent "assistant."

Instead of implementing a natural language interface, the speech recognition system (SRS) utilized a hierarchical grammar structure to enhance recognition accuracy. The hierarchy consisted of three-word commands for controlling the Synthetic Vision System (SVS) displays (PFD, HUD, and ND) as shown in Figure 2, with the first word representing the display device, the second word indicating the function or element to be controlled, and the third word specifying the value or modifier. For instance, the command "NAV RANGE 5" adjusted the navigational display range to 5 nautical miles. Additionally, two "exceptions" were programmed: "cancel" to undo the previous command and "repeat" to reissue the last command.

Some words in the SRS grammar had alternative pronunciations, allowing users to choose between saying the word or spelling out each letter. For example, "HUD" could be pronounced as a word or spelled out as "H-U-D," and "NAV" could be said as a word or spelled out as "N-D." Similarly, "field-of-view" could be articulated as "FO-V."

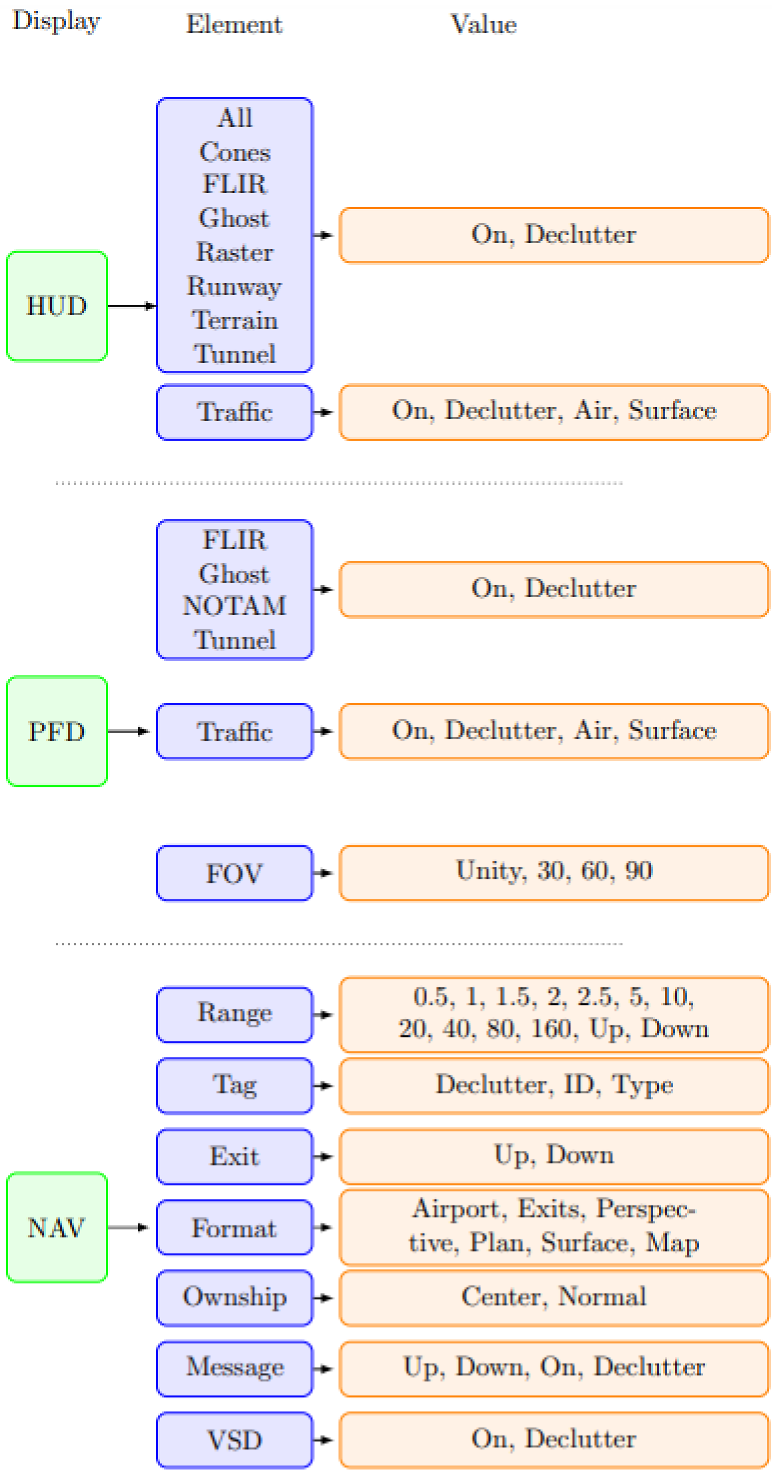

Fig. 2. Hierarchical Grammar for GVSITE. The 3 tier grammar structure: 1) Display device, 2) Display element and 3) State.

The SRS provided positive visual feedback during its operation. While the EP depressed the push-to-listen button and vocalized a command, the SRS interpreted it. Throughout the button's depression, a box featuring plus signs was displayed at the bottom of the PFD and HUD (Fig. 3). Should the SRS hold a minimum 40% confidence level in its interpretation, it transmitted the command to the displays. The interpreted command was then briefly showcased to the pilot for confirmation (Fig. 3). Conversely, if the SRS held less than a 40% confidence level, a box containing minus signs was briefly presented at the bottom of the PFD (Fig. 4). The 40% confidence threshold was established based on preliminary testing conducted prior to the evaluation flights.

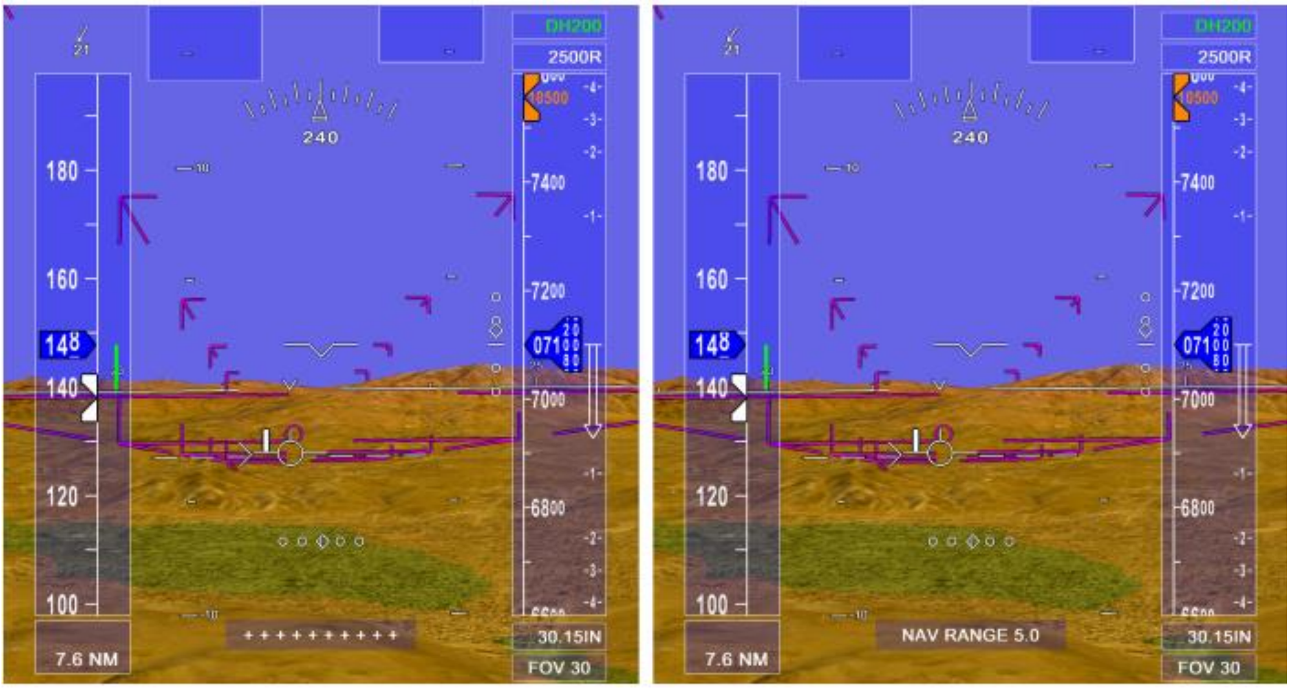

Fig. 3. The SRS box awaiting spoken command (left) and displaying the recognized command(right).

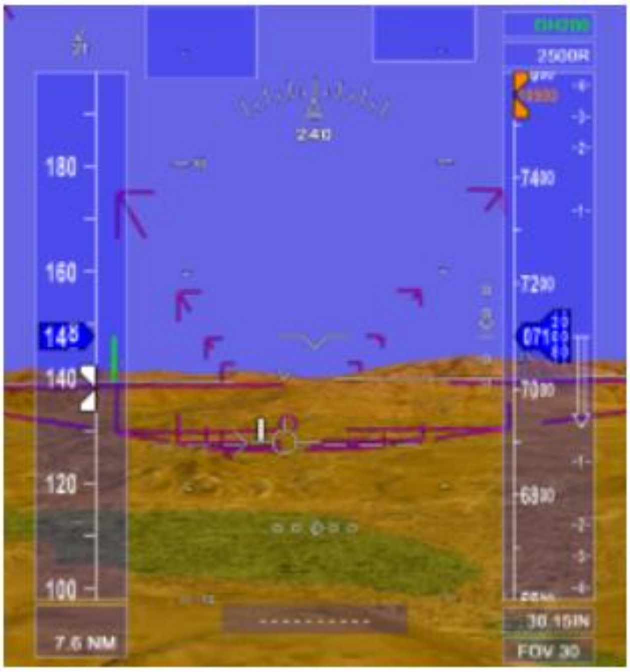

Fig. 4. The PFD display when the SRS was not confident in its interpretation.

The complete grammar setup for GVSITE is provided in Appendix A. Additionally, synonyms were permitted, such as using "NAV" interchangeably with "ND." Before the evaluation flights, it was noted that the phonetic similarity between "off" and "on" led to poor recognition rates when both were included in the grammar. Therefore, "declutter" was substituted for "off." In the context of aviation displays, "declutter" essentially means the same as "off" – to remove symbols from a display. Although "off" would have been preferred by the EPs as the natural opposite of "on," "declutter" was acceptable and became easy to remember and use after training.

4.4. Results

Throughout the entire flight test period, pilots issued a total of 505 verifiable SRS commands, achieving an overall success rate of 84%. This means there were 425 correctly recognized commands and 80 incorrectly recognized ones. Despite this, the SRS software's reported accuracy rate is 96%.

The distribution of commands pertaining to each display is presented in Table 1. The data indicates that commands for the PFD and ND were issued almost identically, at a rate four times greater than those for the HUD. It is important to note that the HUD had hardware controls for symbology and raster declutter, which were mounted on the EP's yoke. While the SRS commands for the HUD could adjust symbology groups, the hardware controls toggled the entire stroke or raster HUD components.

