BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF INFORMATION TECHNOLOGY
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

PILOT PROFICIENCY CLASSIFICATION FROM GAZE

KLASIFIKACE ZPŮSOBILOSTI PILOTA Z POHLEDU

MASTER'S THESIS
DIPLOMOVÁ PRÁCE

AUTHOR **Bc. DOMINIK RUTA**
AUTOR PRÁCE

SUPERVISOR **doc. Ing. PETER CHUDÝ, Ph.D., MBA**
VEDOUCÍ PRÁCE

BRNO 2022

Brno University of Technology
Faculty of Information Technology

Department of Computer Graphics and Multimedia (DCGM) Academic year 2021/2022

Master's Thesis Specification

Student: **Ruta Dominik, Bc.**
Programme: Information Technology and Artificial Intelligence
Specialization: Computer Vision
Title: **Pilot proficiency classification from gaze**
Category: Artificial Intelligence
Assignment:

1. Get familiar with aircraft piloting techniques and procedures.
2. Research gaze and state classification techniques.
3. Design and implement framework for pilot proficiency classification.
4. Perform testing under various operational conditions and evaluate achieved results.
5. Discuss potential further improvements.

Recommended literature:

- Bishop, C. M.: Neural Networks for Pattern Recognition. Oxford, UK: Clarendon Press, 1996, ISBN 0198538642
- Pennington, J. E.: Single Pilot Scanning Behavior in Simulated Instrument Flight, Hampton, VA, NASA Langley Research Center, 1979.
- Tole, J. R., Stephens, A. T., Vivaudou, M., Ephrath, A., & Young, L. R.: Visual Scanning Behavior and Pilot Workload, Cambridge, MA: Massachusetts Institute of Technology, 1983.

Detailed formal requirements can be found at https://www.fit.vut.cz/study/theses/

Supervisor: **Chudý Peter, doc. Ing., Ph.D. MBA**
Head of Department: Černocký Jan, doc. Dr. Ing.
Beginning of work: November 1, 2021
Submission deadline: May 18, 2022
Approval date: May 17, 2022

Abstract

This work deals with the classification of pilot proficiency level and basic flight maneuvers from gaze. The goal is to provide additional valuable tool for aviation instructors to evaluate proficiency of pilot students and provides them with feedback. This idea is based on results of numerous relevant studies, which discovered correlation between effective scanning patterns and domain performance. This thesis considers two proficiency levels — amateur and experienced.

This work utilizes common analysis metrics of visual scanning and machine-learning classification techniques. The Support Vector Machine algorithm is used for the proficiency classification and Hidden Markov Models are utilized in basic flight maneuvers classification. The result of this thesis is a high accuracy proficiency classification and good ability to distinguish between individual basic flight maneuvers performed by pilots.

Abstrakt

Tato práce se zabývá klasifikací úrovně odbornosti pilota a leteckých manévrů z pohledu očí. Cílem je poskytnout další cenný nástroj pro hodnocení pilotáže leteckými instruktory a poskytnout tak zpětnou vazbu trénovaným pilotům. Tato myšlenka je založena na základě výsledků relevantních studií, které objevily korelaci mezi užíváním efektivních skenovacích vzorů a doménové výkonnosti. V této práci jsou uvažovány dvě třídy odbornosti — piloti a nováčci.

Tato práce využívá běžné metriky pro analýzu pohledu očí. Dále jsou v této práci využity klasifikační techniky strojového učení. Metoda podpůrných vektorů je využita pro klasifikaci úrovně způsobilosti, zatímco pro klasifikaci leteckých manévrů jsou využity skryté Markovovy modely. Výsledkem práce je vysoce přesná klasifikace úrovně odbornosti pilot a dobrá schopnost rozeznat individuální letecké manévry provedené piloty.

Keywords

classification, machine-learning, supervised learning, Support Vector Machine,
Hidden Markov Models, airplane piloting, pilot proficiency, scanning patterns, visual scanning, gaze-tracking, flight maneuvers

Klíčová slova

klasifikace, strojové učení, pilotáž, způsobilost pilota, skenovací vzory, sledování očí, skryté Markovovy modely, letecké manévry, učení s učitelem,

Reference

RUTA, Dominik. *Pilot proficiency classification from gaze.* Brno, 2022. Master's thesis. Brno University of Technology, Faculty of Information Technology. Supervisor doc. Ing. Peter Chudý, Ph.D., MBA

Rozšířený abstrakt

Důležitým předpokladem pro efektivní provedení letových manévrů je efektivní využití jednotlivých letových přístrojů. Schopnost efektivního využití přístrojů postrádají zejména nováčci. Proto je předmětem tréninku také studie efektivních sledovacích vzorů. Instruktoři však nemají kvalitní nástroj pro evaluaci použitých technik trénovaných pilotů. Pokud by takový nástroj měli k dispozici, věděli by, kde jsou slabá místa trénovaného pilota. Díky této znalosti by pak mohli přizpůsobit jejich letecký trénink za účelem větší efektivity. Hlavním cílem této práce je tak návrh klasifikátoru odbornosti pilota a kvality provedených manévrů na základě pohledu očí a využitých sledovacích technik jednotlivých přístrojů. Celkově dvě třídy odbornosti jsou v této práci uvažovány — zkušený pilot a amatérský pilot.

Úvodní část této práce shrnuje znalosti pro správnou techniku pilotáže se zaměřením na přístrojové létání. Zaměření na přístrojové létání je důležité z toho důvodu, že klasifikace způsobilosti pilota je postavena na hodnocení správného používání jednotlivých přístrojů. Proto jsou v této kapitole blíže popsány základní letové přístroje. Dále jsou popsány dva běžné způsoby využití jednotlivých přístrojů pro správné řízení letadla. Rovněž jsou popsány běžně používané efektivní sledovací vzory, které zkušení pilot využívají při provádění jednotlivých manévrů. Informace ohledně sledovacích vzorů jsou cenné zejména pro finální analýzu dosažených výsleků. Na konci této kapitoly jsou pak blíže popsány základní letecké manévry, které jsou v této práci předmětem klasifikace.

Další část blíže popisuje problematiku sledování pohledu očí. Nejdříve jsou shrnuty poznatky z prací a výzkumů, které se zabývaly podobnou tématikou jako tato práce, a sice studovaní sledovacích vzorů v různých doménách. Hojná část těchto prací se zaměřovala právě na studii sledovacích vzorů pílotů a jejich rozdíl v rámci různých úrovní zkušeností. Následně jsou popsány základní termíny z oblasti měření sledování očí. Znalost těchto pojmů je klíčová pro správné nastavení měřících přístrojů, efektivní zpracování naměřených dat, a také implementaci a správné nastavní jednotlivých klasifikátorů. Nakonec je uveden výčet běžně používaných metrik pro analýzu sledovacích vzorů očí. Blíže jsou pak popsány přechodové matice, které jsou využity v rámci klasifikace úrovně odpornosti pilota.

Další podstatnou část tvoří kapitola zabývající se studií klasifikačních technik. Jelikož problém klasifikace spadá do okruhu strojového učení, jsou proto nejdříve uvedeny základní principy strojového učení. Následně je blíže uvedena teorie klasifikace. Jsou popsány postupy pro efektivní trénování a evaluci klasifikátorů. Tyto znalosti jsou využity při návrhu a implementaci samotného klasifikačního rámce. Následně jsou blíže popsány dvě klasifikační metody, které byly v této využity. První z nich je metoda podpůrných vektorů. Jedná se o jednoduchý klasifikátor, který je efektivní pro klasifikací dvou tříd. Dále jeho výhoda leží ve schopnosti vypořádání se z daty vysoké dimenze. Zejména z těchto důvodů byl tento klasifikátor vybrán pro účely klasifikace úrovně odbornosti pilota. Další z popsaných technik jsou skryté Markovovy modely. Tato metoda klasifikace byla vybrána pro klasifikaci základních leteckých manévrů. Důvodem této volby je schopnost efektivního zpracování datových řad.

Nejdůležitější části této práce je věnována praktické části. Ta nejdříve popisuje návrh samotného experimentu pro sběr dat. Popsán je letový vzorec, který každý z účastníků zalétl celkem 4-krát. Vzorec se skládá z kombinace základních leteckých manévrů. Zahrnuty jsou všechny manévry popsány v kapitole zabývající se technikou pilotáže. Následně je příblížen simulační rámec pro měření a sběr dat. V bližším detailu je popsáno zařízení pro měření pohledu očí. Následně jsou popsány způsoby zpracování naměřených dat. Toto zpracování zahrnuje extrakci dat z jednotlivých letových fází experimentu. Tato data následně tvoří datové sady pro trénování klasifikátorů. Velká část popisu zpracování je věnována

technice přiřazování fixačních oblastí zájmů jednotlivým fixacím. Je zhodnocen původní návrh rozložení těchto oblastí. Výsledkem je, že původní návrh nebyl dostatečně silný pro efektivní rozlišení jednotlivých leteckých manévrů. Z toho důvodu byla použitá technika dělení některých oblastí na dílčí oblasti zvýrazňující konkrétní elementy některých přístrojů. Nově vzniklé kombinace vedly k uspokojujícím výsledkům. Celkově byly vytvořeny dvě různé rozložení. V poslední části je popsán návrh a implemntace klasifikačního rámce pro klasifikaci leteckých manévrů a úrovně odbornosti pilota. Rámce nejdříve klasifikuje úroveň odbornosti do jedné ze dvou tříd, a následně vybere model pro klasifikaci leteckým manévrů. Důvodem tohoto dělení je, že nelze porovnávat manévry provedené zkušenými a nezkušenými piloty v rámci jednoho klasifikačního modelu. Výsledky by nebyly relevantní. Model pro klasifikaci manévrů je navržen tak, že lze klasifikovat různé třídy manévrů, a nikoliv pouze jednotlivé manévry.

Další část se věnuje evaluaci klasifikátorů pro tři různé případy. V prvním případě byly klasifikovány dvě třídy manévrů — podélné a stranové. V tomto případě byla klasifikace manévrů v rámci obou tříd odbornosti vysoká. Přesnost klasifikace odobornosti pilota byla rovněž vysoká. Při této evaluaci bylo použito jedno ze dvou rozložení fixačních oblastí zájmu. V dalším případě byly manévry rozděleny do 4 tříd, kdy vertikální zatáčky nebyly uvažovány. V případě zkušených pilotů přesnost klasifikace byla rovněž vysoká. U nezkušených pilotů však byla vnesena nejistota a přesnost byla nižší. V posledním případě bylo klasifikováno všech 6 manévrů. V obou případech byl problém zejména při rozlišování mezi horizontálními a vertikálními zatáčkami.

Poslední část práce pak shrunuje dosažené výsledky a diskutuje další možný vývoj.

Pilot proficiency classification from gaze

Declaration

I hereby declare that this Master's thesis was prepared as an original work by the author under the supervision of Mr. doc. Ing. Petera Chudého, Ph.D., MBA. I have listed all the literary sources, publications and other sources, which were used during the preparation of this thesis.

.....
Dominik Ruta
May 25, 2022

Acknowledgements

First of all I would like to express my gratitude to my supervisor Mr. doc. Ing. Petr Chudý, Ph.D., MBA for his tremendous support and invaluable advice provided during planning and development of this thesis. I can not express how grateful I am for his willingness to provide his time and knowledge. Next, I would like to thank my colleagues from the research group Aeroworks for their assistance during preparations and execution of flight experiments. Many thanks also belong to all participants, who were willing to invest their time into the flight experiments and provide their valuable skills. I can not forget to express my gratitude to the Technology Agency of the Czech Republic for funding the National Competence Centre for Aeronautics and Space TN01000029, subproject TN01000029-16 Tactical Cognitive Agent. Finally, I have to express my many thanks to my girlfriend Anna and my family, for their endless patience and mental support.

Contents

List of Figures

List of Tables

List of Symbols

List of Abbreviations

Abbreviation	Meaning
ADI	Attitude Direction Indicator
AGL	Above Ground Level
AI	Attitude Indicator
AOI	Area of Interest
API	Application Programming Interface
ASI	Airspeed Indicator
CAS	Calibrated Airspeed
CG	Center of Gravity
DME	Distance Measuring Equipment
FAA	The Federal Aviation Administration
FDS	Flight Director System
GPS	The Global Positioning System
GNSS	Global Navigation Satellite System
GS	Groundspeed
HI	Heading Indicator
HMM	Hidden Markov Model
HSI	Horizontal Situation Indicator
IAS	Indicated Airspeed
ICAO	The International Civil Aviation Organization
I-DT	Dispersion-Threshold algorithm
IFR	Instrument Flight Rules
ILS	Instrument Landing System
IVSI	Instantaneous Vertical Speed Indicator
KTAS	True Airspeed in Knots
KNN	K-Nearest Neighbours

Abbreviation	**Meaning**
LSA	Light Sports Aircraft
NASA	National Aeronautics and Space Administration
NN	Neural Network
PCA	Principal Component Analysis
PDT	Proportional Dwell Time
PFD	Primary Flight Data
RPM	Revolutions Per Minute
SDP	Standard Datum Plane
SVM	Support Vector Machine
SEP	Single-Engine Propeller
TAS	True Airspeed
TC	Turn Coordinator
TMD	Transition Matrix Density
UDP	User Datagram Protocol
VFR	Visual Flight Rules
VOR	Very high frequency Omni-directional Range
VSI	Vertical Speed Indicator

List of Symbols

Symbol	Meaning
$\alpha_t(i)$	Forward variable
$\beta_t(i)$	Backward variable
$\gamma_t(i)$	Probability of being in i-th state given observation sequence O
$\delta_t(i)$	Highest probability for hidden state path
λ	Set of HMM parameters
$\hat{\lambda}$	Re-estimated HMM parameters
ξ_i	Slack variable of i-th feature vector
$\xi_t(i,j)$	Probability of being in i-th state at time t and j-th state at time $t+1$ given observation sequence O
π_i	Initial probability for i-th state
$\hat{\pi}_i$	Most likely estimate of π_i
ϕ_S	Angle of bank for standard rate of turn
$\Psi_t(i)$	Argument maximizing $\delta_t(i)$
$\boldsymbol{\pi}$	Initial state probability distribution
a_{ij}	Probability of transition from i-th state to j-th state
$\hat{a}_{ij}$	Most likely estimate of a_{ij}
$b_i(j)$	Probability of observing j-th symbol in i-th state
$\hat{b}_i(j)$	Most likely estimate of $b_i(j)$
C	Regularization parameter
D	Number of dimensions
D_m	Mean duty cycle
E	Set of observable symbols
E_i	i-th observable symbol
H	Hidden Markov Model
h	Altitude
N	Number of dwells / AOIs / feature vectors / states
M	Number of observable symbols
n_{ij}	Frequency of transition from i-th state to j-th state
$m_i(j)$	Frequency of observing j-th symbol in i-th state
O	Observation sequence
o_t	Observation sequence random variable
P_{ij}	Probability of transition from i-th state to j-th state

Symbol	Meaning
P_{ij_1}	Probability of one-way transition occurrence between i-th and j-th state
P_{ij_2}	Probability of two-way transition occurrence between i-th and j-th state
p_i	Period between i-th and next dwell
S	Set of possible hidden states
S_i	i-th hidden state
s	Distance
T	Length of the observation sequence
t	Time
t_i	Dwell time of i-th dwell
V_Y	Best rate of climb
V_X	Best angle of climb
X	Hidden state path
x_i	Feature of feature vector
x_t	Hidden state path random variable
y_C	Maneuver class inference result
y_i	Feature vector label
y_L	Level of proficiency inference result
$\mathbf{A}$	State transition matrix
$\mathbf{B}$	Emission matrix
$\mathbf{x_i}$	Feature vector
$\mathbf{w}$	Hyperplane normal vector

Chapter 1

Introduction

Gaze study is an interesting discipline, which has found utilization in numerous applications such as interface design, fatigue detection, medical research or virtual reality. For instance, study on visual scanning strategies of pilots during various maneuvers and situations, resulted in the design of an optimal arrangement of instruments in the cockpit, which lately became a world wide standard. Gaze-tracking is a powerful tool which can be used to discover and better understand efficiency of visual scanning strategies used by domain experts. It has been proved there exists a strong correlation between performance and efficient scanning strategies. Thus, if we have means to study effective patterns, we can use the acquired knowledge, among other possible utilization, for an effective training of domain novices and detect, whether the individual is approaching the level of an expert or not. Gaze analysis is also a valuable information for the instructor, as she/he can give valuable feedback to trainee on a potentially ineffective scanning pattern. The overall evaluation would thus take into account how effective in processing of visual information the pilot is.

This approach could also be used in pilot training. Effective scan patterns play a crucial part in the overall performance. However, this criteria is usually not, if ever, taken into account in the overall evaluation. Evaluation of quality of performed flight maneuvers and other piloting techniques, is mainly based on performance and subjective rating of the instructor. Although the instructor places great emphasis on the proper use of instruments, he does not have required tools to evaluate scanning patterns to classify pilot proficiency.