Table 1. SRS Commands per Display

Display	**Commands Spoken**	**Percentage**
PFD	222	44%
ND	227	45%
HUD	56	11%
All	**505**	**100%**

Each EP averaged 34 SRS commands per test flight, with a maximum of 64 and a minimum of 12. Figure 5 illustrates the total commands spoken during the flight test for each EP.

There were two types of incorrect recognitions by the SRS:

1) Instances where the SRS was not confident in matching to any command, resulting in rejection due to recognition falling below the threshold level of 40%. This occurred regardless of whether the utterance was correctly interpreted.

2) Cases where the SRS incorrectly interpreted a command, with recognition exceeding the threshold level of 40%, but the recognized utterance did not match the spoken command. For example, "PFD FOV 60" was interpreted as "NAV RANGE 60".

As depicted in Figure 6, the error rates varied considerably among EPs. Two EPs had error rates of 42% and 37%, while the others were closer to 10% error rates. Among the commands where the SRS exceeded the 40% threshold, the accuracy rate was 96%. This means that only 20 of the 80 incorrect recognitions were

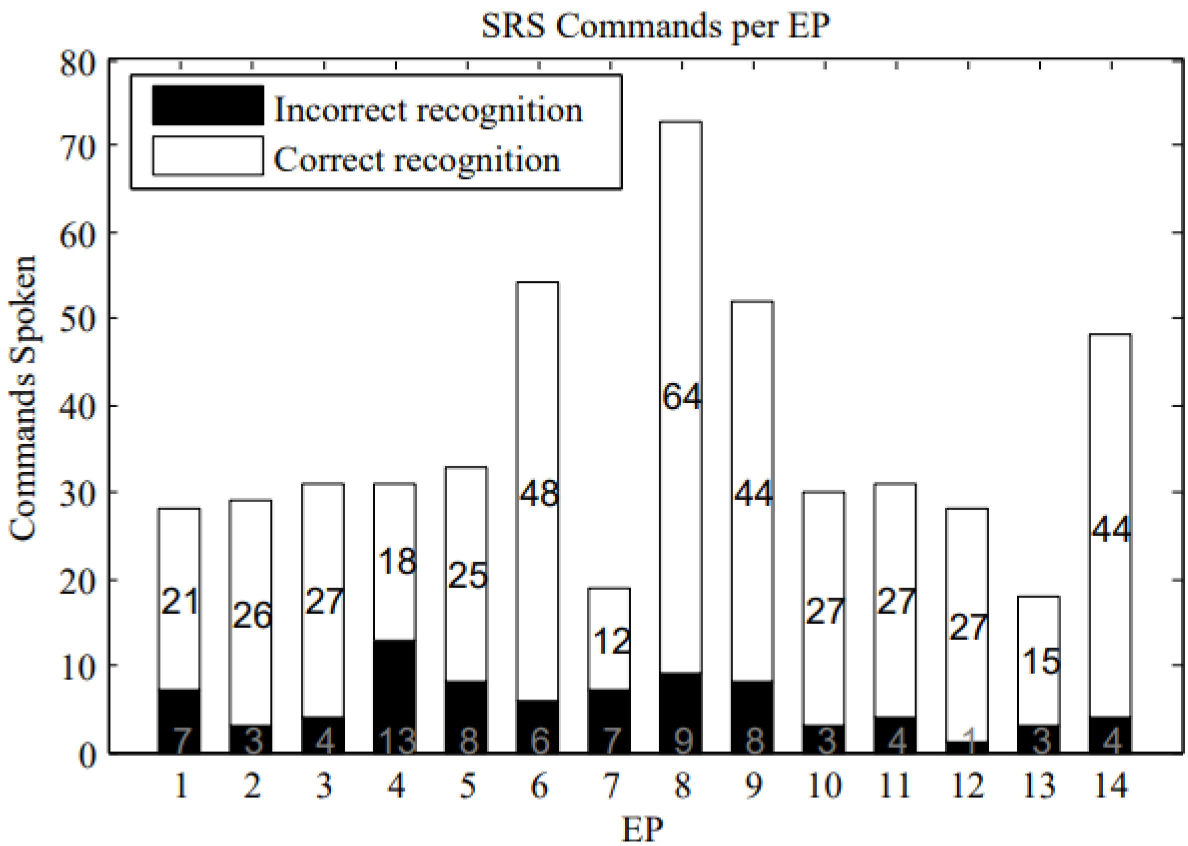

Fig. 5. Number of SRS commands spoken by each EP.

misinterpretations. The remaining 60 incorrect recognitions were confidence-related, indicating that the recognizer performance did not surpass the 40% threshold level, leading to no recognition action being taken (refer to Table 2).

Table 2. Incorrect SRS Commands.

	Incorrect Command	Percentage
Below 40% confidence	60	75%
Misinterpretation	20	25%
Total	**80**	**100%**

The errors were linked to the specific display to which the utterance was directed. These details are illustrated in Figures 6 to 8. Specifically, the PFD command comprised 67 out of the total 80 (84%) incorrect recognitions. Despite both the ND and PFD being addressed an equal number of times, the PFD accounted for a disproportionately high share of the errors.

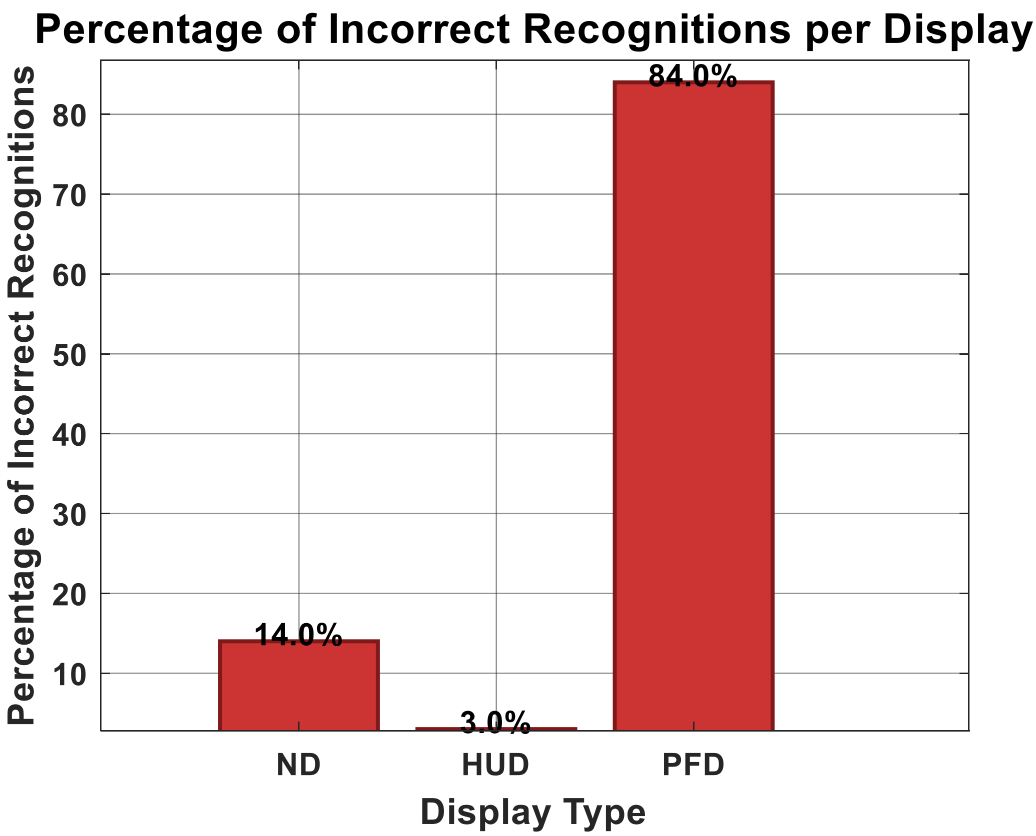

Fig. 6. Percentage of Incorrect Recognitions per Display.

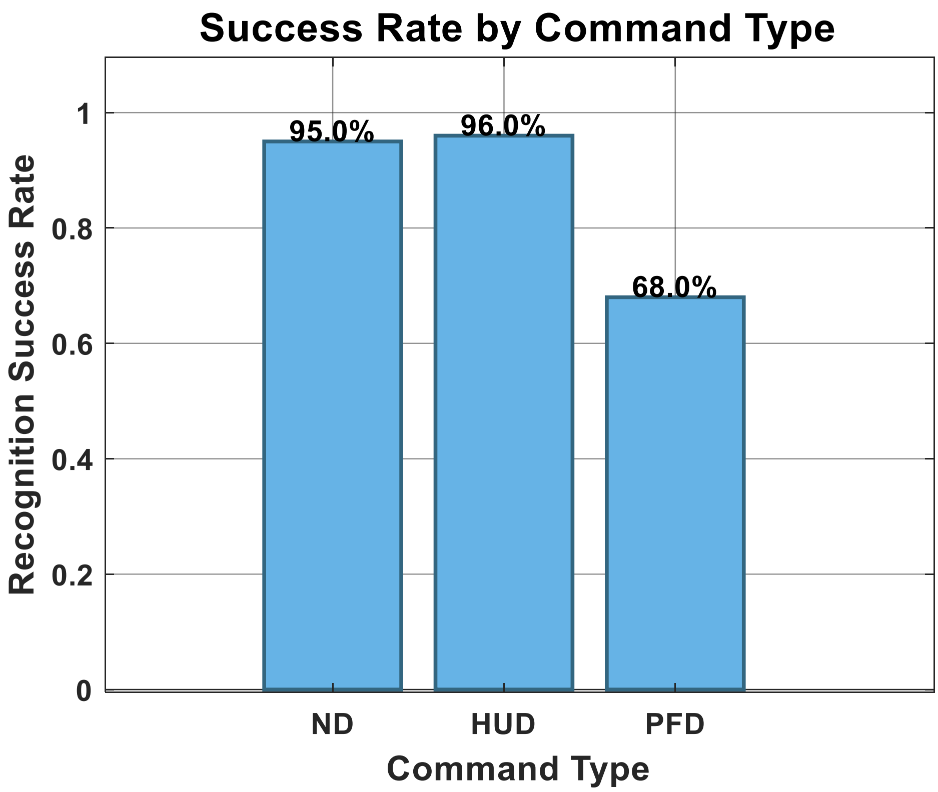

Fig. 7. Commands Correctly Recognized per Display.

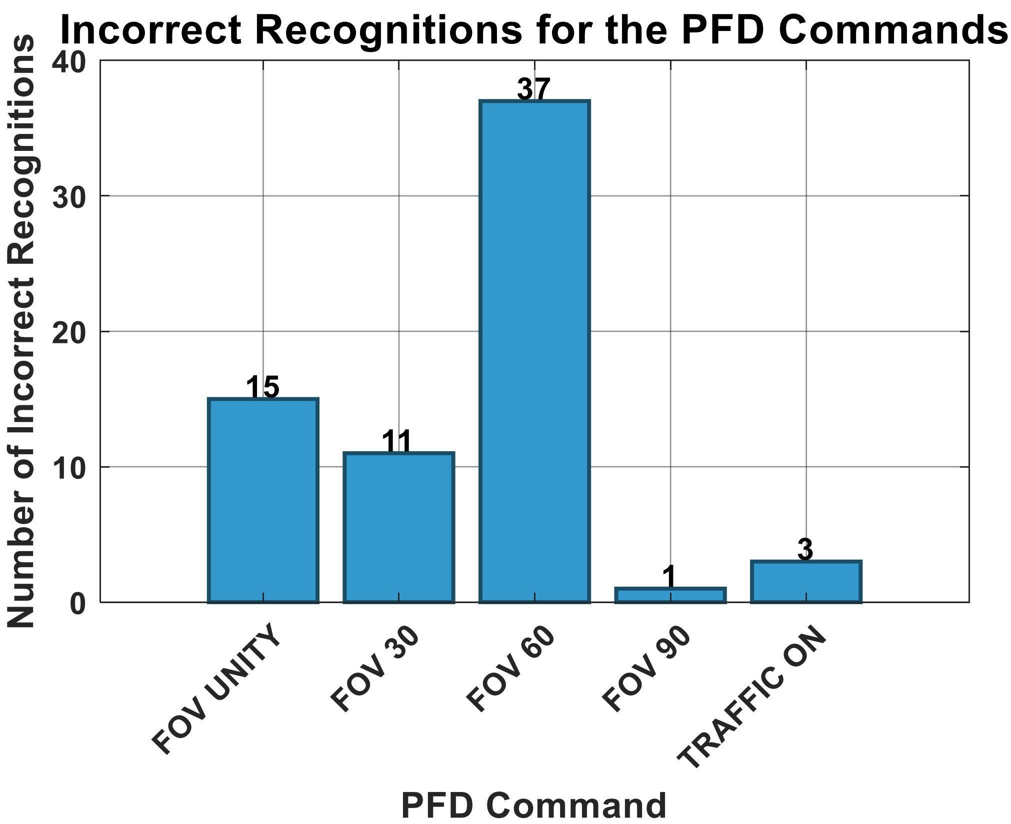

Fig. 8. Incorrect Recognitions for the PFD.

4.5. Summarising

Across all GVSITE data flights, the SRS achieved an overall success rate of 84%, correctly interpreting 84% of the 505 total SRS commands. Eighty commands were either recognized with insufficient confidence or incorrectly interpreted. Specifically, for 60 out of these 80 incorrect commands, the SRS had less than 40% confidence in its interpretation. The remaining 20 incorrect commands were misinterpretations, such as interpreting “NAV RANGE 5” as “NAV RANGE 20.”

When the success rate was broken down by display type, the ND commands had a 95% success rate, the HUD commands had a 96% success rate, and the PFD commands had a lower success rate of 68%. The SRS recognition engine’s practical success rate is known to be 96%, aligning with the success rates for the ND and HUD commands.

The incorrect recognitions for the PFD commands, detailed in Figure 8, revealed that the majority of errors were due to the “PFD FOV” commands. Most of the time, the SRS lacked confidence in interpreting these commands. If “PFD FOV” commands were excluded from the analysis, the overall success rate of the SRS would match the published accuracy rate of 96%.

The poor performance for PFD commands was primarily due to the noise-attenuating microphones and headsets used in the G-V aircraft. These microphones canceled out ambient cockpit noise when no voice input was detected, but caused a response lag, making the initial part of the pilot’s speech sound truncated. Consequently, commands like “HUD” and “NAV” became nearly phonetically equivalent to their truncated versions “UD” and “AV,” whereas “PFD” did not have a similar phonetic equivalent, making “FD” harder to recognize. This issue was more pronounced when pilots spoke quickly, complicating recognition. This hypothesis should be experimentally verified, and if confirmed, several changes could improve the system: a) tailoring the grammars to include truncated phonetic equivalents, b) modifying commands to compensate for this effect, or c) training pilots to utter a sound before pressing the push-to-listen button.

Additionally, another significant factor affecting SRS performance was the audio input volume and quality from the EPs. While audio input was checked during ground tests before each flight, there was no real-time monitoring of SRS volume or quality. This oversight should have been addressed. The audio volume is crucial for SRS performance, but it was an uncontrolled variable during the tests since EPs often adjusted their intercom volumes and microphone positions, affecting SRS performance. Occasionally, the audio input was too weak or overly saturated. To resolve this, real-time audio volume monitoring should be implemented, ideally with a series of lights indicating the audio input volume and its status (e.g., “high-medium-low”).

4.6. EP Recommendations

After analyzing the results of the SRS work and the pilots' feedback, I can make the following suggestions:

1. Introduce shortcuts for common commands. For instance, allow "RANGE 5" to replace "NAV RANGE 5" and "VIEW 30" for "PFD FOV 30." Although the hierarchical structure methodology was clear, "range" only applied to the Navigation Display and "view" to the PFD. Thus, including "NAV" and "PFD" was unnecessary for most EPs.

2. Adjust the cadence of certain commands to create a uniform cadence across all commands, if feasible. For example, change "PFD FOV" to a cadence similar to "NAV RANGE." This could involve changing the command "PFD" to a single word (possibly "Primary"), although a one-syllable word would be preferable.

3. Replace "FOV" with a single word (perhaps "View"). Using "Field-of-View" or even "F-O-V" was verbally cumbersome compared to a single word like "View."

4. Ensure consistency in using "UP/DOWN" and "INCREASE/DECREASE." The increase/up and decrease/down commands were not always intuitively obvious and were not consistently programmed to be synonymous.

5. Nearly all EPs desired a higher accuracy rate for the SRS, aiming for a 99+% recognition performance.

Chapter 5

Laboratory SRS Experiment

Following the GVSITE flight test, a laboratory study was conducted to establish baseline performance of the speaker-independent speech recognition technology. Participants were asked to speak words and phrases commonly used in standard aviation dialogue. The study was divided into three segments: single word utterances, short command phrases, and longer ATC clearance phrases. This study collected recognition data, assessed the basic accuracy of the recognition, and recorded a confidence factor in the recognition output by the SRS. Notably, the laboratory study employed a different SRS recognizer than the one used in the GVSITE flight test.

5.1. Participants

A total of 25 native US English-speaking individuals (18 males and 7 females) participated in the laboratory study. No additional information was collected about the participants. Each participant spent approximately 10 minutes completing the study.

5.2. Equipment

A laptop with standard microphone and earphone connections was used for the study. The headset employed was an Andrea ANC-700, featuring an active noise-canceling microphone optimized for speech recognition.

- Specifications of the Andrea ANC-700 Microphone:
- Noise Cancellation: 6 dB/octave
- Frequency Range: 100-10,000 Hz
- Impedance at 1 kHz (SoundBlaster Interface): 300 ohm
- Electrical Signal-to-Noise Ratio: 60 dB
- Sensitivity at 1 kHz (0 dB = 1 V/Pa) SoundBlaster Interface: -36 dB
- Current Consumption (SoundBlaster Interface): 0.500 mA

5.3. Method

Participants were instructed to read words displayed on a screen, divided into three segments of the study: single word phonetics, short commands, and ATC clearances. The utterances were assessed for accurate recognition and a confidence factor.

Each participant spoke a total of 71 utterances (Table 3), consisting of 26 single word utterances (e.g., the aviation phonetic alphabet from Alpha to Zulu) and 45 phrase utterances. The phrase utterances included 39 short phrases and 6 long phrases. The short command phrases were typical flight deck and display management commands used in the previous GVSITE flight test, such as “NAV RANGE 20.” The longer phrase utterances were taxi clearances, with the longest one comprising 14 words (19 syllables).

Table 3. All of the 71 utterances each participant spoke.

1	Alpha	27	NAV range 1	53	Checklist Takeoff
2	Bravo	28	NAV range 2	54	Checklist Climb
3	Charlie	29	NAV range 5	55	Checklist Cruise
4	Delta	30	NAV range 10	56	Checklist Descent
5	Echo	31	NAV range 20	57	Checklist Landing
6	Foxtrot	32	NAV range 50	58	Checklist After Landing
7	Golf	33	NAV range 100	59	Before Takeoff Checklist
8	Hotel	34	NAV range 200	60	Takeoff Checklist
9	India	35	NAV zoom out	61	Climb Checklist
10	Juliet	36	NAV zoom in	62	Cruise Checklist
11	Kilo	37	NAV range back	63	Descent Checklist
12	Lima	38	P F D Field of view unity	64	Landing Checklist
13	Mike	39	P F D Field of view 30	65	After Landing Checklist
14	November	40	P F D Field of view 60	66	NASA 557 Taxi To Runway 23 via D F T L
15	Oscar	41	P F D Field of view 90	67	United 231 Taxi At Concourse D via E B A
16	Papa	42	P F D F O V unity	68	NASA 557 Hold Short Of Runway 14 Rt at D
17	Quebec	43	P F D F O V 30	69	United 231 Taxi To Runway 14 Lt via T O B W
18	Romeo	44	P F D F O V 60	70	NASA 557 Hold At Gate K
19	Sierra	45	P F D F O V 90	71	United 231 Hold At Concourse J
20	Tango	46	P F D declutter		
21	Uniform	47	P F D traffic on		
22	Victor	48	P F D traffic off		
23	Whiskey	49	HUD declutter		
24	X-Ray	50	HUD traffic on		
25	Yankee	51	HUD traffic off		
26	Zulu	52	Checklist Before Takeoff		

In all three segments of the study, the independent variable was the utterance. The dependent variables were accuracy (correct or incorrect) and the confidence factor (ranging from 0 to 100). In the first segment, utterance numbers 1 through 26 were used, corresponding to the aviation phonetic alphabet from Alpha to Zulu. In the second segment, the utterances were approximately 3-5 words long, with utterance numbers 27 through 65 being used. Additionally, in this segment, two different command sets were compared and evaluated for accuracy. One command set began with "PFD Field-of-View view angle" and was compared to the set beginning with "PFD FOV view angle." Another comparison was made between commands starting with "Checklist checklist name" versus "checklist name Checklist." The third segment used utterance numbers 65 through 71, which were modeled after typical ATC ground control clearances.

5.4. Procedure

Each participant was equipped with a headset, and the microphone was adjusted to a distance proportional to their normal speaking volume. A volume level meter within the software ensured consistent microphone positioning and input levels. Additionally, the microphone height was set below the "Puff line" to minimize wind noise during the pronunciation of words containing the letter "P."

Participants were instructed to speak the designated word or phrase while the speech recognition software captured the audio and processed it using its recognition algorithms. The recognized utterance was then displayed to the participant, who indicated whether it was correctly recognized. This process continued until all utterances in each segment of the study were completed.

5.5. Results

The overall recognition rate for all 71 utterances by all 25 participants (1775 utterances) was 95.5% correct. Per participant, the median was 96%, the maximum was 100%, the minimum was 77% and the standard deviation was 5.52.