The expansion of technology has also affected camera technology, thanks to which cameras can be effectively installed for gaze-tracking in an aircraft cockpit or directly worn by the pilot on the headset. Classification technologies of effective visual scanning patterns can thus be deployed. These could be designed and implemented to asses multiple aspects of the pilot's scanning strategy during the flight. At first, the overall visual scanning strategy during individual flight maneuvers could be analyzed. Based on the analysis, pilot proficiency level would be determined, i.e. if the scanning strategy is advanced enough so more complex techniques and procedures could be trained, or rather the strategy should be subject of further training as it is not effective enough and is more similar to one used by inexperienced pilots. In another step, the scanning strategy could be utilized for the purpose off flight maneuver classification. Result of this type of classification would indicate, if utilized scanning strategy corresponds to those commonly used by experienced pilots during individual flight maneuvers.

This thesis deals with the classification of pilot proficiency and performed basic flight maneuvers. Overall, two proficiency levels are considered — experienced and amateur. Chapter 2, Airplane Piloting Techniques and Procedures, deals with the description of principles of instrument flight. This includes description of basic flight instruments, common control and scanning strategies, and finally, basic flight maneuvers. Following chapter 3, Gaze Classification Techniques, provides study of commonly used visual scanning strategy metrics. Part of this section also includes research on related work and basic gaze-tracking terminology. Chapter 4, State Classification Techniques, serves as an introduction into classification domain. Furthermore, this chapter introduces theoretical background of two algorithms utilized in this thesis. These are Support Vector Machine and Hidden Markov Model algorithms. The chapter 5, Design and Implementation of the Classification Framework, provides design and implementation details of the classification framework along with experiment design. Chapter 6, Evaluation, provides and discusses achieved results.

Chapter 2

Airplane Piloting Techniques and Procedures

In this chapter, necessary knowledge on piloting techniques and common procedures will be provided. It is crucial to understand the problematic, in order to properly design respective experiments and to analyze obtained results. The emphasis is placed on the proper understanding of principles of instrument flight and their utilization in basic flight maneuvers and procedures.

At first, a brief introduction into the concept of the instrument flight will be provided. Next, basic flight instruments will be described. This part is important to understand, especially the purpose of each instrument, as visual scanning used for classification of flight maneuvers and proficiency, are composed of sequences of flight instruments. Next part of this chapter focuses on the introduction of basic flight maneuvers the pilot should be able to manage, in order to perform a successful flight. At the end of this chapter, examples of various flight patterns, on which a beginner pilot trains and improves basic knowledge and piloting skills, will be provided.

2.1 Rules of Flight

Piloting an aircraft is a complex task, which requires a pilot mastering multiple disciplines together. In order to perform safe flight, the pilot needs to be capable of the attitude control, navigation and collision avoidance in any situation. This capability however is highly influenced by the current weather conditions. Thus, two sets of regulations are defined, each for different weather conditions, which declare rules for civil flight

First set of regulations is called Visual Flight Rules (VFR). Using these rules, pilot solely uses references to outside visual cues such as natural horizon, buildings and terrain features. However, operating aircraft in such a way is possible only in relatively clear weather conditions. The pilot must be able to clearly see the outside references and other aircraft in the surrounding airspace from a sufficient distance. Otherwise, the flight would be highly unsafe [9].

If weather conditions are not favorable for a VFR flight, the pilot operates the aircraft using the Instrument Flight Rules (IFR). In this case, the pilot uses solely instruments available in the flying deck for a proper aircrafft control and electronic signals to accomplish navigation. Most of the IFR navigation is based on the use of multiple satellite-based systems (GPS/GNSS) and ground-based systems (DME/VORs, ILS) [9].

The IFR flight is a demanding discipline for which several conditions must be met. Most important are [9]:

- The airplane must be equipped with certified and properly tested instruments.
- The pilot must have an instrument rating - required qualification for IFR flight.
- Before each flight, the pilot must fill IFR flight plan and pass it to eligible entity.

Reaching the qualification for IFR flight requires pilot to undergo sufficient training aimed on mastering multiple disciplines. These disciplines are performing basic flight maneuvers using instruments only, instrument radio navigation, instrument takeoff and landing, solving unusual flight positions, IFR pre-flight planning or finnishing the flight with a reduced number of instruments [9].

2.2 Basic Flight Instruments

Flight instruments play an important role in making of pilot situational awareness, as they provide all necessary flight data defining the actual flight state of an aircraft. It is crucial for the pilot to understand how to interpret and manipulate flight instruments in order to perform safe flight. If the pilot would not have access to a spectrum of flight data, flying the aircraft would become under adverse situations dangerous.

Flight instruments are located in an airplane cockpit, and provide multiple flight data information. Among the most basic and crucial flight data are:

- airspeed,
- attitude,
- altitude,
- vertical speed,
- heading,
- turn rate.

As the aviation evolved over the years, more and more information were provided to the pilot. As the reading of data become more challenging, in order to make it more convenient for pilots, new principles of visualisation and interpretation on flight instruments were developed. Originally used mechanical instruments were gradually replaced by electro-mechanical, and later with electro-optical instruments. Single-purpose instruments evolved into much complex multipurpose ones, providing multiple clustered data on a single instrument or device. In case of electro-optical instruments, data were primarily displayed on the Primary Flight Displays (PFD) [25].

Six instruments will be described in the following part of this section. These instruments are referred to as the “aviation six pack” or “basic six”, and are the most basic instruments, and are in some way present in all airplanes [38].

Figure 2.1: Single engine aircraft cockpit [47].

2.2.1 Altimeter

Altimeter (ALT) is one of the most important instruments in the cockpit, as it is the only instrument, which provides altitude information. It provides an indicated height above a given pressure datum. By default, altimeters are calibrated to indicate height above the sea level. There are five major types of altitude the pilot should be familiar with [10][19]:

- *Indicated Altitude* — is directly obtained from the altimeter. Indicated value is usually affected by multiple errors (installation, instrument). This is something the pilot has to be aware of and be able to work with.
- *True Altitude* — is the actual altitude of the airplane. It is a vertical distance of the airplane above the sea level.
- *Absolute Altitude* — is the vertical distance of the airplane above the ground. It is usually referenced to as above ground level (AGL).
- *Pressure Altitude* — is the height above standard datum plane (SDP), which is a theoretical level, where air pressure equals to 29.92 "Hg, or 1 013.4 hPa. It is used to measure density altitude, true altitude, and other performance data.
- *Density Altitude* — represents the pressure altitude corrected for multiple variations from standard temperature.

A typical altimeter uses a dial with numbers arranged from zero to nine in a clockwise direction to display altitude. The method of indicating the measured height depends very much on the manufacturer. The most common way of indicating altitude however, is to use three hands of different lengths [10]. An example of a common altimeter gauge is depicted in figure 2.2.

The altimeter is a part of the pitot-static system, together with the airspeed indicator and the vertical speed indicator. This system provides both, dynamic and static, pressures necessary to indicate respective state quantities. In case of the altimeter, only a static pressure is used. Since in various situations static pressure is not always the same, in order

Figure 2.2: An example of a common altimeter gauge.

to get valid altitude measurement, pilot has to be able to configure the altimeter correctly depending on the conditions [19].

2.2.2 Airspeed Indicator

Airspeed Indicator (ASI) is an another crucial instrument present on the dashboard. As the aircraft performance depends highly on the airspeed, for pilot it is absolutely necessary to know a current airspeed, in order to perform a safe flight. Thanks to the airspeed information, the pilot, for instance, knows whether the aircraft has sufficient speed at takeoff or if the aircraft is approaching the stalling speed. Whether it can deploy or, on the contrary, must close flaps or landing gear, or knows when the aircraft is approaching its maximum flight speed [19].

As was the case with an altimeter, also here the pilot must be familiar with multiple types of velocities [10]:

- *Indicated Airspeed (IAS)* — is directly obtained from the indicator. Displayed value is affected by both, installation and instrument errors. It is uncorrected for variations in atmospheric density.
- *Calibrated Airspeed (CAS)* — is indicated airspeed with minimized instrument and installation errors. While at high speeds CAS and IAS are almost identical, at low speeds, the difference between the two types may be of several units.
- *True Airspeed (TAS)* — is calibrated airspeed corrected for an altitude and nonstandard temperature.
- *Groundspeed (GS)* — is true airspeed adjusted for the current wind. It measures actual speed of an airplane over the ground.

Figure 2.3: Airspeed Indicator gauge.

Typical ASI is designed as a dial with a single hand moving in the clockwise direction. It has been standardized, that the ASI scale has to include multiple coloured arcs and lines to indicate various speed limits. The arcs are used to indicate operating speed ranges, whereas radial lines indicate limiting airspeeds. Among the most important markings belong [10][19]:

- *White arc* — indicates flap operating range. Range in which flaps could be operated.
- V_{S0} — denotes the stalling speed or the minimum steady flight airspeed in the landing configuration (gear and flaps down). It is located at the lower limit of the white arc.
- V_{FE} — indicates maximum speed, at which flaps can be extended. It is located at the upper limit of the white arc.
- *Green arc* — indicates normal operating airspeed range of the airplane.
- V_{S1} — denotes the stalling speed for the airplane in the clean configuration (gear and flaps up).
- V_{N0} — is the maximum structural cruising speed. Speed should not exceed this limit, except in smooth air. It is located at the upper limit of the green arc.
- *Yellow arc* — is an indicator of range, in which the pilot should fly with caution and only in a smooth air.
- *Red line* — indicates the airspeed pilot should never exceed. High risk of structural damage is taken when flying in higher airspeed.

ASI is the only instrument of the pitot-static instrument using both dynamic and static pressure to obtain measured value. The ASI gauge with colored arcs can be seen in figure 2.3.

2.2.3 Vertical Speed Indicator

Third, and the last instrument using pitot-static system is Vertical Speed Indicator (VSI). As in case of the altimeter, it utilizes only the static pressure. By looking at VSI, pilot is basically able to determine two information. First, the information whether the airplane climbs, descends or is in the level flight, and second, the actual rate of climb or descent. The rate is usually measured in units per minute [19].

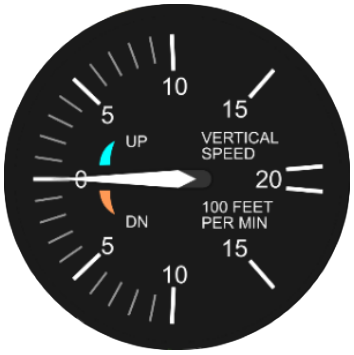


Figure 2.4: Vertical Speed Indicator gauge.

Dial of the VSI consists of a single hand moving in the clockwise direction in case of climbing, while in the descent, the hand moves in the opposite counter-clockwise direction. When in the level flight, the hand indicates zero on the indicator [10].

Since the basic VSI indicates actual rate with some lag, it is sometimes replaced by Instantaneous Vertical Speed Indicator (IVSI) compensating initial lag present in the basic VSI. The period between the start of the climb/descent and the rate displayed on the VSI is in order of seconds [19]. A typical analog VSI is depicted in figure 2.4.

Figure 2.5: Heading Indicator with a heading bug set onto the North.

2.2.4 Heading Indicator

Heading Indicator (HI), also known as Directional Gyro (DG), is one of the three gyroscopic flight instruments. It basically a mechanical instrument indicating airplane heading designed to facilitate the use of the classical magnetic compass. Nevertheless, magnetic compass is still used to align the HI towards the reference, magnetic North, in order to get valid information from the HI. This procedure is necessary, because the HI loses accuracy over time due to the gyroscopic drift. Thus, the pilot has to provide correction to the HI periodically [10].

To prevent these errors, the two instruments were combined into a single instrument, which constantly adjusts reference to the magnetic north. The instrument is called remote

indicating compass and it utilizes Earth's magnetic filed and its magnetic flux for constant adjustment [10]. HI usually allows the pilot to set so called "heading bug" onto the desired heading, so does not to remember it all the time. An example of the HI can be seen in figure 2.5.

2.2.5 Attitude Indicator

Attitude Indicator (AI), sometimes referred to as the artificial horizon, is an instrument whose purpose is to provide the pilot with an information of the airplane attitude with respect to the Earth's horizon. That is, orientation in both roll and pitch axis. The AI is another gyroscopic flight instruments representative. It is very sensitive, so it is able to indicate even small changes in pitch or bank angle.

Display of the AI consist of the two coloured areas separated by vertical line representing the horizon. In the middle is located airplane representation. Instrument provides both pitch and bank scales in degrees. A classic look AI is presented in figure 2.6.

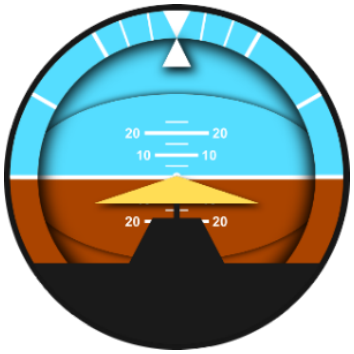


Figure 2.6: Classic look Attitude Indicator.

In further development, classic AI has been merged together with Flight Director System (FDS) into a single instrument referred to as Attitude Direction Indicator (ADI). Besides displaying the airplane attitude, ADI utilizes received radio navigational data, in order to provide lateral and vertical navigational deviations [19].

2.2.6 Turn Indicators

On aircraft's instrument panel can be mounted one of two, or both, instruments to indicate turn direction, its rate and overall „quality", or coordination, of the turn. They are referenced to as Turn and Bank (Turn and Slip) Indicator and Turn Coordinator (TC) respectively [10].

Turn and Bank Indicator displays the turn direction in the horizontal plane. In addition, it provides the pilot with a rate of turn information. The rate is indicated by a needle on the scale usually consisting of three bars. Two extreme bars indicate standard-rate turn. Finally, the instrument is equipped with an inclinometer, which detects airplane yaw. The inclinometer consists of a tube and ball in it. In order to perform coordinated turn, the ball should stay in the middle of the tube, otherwise, rudder has to be used to compensate slip or skid [10].

Turn Coordinator is the further development of the Turn and Bank Indicator. It provides additional information in the form of the roll rate. This is indicated by an airplane

Figure 2.7: A typical Turn Coordinator with an inclinometer.

marking. Sometimes it is confused with the AI, but except the AI, the TC does not provide pitch angle information [10]. This instrument can be viewed in figure 2.7.

2.2.7 Instrument arrangements

All six instruments described above are coordinated instruments, the use of which enables the pilot to determine the state of the aircraft. In order to make instruments reading for pilot intuitive and less demanding, it is important to arrange those instruments on the dashboard into a reasonable layout [38].

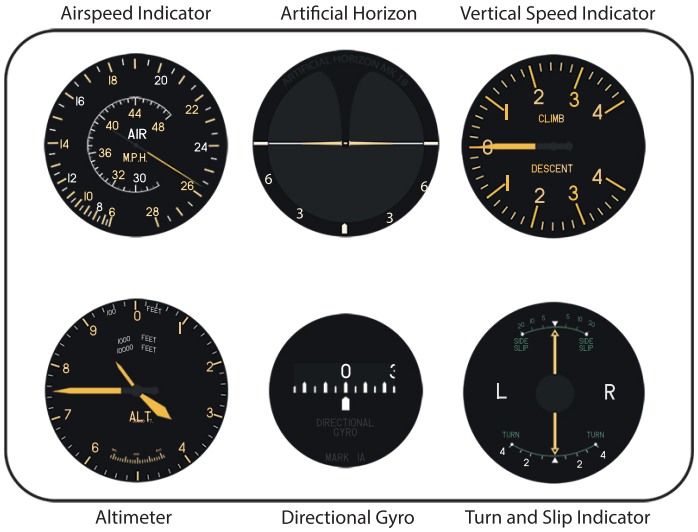


Figure 2.8: "Basic six" arrangement used in the cockpit of a Supermarine Spitfire.

The first real attempt to design instrument layout resulted in an arrangement called "basic six", or "six pack". In this arrangement are instruments aligned into two rows and three columns. Middle top instrument is the AI (or artificial horizon) as it is referred to as the master instrument, since it indicates pitch and roll of the airplane. Other three instruments: ASI, VI and ALT, surround the AI. The reason for this arrangement the is fact, that attitude is directly related to the control of airspeed and altitude. Thus, the arrangement of these three instruments around the AI supports the interpretation of pitch

attitude. The HI is placed bellow the AI and supports the roll attitude interpretation. Finally, the turn indicator is placed next to the heading indicator as it also relates to roll attitude and heading indication [38]. In figure 2.8, an example of the "basic six" arrangement is presented. This particular arrangement of instruments was used in the famous Supermarine Spitfire.

As the aviation evolved over period of time, new and more complex instruments were introduced. Due to this evolution, the original "six pack" had to be reviewed and new layout called
"T-arrangement", or "basic T" has been designed. In this arrangement, four main instruments are considered and placed within a T shape. These are ASI, ALT and AI (or ADI) forming the horizontal bar of the T, and HI (or HSI) forming the vertical bar. Other two, TC and VSI are placed each on both sides of the HI. The arrangement is depicted in figure 2.9.