5.5.1. Single Word Utterance

The overall recognition rate for all aviation phonetic utterances across all 25 participants (totaling 650 utterances) was 94.8% accurate, as outlined in Table 4. Participant performance varied, with recognition rates ranging from 100% to 80% for phonetics such as ‘A’ and ‘P.’

Table 5 breaks down the recognizer’s confidence level (refer to Appendix E) for each aviation phonetic alphabet utterance (single words) across all 25 participants (650 utterances). The data indicates that the mean standard deviation for confidence was approximately 8.0, with the utterance “Tango” exhibiting the greatest variability (standard deviation).

Additionally, Table 6 provides the percentage of correct recognitions by participant for single-word utterances. Eight participants achieved a perfect recognition score, while one participant had a recognition performance of only 77%.

Table 4. Mean Percentage of Correct Recognition, All Participants (N=25).

Utterance	Mean %
Foxtrot, India, Juliet, Quebec, Sierra, Uniform, Victor, X-Ray, Yankee	100
Charlie, Delta, Echo, Hotel, Kilo, Lima, Mike, Oscar, Romeo, Whiskey	96
November, Tango	92
Bravo, Zulu	88
Golf	84
Alpha, Papa	80

5.5.2. Short Phrase Utterance

Within the second segment, two different command sets were evaluated to determine which set to use. The confidence level for these phrases is tabulated in Table 7, as well as the mean correct recognition rate. The “PFD Field of View number” versus “PFD FOV number” set both were recognized 100% of the time. Similarly, the “Checklist checklist name” versus “checklist name Checklist” command set was only different by 1%. Since the accuracy data revealed no clear advantage, the more natural speech data sets will be used; “PFD Field of View number”, and “checklist name Checklist”. Finally, the percentage correct by participant for the short phrase word utterances is given in Table 8. Fourteen participants obtained perfect recognition score. One participant only had 85% recognition performance, whereas they had 92% performance in the single word utterance test. The participant with the worst performance in the single word utterances, scored 95% in the short phrase utterances.

5.5.3. ATC Long Phrase Utterance

Segment 3 was included as an initial exploration into potential future studies focusing on SRS applications in cockpit interactions with ATC communications.

Table 9 provides the confidence levels associated with the long phrase utterances, while Table 10 displays the percentage of correct recognitions by participant for these lengthy phrases.

Analysis of the percentage of correct recognitions by participant reveals that 8 participants achieved perfect recognition scores. However, 4 participants exhibited a recognition performance of only 67%, resulting in a mean recognition rate of 86% across all participants. The majority of ATC phrase utterances were accurately recognized, with only one word being incorrectly interpreted. Specifically, the word 'Alpha' in the utterance "United 231 Taxi to Concourse Delta via Echo Bravo Alpha" was misinterpreted five times, reflecting a consistent error rate observed with 'Alpha' from the initial phonetic segment. Additionally, it was observed that short syllable words (such as 'at', 'and', 'to') were frequently omitted.

Table 11 presents a summary of correct recognitions across all segments.

Table 5. Segment 1: Confidence of Phonetic, All Participants (N=25).

Utterance	Confidence Mean	Median	Max	Min	SD	Recognized Mean Correct
Alpha	64.76	67	79	36	12.0	80
Bravo	68.56	71	80	45	8.4	88
Charlie	66.32	67	87	42	8.9	96
Delta	80.04	82	89	48	8.4	96
Echo	75.12	77	85	57	6.6	96
Foxtrot	66.96	69	80	48	8.0	100
Golf	65.76	69	80	47	9.3	84
Hotel	72.76	75	88	34	12.3	96
India	80.60	80	88	67	5.0	100
Juliet	73.20	74	85	56	6.4	100
Kilo	66.68	69	82	44	8.9	96
Lima	75.08	75	89	55	9.4	96
Mike	77.56	78	88	49	8.4	96
November	73.20	77	86	34	12.5	92
Oscar	70.20	72	80	45	7.5	96
Papa	66.44	69	77	50	7.2	80
Quebec	62.40	62	76	46	7.3	100
Romeo	73.32	75	87	45	9.8	96
Sierra	61.12	60	72	47	6.7	100
Tango	70.64	75	85	0	16.9	92
Uniform	68.72	68	83	51	7.2	100
Victor	71.04	70	85	49	8.3	100
Whiskey	79.44	79	88	71	3.5	96
X-Ray	72.56	73	87	54	8.4	100
Yankee	69.44	73	85	52	10.3	100
Zulu	60.36	63	72	31	9.2	88

Table 6. Segment 1: Phonetics Percent Correct per Participant Sorted by Incorrect Recognitions.

Participant	Correct	Incorrect	% Correct	Dev from avg	Misrecognized
1	19	7	76.9	-18.0	B, C, H, N, O, T, Y
2	22	4	84.6	-10.3	E, K, T, Z
3	23	3	88.5	-6.5	A, P, Z
4	24	2	92.3	-2.6	B, O
5	24	2	92.3	-2.6	A, G
6	24	2	92.3	-2.6	G, M
7	24	2	92.3	-2.6	P, R
8	24	2	92.3	-2.6	A, L
9	24	2	92.3	-2.6	A, P
10	25	1	96.2	1.2	A
11	25	1	96.2	1.2	D
12	25	1	96.2	1.2	G
13	25	1	96.2	1.2	P
14	25	1	96.2	1.2	P
15	25	1	96.2	1.2	B
16	25	1	96.2	1.2	G
17	25	1	96.2	1.2	Z
18	26	0	100.0	5.1	
19	26	0	100.0	5.1	
20	26	0	100.0	5.1	
21	26	0	100.0	5.1	
22	26	0	100.0	5.1	
23	26	0	100.0	5.1	
24	26	0	100.0	5.1	
25	26	0	100.0	5.1	
		Mean	**94.92**		
		Median	**96.15**		
		SD	**5.52**		
		Max	**100.00**		
		Min	**76.92**		

Table 7. Segment 2: Confidence of Command, All Participants (N=25).

Utterance	Confidence Mean	Median	Max	Min	SD	Recognized Mean Correct
NAV range 1	72.56	75	81	44	7.4838	96
NAV range 2	79.56	80	86	67	4.4355	100
NAV range 5	72.68	75	82	37	9.5904	96
NAV range 10	77.36	78	83	71	3.8824	100
NAV range 20	71.76	71	78	59	4.684	100
NAV range 50	72.76	73	84	66	4.6123	100
NAV range 100	68.68	69	79	60	4.58	100
NAV range 200	70.60	72	79	35	8.5147	96
NAV zoom out	70.44	73	78	35	8.3869	96
NAV zoom in	72.16	72	80	64	4.5797	96
NAV range back	66.44	73	84	0	19.929	84
P F D Field of view unity	70.40	71	79	57	5.7591	100
P F D Field of view 30	68.40	71	82	40	9.3986	100
P F D Field of view 60	67.76	70	81	36	9.2298	100
P F D Field of view 90	69.16	71	76	57	5.735	100
P F D F O V unity	71.96	72	81	63	4.8087	100
P F D F O V 30	73.92	74	85	60	5.7076	100
P F D F O V 60	72.76	73	82	43	7.5899	100
P F D F O V 90	72.84	73	82	57	5.5579	100
P F D declutter	66.76	69	78	37	7.7421	96
P F D traffic on	71.20	73	81	56	6.7144	100
P F D traffic off	69.28	70	80	56	6.4841	88
HUD declutter	63.96	64	74	56	4.8346	100
HUD traffic on	69.92	70	79	58	5.4077	100
HUD traffic off	67.52	69	78	52	6.7769	88
Checklist Before Takeoff	70.92	73	78	48	6.1841	100
Checklist Takeoff	69.72	72	78	54	7.3116	100
Checklist Climb	73.64	74	82	60	5.322	100
Checklist Cruise	73.08	75	79	53	6.4026	100
Checklist Descent	71.28	73	85	46	7.8396	96
Checklist Landing	69.84	73	81	0	15.184	96
Checklist After Landing	67.92	71	81	36	9.1511	96
Before Takeoff Checklist	72.56	74	82	42	8.1705	96
Takeoff Checklist	72.24	75	82	57	6.6538	96
Climb Checklist	72.08	72	79	64	3.9887	100
Cruise Checklist	67.72	68	78	57	5.712	96
Descent Checklist	71.60	73	82	56	6.1779	100
Landing Checklist	73.44	75	83	61	5.6648	92
After Landing Checklist	70.72	71	81	59	5.8489	100

Table 8. Segment 2: Commands, Percent Correct per Participant Sorted by Incorrect Recognitions.

Participant	Correct	Incorrect	% correct	Dev from avg
1	33	6	84.6	-12.9
2	35	4	89.7	-7.8
3	36	3	92.3	-5.2
4	36	3	92.3	-5.2
5	37	2	94.9	-2.7
6	38	1	97.4	-0.1
7	38	1	97.4	-0.1
8	38	1	97.4	-0.1
9	38	1	97.4	-0.1
10	38	1	97.4	-0.1
11	38	1	97.4	-0.1
12	39	0	100.0	2.5
13	39	0	100.0	2.5
14	39	0	100.0	2.5
15	39	0	100.0	2.5
16	39	0	100.0	2.5
17	39	0	100.0	2.5
18	39	0	100.0	2.5
19	39	0	100.0	2.5
20	39	0	100.0	2.5
21	39	0	100.0	2.5
22	39	0	100.0	2.5
23	39	0	100.0	2.5
24	39	0	100.0	2.5
25	39	0	100.0	2.5
		Mean	**97.54**	
		Median	**100.00**	
		SD	**3.95**	
		Max	**100.00**	
		Min	**84.62**	

Table 9. Segment 3: Confidence by ATC Phrase, All Participants (N=25).

Utterance	Mean	Median	Max	Min	SD
NASA 557 Taxi To Runway 23 via D F T L	100	73	80	65	4.32
United 231 Taxi to Concourse D via E B A	72	69	79	61	4.28
NASA 557 Hold Short Of Runway 14R at D	92	67	76	59	4.12
United 231 Taxi to Runway 14L via T O B W	92	72	80	61	4.57
NASA 557 Hold at Gate K	80	65	70	63	2.33
United 231 Hold at Concourse J	80	65	78	57	5.10

5.6. Optimization

An utterance was deemed correct if the participant confirmed the recognizer's guess as accurate. The recognizer utilized an internal algorithm to assess recognition correctness based on an "utterance score," which needed to meet or exceed the predetermined utterance threshold of 50. Alongside the participant's assessment, the recognizer's score, derived from the utterance score, was also recorded. This dataset underwent analysis to identify an optimal utterance score threshold that would enhance recognition rates using the recognizer's utterance score.

Out of all single-word phonetic utterances (totaling 650), there were 7 instances (1.1%) where the SRS marked a correct recognition despite the confidence threshold being below 50, resulting in a recorded incorrect recognition. Conversely, there were 15 occurrences (2.3%) where the utterance was actually incorrect but was incorrectly deemed correct by the SRS.

In digital avionics design, priorities often prioritize error detection over error correction. In essence, it is preferable to receive no data than to receive erroneous data. For example, the ARINC 429 digital data bus lacks error correction capability but transmits data (for error detection) to determine if a data packet was received accurately. Following a similar principle, SRS optimization may focus on achieving a lower false positive rate than an overall recognition rate.

Adjusting the threshold setting allows for some recognition optimization. A "false positive" occurs when the utterance score exceeds the threshold and is considered correct, despite being incorrect. To minimize false positive recognitions, the threshold could be increased. For instance, resetting the threshold to 52 reduces the false positive rate by 0.5%, albeit with a corresponding 0.6% decrease in the overall recognition rate (refer to Table 12). Conversely, resetting the threshold to 48 increases the correct recognition rate by 1.2%, but also raises the false positive rate by 0.6%. Depending on priorities, SRS optimization through threshold setting adjustment is feasible within a narrow range.

Table 10. Segment 3: ATC Phrase, Percent Correct per Participant sorted by Incorrect Recognitions.

Participant	Correct	Incorrect	% Correct	Dev from avg
1	4	2	66.7	-19.3
2	4	2	66.7	-19.3
3	4	2	66.7	-19.3
4	4	2	66.7	-19.3
5	5	1	83.3	-2.7
6	5	1	83.3	-2.7
7	5	1	83.3	-2.7
8	5	1	83.3	-2.7
9	5	1	83.3	-2.7
10	5	1	83.3	-2.7
11	5	1	83.3	-2.7
12	5	1	83.3	-2.7
13	5	1	83.3	-2.7
14	5	1	83.3	-2.7
15	5	1	83.3	-2.7
16	5	1	83.3	-2.7
17	5	1	83.3	-2.7
18	6	0	100.0	14.0
19	6	0	100.0	14.0
20	6	0	100.0	14.0
21	6	0	100.0	14.0
22	6	0	100.0	14.0
23	6	0	100.0	14.0
24	6	0	100.0	14.0
25	6	0	100.0	14.0
		Mean	**86.00**	
		Median	**83.33**	
		SD	**11.47**	
		Max	**100.00**	
		Min	**66.67**	

Table 11. Total Correct Recognition for All Participants (N=25).

Segment	Phonetic (26)	Short Phrase (39)	ATC Phrase (6)	Total (71)
% Correct	94.9	97.5	86.0	95.5

Table 12. Optimization analysis of Confidence Threshold setting

Confidence Threshold Setting	Marked Correct	Marked Incorrect	% Correct	Correct but marked incorrect	Incorrect but marked correct	Total
52	618	32	95.1	11 (1.7%)	12 (1.8%)	23
51	621	29	95.5	9 (1.4%)	13 (2.0%)	22
50	625	25	96.2	7 (1.1%)	15 (2.3%)	22
49	629	21	96.7	5 (0.7%)	17 (2.6%)	22
48	633	17	97.4	3 (0.5%)	19 (2.9%)	22
47	636	14	97.8	2 (0.3%)	21 (3.2%)	23

5.7. Summarising

The SRS engine successfully demonstrated its capability to recognize voice commands independently and process natural continuous speech. Across 1775 utterances spoken by 25 different participants, the SRS achieved a recognition rate exceeding 95%.

However, for in-flight applications, enhancing microphone quality and noise cancellation features are imperative to ensure optimal input audio signal quality. Discrepancies were observed between the laboratory test and the aircraft SRS performance, with the latter falling short of the expected recognition rate. Two significant disparities between the flight test and laboratory conditions were noted: firstly, the presence of higher ambient noise levels during the flight test compared to the controlled laboratory environment, and secondly, the absence of a volume display for the flight test setup. Addressing these limitations could potentially improve the recognition rate to a maximum of 96%. Nevertheless, pilots have emphasized the necessity for the SRS to achieve closer to a 99.99+% correct recognition rate.

Conclution

The data underscores the obstacles and complexities associated with developing a speech recognition system tailored for aviation, pinpointing specific challenges like the integration of the aviation phonetic alphabet.

The data indicates a pressing need for substantial research and development efforts. Generally, the recognition rate standards for commercial speech recognition systems fall significantly short of those necessary for aviation applications. Despite the increasing adoption of Speech Recognition Systems (SRSs) in consumer electronics like Siri™, Cortana™, and Amazon Echo™, aviation communication differs significantly from natural language, necessitating tailored solutions for this unique context. To enhance recognition rates, it's essential to implement structured or restricted grammars, hierarchical structures, speaker-dependent models, and context-specific adjustments in SRSs for aviation. For instance, real-time correlation of waypoint names and their pronunciation with aircraft position could improve recognition rates, considering that a pilot in Virginia might not reference waypoints in California.

The increasing necessity for speech recognition systems in aviation is becoming increasingly urgent. This demand is primarily motivated by the growing importance of enhanced data exchange among stakeholders in the National Air Space (NAS). This emphasis particularly concerns digital communication systems such as the Aircraft Communications Addressing and Reporting System (ACARS), controller-pilot data-link communication (CPDLC), and emerging operational frameworks known as "Net-Centric Operations." These frameworks enable new modes of operation by facilitating the exchange of status, intentions, and performance data among all users, fostering cooperative and coordinated flight operations.

In these operational scenarios, human oversight, awareness, and potential intervention remain essential, despite the increasing prevalence of machine-to-machine collaboration. As the volume of information exchanged grows, it poses a challenge for humans-in-the-loop, leading to information overload and clutter. To manage this data effectively, there is a growing need for Increasingly Autonomous Systems (IAS). These systems are designed to provide humans with relevant information such as traffic updates, intentions, and messaging, and enable interaction or intervention when necessary.

IAS operates autonomously, comprehending communications and extracting pertinent information, including path planning, intent, and state data from all aircraft within its range. It employs adaptive capabilities, learning from user input and contextual data through machine learning algorithms. A key aspect of IAS design is its human-centered approach, emphasizing bi-directional communication to ensure effective collaboration between humans and machines. Therefore, speech-based interfaces, including text-to-speech and speech-to-text capabilities, play a crucial role in facilitating natural interaction and creating an intuitive interface for IAS. Research indicates that natural, aural communication is essential for developing a user-friendly and low-workload IAS interface.

IAS are expanding their presence in various aviation applications beyond trajectory planning and execution. Technologies such as machine learning and cognitive computing, typified by IBM Watson, are increasingly recognized for their potential to enhance safety and performance within the aviation sector. However, a key technical challenge lies in developing these increasingly autonomous systems into intelligent machines. This requires leveraging machine learning algorithms while ensuring human involvement and interaction to optimize system performance beyond what either component could achieve individually. The collaboration between humans and autonomy is pivotal for the success of IAS, with speech serving as a natural and intuitive interface crucial for enabling autonomous systems. In future research, I propose to use integrated aviation-specific speech recognition systems with technologies such as IBM Watson to reduce the workloads of the commercial flight cockpit.

References

1. Joint Planning and Development Office. Analysis of the Next-Generation Air Transportation System Integrated Plan. Published by the U.S. Department of Transportation in December 2004.

2. “Evaluation of Speech Recognition Systems for Aircraft Cockpit Voice Control" (Smith et al., 2018)

3. Prinzo, O.V. Examination of Pilot-Controller Voice Communications during Approach Control. Technical Report DOT/FAA/AM-96/26, issued by the FAA Civil Aeromedical Institute in 1996.

4. Cardosi, K.M. Study on En Route Controller-Pilot Voice Communications. Technical Report DOT/FAA/RD - 93/11, published by the Department of Transportation, Federal Aviation Administration, in 1993.

5. Hutchins, E., Holder, B.E., and Perez, R.A. Research on Culture and Flight Deck Operations. Technical Report Sponsored Research Agreement 22-5003, conducted at the University of California San Diego, January 2002.

6. Federal Aviation Administration (FAA). Investigation into the Interfaces between Flight Crews and Modern Flight Deck Systems. Technical report issued by the FAA Human Factors Team in June 1996.

7. Soeters, J.L., & Boer, P.C. Examination of Culture and Flight Safety in Military Aviation. Published in the International Journal of Aviation Psychology, 10(2), 111–133, 2000.

8. Croft, J. Exploration of Opportunities for Expatriate Pilots in the Asia-Pacific Region. Published in Aviation Week & Space Technology, February 2015.

9. Orasanu, J., Davison, J., & Fischer, U. Addressing Culture and Language Barriers in Global Aviation Communication. Presented at the 9th International Symposium on Aviation Psychology, April 1997.

10. Flight Safety Foundation. Article on the Significance of Language Proficiency for Flight Safety. Published in Flight Safety Digest, Jan-Feb 2006.

11. Eurocontrol. Initial Communications Operating Concepts and Requirements for Future Radio Systems. Technical report prepared by

Eurocontrol/FAA Future Communications Study Operational Concepts and Requirements, January 2005.

12. Kerns, K. Review and Synthesis of Simulation Literature on Data-Link Communication between Controllers and Pilots. Published in the International Journal of Aviation Psychology, 1(3), 181–204, 1991.

13. Federal Aviation Administration (FAA). Report on the User Benefits of Two-Way Data Link ATC Communications. Technical Report FAA/CT-95-4, issued by the Federal Aviation Administration Data Link Benefits Study Team, February 1995.

14. Federal Aviation Administration (FAA). Study on the Benefits of Controller-Pilot Data Link ATC Communications in Terminal Airspace. Technical Report FAA/CT-96-3, conducted by the Federal Aviation Administration Data Link Benefits Study Team, September 1996.

15. Lozito, S.A., McGann, S.A., & Corker, K. Experimentation on Data Link Air Traffic Control and Flight Deck Environments: Insights into Flight Crew Performance. Presented at the 7th International Symposium on Aviation Psychology, April 1993.

16. Corwin, W.H., & McCauley, H. Considerations for Retrofitting Data Link Systems. Presented at the Aerospace Technology Conference and Exposition, October 1990.

17. Andre, A.D., Lins, J.M.C., & Wilson, J. Analysis of Conveying Message Criticality via Datalink. Presented at the Twelfth International Symposium on Aviation Psychology, 2003.

18. Talotta, N.J., & Shingledecker, C. Evaluation of Initial Data Link Terminal Air Traffic Control Services: Mini-Study 3. Technical Report DOT/FAA/CT-92/18, Volume I, issued by the U.S. Department of Transportation, Federal Aviation Administration, 1992.

19. Wickens, C.D., Miller, S., & Tham, M. Implications of Data Link for Representing Pilot Request Information on 2-D and 3-D Air Traffic Control Displays. Published in the International Journal of Industrial Ergonomics, 18, 283–293, 1996.

20. Waller, M.C., & Lohr, G.W. Simulation of Data Link ATC Message Exchange. Technical Report NASA TP-2859, prepared by NASA Langley Research Center, 1989.