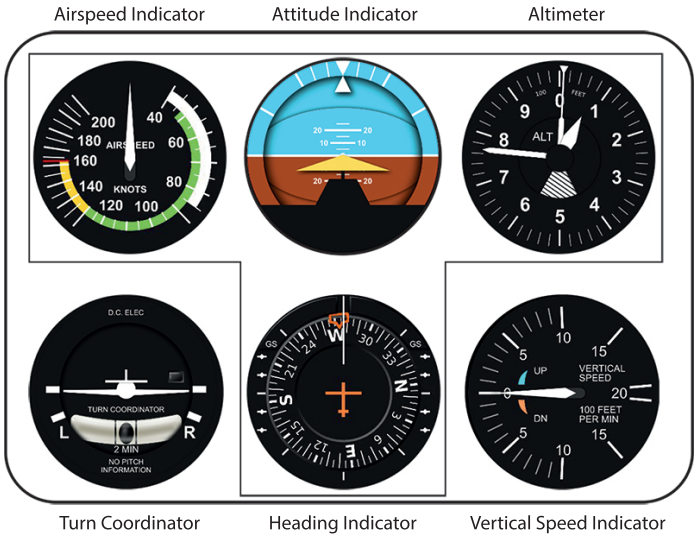


Figure 2.9: Typical example of the T-arrangement.

2.3 Instrument Flying Basics

When flying instrument flight, aircraft attitude and its navigation is performed by using solely on-board instruments rather than outside references. For a proper aircraft control, pilot has to be able to correctly interpret each instrument, especially those of the "basic six". Proper interpretation provides the pilot same information, as he would get by using outside references in the VFR.

Overall, two approaches of flight instruments interpretation are being commonly used. Depending on the personal preference and controlled aircraft, suitable approach is selected.

2.3.1 Control and Performance Method

Basic concept of this method is, that the aircraft is controlled by properly setting aircraft attitude (control) and power (performance). All instruments are divided into three groups [35]:

- *Control* — is a group of instruments composed of AI and power instruments. Power instruments vary across airplanes, but the most common are tachometer and manifold pressure.
- *Performance* — is a group of instruments indicating actual performance state of the aircraft. These instruments are ALT, ASI and VSI.
- *Navigation* — group includes instruments used to navigate the aircraft onto the selected positions. Set of instruments includes various course indicators, range indicators, glideslope indicators, and bearing pointers.

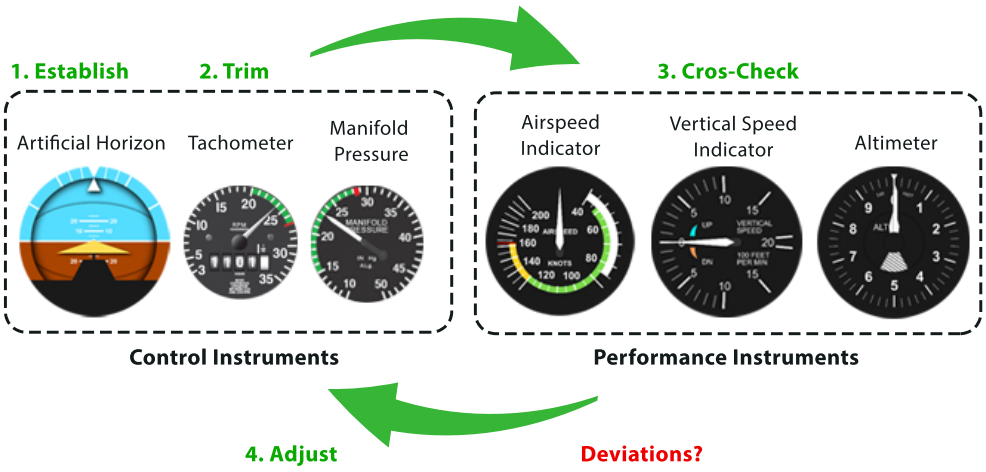


Figure 2.10: Scheme of the control and performance method [35].

The concept of establishing desired aircraft attitude utilizing this method is graphically presented in figure 2.10. Overall, it is composed of 4 steps [35]:

1. Establish desired attitude and power settings.
2. Trim control surfaces.
3. Cross-check performance instruments if established attitude and power settings provide desired performance.
4. If there are any deviations, adjust attitude and power settings.

2.3.2 Primary and Supporting Method

Another method of the aircraft attitude control works with the concept of assigning primary and supporting instruments to each component (function) of the attitude control. These components are pitch, bank and power. Groups are defined as follows [35]:

- *Primary* — is group of three instruments providing for each maneuver direct performance information for each out of three attitude functions - pitch, bank and power.
- *Supporting* — is group of instruments representing backup for the primary instrument. For each function, there may be more than one supporting instrument.

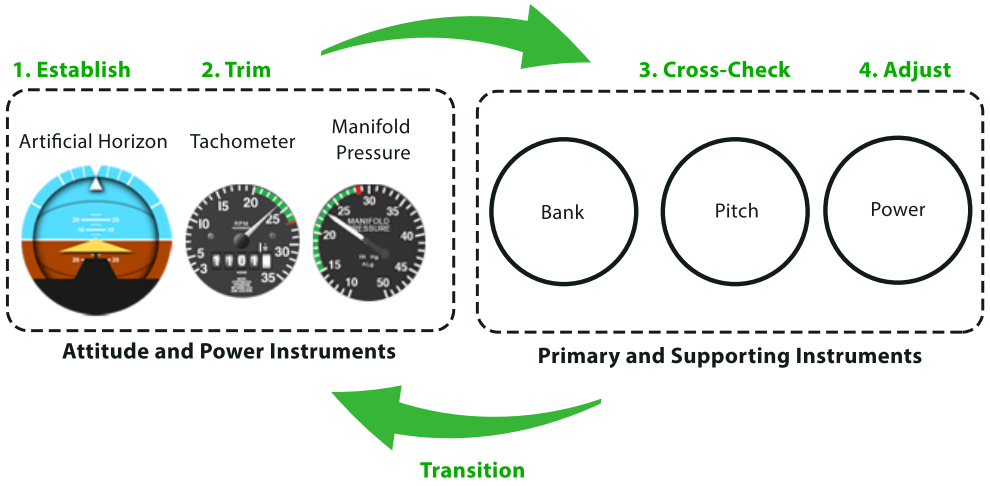


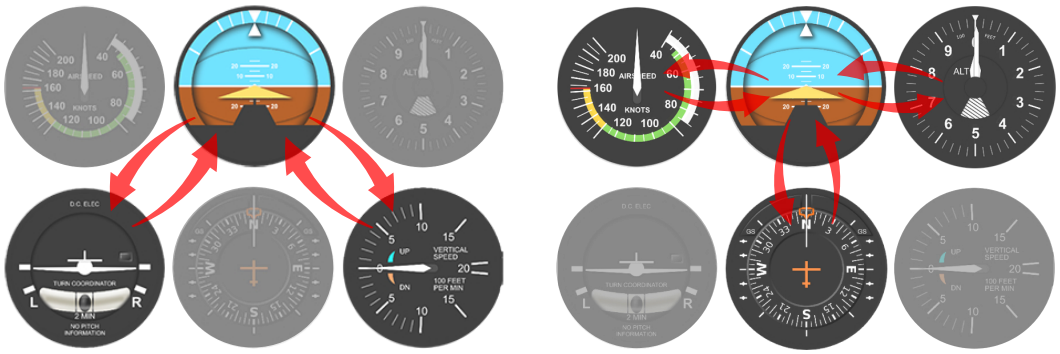
Figure 2.11: An illustration of the principle of the primary and supporting method [35].

The concept of using instruments during particular flight maneuvers while utilizing primary and supporting method, is the following [35]:

1. During the transitions between maneuvers, AI and power instruments are used.
2. After the attitude and power setting are established, use appropriate primary and supporting instruments to maintain the attitude.

2.4 Instrument Scanning Strategies

In order to produce desired performance in the attitude and power control, pilot uses his scanning strategy, i.e. cross-check, to observe and interpret instruments, that seems to be the most effective from his perspective. Furthermore, he may have utilized multiple strategies, each for a different maneuver. Although the scanning patterns are highly self preference matter, it is recommended to use primarily instruments giving the instant information relevant to the maneuver and attitude function. Four o the scanning strategies, which are widely used between expert pilots, are described in this section. Each of the four strategies fits to different maneuvers. All scans stick to the rule: attitude plus power equals performance. Thus, AI is always in the centre of the scan [9].



(a) Inverted V-scan strategy.

(b) T-scan strategy.

Figure 2.12: Instrument scanning strategies utilizing only a subset of instruments.

2.4.1 Inverted V-Scan

This scan involves primarily three instruments - AI, TC and VSI. Center of the scan is the AI. Initial scan of the AI is followed by transitions to the TC and back. Finally, vertical speed is cross-checked. Inverted V-scan is useful to validate, if all instruments, or rather systems they are powered with, work properly. This comes from the fact, that all three instruments are driven by different systems (vacuum, electric and pitot-static system). In terms of maneuvers, with ALT included, this scan is suitable for turns [9][35].

2.4.2 T-Scan

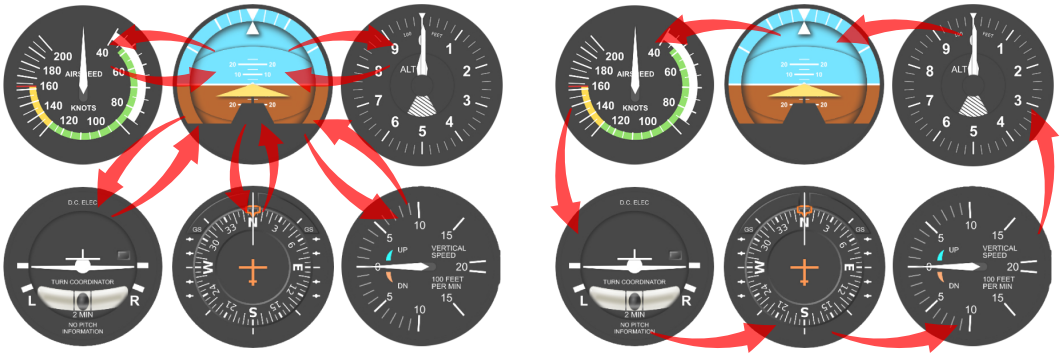
The T-scan is a scanning technique involving instruments of major performance parameters - airspeed, heading and altitude. Center of the scan is the AI. Then, scan continues to the ASI and back, from the AI to the HI and back, and finally, to the ALT and back to the AI. This scan is useful in straight-and-level flight, as the pilot checks correct heading and flight level, while trying to maintain stable airspeed [35].

2.4.3 Selective Radial Scan

Selective radial scan, sometimes also referred to as Hub and Spoke, is the combination of two previous scans, and is the most commonly used scanning technique. “Selective” because pilot selects only the most relevant instruments for the maneuver, and “radial” because the scan is centered around the AI (Hub) and eyes radially moves to other instrument (Spoke) and back. Statistically, the AI is in focus in 80-90 % of the flight time. The combination of predominantly looked at instruments determine performed maneuver [9][35].

2.4.4 Rectangular Scan

The last common scanning technique is the rectangular scan. Scan goes in the following order: first scan top three instruments from left to right or vice versa. Then drop to the bottom three and continue in the established direction to enclose the rectangle. The scan may also be initiated in the AI and continues in the rectangular path either in the clockwise or counter-clockwise direction. This scan is useful in situations, when rapid scan of all instruments is needed [9][35].



(a) Selective Radial Scan strategy. (b) Rectangular Scan strategy.

Figure 2.13: Instrument scanning strategies utilizing all instruments.

2.5 Longitudinal Basic Flight Maneuvers

First group of flight maneuvers are longitudinal maneuvers. Among longitudinal basic flight maneuvers are straight-and-level flight, climb and descent. In this type of maneuvers, heading of an aircraft is maintained, while the emphasis is placed on the pitch attitude control [11].

2.5.1 Straight-and-Level Flight

Straight-and-Level flight is one of the four basic flight maneuvers. During the maneuver, pilot should maintain desired altitude and heading. Straight-and-level flight perfection lies in the correct utilization of flight controls and flight instruments. The pilot also has to be able to make effective corrections of any deviations from desired airplane's behavior [11].

Pitch Control

To maintain level flight, that is maintaining constant altitude, proper pitch attitude has to be established. Direct indication of the pitch attitude is provided by the AI. Instruments providing indirect pitch attitude are ALT, VSI and [9].

Since the altitude should remain the same, change in altitude indicates change in pitch attitude. Direction of needles movement indicates whether the plane pitches up or down. Additionally, the rate of altitude change tells the pilot the amount of pitch angle [9].

Similar applies to the VSI. The Pilot is able to determine direction of the pitch from needle movement and it's amount by looking at rate of movement of the needle [9].

In case of the constant power and pitch attitude during the level flight, airspeed remains constant. As the pitch attitude changes, airspeed increases or decreases depending on the pitch direction. The ASI is, therefore, used as an instrument of an indirect pitch attitude indication in the straight-and-level flight [9].

Checking pitch as an indication of level flight however, is relevant only in case of constant airspeed. If further pitch adjustment is needed for some reason, the altimeter is used as a primary instrument, since it directly indicates the altitude which has to be maintained. The altimeter is therefore generally used as a primary instrument for the pitch control in the straight-and-level flight, while the couple of the AI and the VSI are supporting instruments [9].

Bank Control

Straight flight means to maintain constant heading. In order to maintain set heading, proper bank control should be provided, as the bank angle other than zero results in heading change. Here, as in case of the pitch, the bank angle is directly provided by the AI. It provides pilot instant information of bank attitude by placing pointer along the scale on the top of the instrument, and by showing the relationship of the airplane miniature with respect to the horizon [9].

Other two instruments, the HI and the TC (or the turn-and-slip indicator), are used as indirect indicators of the bank attitude. In case of the HI, rapid movement of the rose indicates high bank angle, whereas slow movement indicates low bank angle. Indirect indication of the bank attitude also provides the TC by turning the aircraft miniature into the appropriate direction of turn. This instrument is also able to provide an indirect information of the heading change since it detects both, roll and turn. The inclinometer is also capable of indicating change of the heading, as it indicates if there are any deviations from the desired heading due to the improperly trimmed aircraft [9].

In general, the HI is used as a primary instrument for the bank control in the straight-and-level flight, as it directly indicates if the desired heading is maintained. The AI and the TC are supporting instruments [9].

Power Control

Depending on the situation, primary instrument for the power control is selected. In the straight-and-level flight is in most cases used the ASI as the primary instrument for power control. In exceptional cases, when decreasing airspeed for instance, manifold pressure gauge is used as a primary instrument [9].

In a straight-and-level flight with a constant airspeed and power, the T-scan shows as an optimal scanning pattern [35].

2.5.2 Climb

Climb is a maneuver of increasing the altitude of an aircraft. This maneuver can be performed in two ways, maintaining constant airspeed or constant rate of climb. The airspeed is selected depending on needs of climbing performance. Overall, there are three general types of climb the pilot should be familiar with [11]:

- *Normal climb* — sometimes referred to as cruise climb, is type of climb performed at an airspeed recommended by the manufacturer. In this type of climb, for the pilot it is easier to see over the nose than in other types of climb. In the normal climb, the airplane flies at higher airspeed than in the best rate of climb.
- *Best rate of climb* — is the airspeed that allows the airplane to reach the desired altitude in the shortest possible time. It takes more time to reach the altitude when climbing either higher or lower airspeed. The best rate of climb is sometimes referred to as V_Y and it is the ratio of altitude h over time t:

$$V_Y = \frac{h}{t} \tag{2.1}$$

- *Best angle of climb* — is an airspeed which is the most effective in terms of a flight distance. In other words, the desired altitude is reached in the shortest distance. It is referred to as V_X and is expressed as the ratio of altitude h over distance s:

$$V_X = \frac{h}{s} \tag{2.2}$$

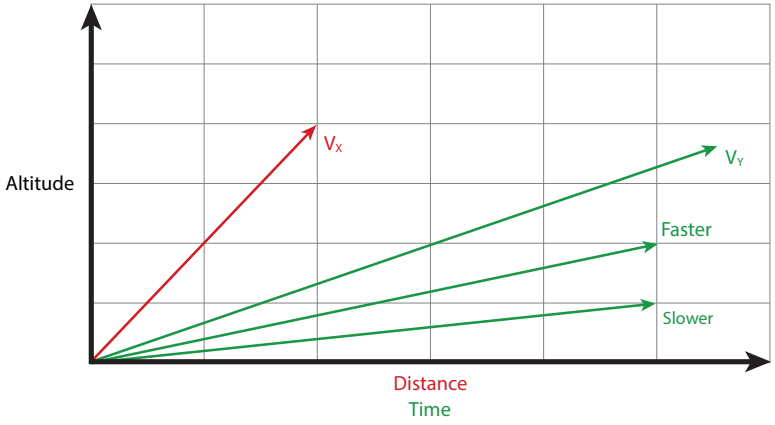


Figure 2.14: Visualization of the relationship between the best rate of climb and the best angle of climb.

Procedure of climbing maneuver is divided into three phases - (1) enter, (2) stabilized climb and (3) leveling off. When entering the climb, pitch attitude is established to reach predetermined vertical speed. For this purpose, the AI is used as a primary instrument. Engine instruments and the HI are primary indicators for power and bank control respectively [9].

In the stabilized climb, depending on the selected performance, primary instrument for the pitch control is selected. In case of stabilized constant airspeed climb, the ASI is selected. For the constant rate climb, primary indicator for the pitch control is the VSI, and the ASI becomes primary instrument for the power control [9].