21. Groce, J.L., & Boucek, G.P. Study on Air Transport Crew Tasking in an ATC Data Link Environment. Technical Paper 871764, published by SAE International, 1987.

22. Van Gent, R.N.H.W. Human Factors Issues with Airborne Data Link for En-Route and Terminal Flight Operations. Technical Report NLR TP 95666, conducted at the National Aerospace Laboratory, Amsterdam, NL, 1995.

23.Midkiff, A.H., & Hansman Jr., R.J. Identification of Important "Party Line" Information Elements and Implications for Situational Awareness in the Datalink Environment. Published in Air Traffic Control Quarterly, 1(1), 5–30, 1993.

24. Pritchett, A.R., & Hansman, R.J. Variations in "Party Line" Information Importance among Pilots. Presented at the 8th International Symposium on Aviation Psychology, April 1995.

25. Knox, C.E., & Scanlon, C.H. Flight Tests with Data Link for Air Traffic Control Information Exchange. Technical Report NASA Tech. Paper 3135, issued by NASA Langley Research Center, 1991.

26. Smith, N., Moses, J., Romahn, S., Polson, P., Brown, J., Dunbar, M., Palmer, E., & Lozito, S. Assessment of Flight Crew Experiences with FANS-1 ATC Data Link. Presented at the Tenth International Symposium on Aviation Psychology, 1999.

27. Hansen, J.H.L. Analysis and Compensation of Speech under Stress and Noise for Environmental Robustness in Speech Recognition. Published in Speech Communication, 20, 151–173, November 1996.

28. South, A.J. Characteristics of Speech Produced under High G-Force and Pressure Breathing. Presented at the IEEE International Conference on Acoustics, Speech, and Signal Processing, March 1999.

29. Werkowitz, E.B. Speech Recognition in the Tactical Environment: AFTI/F-16 Voice Command Flight Test. Presented at Speech Tech '84 Voice Input/Output Applications Show and Conference, New York, 1984.

30. Riley, V. Development of a Pilot-Centered Autoflight Interface. Presented at the 2000 World Aviation Conference, October 10 - 12, 1999.

31. Rankin, J., & Mattson, P. Controller Interface for Controller-Pilot Data Link Communications. Presented at the 16th DASC, October 1997.

32. Faerber, R.A., & Garloch, J.L. Usability Evaluation of Speech Synthesis and Recognition for Improving the Human Interface to Next Generation Data Link Communication Systems. Presented at the 19th Digital Avionics Systems Conference, October 2000.

33. Lechner, A., Mattson, P., & Ecker, K. Voice Recognition: Software Solutions in Real-Time ATC Workstations. Published in Aerospace and Electronic Systems Magazine, IEEE, 17, 11–16, November 2002.

34. Noyes, J.M., & Starr, A.F. Comparison of Speech Input and Touch Screen for Executing Checklists in an Avionics Application. Published in the International Journal of Aviation Psychology, 17, 299–315, 2007.

35. Damiani, S., Deregibus, E., & Andreone, L. Driver-Vehicle Interfaces and Interaction: Future Directions. Published in European Transport Research Review, 1(2), 87–96, 2009.

36. Zhang, H., & Ng, W.L. Speech Recognition Interface Design for In-Vehicle Systems. Presented at the 2nd International Conference on Automotive User Interfaces and Interactive Vehicular Applications, October 2010.

37. Draper, M., Calhoun, G., Ruff, H., Williamson, D., & Barry, T. Manual versus Speech Input for Unmanned Aerial Vehicle Control Station Operations. Presented at the Human Factors and Ergonomics Society Annual Meeting, October 2003.

38. Osgood, R.K. Jr. JSF Integrated Helmet Audio-Visual System Technology Demonstration Results. Presented at the Head-Mounted Displays II Conference, April 1997.

39. Nash, T. Eurofighter Typhoon: A Record of Continuing Achievements. Published in Air & Space Europe, 1(3), 40–45, 1999.

40. Reynolds, T.G., Hansman, R.J., Bolczak, R., & Tarakan, R. Improving Surveillance of Clearances in Future Air Traffic Control Systems. Technical Report AIAA 2004-6391, published by AIAA, September 2004.

41. McCann, R.S., Hooey, B.L., Parke, B., Foyle, D.C., Andre, A.D., & Kanki, B. Evaluation of the Taxiway Navigation and Situation Awareness (T-NASA) System in High-Fidelity Simulation. Published in SAE Transactions: Journal of Aerospace, 107, 1612–1625, 1998.

42. Nass, C., & Brave, S. Book titled "Wired for Speech: How Voice Activates and Advances the Human-Computer Relationship." Published by The MIT Press, 2005.

43. Kramer, L.J., Arthur III, J.J., Bailey, R.E., & Prinzel III, L.J. Flight Testing an Integrated Synthetic Vision System. Presented at the Enhanced and Synthetic Vision Proceedings of SPIE Conference, 2005.

44. Ferrucci, D.A. Introduction to "This is Watson." Published in the IBM Journal of Research and Development, 56(3.4), 1:1–1:15, 2012.

Appendix A

Available commands for the GVSITE flight test

The tables provided outline the commands accessible within the Speech Recognition System (SRS). Moreover, Evaluation Pilots (EPs) had access to shortcut commands, namely CANCEL and REPEAT. CANCEL was utilized to revoke the last command issued, while REPEAT was employed to replicate the previously spoken command.

Table A1. HUD Commands for GVSITE flight test.

Display	Attribute	State
HUD	ALL	DECLUTTER
HUD	ALL	ON
HUD	TUNNEL	DECLUTTER
HUD	TUNNEL	ON
HUD	TERRAIN	DECLUTTER
HUD	TERRAIN	ON
HUD	GHOST	DECLUTTER
HUD	GHOST	ON
HUD	TRAFFIC	DECLUTTER
HUD	TRAFFIC	ON
HUD	TRAFFIC	AIR
HUD	TRAFFIC	SURFACE
HUD	CONES	DECLUTTER
HUD	CONES	ON
HUD	FLIR	DECLUTTER
HUD	FLIR	ON
HUD	RASTER	DECLUTTER
HUD	RASTER	ON
HUD	RUNWAY	DECLUTTER
HUD	RUNWAY	ON
HUD	INSERT	

Table A2. PFD Commands for GVSITE flight test.

Display	Attribute	State
PFD	TUNNEL	DECLUTTER
PFD	TUNNEL	ON
PFD	GHOST	DECLUTTER
PFD	GHOST	ON
PFD	TRAFFIC	DECLUTTER
PFD	TRAFFIC	ON
PFD	TRAFFIC	AIR
PFD	TRAFFIC	SURFACE
PFD	FOV	UNITY
PFD	FOV	30
PFD	FOV	60
PFD	FOV	90
PFD	FLIR	DECLUTTER
PFD	FLIR	ON
PFD	NOTAM	DECLUTTER
PFD	NOTAM	ON
PFD	CHANNEL	DECLUTTER
PFD	CHANNEL	BOTTOM
PFD	CHANNEL	TOP

Table A3. NAV Commands for GVSITE flight test.

Display	Attribute	State
NAV	OWNSHIP	CENTER
NAV	OWNSHIP	NORMAL
NAV	TAG	DECLUTTER
NAV	TAG	ID
NAV	TAG	TYPE
NAV	RANGE	0.5
NAV	RANGE	1
NAV	RANGE	1.5
NAV	RANGE	2
NAV	RANGE	2.5
NAV	RANGE	5
NAV	RANGE	10
NAV	RANGE	20
NAV	RANGE	40
NAV	RANGE	60
NAV	RANGE	80
NAV	RANGE	160
NAV	RANGE	DOWN
NAV	RANGE	UP
NAV	MESSAGE	DECLUTTER
NAV	MESSAGE	ON
NAV	MESSAGE	DOWN
NAV	MESSAGE	UP
NAV	FORMAT	AIRPORT
NAV	FORMAT	PERSPECTIVE
NAV	FORMAT	MAP
NAV	FORMAT	SURFACE
NAV	FORMAT	PLAN
NAV	FORMAT	EXITS
NAV	FORMAT	ANIMATE
NAV	VSD	DECLUTTER
NAV	VSD	ON
NAV	EXIT	DOWN
NAV	EXIT	UP
NAV	CLEARANCE	
NAV	DIRECTOR	
NAV	MAP	
NAV	CONFORMAL	
NAV	ALIGNMENT	
NAV	BORE SIGHT	

Table A4. EFB Commands for GVSITE flight test.

Display	Attribute	State
EPAD	OWNSHIP	CENTER
EPAD	OWNSHIP	NORMAL
EPAD	TAG	DECLUTTER
EPAD	TAG	ID
EPAD	TAG	TYPE
EPAD	RANGE	0.5
EPAD	RANGE	1
EPAD	RANGE	1.5
EPAD	RANGE	2
EPAD	RANGE	2.5
EPAD	RANGE	5
EPAD	RANGE	DOWN
EPAD	RANGE	UP
EPAD	MESSAGE	DECLUTTER
EPAD	MESSAGE	ON
EPAD	MESSAGE	DOWN
EPAD	MESSAGE	UP
EPAD	FORMAT	AIRPORT
EPAD	FORMAT	SURFACE
EPAD	FORMAT	EXITS
EPAD	EXIT	DOWN
EPAD	EXIT	UP

Appendix B

Custom Speech Application Software

The evaluation software was created within the Microsoft Visual Studio 2005 development environment, utilizing the C# programming language and the Fonix C# Application Programmers Interface (API). Around 2,000 lines of code were dedicated to this project. Below are screenshots depicting the three segments utilized in the study.

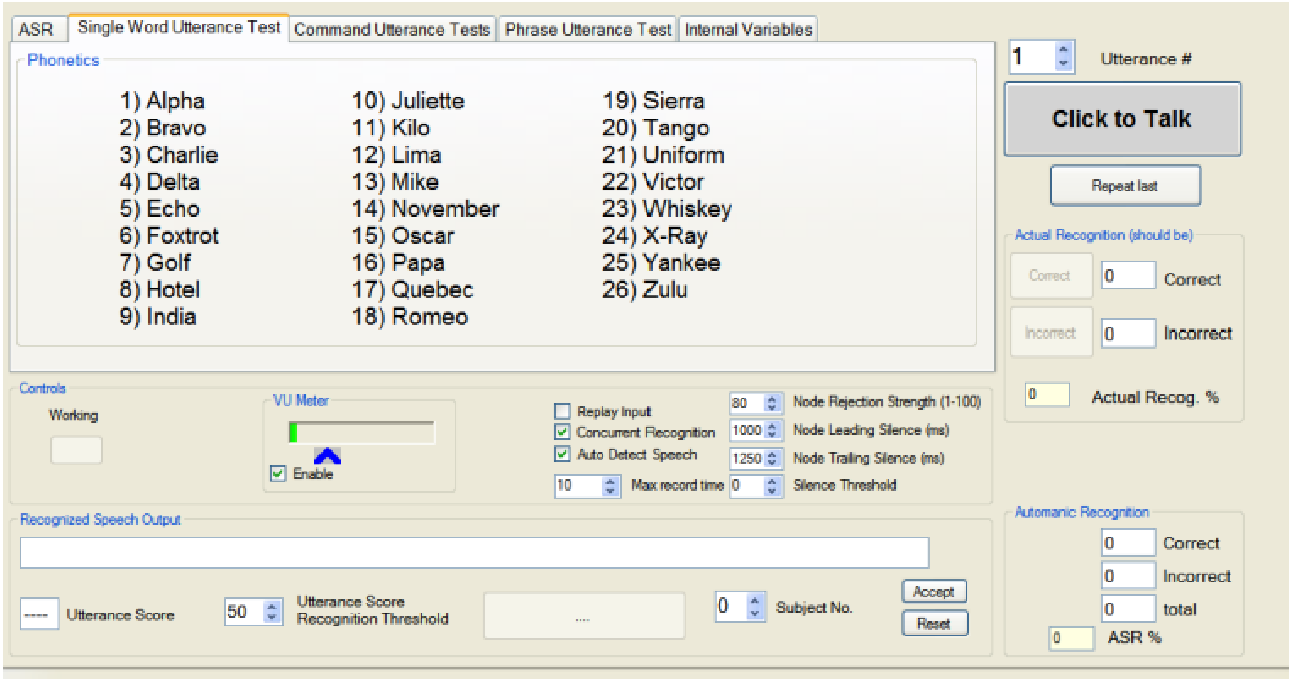

Figure B1. Single word utterance segment screen image.

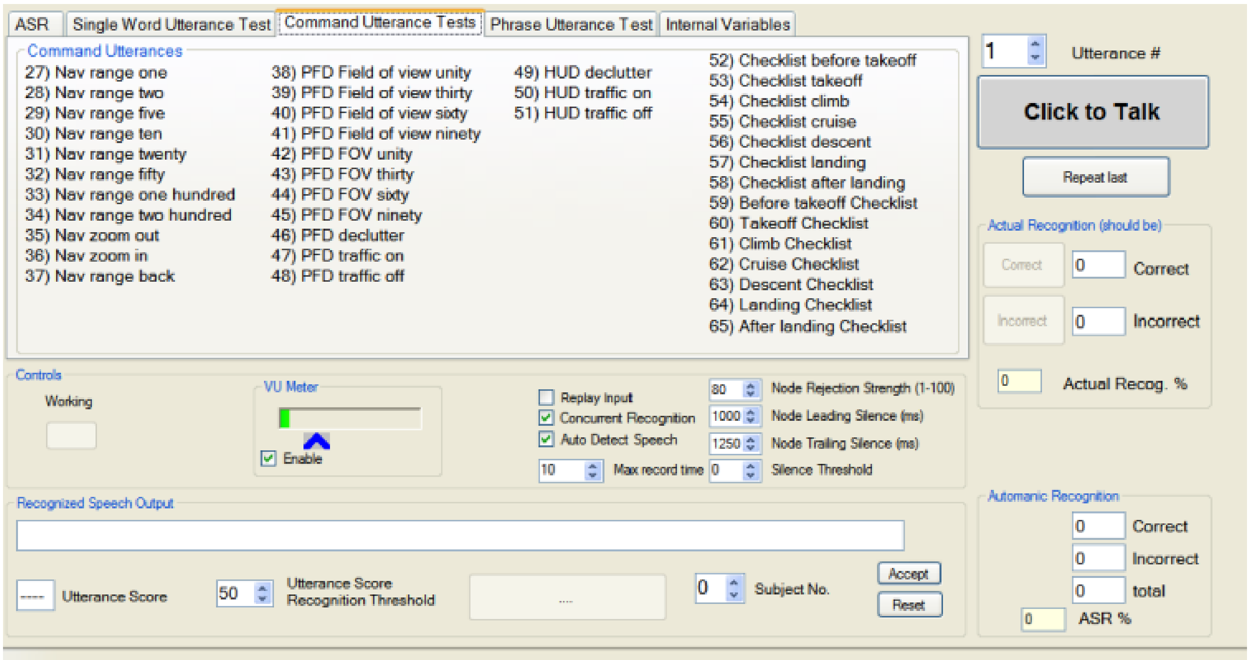

Figure B2. Short (command) phrase utterance segment screen image.

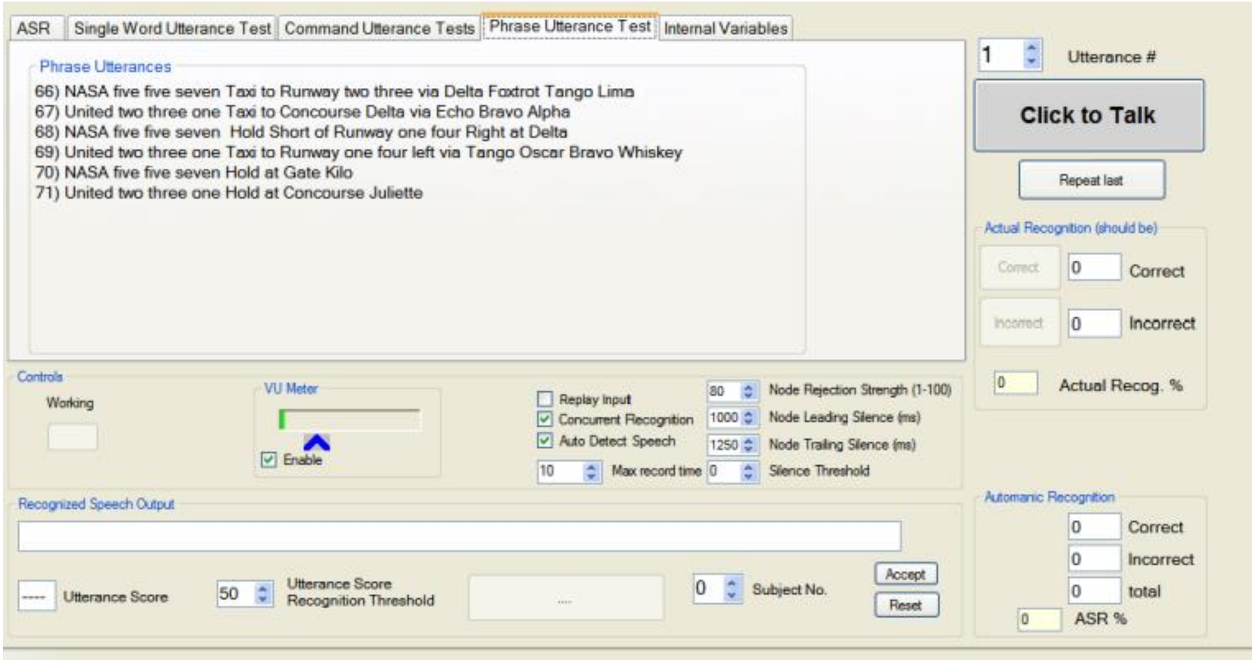

Figure B3. Long phrase utterance segment screen image.

Appendix C

Speaker-independent engine

A speech application is structured into nodes, each representing the vocabulary and other recognition configurations utilized during speech recognition. The Software Development Kit (SDK) facilitates word-spotting and grammar nodes, eliminating the need for training. The diagram below illustrates a speech utterance and outlines the audio attributes associated with each node. These attributes dictate how the speech detector frames the utterance before transmitting it to the recognizer.

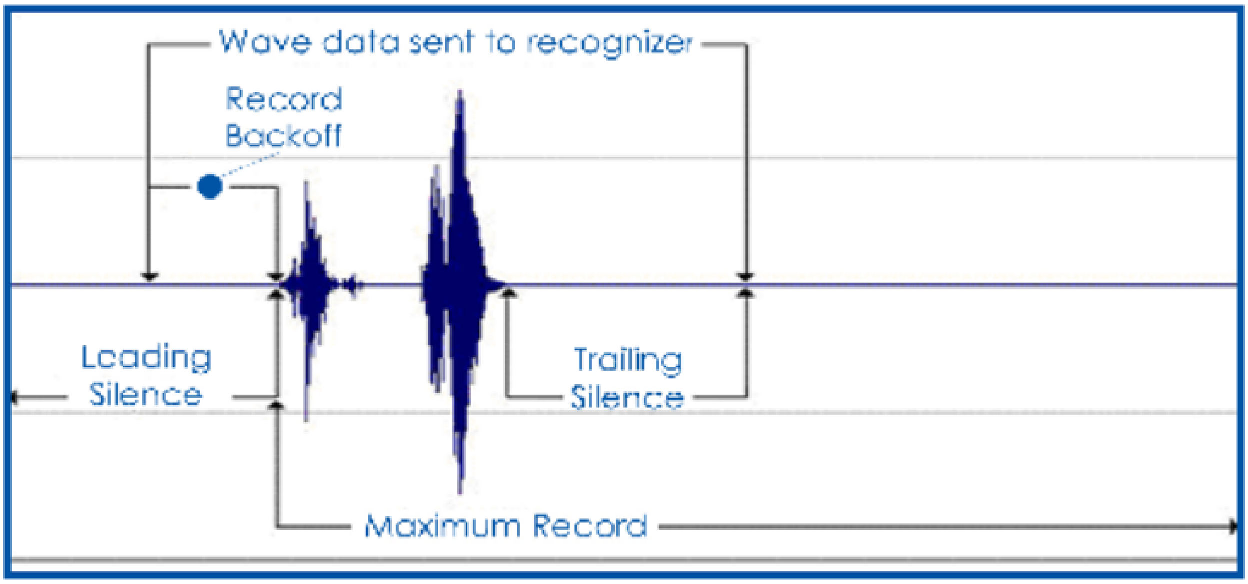

Figure C1. Speech waveform and attributes.

The speaker-independent SRS engine undertakes the following operations:

1. Audio collection: Raw audio data is gathered from an input source, such as a microphone, and forwarded to the Audio Processing component.

2. Audio processing: The audio input is segmented into "frames" using predetermined parameters, ensuring that only relevant audio data for recognition is retained. This processed data is then directed to the Feature Extraction component.

3. Feature extraction: Frequency components are extracted from the processed audio data at intervals of 10 milliseconds. These frequency components are then passed on to the Neural Networks component.

4. Neural networks: Phoneme probability estimates are derived from the frequency components by the neural networks. These estimates are subsequently transmitted to the

Continuous Word Decoder. Neural networks play a pivotal role in the speech recognition technology.

5. Continuous word decoder: The collection of phoneme probabilities is compared against the dictionary, resulting in a list of word probabilities arranged in descending order of likelihood.

Appendix D

Software Implementation

Each segment of the study had its own configuration for grammar structure and candidate word dictionary. The Application Programmers Interface (API) employed a straightforward script language to define the grammar structure. In this language, a "vertical line" denotes a logical OR, a "space" signifies a logical AND, square brackets "[ ]" indicate optional elements, and parentheses "( )" are used for grouping.