As the pilot starts to leveling off the aircraft, the ALT is primarily used to control the pitch attitude, and the ASI to control power to establish desired airspeed [9].

2.5.3 Descent

Descent is the opposite maneuver to the climb, when the airplane altitude is decreased. It can be performed at variety of airspeeds and altitudes. As in the case of the climb maneuver, the pilot should be familiar with the following three types of descents [11]:

- *Partial power descent* — is type of descent performed with partial power. It is sometimes referred to as the en route descent. During this type of descent, airspeed, power and pitch attitude should be kept constant. Usually the rate of descent is maintained at 500 ft $\cdot$ min^{-1}.

- *Descent at minimum safe airspeed* — is descent type, at which airspeed is normally maintained no greater than 1.3 of the stalling speed. A steeper descent angle is usually maintained during this type of descent.

- *Emergency descent* — is a special type of descent with high drag and high airspeed, requiring a specific airplane configuration.

Same as the climb, descent is divided into three phases — (1) entry, (2) stabilized descent and (3) leveling off. The assignment of primary instruments is also similar. For the constant airspeed descent, primary instrument for the pitch control is the ASI, while in case of the constant rate descent, pitch is primarily controlled by the VSI.

2.6 Lateral Basic Flight Maneuvers

Second type of flight maneuvers are lateral maneuvers. The only representative of lateral basic flight maneuver here is level turn. In lateral maneuver, the main emphasis is placed on the roll attitude control [11].

2.6.1 Level Turn

Another basic flight maneuver each pilot should learn is level turn. Level turn is a maneuver, when the pilot changes airplane heading while maintaining its current altitude. To perform the turn, the pilot has to initiate banking by moving control stick either to the left or right, depending on the direction he wants to fly. If the pilot moves the stick to the right causing ailerons deflections, the right wing will be lowered and the airplane begins to turn to the right. Same applies for the opposite direction [9].

Ailerons are used in combination with rudder, in order to perform coordinated turn. Additionally, pilot should pitch up or add more power in order to maintain current level [9].

Turn types

Level turns are divided into three categories according to the angle of bank [11]:

- *Shallow turns* — with the bank angle less than 20 °. In most cases, when the pilot releases pressure exerted on ailerons control, the airplane tends to roll back to the bank angle equal to zero.
- *Medium turns* — with the bank angle between 20 ° and 45 °. When the pilot performs medium turn, he can release pressure on ailerons to neutralize them. With ailerons neutralized, the airplane should be maintaining constant bank angle without any control inputs.
- *Steep turns* — with the bank angle higher than 45 °. When airplane reaches bank angle higher than 45 °, it tends to continue in the banking even with ailerons neutralized. To stop banking, pilot must provide pressure on the control stick in the opposite direction.

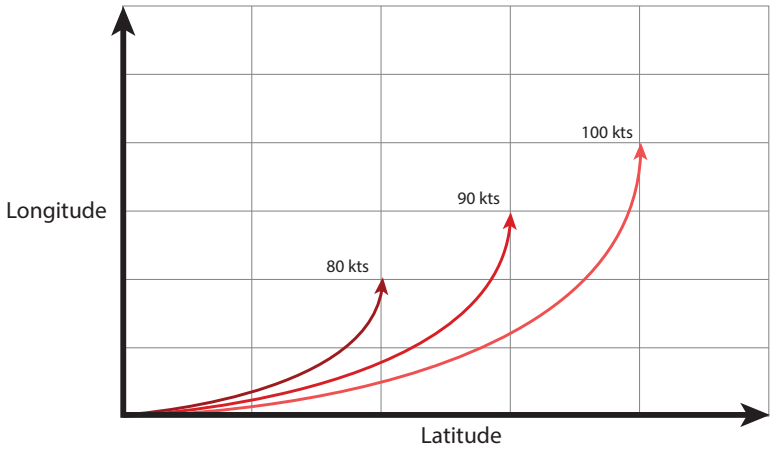


Figure 2.15: Change of turn radius with variable angle of bank and constant airspeed. Inspired from [11].

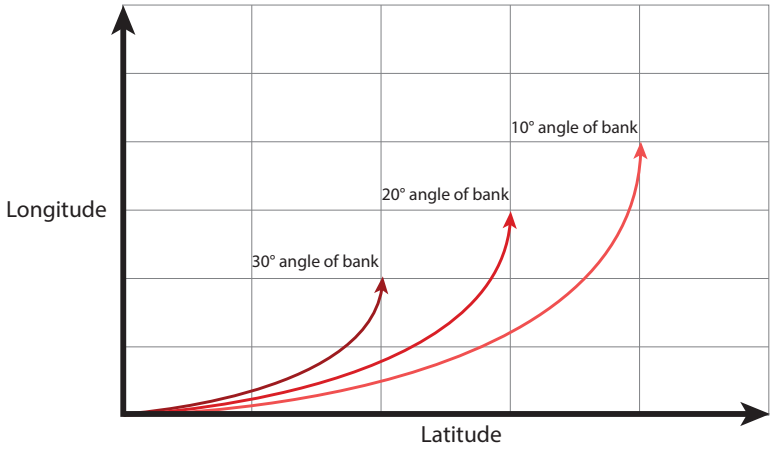


Figure 2.16: Change of turn radius with variable airspeed and constant angle of bank. Inspired from [11].

Important parameter of turn maneuvers is also turn rate as it directly affects turn radius. Turn rate tells the pilot how fast the airplane changes its heading. The turn rate is expressed by default in degrees per second. Amount of the turn rate is directly affected by a combination of the bank angle and the airspeed, where following rules are generally applied [11]:

- At constant airspeed, the rate of turn increases as the bank increases.
- At constant bank angle, the rate of turn decreases as the airspeed increases.

Common flight maneuvering is referenced to standard rate turn, when the aircraft performs the full 360° turn in 2 minutes. That equals to 3 °/s of rate of turn. Following rule of thumb is used, in order to get appropriate angel of bank for the standard rate turn θ_S [11]:

$$\phi_S = (\frac{\text{KTAS}}{10}) + 5 \tag{2.3}$$

where KTAS denotes TAS in knots.

Turns to Predetermined Headings

Turns of this type are established in the standard rate turn, and are maintained until desired heading is reached. Regarding the primary and supporting instruments, primary instruments are TC, ALT and ASI to control bank, pitch and power respectively. Small exception is the AI, which is considered as the primary instrument for the pitch control in establishing turn [35].

For the bank control, the supporting instrument is the AI, for the pitch control these are the AI together with the VSI, and engine instruments for the power [9].

Timed Turns

Timed turn is type of turn, when pilot practices his skills of performing accurate standard rate turns without reference to the HI. Based on given time information and direction, he should be able to determine target heading, as he knows how many degrees to turn while turning in standard rate. For instance, 15 second standard rate turn yields 45 ° turn [35].

In this case, primary instrument for the bank control is the TC. For the pitch and the power control, primary instruments are the same as for turns on predetermined headings. Furthermore, the AI is again primary instrument for the pitch control at the beginning of the maneuver [35].

Compass Turns

Purpose of compass turns are essentially the same as timed turns. Their purpose is to teach the pilot the correct utilization of instrument panel without the reference to the HI. Turning to the desired heading using primarily magnetic compass is not as easy as may seems at the first sight. During various maneuvers, behavior of the compass can be confusing to the pilot. Thus, it is necessary to get familiar with lags and a non-linear behavior of the compass. Set of primary and supporting instruments is similar to timed turns [35].

2.7 Combined Flight Maneuvers

Most common combined maneuvers performed in an air traffic are vertical turns including climbing and descending turns. In case of longitudinal and lateral maneuvers, the movement of the aircraft was controlled either in a horizontal or vertical plane. During combined maneuvers, the aircraft is controlled in both planes simultaneously.

2.7.1 Climbing Turn

Climbing turn is a combination of climb and turn maneuvers. This type of maneuver requires simultaneous control of multiple components, making this maneuver difficult to perform [11]. Due to its complexity, it provides suitable test of pilot proficiency.

Climbing turns should be initiated in way, that first comes climb and than turn. Alternatively, turn and climb are initiated simultaneously. For climbing turns applies the same as for level turns. To reach the most efficient rate of climb, it is recommended to establish shallow angle of bank [11].

2.7.2 Descending Turn

Descending turns are similar to climbing turns but in opposite direction using descending principles. First, pitch is established and then turn is initiated [9]. As the climbing turn, this type of maneuver is due to its complexity an optimal option for proficiency testing.

2.8 Basic Instrument Flight Patterns

In this section, multiple flight patterns are introduced, as an example of possible maneuver combinations to use in the design of the experiment scenarios. These patterns are commonly used in order to train basic flight maneuvers in the instrument flight. These patterns, depending on the complexity, involve multiple flight maneuvers and their variations, introduced in the previous sections. Following list is just a selection of many possible patterns:

- *Procedure Turn* — is a flight maneuver, that is used to reverse flight direction. They are used especially in cases, when intercepting the inbound course is required, prior to the final approach fix during the approach procedure. Rate of turn should be held at 3 °/s. The entire maneuver is flown in level flight. Overall, three common types of procedure turns are used: standard 45° procedure turn, 80/260 procedure turn, and teardrop pattern. In addition, three teardrop patterns are defined, differing in degrees of the initial turn.

- *S-Turns*: — is a series of level turns, originally designed as a type of ground reference maneuvers. This is due to an option of a pilot to use outside references in order to perform turn with a constant radius. However, S-turns can be also included in the instrument flying training. In combination with configuration changes, it can be used to speed up scanning rate, and to improve power and altitude control required during instrument approaches. It is performed in such a way, that while the pilot gradually alternates left and right turns, he also during each turn adjusts an airspeed and changes aircraft configuration. Basic S-Turns pattern is depicted in figure 2.17.

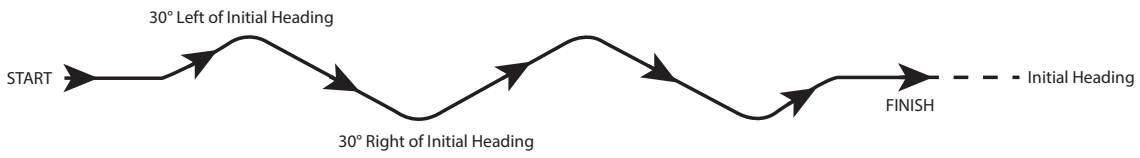


Figure 2.17: Basic S-Turn pattern [35].

- *Vertical S's* — combines climbs and descents together into series of these two maneuvers. Different rates of climb/descent are applied to each pair of these maneuvers. Starting with the rate of 500 ft $\cdot$ min^{-1}, pilot gradually performs four consecutive climb-descent procedures, ending at 200 ft $\cdot$ min^{-1}. Each climb and descent lasts for a one minute. This type of pattern is shown in figure 2.18.

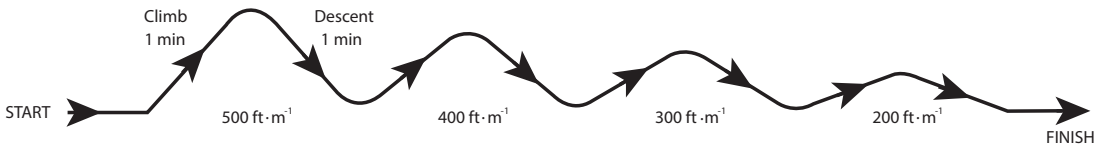


Figure 2.18: Vertical S's pattern [35].

- *Racetrack Pattern* — is a pattern combining 180° standard-rate turns and timed straight-and-level flights. In an air traffic, this type of pattern is usually used as a holding pattern. For training purposes, it is possible to combine two racetracks together into a single pattern. In this combination, each of two racetracks would be flown in a different direction. Combined pattern of left and right hand racetrack is depicted in figure 2.19.

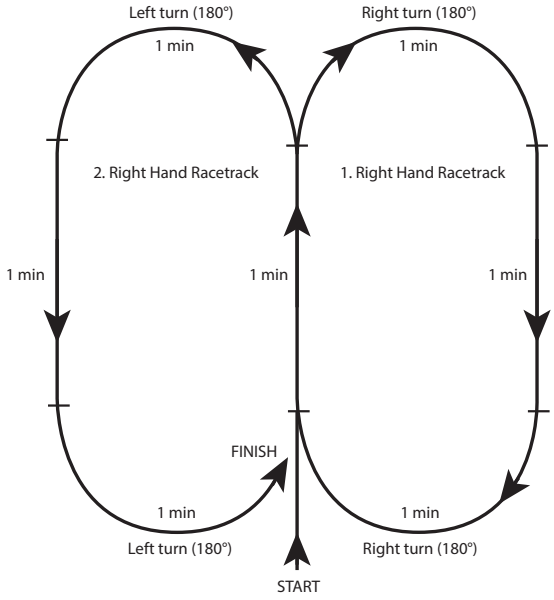


Figure 2.19: Combined racetrack flight pattern.

Chapter 3

Gaze Classification Techniques

As this thesis deals with a classification problem based on visual scanning strategies, theory of gaze-tracking analysis should be provided. This includes introduction into the basic terminology of the gaze-tracking, followed by definition of multiple metrics commonly used in gaze analysis. At the beginning of this chapter however, related work on the topic of human scanning behavior analysis is presented first.

3.1 Related Work

There has already been many studies on the topic of human scanning behavior. Some of these studies focused on the finding of relationship between the scanning strategies and the level of proficiency in a specific domain. Naturally, one could say that the way human uses eyes indicates the strategy and patterns in obtaining the information related to some task. If the human is a domain expert, he/she knows where to focus her/his gaze attention, to obtain necessary information in the most efficient way. Thus, she/he uses some strategy, that is most likely efficient in obtaining the information to reach high task performance.

Results of several studies support the idea that an efficient scanning strategy lead to a better task performance. One of such studies [48] came up with a testing of effects of scanning strategies on performance in videogames. There were two groups of players, where one group received training on visual scanning patterns, while the other group did not. As a result, players trained on efficient scanning strategies outperformed players with random or no training. Additional knowledge gained from the result of this experiment is, that scanning patterns can be trained, leading to a better performance. Related studies focusing on different domains such as medicine, chess, sport or gun firing [36][44][51] came to similar conclusions.

Concerning an aeronautical domain, there has been multiple studies showing that scanning behavior of the pilot is an indication of different proficiency levels [6][20][21][32][53]. Some of them also revealed, that experienced pilots use shorter dwells and use higher frequency of visiting different instruments in the cockpit [3][31][32]. Results of the [3] also showed, that expert pilots are capable of flexible adaptation of their scanning patterns on changing task demands. Another study [26] also came to the conclusion, that as opposed to novices, experienced pilots have more structured patterns. According to results of related research focusing on the relation of scanning strategies and pilot expertise [32], scanning patterns are more complex and elaborate in case of expert pilots. Results also showed, experts tend to have better distribution of attention than novices.

3.2 Gaze Measure Theory

Prior to metrics description, brief introduction of measured features during scanning behavior analysis should be provided. There are numerous features which are from a statistical point of view interesting indicators of the internal human state. As has been mentioned in the previous section, these features, for instance, can distinct between levels of expertise in specific domain, or might change in various conditions.

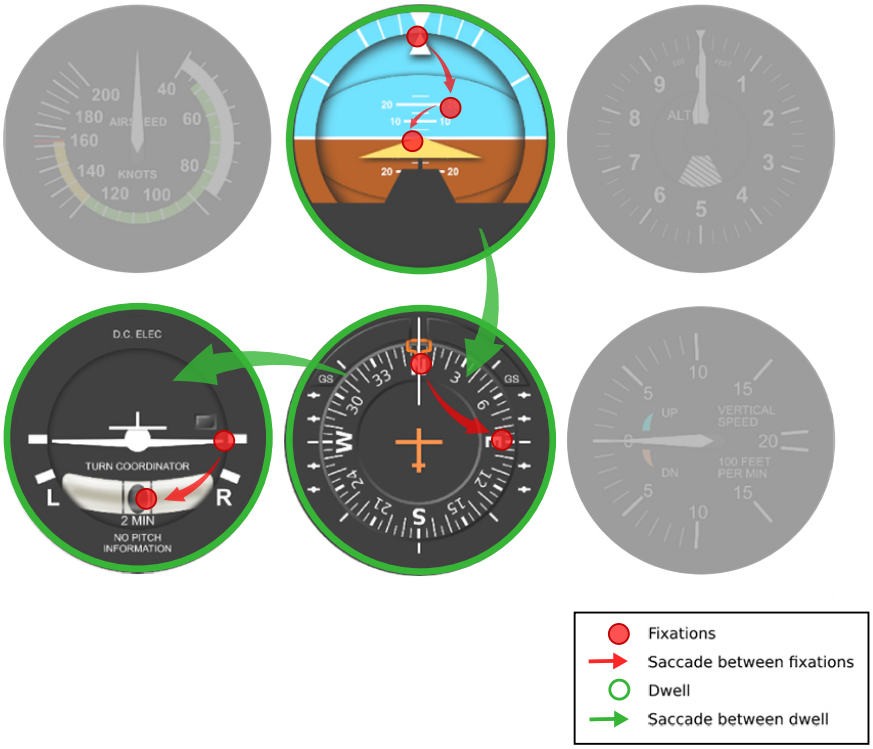


Figure 3.1: Graphical representation of the relationship between fixations, saccades and dwells.