In the first segment, the dictionary consisted solely of the 26 phonetics.

$phonetics = (Alpha%A | Bravo%B | Charlie%C | Delta%D | Echo%E | Foxtrot%F | Golf%G | Hotel%H | India%I | Juliette%J | Kilo%K | Lima%L | Mike%M | November%N | Oscar%O | Papa%P | Quebec%Q | Romeo%R | Sierria%S | Tango%T | Uniform%U | Victor%V | Wiskey%W | X-Ray%X | Yankee%Y | Zulu%Z);

$grammar = $phonetics;

In the second segment, which focused on short command utterances, the structure was as follows:

```
$navcommand = NAV (declutter |
(zoom (in | out)) |
 ((range ( back | one%1 | two%2 | five%5| ten%10 |
twenty%20 | fifty%50 | one-hundred%100 |
two-hundred%200)))) ;
$pfdcommand = PFD (declutter |
(traffic (on | off)) |
(((Field of view) | (F O V))
(unity | thirty%30 | sixty%60 | ninety%90)));
$hudcommand = HUD (declutter | (traffic (on | off)));
$chklstcommand = [Checklist] ((Before Takeoff) |
Takeoff | Climb | Cruise | Descent |
Landing | (After Landing)) [Checklist];
$grammar = $pfdcommand | $navcommand | $hudcommand | $chklstcommand;
```

The third segment, (long phrase utterance) contained the following structure:

$command = (Hold | (Hold at) | (Hold Short) | (Hold Short Of) | Taxi);

$modifiers = (To|At);

$dest = (Ramp | Gate | Concourse | (Runway One%1 Four%4 Left%Lt)| (Runway One%1 Four%4 Right%Rt)| (Runway two%2 three%3) | (Runway one%1 six%6));

$modifiers2 = (via|at);

$grammar = $callsign $command [$modifiers] $dest [$phonetics] [$modifiers2] {$phonetics};

Appendix E
Software controls definitions

The Utterance Score is an integer ranging from 0 to 100, provided by the SRS algorithm, indicating the confidence level in recognizing the last spoken utterance.

The Utterance Score Recognition Threshold is also an integer ranging from 0 to 100, used to compare with the Utterance Score. If the Utterance Score exceeds or equals this threshold, the utterance is considered recognized. Throughout the study, this threshold was consistently set to 50.

Node Rejection Strength, another integer ranging from 0 to 100, is a setting in the SRS algorithm determining the threshold for recognizing or rejecting out-of-vocabulary words. Increasing this value makes word recognition more stringent, resulting in more rejections. Conversely, decreasing this value makes word recognition less strict but increases the chance of accepting out-of-vocabulary words. For this study, the value was set to 80.

Node Leading Silence is an integer setting ranging from -1 to 10000 milliseconds used in the SRS algorithm. It measures the duration of silence between the start of recording and the detection of speech. If no speech is detected within the leading silence time, recording stops.

Node Tailing Silence is also an integer setting (-1 to 2000 milliseconds) used in the SRS algorithm to determine the maximum length of silence the speech detector waits for before recognizing the end of speech. This allows for natural pauses in speech. Setting a lower value results in faster recognition results. Trailing silence begins after speech detection stops, and recording stops when the trailing silence time is reached. In this study, the value was consistently set to 1250 milliseconds.

Maximum Record Time is an integer setting (0 to 120 seconds) that defines the maximum recording time after speech detection. For this study, the maximum record time was set to 10 seconds.

Silence Threshold is an integer setting (0 to 500) designed for high noise environments where speech may be detected prematurely. However, this threshold is only applied if recognition is lower than expected. The program dynamically adjusts the silence threshold.

Auto Detect Speech is a discrete setting that toggles speech detection on or off during audio collection. When auto speech detect is enabled, only detected speech (including trailing silence) is sent to the recognizer. This setting was enabled throughout the study.

Concurrent Recognition is another discrete setting that allows concurrent recognition and audio acquisition. This feature requires a fast processor to perform speech recognition while collecting audio. It was enabled throughout the study.

Record Back-off refers to the duration of time before speech detection begins, which is incorporated into the data transmitted to the recognizer. This feature helps prevent clipping at the start of an utterance. Throughout the study, the record back-off setting remained at its default value of 250 milliseconds.

Appendix F

Visualization of results using Matlab

Percentage of Incorrect Recognitions per Display:

```
% Percentage of Incorrect Recognitions per Display
% Data
display_types = {'ND', 'HUD', 'PFD'};
incorrect_rates = [14, 3, 84]; % Incorrect recognition rates for ND, HUD, and PFD in percentage

% Create a bar chart
figure;
bar(incorrect_rates, 'FaceColor', [0.8, 0.2, 0.2], 'EdgeColor', [0.5, 0.1, 0.1], 'LineWidth', 1.5);
set(gca, 'XTickLabel', display_types, 'FontSize', 12, 'FontWeight', 'bold');
xlabel('Display Type', 'FontSize', 14, 'FontWeight', 'bold');
ylabel('Percentage of Incorrect Recognitions', 'FontSize', 14, 'FontWeight', 'bold');
title('Percentage of Incorrect Recognitions per Display', 'FontSize', 16, 'FontWeight', 'bold');
ylim([0 40]);
grid on;
ax = gca;
ax.GridAlpha = 0.5;

% Add percentage labels above bars
for i = 1:length(incorrect_rates)
    text(i, incorrect_rates(i) + 1, sprintf('%.1f%%', incorrect_rates(i)), ...
        'HorizontalAlignment', 'center', 'FontSize', 12, 'FontWeight', 'bold');
end

% Save the figure
saveas(gcf, 'incorrect_recognitions_per_display.png');
```

Commands Correctly Recognized per Display:

```
% Success Rate by Command Type
     % Data
     commands_types = {'ND', 'HUD', 'PFD'};
     success_rates = [0.95, 0.96, 0.68]; % Success rates for ND, HUD, and PFD

     % Create a bar chart
     figure;
     bar(success_rates, 'FaceColor', [0.4, 0.7, 0.9], 'EdgeColor', [0.2, 0.4, 0.5], 'LineWidth', 1.5);
     set(gca, 'XTickLabel', commands_types, 'FontSize', 12, 'FontWeight', 'bold');
     xlabel('Command Type', 'FontSize', 14, 'FontWeight', 'bold');
     ylabel('Recognition Success Rate', 'FontSize', 14, 'FontWeight', 'bold');
     title('Success Rate by Command Type', 'FontSize', 16, 'FontWeight', 'bold');
     ylim([0 1]);
     grid on;
     ax = gca;
     ax.GridAlpha = 0.5;

     % Add percentage labels above bars
     for i = 1:length(success_rates)
         text(i, success_rates(i) + 0.02, sprintf('%.1f%%', success_rates(i) * 100), ...
             'HorizontalAlignment', 'center', 'FontSize', 12, 'FontWeight', 'bold');
     end

     % Save the figure
     saveas(gcf, 'command_type_success_rates.png');
```

Incorrect Recognitions for the PFD:

```
% Data
pfd_commands = {'FOV UNITY', 'FOV 30', 'FOV 60', 'FOV 90', 'TRAFFIC ON'};
incorrect_counts = [15, 11, 37, 1, 3]; % Number of incorrect recognitions for each PFD command

% Create a bar chart
figure;
bar(incorrect_counts, 'FaceColor', [0.2, 0.6, 0.8], 'EdgeColor', [0.1, 0.3, 0.4], 'LineWidth', 1.5);
set(gca, 'XTickLabel', pfd_commands, 'FontSize', 12, 'FontWeight', 'bold', 'XTickLabelRotation', 45);
xlabel('PFD Command', 'FontSize', 14, 'FontWeight', 'bold');
ylabel('Number of Incorrect Recognitions', 'FontSize', 14, 'FontWeight', 'bold');
title('Incorrect Recognitions for the PFD Commands', 'FontSize', 16, 'FontWeight', 'bold');
ylim([0 40]);
grid on;
ax = gca;
ax.GridAlpha = 0.5;

% Add counts above bars
for i = 1:length(incorrect_counts)
    text(i, incorrect_counts(i) + 1, sprintf('%d', incorrect_counts(i)), ...
        'HorizontalAlignment', 'center', 'FontSize', 12, 'FontWeight', 'bold');
end

% Save the figure
saveas(gcf, 'incorrect_recognitions_pfd.png');
```

Appendix G

Basic code of the voice command system for on-board control systems.

Creating a complete voice command system for on-board control systems involves several components, including speech recognition, command parsing, and interfacing with the control systems. Here's a frame using Python and the SpeechRecognition library for speech recognition and command parsing. This code assumes a basic setup and does not include integration with actual aircraft systems, which would require more specific and robust implementation:

Install Required Libraries:

```
pip install SpeechRecognition pyttsx3
```

Import Libraries and Initialize Components:

```
import speech_recognition as sr
import pyttsx3

# Initialize the recognizer and text-to-speech engine
recognizer = sr.Recognizer()
tts_engine = pyttsx3.init()

# Function to convert text to speech
def speak(text):
    tts_engine.say(text)
    tts_engine.runAndWait()

# Function to recognize speech
def recognize_speech():
    with sr.Microphone() as source:
        print("Listening...")
        audio = recognizer.listen(source)
        try:
            command = recognizer.recognize_google(audio)
            print(f"Recognized command: {command}")
            return command.lower()
        except sr.UnknownValueError:
            print("Could not understand the command")
```

```
        return None
    except sr.RequestError:
        print("Error with the recognition service")
        return None
```

Define Command Parsing and Execution:

```
# Define command functions
def set_nav_range(value):
    speak(f"Setting navigation range to {value} nautical miles")
    # Here you would add the code to interact with the actual aircraft systems

def declutter_display():
    speak("Decluttering the display")
    # Here you would add the code to interact with the actual aircraft systems

def execute_command(command):
    if command:
        words = command.split()
        if len(words) < 2:
            speak("Incomplete command")
            return

        device = words[0]
        action = words[1]

        if device == "nav" and action == "range":
            if len(words) == 3 and words[2].isdigit():
                set_nav_range(words[2])
            else:
                speak("Please specify the range value")
        elif device == "display" and action == "declutter":
            declutter_display()
        else:
            speak("Unknown command")
    else:
        speak("No command detected")
```

Main Loop to Continuously Listen and Execute Commands:

```
def main():
    speak("Voice command system activated. Awaiting your commands.")
    while True:
        command = recognize_speech()
        execute_command(command)

if __name__ == "__main__":
    main()
```

1) Speech Recognition: The recognize_speech() function listens for audio input and uses Google's speech recognition service to convert it to text.
2) Text-to-Speech Feedback: The speak() function provides feedback to the user through text-to-speech.
3) Command Parsing: The execute_command() function parses the recognized text and executes the appropriate function based on the parsed command.
4) Command Functions: Example command functions like set_nav_range() and declutter_display() simulate actions that would be taken in a real system.
5) Main Loop: The main() function continuously listens for commands and processes them.