First basic term in the gaze measure domain is a fixation. Fixation or visual fixation is a period, when the human eye is focused on a single location. During these fixations, the visual information is extracted from the fixated location. In gaze analysis, interesting features are fixation location and fixation duration [43]. According to relevant study, mean fixation duration, depending on the task, ranges between 200-400 ms, while the shortest measured duration has been around 40 ms and the longest up to 800 ms [43].

Human eye is capable of basically two types of movements. The first type is called saccadic movement. Saccade is a rapid simultaneous movement of both eyes between two consecutive fixations, reaching velocity of up to 500 °/s. This makes of the saccade one of the fastest movements of a human body. During saccades, the eye does not retrieve any visual information. This phenomena is called saccadic suppression. As in the case of fixations, we can also measure saccade duration, however this information is usually not in the center of interest. Interesting parameter of the saccade movement could be saccade amplitude, which is the angular distance the eye travels in the direction of the movement. Another type of movement is called smooth pursuit. It is a movement, when human eyes

remain aligned with the focused moving object, resulting in a smooth eye movement [43]. From the perspective of pilot scanning behavior analysis, saccades are more interesting since they are able to distinguish fixations between different instruments in the cockpit.

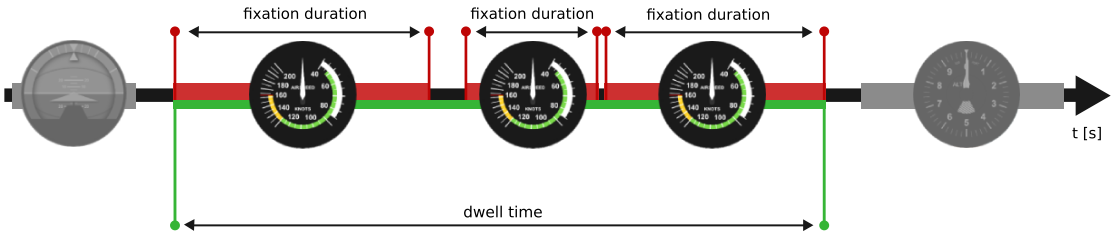


Figure 3.2: Graphical representation of the fixation duration and dwell time.

When dealing with fixations, we often want to know, what is the subject of gaze interest. For instance, in terms of pilot visual behavior, we want to know what instrument she/he is looking at. Typically, the object on which the eye is focused, occupies some area of the visual field. In addition, the object can be part of some group of related objects occupying even larger area. In general, regions of visual field, where we want to know if eyes are focused on that area, i.e. if the fixation is inside the area, are called Areas of Interests (AOI) [14]. During the gaze measurement, we can register multiple consecutive fixations in that area. In this case, group of these fixations are usually referred to as dwell. Interesting factor widely used in the recent studies on scanning behavior is dwell time. Dwell time is an information of what is the amount of time human have spent looking at a particular AOI. Graphical representation of both, fixation duration and dwell time, can be seen in figure 3.2. Note that individual fixations do not have to follow each other exactly, but there is a gap between them. Another valid numerical measures in the scanning behavior analysis are frequencies of fixations and dwells [14].

When particular dwells are observed, series of all dwells is called visual scanning. Having such visual scanning, one could be interested into finding visual scanning patterns, i.e. determining scanning strategies revealing many interesting characteristics about the observed person. Under the term visual scanning patterns we understand repeated sequences of dwells in the visual scanning.

3.3 Basic Eye Movement Metrics

Now, when the brief introduction into the basics of gaze measuring has been provided, it is possible to begin with the description of the state of the art of scanning behavior metrics used in related studies. This section in particular, is dedicated to relevant basic measures which can be obtained from eye movement measurements. More complex metrics focusing on the study of patterns will be described in the following section.

3.3.1 Dwell Time

Dwell time is the most common measure in the relevant studies. It tells us, what is the total time spent in some region for a single dwell. In the cockpit, each region would typically represent a single, or group of related instruments. In addition to the cockpit instruments,

area of windscreen is usually also included. As has been mentioned, dwells are composed of multiple fixations, thus it is relevant, that multiple studies analyse fixations durations as well. For the sake of simplicity however, only dwells are considered in this chapter.

Dwell time (or dwell duration) proved to be a good indication of domain expertise and mental workload changes. In the 3.1 has been mentioned, that studies dealing with the finding of relationship between the visual scanning and pilot expertise found, that shorter dwell times is the sign of an experienced pilot behavior [3]. Dwell time also vary across different instruments [34] and over different phases of flight [27].

Researches of National Aeronautics and Space Administration (NASA) dedicated to the workload also discovered the fact, that dwell time increases with the increasing task difficulty and workload [18].

3.3.2 Proportional and Average Dwell Time

Another valuable property of scanning behavior we can obtain from the gaze data, are proportional dwell times for all AOIs. To obtain Proportional Dwell Time (PDT), or simply percentage of the area utilization, we simply sum all dwell times of a particular region, and divide it by the sum of all dwell times [14] of all regions. By summing all dwell times of a particular region together, we obtain total dwell times, i.e. amount of time the person have spent looking at each region in total, which sometimes might also be an interesting value for the analysis.

Having the PDT, we can observe what is the most important and valuable source of information for the person in a given task. In terms of flight tasks, it was discovered that percentage of instruments usage vary across different phases of flight, since each type of maneuver requires slightly different instrument information processing [7]. Another interesting point that was found observing PDT in the related study is, that experts have different patterns in PDT than novices. Same study also found the relationship between the PDT and fuel consumption [31].

Sometimes we can be interested into average dwell time the pilot have spent looking at each instrument. Average dwell time is simply obtained by dividing the total dwell time of each instrument by the number of dwells of respective instrument.

3.3.3 Dwell Frequency

Dwell frequency is another interesting feature in the eye measurement. It describes how often particular AOIs are visited [14]. In relation to the pilot proficiency, correlation between the expertise and dwell frequency was found. Relevant studies discovered, that expert pilots have significantly higher dwells frequency than novices. One study also showed different frequencies across different instruments [3]. Relationship between the performance and dwells frequency was also observed [26].

3.3.4 Duty cycle

Pennington (1979) in his study on pilot scanning behavior in instrument flight, used duty cycle as an another metric for the purpose of scanning behavior analysis for various flight maneuvers. Having N dwells on particular instrument in the whole run, duty cycle is defined as a ratio of i-th dwell time t_i on that instrument, and period p_i between the i-th dwell and next dwell (returning) on that instrument: $\frac{t_i}{p_i}$. Overall, for N dwells on some

instrument, $N-1$ duty cycles in total are acquired. Mean duty cycle D_m of some instrument is obtained as follows [40]:

$$D_m = \frac{1}{N-1} \sum_{i=1}^{N-1} \frac{t_i}{p_i} \tag{3.1}$$

while holds $p_i - t_i > 0$. Results of the study showed, that the duty cycle is, as in the case for dwell time, different for each instrument. Additionally, an idea was proposed, that low duty cycle of some instrument is an indication of low need for an information of that instrument and vice versa [40].

3.4 Scanning Pattern Metrics

Simple statistical metrics introduced in the previous section, are able to describe some characteristics on the high-level scope. However, we are usually interested in more complex features of the gaze, such as visual scanning strategies or cognitive state of the person. In this domain, basic metrics are not able of such a description. To describe these features, we have to use more complex metrics, than just looking at statistics. Many related studies did come up with various approaches to define these complex characteristics of the gaze. In general, three different classes of metrics are available. Each class of metrics looks at the problem from a different perspective. Together, they form relatively powerful view on the visual scanning behavior of the human. These classes are:

- Transition Matrices — are intuitive tool used to discover importance of individual objects of the observed scene and what are the relationships between them. Multiple types of transition matrices can be created, each relating to different aspect of the visual scanning [32].
- Sequence Analysis — is a group of metrics used to describe the complexity and orderliness of the visual scanning. Typical representative of this group commonly used in related studies is entropy, describing how ordered and complex of each individual the visual scanning is. Similar metric utilized for this purpose is Lempel-Ziv Complexity (LZC), indicating visual scanning complexity based on the number of unique sequences [32].
- Attentional Modes — describe the way how each individual scans the environment, i.e. which attentional mode is mainly utilized. Two modes are defined — focal and ambient. While the ambient mode is characterized by low fixation durations and large saccade amplitudes, the focal mode is characterized with the exact opposite. Metric used for this type of analysis is called $\mathcal{K}$ *coefficient*. [29]

3.4.1 Transition Matrices

In this chapter, as advanced metrics only transition matrices are described in more detail, as they are the only metric utilized in this thesis. They proved to be robust and flexible metric independent on the flight maneuver type. Transition matrices use properties of Markov chains. Detailed description of the Markov chains will be provided in the chapter 4.

Transition matrix

In general, transition matrices are used to describe the relationships between particular states in terms of dependency of the current state upon previously visited states. In our case, states are represented by AOIs. Transition matrices are commonly used as a second-order Markov transition matrices. This means, that current state depends solely on the previously visited state. Thus, the second-order transition matrix has two dimensions. If we extend the transition matrix by the third dimension, current state depends on the two previously visited states [37].

$$\begin{array}{cc} & \text{to } j\text{-th AOI} \\ & \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \text{from } i\text{-th AOI} & \begin{array}{c} 1 \\ 2 \\ 3 \end{array} \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix} \end{array} \tag{3.2}$$

Transition matrix is an $N \times N$ matrix, where N is the number of AOIs. Equation 3.2 shows the general example of a transition matrix for 3 AOIs. Value of each cell contains the probability of transition from the current AOI to one of the possible AOIs. The probability is computed as:

$$P_{ij} = \frac{n_{ij}}{\sum_{j=1}^{N} n_{ij}} \tag{3.3}$$

where n_{ij} denotes frequency of transition from i-th AOI to j-th AOI. Sum of each row is equal to 1. Depending on the application, self transitions in the visual scanning are sometimes omitted, thus, values on the diagonal are equal to zero.

Transition matrices are frequently used tool for finding scanning patterns. For example, it was discovered, that transition matrices are different for various flight maneuvers [13]. This might be limitation in determination of any general scanning strategy for a complex task. There was a study, which on the basis of transition matrices built Hidden Markov Models for different flight tasks [22]. Interesting discovering was made by related works, that transition matrices are able to distinguish between expert pilots and novices.

Transition occurrence matrix

Slightly different approach of study of visual scanning patterns used Fitts et al. (1949) in their studies, dedicated to finding an optimal cockpit instrument arrangements. Among other statistics, they were also interested in occurrences of transitions between individual instruments, their percentage utilization in particular [12].

To obtain these percentages, number of occurrences of all transitions (i.e. dwell transition frequencies) between all instruments ($N \times N$ transitions in total) are needed first. Then, each frequency is divided by the number of all transitions in the visual scanning. When analyzing these transitions, they did not considered self transitions to the same instruments. In this manner, probability of occurrence of a particular one-way transition is obtained [17]:

$$P_{ij_1} = \frac{n_{ij}}{\sum_{i=1}^{N} \sum_{j=1,s.t.i\neq j}^{N} n_{ij}} \tag{3.4}$$

If we put all probabilities into the matrix, we would get matrix with probabilities of transition occurrences, thus transition occurrence matrix. Sum of all elements of the matrix is equal to 1. Note, that as self transitions are not considered, this matrix have zeroes on its diagonal.

In the study however, they utilized two-way transitions, where they considered transitions in both directions. Thus, we get symmetric matrix. We obtain the probability of a two-way transition occurrence by using the following equation [17]:

$$P_{ij_2} = \frac{n_{ij} + n_{ji}}{\sum_{i=1}^{N} \sum_{j=1,s.t.i\neq j}^{N} n_{ij}} \tag{3.5}$$

Transition matrix density

Transition Matrix Density (TMD) is a metric able to describe level of complexity and efficiency of the scanning pattern of a pilot with a single value. This metric was first introduced in the study dedicated to evaluation of computer interface [15]. The density is computed as a count of all transitions occurred at least once, divided by the total number of all possible transitions, that is $N \times N$:

$$\text{TMD} = \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} c_{ij}}{N^2} \tag{3.6}$$

where

$$c_{ij} = \begin{cases} 1, & \text{if } n_{ij} \geq 1 \\ 0, & \text{otherwise} \end{cases}$$

If the transition matrix has high density, it could be an indication of a dispersed and lengthy scanpath, while sparse transition matrix may indicate an efficient search [15]. However, sparse matrix may also indicate some negative aspects. For example, when a novice pilot is excessively engaging his visual attention on a single instrument [52]. The TMD is valuable in situations, when all AOIs are not used in measured epoch, otherwise the value might not be very useful when comparing with other pilots [32].

Chapter 4

State Classification Techniques

Since the goal of this thesis is to design and implement basic flight maneuver and pilot proficiency classifier, theoretical background of the machine learning techniques, which are heavily utilized for this purpose,is introduced. In the first part of this chapter, a brief description of the machine learning will be provided together with general classification of the learning techniques followed by theoretical basics of classification. Special attention in the remaining part of this section is dedicated to Support Vector Machine (SVM) algorithm and Hidden Markov Model (HMM), based on which the classification framework is designed and implemented.

4.1 Machine Learning

Machine learning is one of the many subfields of computer science. It deals with building algorithms solving practical problems. These algorithms improve over time based on collected data, called training data, containing examples of some phenomena associated with the respective problem.

Result of the machine learning process can be understood as a function $f(x)$ which, based on input x, generates output y. For instance, x is an image of either dog or cat, and the function f generates an answer y to a question: "Does the image contains dog or cat?". All examples contained in respective datasets can come from nature, be handcrafted by humans or generated by another algorithms [4][8].

Generally, machine learning algorithms are divided into four classes: supervised, unsupervised, semi-supervised and reinforcement learning [8].

4.1.1 Supervised Learning

Algorithms built upon supervised learning principles are such algorithms, which use training datasets consisting of N pairs $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$. Each pair consists of feature vector $\mathbf{x}_i$ and label y_i. Feature vector is a vector of $1, \ldots, D$ features, which describe the example of the phenomena. For instance, in case of some person, these features could be age, height, weight, or sex. In case of an image, feature vector would consists of all image pixels. Each feature vector is coupled with label y_i telling the algorithm, what is the desired output on the input $\mathbf{x}_i$. Usually, label is value from a finite set of classes or real value. It can be represented by more complex forms however [8].

Because the learning process is based on known classes, typical problem solved by the supervised learning is a classification. Examples of classification are prediction of an animal

from picture or level of pilot proficiency. Another type of problem solved by the supervised learning is regression, where the output is some real or continuous value. Common regression problem is a stock value prediction [8].

4.1.2 Unsupervised Learning

Unlike the supervised learning, unsupervised learning algorithms do not use labels of feature vectors, thus the desired outputs have to be discovered by finding patterns and grouping similar examples together. The dataset $\{(\mathbf{x}_i)\}_{i=1}^{N}$ consists of N feature vectors, which are transformed either to a single value or another vector [8].

Unsupervised learning includes algorithms, which have found utilization in solving problems of clustering, density estimation or dimension reduction [8].

4.1.3 Semi-Supervised Learning

Semi-supervised learning is a combination of both, supervised and unsupervised learning. The dataset of the algorithm is comprised of labeled and unlabeled examples. Typically, there is much more unlabeled samples than labeled present in the dataset. This brings more information about the problem leading to higher learning accuracy [8].

4.1.4 Reinforcement Learning

Reinforcement learning is an another machine learning approach of solving some problem. As opposed to supervised learning, reinforcement learning algorithms do not accept any optimal outputs, but they rather discover them by trial and error procedure [8].

The basic idea of the reinforcement learning lies in maximizing rewards of the algorithm for taking actions in particular states of the environment it is located in, in order to achieve as optimal solution of the problem as possible. The goal of the algorithm is to learn the policy, which is the function that takes state (feature vector) as an input and outputs the most optimal action to take based on the current experiences. Each action the algorithm takes in some particular state, is evaluated by different reward and leads to the discovery of new states. Initially, actions are taken totally random, however, over time the algorithm tends to use known actions with high rewards more frequently. These two attitudes are called exploration and exploitation, and it is crucial not to focus on either of them too much [8].

Reinforcement learning algorithms are used in many disciplines such game theory, control theory, statistics, multi-agent systems and much more [8].

4.2 Classification Theory

As has been mentioned in the previous section, classification is a typical problem solved by supervised learning. The assumption in solving classification problem is, we exactly know what are the classes we want input data to classify into. These classes fall into a finite set of possible classes. Depending on the size of the set, we define the type of classification problem. If data are classified only into two classes, we speak of binary (or binomial) classification. Classification into three or more classes is called multiclass (or multinomial) classification [8].

4.2.1 Learning Process

The core idea of the classifier learning process is to take training and find appropriate rules to build knowledge from them, so the classifier is able to identify class of some new unlabeled example. In general, the supervised machine learning process is divided into three phases — (1) training phase, (2) validation phase, and (3) testing phase [50].

Training Phase

During this phase, parameters of the classification model are being estimated, approximated and optimized based on the training set consisting of labeled data. To reach an optimal generalization, the training set should be large enough to cover the whole feature space with an appropriate amount of examples. Training set should represent the majority of the whole dataset. Usually, it takes 70 % or more out of the whole dataset. Exact percentage depends on the size of the available dataset [50].

Validation Phase

This phase provides the feedback on the classification model effectiveness. The goal is to evaluate how well the model performs on unknown data in the process of model selection, i.e. hyperparameters tuning. Hyperparameters are parameters of the model which are not object of training process, as they are used to control the learning process itself [8]. For the purpose of validation, validation set is used. Multiple model validation techniques can be selected. The selection depends on preferences and dataset availability [50].

Most common technique is called Hold-Out. Using this approach, the dataset is divided into three groups — training, validation and test set, usually in ratio 70:15:15. Model is validated on validation set consisting of unknown data. Another option is to divide dataset only into training and test dataset, and model is validated on portion of randomly selected samples of the training set. Validation and test sets do not affect training phase directly, thus Hold-Out. Hold-Out technique is sensitive to a dataset division, so model can become biased towards training set. Moreover, this method is not suitable for small dataset [50].

Another technique is called cross-validation, where the model is trained and validated on combination of known and unknown data. There are multiple cross-validation algorithms. Most common cross-validation technique is called K-Fold Cross-Validation. Training set is split into k groups, where k-1 groups are used to train, while the rest is used for validation. Model score is then recorded, and the process is repeated, until each group is used as a validation set. At the end, average score of all k recorded scores is obtained. This approach is used especially in cases, when only a dataset of limited size is available [50].

If the result of the is not satisfactory, model parameters are not optimized well enough, thus, another round of training process is needed. During this phase, over-fitting problem can also be controlled and corrected [50].

Testing Phase

In this phase, the final model is evaluated on the independent set called training set. This set was not used either during the training nor the validation phase. So called qualitative measures are used to evaluate the model, such as accuracy, sensitivity, specificity and precision [50].

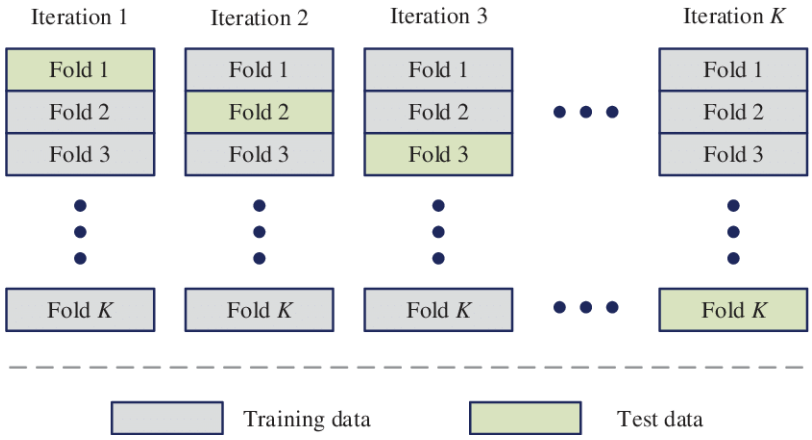


Figure 4.1: Illustration of the K-Fold Cross-Validation technique [45].

4.2.2 Learning Algorithms Families

Overall, there are two families of supervised learning algorithms: model-based and instance-based algorithms. Model-based algorithms, representing the larger of two groups, are such algorithms, which build models and optimize their parameters from available training data. Effectiveness of a classifier depends solely on the setting of these parameters. On the other hand, instance-based algorithms do not build any model and optimize its parameters. Instead, the model is represented by the dataset itself. In terms of classification, a typical representative of the model-based algorithm group is an algorithm called Support Vector Machine (SVM). Example of an instance-based classification algorithm is the K-Nearest Neighbours (KNN) [8].

4.3 Support Vector Machine

Support Vector Machine (SVM) is a highly popular algorithm (probably to the most popular of all machine learning algorithms) used for classification and regression problems. They are known for their simplicity, high robustness and ability of an efficient generalization. SVMs are capable of dealing with outliers. Another advantage is their ability to account for high dimensional data. In fact, there is a whole group of related algorithms for classification and regression referred to as Support Vector Machines (SVMs) that have found utilization in many applications such as handwriting identification, stock and weather prediction, or video and audio processing [1].

The SVM in its simplest form is capable of binary classification based on a linear classifier. However, many other techniques were also introduced, so linearly non-separable data can be classified or multiclass classification can be performed [8].

4.3.1 The Basic Idea

Given a set of data points, where each sample belongs to one of two classes, the SVM tries to find such a boundary in an D-dimensional space, that would distinctively separate these data points. Then, when new data point is observed, the SVM is capable of correct classification. The boundary is referred to as a hyperplane. It separates the D-dimensional space into two regions, where D corresponds to the number of features. For instance, in case of 2-dimensional space, the hyperplane is a simple line [8].

Actually, there exist multiple hyperplanes separating the data. The SVM is capable of finding such a hyperplane, that maximizes the so called margin, i.e. the distance between two data points from each class. This is the advantage over other algorithms such as neural networks, which are only capable of finding local minima. Thus, for the same dataset, the resulted hyperplane would be different every time [24]. Maximal margin leads to a better generalization. The margin is determined by the position of so called support vectors. These are points from each class, which are closest to the hyperplane. Deleting some support vector would result into a hyperplane change. The classifier based on the maximum margin is called Maximum Margin Linear Classifier (MMLC) [1] [24]. A principle of the hyperplane, maximized margin and support vectors is depicted in figure 4.2.

If the data cannot be separated with any hyperplane in the D-dimensional space by default, a so called "kernel trick" technique is applied, which maps data into higher-dimensional space, where a linear classifier can be applied [1].

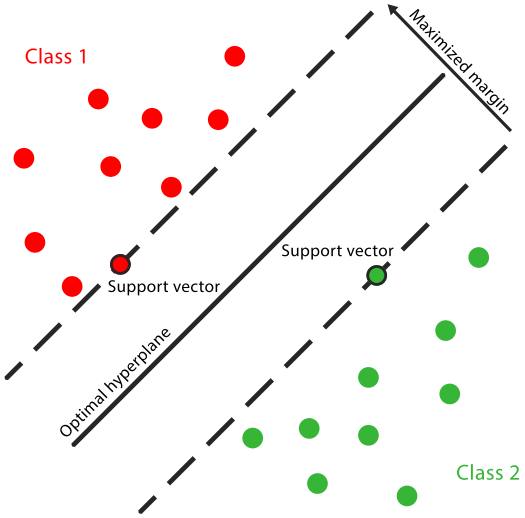


Figure 4.2: Support Vector Machine — Hard-Margin Classifier. Inspired by [33].

4.3.2 Linear Classifier

Linear classifier is the simplest SVM classifier example. Here we assume, that data in original D-dimensional space can be linearly separated. We have a set of N data points $\{(\mathbf{x_i}, y_i)\}_{i=1}^{N}$, where y_i belongs to one of the classes $\{-1, 1\}$.

Each $\mathbf{x_i}$ has D features. The goal is to find such a hyperplane, that is able to linearly separate both groups with maximum margin. The hyperplane can be expressed by the following equation [8]:

$$\mathbf{w}^T\mathbf{x} - b = 0 \tag{4.1}$$

where $\mathbf{w}$ denotes normal vector to the hyperplane and $\mathbf{x}$ is an input feature vector. Label of some feature vector $\mathbf{x}$ is defined as [8]:

$$y = \text{sign}(\mathbf{w}^T\mathbf{x} - b) \tag{4.2}$$

Hard-Margin

Constraints for finding the hyperplane with a maximum margin, i.e. finding optimal values of $\mathbf{w}$ and b, are following [8]:

$$\begin{aligned} \mathbf{w}^T\mathbf{x}_i - b \geq +1 \quad &\text{if } \; y_i = +1 \\ \mathbf{w}^T\mathbf{x}_i - b \leq -1 \quad &\text{if } \; y_i = -1 \end{aligned} \tag{4.3}$$

or simply $y_i(\mathbf{w}^T\mathbf{x}_i - b) \geq 1$, where, from the geometrical perspective, $\mathbf{w}^T\mathbf{x} - b = 1$ and $\mathbf{w}^T\mathbf{x} - b = -1$ are equations of two parallel hyperplanes defining boundaries for both classes. A space between these two hyperplanes is the margin we want to maximize. The distance between these two is equal to $\frac{2}{\|\mathbf{w}\|}$. Note that it is also the distance between the closest points of the two classes [8].

So the subject to machine learning here is the optimization problem of maximizing objective function [1]:

$$J(\mathbf{w}, b) = \frac{2}{\|\mathbf{w}\|} \tag{4.4}$$

subject the constraints from equation 4.3. This can be also viewed as a minimization problem by transforming objective function into [1]:

$$J(\mathbf{w}, b) = \frac{1}{2}\|\mathbf{w}\|^2 \tag{4.5}$$

This is the example of the SVM linear classifier referred to as hard-margin SVM. It does not allow to have any data points within the margin, so they can be easily separated by a linear function. This is also the example of the SVM showed in figure 4.2. However, in real world this is not always the case. For this purpose, the so called soft-margin SVM has been introduced [1].

Soft-Margin

In real world, there is usually some noise present in the data. This represents a problem for hard-margin classifier presented above, as the data cannot be linearly separated. Instead, we allow some data points to be within the margin [1].

First, a hinge loss function is introduced [8]:

$$(\mathbf{x}_i, y_i, g(\mathbf{x}_i)) = \max(0, 1 - y_i \cdot g(\mathbf{x}_i)) \tag{4.6}$$

where $g(\mathbf{x}_i)$ is the function of the hyperplane from 4.1. If constraints from 4.3 are satisfied, result of the hinge loss function is equal to zero. However, if these constraints are

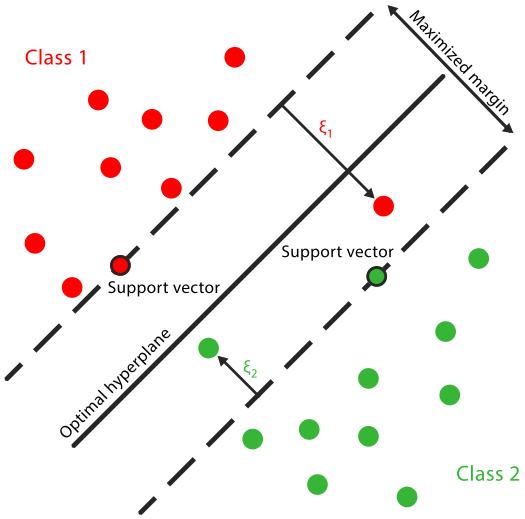


Figure 4.3: Support Vector Machine — soft-margin classifier [33].

violated, result of the function is proportional to the distance on the wrong side of the class boundary the input feature vector belongs to [8]. For each x_i, the result of the hing loss function is represented by a new variable ξ_i called the slack variable [1].

The optimization problem here is expressed as a minimization of the following objective function [1]:

$$J(\mathbf{w}, b, \xi) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i \tag{4.7}$$

subject to $y_i(\mathbf{w}^T\mathbf{x}_i - b) \geq 1 - \xi_i$ and $\xi_i \geq 0$. C represents the regularization parameter, determining the trade-off between the number of misclassification and the size of the margin. C parameter value depends on the preferences for the optimization problem. If the value is lower, margin gets bigger, leading to an increased number of misclassifications. Otherwise, if the value of the parameter is increased, more emphasis is placed on correct classifications, because margin gets tighter [1].

Illustration of the soft-margin classifier can be observed in figure 4.3.

4.3.3 Kernel Classifier

In case data points are not linearly separable, no hyperplane in the original D-dimensional space can be found, which could guarantee an optimal generalization. For this reason, a kernel trick is used. It transforms data in higher dimension, so they can be eventually separated [8]. The effect of the kernel trick is depicted in figure 4.4.

Note that the data in the original space cannot be linearly separated. After the kernel is applied, data are transformed into higher dimension, where data can be easily separated. A new space is called the feature space [8].

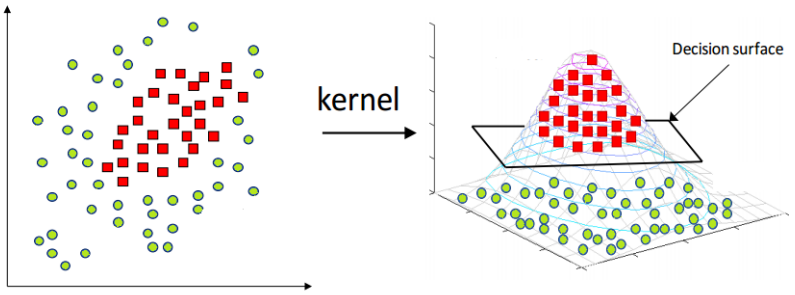


Figure 4.4: Effect of the kernel trick application on non-separable data [49].

4.4 Markov Chains

In terms of the system's state classification, Hidden Markov Model (HMM) is one of the most common technique used today to solve such problem, especially, when no knowledge of the internal representation of the system is present. They are well suited for the processing of sequential data and they proved to be very useful and robust in pattern recognition applications. Most common applications, in which HMMs are used are speech recognition, gesture recognition, bioinformatics, stock predictions, etc. [42].

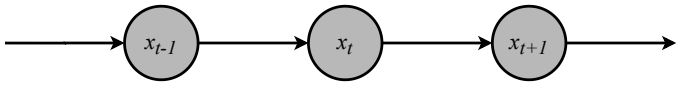


Figure 4.5: Graphical model of the first-order Markov Chain.

The Hidden Markov Model is an extension of a first-order Markov Chain, which is a stochastic model describing randomly changing system. First-order Markov Chain models the system as a system of N states $S = \{S_1, S_2, \ldots, S_N\}$, which at time t is being located only at one of these states, while it is assumed, that each next state at time $t+1$ depends only on the current state. This assumption called Markov Property is formally described as [42]:

$$P(x_t|x_{t-1}, x_{t-2}, \ldots, x_1) = P(x_t|x_{t-1}) \tag{4.8}$$

where x_t denotes system's state at time t. Graphical representation of the model is depicted in figure 4.5. If we consider that right-hand side of equation 4.8 is independent of time, transition matrix $\mathbf{A} = \{a_{ij}\}$ can be created, where [42]:

$$a_{ij} = P(x_{t+1} = S_j|x_t = S_i), \qquad 1 \leq i, j \leq N \tag{4.9}$$

with the following properties [42]:

$$a_{ij} \geq 0 \tag{4.10a}$$

$$\sum_{j=1}^{N} a_{ij} = 1 \tag{4.10b}$$

Therefore, the transition matrix $\mathbf{A}$ tells us what are probabilities of transitions between each pair of all possible states of the system. Additionally, if assumption denoted in equation

4.8 is taken into account, joint distribution of a sequence $x_1, x_2, \ldots, x_T$ of T observations is given by [42]:

$$P(x_1, x_2, \ldots, x_T) = P(x_1) \prod_{t=2}^{T} P(x_t|x_{t-1}) \tag{4.11}$$

where in this case, observation sequence representing the output of the model is a sequence of states of the system over time. Note that while conditional probabilities in the product in equation 4.11 can be obtained from $\mathbf{A}$, marginal probability of the initial state $P(x_1)$ can not however. Thus, the initial state probability distribution matrix $\boldsymbol{\pi} = \{\pi_i\}$ has to be defined, where [42]:

$$\pi_i = P(x_1 = S_i), \qquad 1 \leq i \leq N \tag{4.12}$$

Basic Markov chain assumes, that system's states can be directly observed. In real world however, this is not always the case. In fact, it is more common that the sequence of states the system has gone through, and even it's parameters, are totally hidden. All we can observe are only some physically observable emissions of the system which somehow depend on the hidden-state process. If we have at least some knowledge of how these emissions depend on the hidden states, we can infer these hidden states based on the direct observations. Thus, the HMM is introduced as a powerful tool to solve such problems [42].

4.5 Hidden Markov Models

Hidden Markov Model extends the basic Markov Chain by adding another layer of observation - symbol process producing sequence of emissions $O = o_1, o_2, \ldots, o_T$, were each emission o_t is one of a set of M observation symbols $E = \{E_1, E_2, \ldots, E_M\}$. Note that it is assumed only the finite set of observation symbols, since in this thesis, we focus on the pilot's eye scanning pattern between the finite number of clusters (instruments) forming the whole observation sequence [42].

Another assumption called sensor Markov assumption is introduced, telling that each emission is dependent only on its generating state. It is formally defined as [42]:

$$P(o_t|x_t, x_{t-1}, \ldots, x_1, o_{t-1}, o_{t-2}, \ldots, o_1) = P(o_t|x_t) \tag{4.13}$$

Graphically, the model is shown in figure 4.6. Again, considering the right-hand side of equation 4.13 is being time independent, observable symbol probability distribution (emission) matrix $\mathbf{B} = \{b_j(i)\}$ is created. Each matrix member is described as [42]:

$$b_j(i) = P(o_t = E_i|x_t = S_j), \qquad 1 \leq i \leq M, \quad 1 \leq j \leq N \tag{4.14}$$

denoting that probability of observed symbol E_i at time t is generated by the system in state S_j. Matrix $\mathbf{B}$ has similar properties as transition matrix $\mathbf{A}$ [42]:

$$b_j(i) \geq 0 \tag{4.15a}$$

$$\sum_{i=1}^{M} b_j(i) = 1 \tag{4.15b}$$

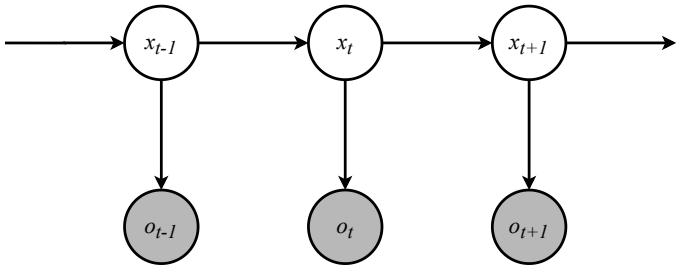


Figure 4.6: Graphical model of the Hidden Markov Model.

Having described all parameters from the above, the HMM can be formally defined as a 5-tuple $H = (S, E, \boldsymbol{\pi}, \mathbf{A}, \mathbf{B})$, where S and E are finite set of states and observable symbols respectively, $\boldsymbol{\pi}$ is the initial probability distribution matrix, and finally, $\mathbf{A}$ and $\mathbf{B}$ are transition and emission matrices respectively. Furthermore, all three matrices form a complete parameter set $\lambda = (\boldsymbol{\pi}, \mathbf{A}, \mathbf{B})$ of the model [42].

4.5.1 The Three Basic Computational Problems

Having some HMM model and observation sequence $O = o_1, o_2, \ldots, o_T$, our goal is to estimate hidden-state sequence of the system based on the sequence O. In order to achieve an optimal output in some way, it is required to solve three computational problems that arise when working with HMM. These are the following [42]:

1. Given the observation sequence $O = o_1, o_2, \ldots, o_T$ and model H with model parameters $\lambda = (\boldsymbol{\pi}, \mathbf{A}, \mathbf{B})$, how to efficiently compute conditional probability $P(O|\lambda)$, i.e. probability the sequence has been generated by the system?

2. Given the observation sequence $O = o_1, o_2, \ldots, o_T$ and model H with model parameters $\lambda = (\boldsymbol{\pi}, \mathbf{A}, \mathbf{B})$, how to choose a sequence of hidden states $X = x_1, x_2, \ldots, x_T$ which is the optimal in defined way?

3. Given observation sequence $O = o_1, o_2, \ldots, o_T$, alternatively together with appropriate labels $Y = y_1, y_2, \ldots, y_T$, and model H with model parameters $\lambda = (\boldsymbol{\pi}, \mathbf{A}, \mathbf{B})$, how to adjust parameters λ to maximize $P(O|\lambda)$, i.e. how to train the model?

In the following sections of this chapter, all three problems will be examined and optimal solutions described.

Model Evaluation

First problem to solve is the model evaluation problem. Given observation sequence and model, we want to compute the probability of the model generating given sequence. If we have multiple models and knowing appropriate likelihoods, we can choose the model that best matches the given sequence. If we know the basics of probability theory, it should not be the problem. The key word here is „efficiently“ however, and this is not the case of classic way [42].

But first, the probability could be computed intuitively by iterating over all possible hidden state sequences $X = x_1, x_2, \ldots, x_T$ of length T [42]:

$$\begin{aligned} P(O|\lambda) &= \sum_{allX} P(O|X,\lambda)P(X|\lambda) \\ &= \sum_{x_1,x_2,\ldots,x_T} \pi_{x_1} b_{x_1}(o_1) a_{x_1,x_2} b_{x_2}(o_2) \\ &\ldots a_{x_{t-1},x_t} b_{x_t}(o_t) \end{aligned} \tag{4.16}$$

It is quite obvious, that this naive approach is highly inefficient. Since there is a large number of calculations as the T increases. For $P(O|X,\lambda)P(X|\lambda)$ there needs to be done $2T-1 \approx 2T$ multiplications. Then, this has to be done for each possible sequence of length T, that is N^T where N denotes number of possible states. Overall, $2T \times N^T$ operations needed to solve the problem. For $N = 6$ states and $T = 100$ observations, it leads to $2 \times 100 \times 6^{100} \approx 10^{80}$ operations [42].

Fortunately, there exists a more efficient algorithm called Forward-Backward algorithm. It is a dynamic programming algorithm utilizing induction in order to solve the whole sequence. It works in two passes. In the first pass, it walks through the sequence forward in time, and then in second pass it goes backward in time, hence forward-backward algorithm [42].

At first, consider new forward variable $\alpha_t(i)$ defined as:

$$\alpha_t(i) = P(o_1, o_2, \ldots, o_t, x_t = S_i|\lambda) \tag{4.17}$$

denoting the probability of partial observation sequence up to the time t and state S_i at that time given model parameters λ. For the first observation, the forward procedure is initialized by computing α for each state according to equation 4.18. Consequently, for each other observation up to $t = T-1$, forward variable is computed using induction in equation 4.19. Finally, the probability $P(O|\lambda)$ is obtained by using equation 4.20 [42].

$$\alpha_1(i) = \pi_i b_i(o_1) \qquad 1 \le i \le N \tag{4.18}$$

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(o_{t+1}) \qquad 1 \le j \le N,\ 1 \le t \le T-1 \tag{4.19}$$

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{4.20}$$

After closer examination of equations 4.18, 4.19 and 4.20, it can be seen that the number of computations has been reduced from $2T \times N^T$ to approximately N^2T. This means that for $N = 6$ states and $T = 100$ observations we would need 3600 computations. This is a rapid reduction [42].

Similarly, the backward variable $\beta_t(i)$ for the purpose of the backward pass, which is together with $\alpha_t(i)$ utilized in solving the third problem the model learning, is defined as [42]:

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \ldots, o_T|x_t = S_i, \lambda) \tag{4.21}$$

which tells the probability of partial observation sequence from time $t+1$ up to T given state S_i at time t and model parameters λ. Backward algorithm is initialized by setting β_T

for all states equal to 1 (equation 4.22). Then, passing back in time, for each of N states S_i at precedent time-step, backward variable is computed according to equation 4.23 [42].

$$\beta_T(i) = 1 \qquad 1 \leq i \leq N \tag{4.22}$$

$$\beta_t(i) = \sum_{j=1}^{N} \beta_{t+1}(j) a_{ij} b_j(o_{t+1}) \qquad 1 \leq i \leq N, \;\; t = t_{T-1}, t_{T-2} \ldots 1 \tag{4.23}$$

One could already see, that the computation difficulty is again equal approximately to N^2T.

Most Likely State Path

Another problem which arises when working with the HMM is the problem of finding the optimal state path matching the observation sequence. However, there is a problem with what is meant by the word „optimal“, since there exist multiple optimal criteria we can apply when solving the second problem of most likely state path. Criteria which is the most commonly used when working with HMMs is to find path maximizing $P(X|O, \lambda)$ or, which is equal, $P(X, O|\lambda)$. To solve this problem, the dynamic programming method called Viterbi algorithm is used [42].

This algorithm is able to find the most probable hidden-state path $X = x_1, x_2, \ldots, x_T$ given a sequence of observations $O = o_1, o_2, \ldots, o_t$. First, a new variable is introduced [42]:

$$\delta_t(i) = \max_{x_1, x_2, \ldots, x_{t-1}} P(x_1, x_2, ..., x_t = S_i, o_1, o_2, \ldots, o_t|\lambda) \tag{4.24}$$

which gives the probability of the path with the highest score up to time $t-1$ while being in state S_i at time t and having first t observations given the model λ. Again, by using induction, we can get δ at next time-step [42]:

$$\delta_{t+1}(j) = \left[\max_i \delta_t(i) a_{ij}\right] b_j(o_{t+1}) \tag{4.25}$$

While evaluating 4.25, the argument that maximizes that δ is stored. At the end of the pass, the path is to be obtained using stored arguments in the backtracking process. Variable denoting the argument is defined as [42]:

$$\Psi_t(j) = \arg\max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] \tag{4.26}$$

The whole process of finding the optimal path according to the Viterbi algorithm has following steps [42]:

1. Initialization:

$$\delta_1(i) = \pi_i b_i(o_1), \qquad 1 \leq i \leq N \tag{4.27a}$$

$$\Psi_1(i) = 0 \tag{4.27b}$$

2. Recursion:

$$\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] \, b_j(o_t), \qquad 1 \leq j \leq N, \;\; 2 \leq t \leq T \tag{4.28a}$$

$$\Psi_t(j) = \arg\max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}], \qquad 1 \leq j \leq N, \;\; 2 \leq t \leq T \tag{4.28b}$$

3. Termination:

$$P^* = \max_{1 \leq i \leq N} [\delta_t(i)] \tag{4.29a}$$

$$x_T^* = \arg \max_{1 \leq i \leq N} [\delta_t(i)] \tag{4.29b}$$

4. Backtracking to determine hidden state sequence:

$$x_t^* = \Psi_{t+1}(x_{t+1}^*), \qquad t = T-1, T-2, \ldots, 1 \tag{4.30}$$

At first, each state's probability of occurring as a first one in the hidden state sequence together with the first observation is computed, and array to store maximizing arguments is initialized. Then, recursively with each next observation maximal probability is evaluated for each state S_j by finding the most probable previous state S_i relating to the given observation sequence. The maximizing argument i is stored into a array. When evaluating final delta, the maximizing probability is obtained with the last state. The final stage is the backtracking process to build the resulting hidden-state sequence matching the input observation sequence as best as possible [42].

Supervised Model Learning

The third, and the last problem to be solved is how to train the model, i.e. how to adjust it's parameters, in order to maximize $P(O|\lambda)$. There are essentially two possible approaches. The first, and the simple one, is supervised learning which assumes, that the dataset contains both, the observation sequence as well as the appropriate hidden-state sequence [42].

Having such dataset, most likely estimate of the transition probability distribution can be computed by just counting the frequency of transitions, i.e. a new transition matrix $A = a_{ij}$ is computed by counting relative frequency of transitions between all states S_i and S_j. To express it more formally, first, the number of occurrences of a transition from state S_i to S_j is given as [5]:

$$n_{ij} = \sum_{t=1}^{T} \|x_t = S_i \wedge x_{t+1} = S_j\| \tag{4.31}$$

where T denotes length of the training sequence and n_{ij} is frequency of transition from state S_i to S_j. Then, we get new most likely estimate $\hat{a}_{ij}$ of a_{ij} as [5]:

$$\hat{a}_{ij} = \frac{n_{ij}}{\sum_{k=1}^{N} n_{ik}} \tag{4.32}$$

where N is the number of all possible states. In a similar manner, the most likely estimate of the emission probability distribution is obtained by counting relative frequency of the system being in state S_j and emitting the observation E_i for each emission function $b_j(i)$. The number of emitted observations E_i in state S_j is formally defined as [5]:

$$m_i(j) = \sum_{t=1}^{T} \|x_t = S_i \wedge o_t = E_j\| \tag{4.33}$$

where $m_i(j)$ is the number of observable symbols E_j emitted by the system in state S_i. Now the most likely estimate $\hat{b}_i(j)$ of $b_i(j)$ is given by [5]:

$$\hat{b}_i(j) = \frac{m_i(j)}{\sum_{k=1}^{M} m_i(k)} \tag{4.34}$$

where M is the number all possible observation symbols. This training approach is convinient, off course, in case appropriate hidden state sequence is available. Due to insufficient dataset however, it is possible that at the end of the training procedure, the model will be underfitted. The cause of this problem is the absence of transitions in the training dataset to a unused state S_j. Thus, there is no way how to reach the state, because the probability of transition to the state S_j is equal to zero. The solution is to bias n_{ij} and $m_i(j)$ by bias values c_{ij} and $c_i(j)$ respectively. Bias values represent our prior knowledge about the probability values. Values can be either small or large integer or real values. The larger the value of the bias is, the more data is required to modify the prior knowledge [5]. Biased counts are then used in equations 4.32 and 4.34 respectively.

Unsupervised Model Learning

Having a complete dataset of observation sequences with appropriate hidden-state sequences is not always the case however. In most cases only the observation sequence is available making the third problem, i.e. model training, the most difficult to solve. In such case the unsupervised learning techniques come to play. Even there is no known technique to solve the problem analytically for the global maxima of the model parameters in a closed-form, still, there exist techniques to maximize the probability $P(O|\lambda)$ locally. The most common algorithm used to train HMM is called Baum-Welch algorithm [42].

It is an iterative procedure which is from of EM (expectation-maximization) algorithm divided into two step, as the name suggests, expectation and maximization. First, parameters λ are randomly generated or build based on prior knowledge about the parameters. Then the algorithm iterates over two steps, the E-step and M-step. E-step estimates expected frequencies, probabilistic equivalents to n_{ij} and $m_i(j)$, based on current probability distributions and forward-backward variables α and β. In the M-step are re-estimated new parameters of the model based on estimates obtained in the E-step [42].

To start with aa formal description of the algorithm, a new variable $\xi_t(i,j)$ is introduced [42]:

$$\begin{aligned} \xi_t(i,j) &= P(x_t = S_i, x_{t+1} = S_j | O, \lambda) \\ &= \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{P(O|\lambda)} \\ &= \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)} \end{aligned} \tag{4.35}$$

denoting the probability of being at time t in state S_i and at time $t+1$ in state S_j given the observation sequence O and model with parameters λ. The probability is computed utilizing forward and backward variables $\alpha_t(i)$ and $\beta_t(i)$ [42].

Second variable, introduced in the formal description of the algorithm defining probability of being in state S_i at time t given parameters λ and observation sequence O, is defined as [42]:

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j) \tag{4.36}$$

If we know values of both variables for each state S_i and S_j, we can sum each variable over time so we get expected number of visiting state S_i and expected number of transitions from state S_i to state S_j [42]:

$$\sum_{t=1}^{T-1} \xi_t(i,j) = \text{expected number of transitions between states } S_i \text{ and } S_j \tag{4.37}$$

$$\sum_{t=1}^{T-1} \gamma_t(i) = \text{ expected number of visiting state } S_i \tag{4.38}$$

Now we have all what is needed to re-estimate new parameters $\hat{\lambda} = (\hat{\boldsymbol{\pi}}, \hat{\mathbf{A}}, \hat{\mathbf{B}})$ as [42]:

$$\hat{\pi}_i = \gamma_1(i) \tag{4.39a}$$

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \tag{4.39b}$$

$$\hat{b}_j(i) = \frac{\sum_{t=1,\, s.t.\ o_t=E_i}^{T-1} \gamma_t(j)}{\sum_{t=1}^{T-1} \gamma_t(j)} \tag{4.39c}$$

It has been proven by Baum [2], that in this manner, we can iteratively estimate new parameters of the HMM and converge to the local maximum if and only if the initial parameters λ do not represent a critical point of the likelihood function. Then applies that $P(O|\hat{\lambda}) > P(O|\lambda)$, $\hat{\lambda} = \lambda$ otherwise [42].

When it comes to comparison of supervised versus unsupervised learning, in several works [5][28] it has been observed, that unsupervised learning significantly overcome supervised learning in terms of performance. Furthermore, unsupervised learning is more flexible, since it does not require appropriate labels. On the other hand advantage of the supervised learning lies in the computational complexity.

Chapter 5

Design and Implementation of the Classification Framework

In the first part of this chapter, the flight scenario flown during experimental measurements is introduced. Next, simulation and data acquisition framework used for experimental measurement purposes is introduced. Finally, proficiency and maneuver classifier design and its implementation are described.

5.1 Flight Scenario

Flight scenario is designed to consist of all basic instrument flight maneuvers. These are straight-and-level flight, level and vertical turns, and climbs and descents. Part of the whole pattern is also the take-off procedure, as the pilot starts each flight from the runway. Overall, the whole flight consists of up to 10 phases. However, first phase, take-off, is not an object of further analysis and classification.

Simulated flight is situated in the area around Brno Turany airport (ICAO code LKTB) with airfield elevation of around 770 ft. The flight starts with a departure from runway 09. Initial climb after the take-off ends when reaching the altitude of 1 000 ft. Predetermined climbing airspeed and correct vertical speed should be established during this phase. Then, initial climb follows up to target pressure altitude of 3 000 ft. After reaching target the altitude, series of basic flight maneuvers is to be performed. The whole pattern can be seen in figure C.1. Each participant performs the introduced scenario four times. The whole flight is performed using solely flight instruments, i.e IFR flight.

Regarding flight the constraints of basic instrument flight maneuvers, all turns are to be performed as a standard-rate turn. During climbs and descents, a vertical rate of climb/descent of $500\,\text{ft} \cdot \text{min}^{-1}$ is to be maintained. A full list of predetermined constraints is below:

- *Airspeed* — during all basic instrument flight maneuvers, the IAS of 100 kts should be reached and maintained. For climbs, the IAS airspeed is set to 75 kts.
- *Altitude* — majority of the flight is performed in pressure altitude range between 2 500 and 3 000 ft.
- *Vertical Speed* — target vertical speed for all maneuvers is set to $\pm 500\,\text{ft} \cdot \text{min}^{-1}$.

- *Turn Rate* — since all turns should be performed as a standard-rate turn, desired turn rate to be maintained during each turn is equal to 3 °/s.

A two way communication between the operator and the pilot takes place during the simulated flight. There are two reasons for the communication:

- Calling maneuvers and parameters to the pilot.
- Labeling of individual maneuvers by the pilot.

As the whole flight pattern is difficult to remember, prior to each maneuver, operator calls the pilot informing her/him what maneuver should be performed and what are the respective flight parameters. The pilot replies with a readback, informing the operator she/he understands the command. Subsequently, the pilot initiates the maneuver. When maneuver is accomplished, the pilot reports the maneuver termination, so the operator could issue request to another maneuver. Every readback and termination calls serve also as labels for individual maneuvers in data processing stage of the experimment.

5.2 Participants

Overall, 16 participants took part in the experiment. Every proficiency class was represent by 8 participants. The "amateur" class consisted of participants with no real flight experiences. However, they were familiar with aircraft control theoretical background, correct interpretation of flight instruments and proved to have high interests in flight experiments. Each recruited certified pilot representing the class "experienced" had in general more than 150 total flight hours (mean = 843.1, std = 1122.6) on Single-Engine Propeller (SEP) aircraft. Three of the certified pilots had previous partial experience with IFR flight, with one the cohort having 100 IFR hours.

Prior to measurements, all participants were asked to perform familiarization free flight exercises and a single trial, to get experienced with the simulator sensitivity and responses.

5.3 SimStar Simulation Framework

For experiments, the SimStar simulation framework developed at the Faculty of Information Technology of the University of Technology in Brno by the Aeroworks research group has been used. The simulation framework consists of multiple functions, which together create a high fidelity illusion of flight. During the simulated flight, the pilot is seated in the cockpit consisting of the the mid-fuselage section of a Light Sports Aircraft (LSA). The cockpit is equipped and instrumented to match the original aircraft cockpit.

The flight dynamics model of the SimStar framework has been adjusted to represent high-wing SEP handling characteristics similar to Cessna 172 aircraft. The model has been in integrated into the X-Plane 11[1] simulator, which provide simulation data to other framework subsystems.

In terms of flight visualization, the simulator features a subsystem consisting of 3 large screens which introduce the simulation to a virtual environment surroundings. Thanks to displays, the pilot is able to navigate his virtual aircraft through the environment with the

[1] X-plane 11: https://x-plane.com/

Figure 5.1: SimStar simulation framework.

use of outside references such as natural horizon, buildings and natural features. Overall, the pilot can enjoy field of view a 115 ° in the horizontal plane and 45 ° in the vertical plane.

Another important aspect of the visualization subsystem is the visualization of the flight instruments on the instrument panel. The pane featuring 6 main flight instruments in the T-arrangement:

- *ASI* — Airspeed Indicator
- *AI* — Attitude Indicator
- *ALT* — Altimeter
- *TC* — Turn Coordinator
- *HI* — Heading Indicator
- *VSI* — Vertical Speed Indicator

On the left side of the instrument panel are two indicators for the elevator and aileron trims respectively. In the central part of the instrument panel is the Timer and Distance Measuring Equipment (DME). The Timer is used to check elapsed time since the beginning of respective maneuver. The tachometer indicating engine's RPM is placed on the right side of the instrument panel. Overall layout of the dashboard is depicted in figure 5.2.

Important part of the framework is the audio subsystem providing spacial reproduction of both, external and internal sounds of the aircraft from the cockpit position. Majority of these sounds are engine and airstream noises. Overall reproduction is provided through

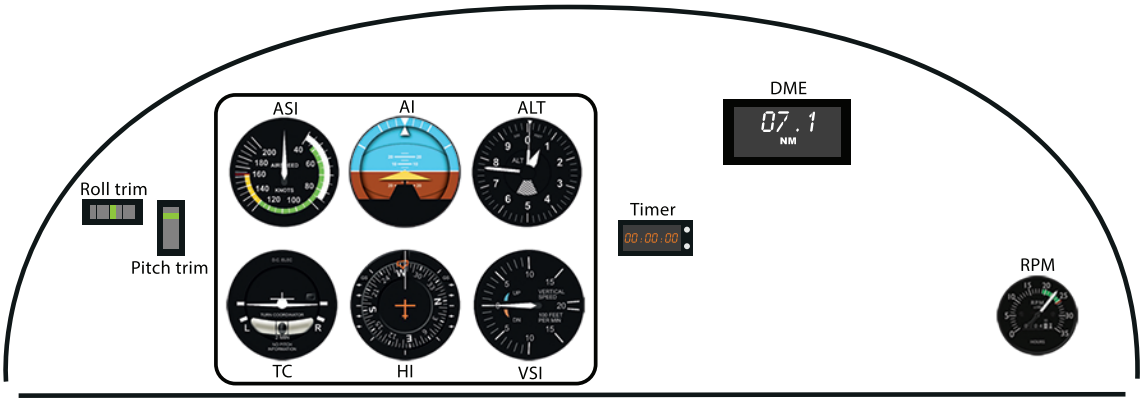


Figure 5.2: SimStar instrument panel.

the high quality headset. Integrated audio system also enables to communication between the pilot and the operator or other crew member.

The last, but not least important part of the simulator, is the haptic subsystem providing the pilot with accurate tactile cues. Thanks to the two-axis active force-feedback mechanism, the pilot is able to experience maneuver relevant forces acting on the control stick. This pressure can be moderated with the use of respective trims. The subsystem contributes to the overall fidelity of the simulator, with the overall experience is shifted to high level closer to authenticity.

5.4 Data Acquisition

Two types of data were measured and recorded — gaze-related data for the purpose of gaze analysis, and flight data used to evaluate pilot performance and for data pre-processing discussed further in this chapter.

5.4.1 Gaze Data

For the purpose of eye movement capture and recording, Pupil Core eye tracker developed by Pupil Labs[2] company was used. Pupil Core is an eye tracking platform designed for a wide range of research purposes. It provides two options of eye tracking headset setups. One of the setups is only capable of pupil tracking for pupillometry purposes with only two eye cameras. Second one, used in this thesis, adds the world camera, so actual gaze data can be recorded. The advantage of this platform is in its modularity and openness, supported by an open source software suite with available Application Programming Interface (API). The software suite also includes data recorder and player for real-time gaze capture and preview.

As has already been mentioned, the eye tracker headset setup with two eye cameras and one world camera has been used. The headset with its component is shown in figure 5.3.

The recording frequency was set to 60 Hz with the video resolution of 720p. Respective eye fixations are detected using dispersion-based method called the Dispersion-Threshold

[2]Pupil Labs: https://pupil-labs.com/

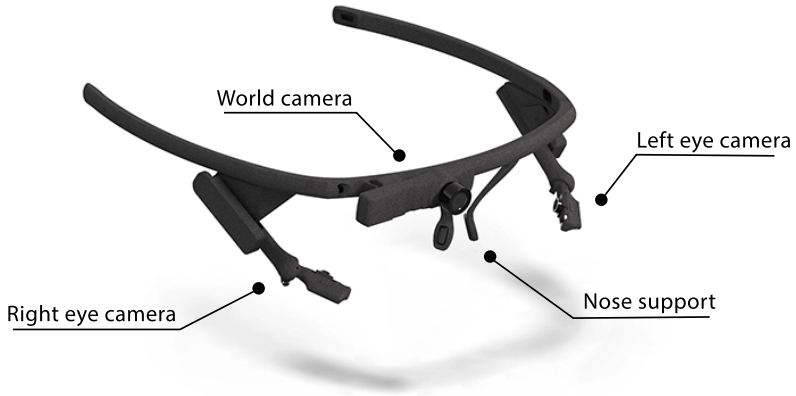


Figure 5.3: Pupil Core eye tracker with two eye cameras and one world camera. Taken from [41].

algorithm (I-DT). Two parameters are required for this method, namely the dispersion and duration thresholds. Without going into details, the duration threshold determines the minimum duration for a single fixation, while the dispersion threshold indicates the maximum distance between two extreme fixation points of a group of consecutive fixation points forming a single fixation. In our case, the dispersion threshold was set to 1.5 °. Regarding the duration setting, it was set to 50 ms, as it provided optimal system results [46].

For a more comfortable gaze data post-processing, the Pupil Core API provides tool to annotate tracked fixations by respective area of interest labels. First, region containing those AOIs has to be defined. In Pupil Labs terminology, this region is referred to as surface, as it represents planar surface in measured environment. The number of these surfaces is not limited. The area of each surface is defined by one or more AprilTag type markers, while a single marker can be part of multiple surfaces. Having these surfaces defined, positions of individual AOIs can be defined within these surfaces. In our case, a single surface is defined, where each AOI represent individual flight instrument. Trim indicators, Timer and DME were ignored in this thesis. Windscreen area is also not considered. Together with the surface region, each AOI has assigned a unique numerical ID and a short name. If the pilot is not fixated within any AOI, but still inside the surface area, ID of the area is assigned instead. Full scheme of the instrument panel and respective instrument AOIs is presented in figure 5.4. Overall, 7 instrument AOIs are defined: (1) ASI, (2) AI, (3) ALT, (4) TC, (5) HI, (6) VSI, (7) RPM.

Fixation positions are first expressed in the world's camera coordinate system with particular AprilTags also tracked by the world camera. Then, based on the knowledge of each marker's position in the coordinate system, homography technique is used to remap fixations into the coordinate system of the corresponding planar surface.

Gaze Tracking Output

The gaze tracking software output from a single measurement is a `*.csv` file containing all relevant information regarding the gaze tracking. Format of the filename containing gaze tracking data has following form: `<prefix>__eyeFixation.csv`.

Overall, the Pupil Core based system outputs 11 measured features. The first feature is a unix timestamp (`Timestamp`) measured in nanoseconds, which indicates starting time

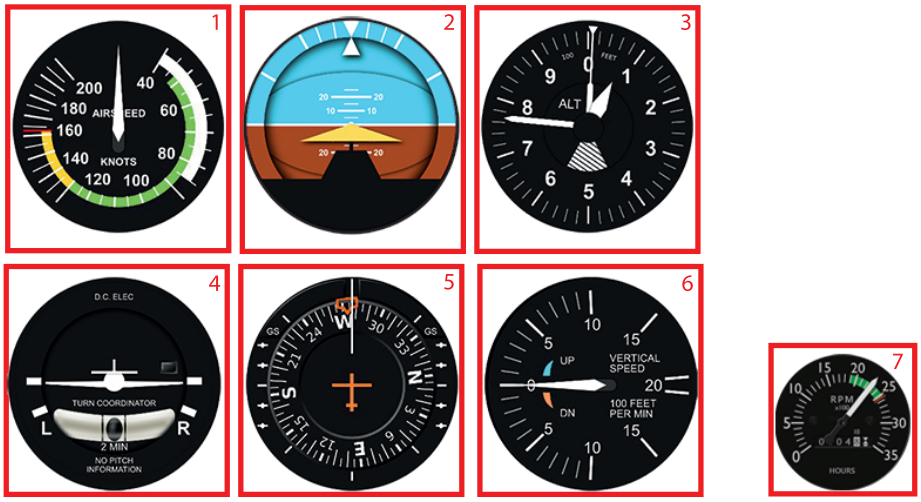


Figure 5.4: Default layout of areas of interest.

of each recorded fixation. Another time information included in the file is the duration of the fixation (`Duration_ms`) measured in microseconds.

Regarding the fixation positions, X and Y coordinates in two coordinate systems are provided. First coordinate system is the original world camera frame (`Norm_pos_x`, `Norm_pos_y`). Second system is the respective planar surface system, into which recorded fixations are remapped, and where all AOIs are defined (`Norm_area_x`, `Norm_area_y`). All coordinates are normalized according to defined width and height of the respective surface.

Another information included in the output file are IDs and short names of both, area and AOI in which the fixation is detected (`Area_ID`, `Area_of_interest`, `Cluster_ID`, `Cluster_name`). Last two features are dispersion and confidence (`Dispersion`, `Confidence`). Dispersion tells what was the dispersion during the fixation detection process, while confidence indicates how certain the algorithm was with left and right eye pupil detection.

5.4.2 Flight Data

The X-plane software provides a large number of various variables that can be logged or set. Individual variables can be accessed using the so called datarefs. A full list of these datarefs with a short description, including variable units and supported X-plane version, is available at the official X-plane website[3].

For performance analysis and data pre-processing, only the most relevant flight data variables were logged during each flight. Set of logged variables is listed in table 5.1. International System (SI) units in the table 5.1 were for the analysis converted into units matching with those on flight instruments.

Data were logged with Aeroworks's in-house script using NASA's XPlaneConnect [4] toolbox. It allows user to access all provided datarefs, and set all writable ones. It consists of XPlaneConnect plugin, which has to be activated in X-Plane, and library of functions used for User Data Protocol (UDP) communication between user program and plugin. Func-

[3]X-plane datarefs: https://developer.x-plane.com/datarefs/
[4]XPlaneConnect plugin: https://github.com/nasa/XPlaneConnect

Table 5.1: Logged flight variables.

Variable	Unit	Dataref
Latitude	°	sim/flightmodel/position/latitude
Longitude	°	sim/flightmodel/position/longitude
Altitude	m	sim/flightmodel/position/elevation
Indicated Airspeed	kts	sim/flightmodel/position/indicated_airspeed
Vertical Speed	$m \cdot s^{-1}$	sim/flightmodel/position/vh_ind
Heading	°	sim/flightmodel/position/true_psi
Turn Rate	°/s	sim/flightmodel/position/R
Bank Angle	°	sim/flightmodel/position/true_phi
Pitch Angle	°	sim/flightmodel/position/true_theta

tions are provided in all major programming languages: C/C++, Java, Python/Python3, and also for Matlab.

The output file with recorded X-plane data is of type `.csv` containing unix timestamp (`Timestamp`) and respective flight variables data, where column name of each flight variable is equal to the respective dataref name defined in table 5.1. Filename format as the following: `<prefix>__Xplane.csv`

5.5 Data Preprocessing

Prior to any analysis and data processing for classification purposes, raw data had to be preprocessed. Preprocessing procedure included the definition of time periods of individual flight phases (maneuvers) according to pilot calls, AOI subclustering and machine learning dataset preparation.

5.5.1 Flight Phases Division

Prior to each experiment, pilots were given instructions to follow the communication with the operator during each flight. Thanks to the communication, appropriate labels were created, so individual flight phases could be defined. For an easier maneuver labels extraction, a special hardware push button was also used as backup solution, with every press recorded. Every time the pilot calls the start and termination of the maneuver, the button was pressed. Signals from the button were recorded, so timestamp of each press became available. During the pre-processing, every two consecutive presses were used to define individual maneuver. Logged flight data were used to double-check, whether all labels correctly define specific maneuver. In case of inappropriate or missing labels (presses), in combination with the communication record, flight data were used to recover those labels.

The result of this pre-processing procedure is another `*.csv` file for each measured experiment, where format of the filename is as following: `<prefix>__Phases.csv`. This file includes timestamp of the maneuver initiation (`Start`), duration of the maneuver in nanoseconds (`Duration`), phase/maneuver name (`Phase`), and phase/maneuver ID (`Phase_ID`) denoting the class of the maneuver used for classification purposes.

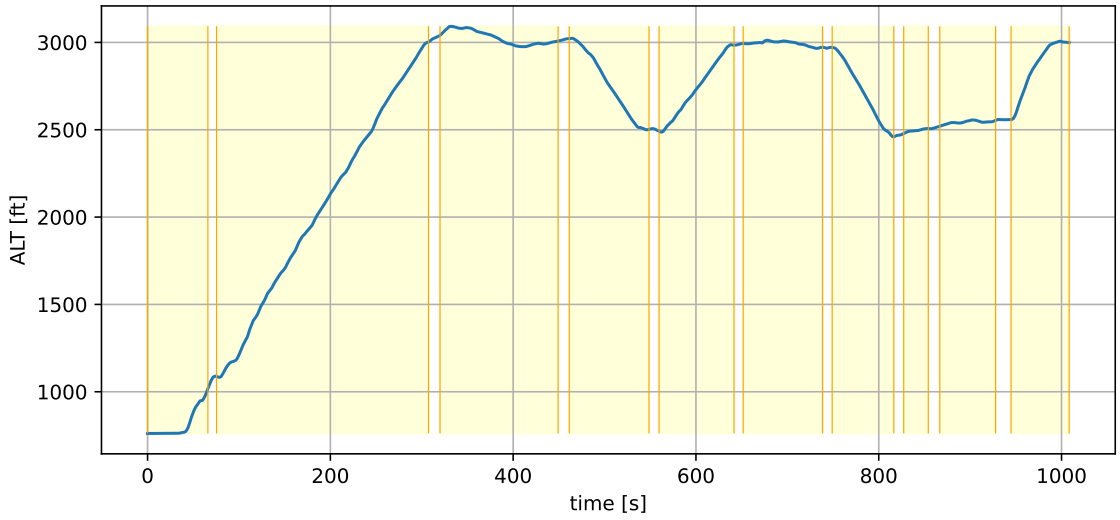


Figure 5.5: Example of a single flight partitioning into individual flight phases according to pilot calls.

5.5.2 Areas of Interest Subdivision

The original idea was to try to classify basic flight maneuvers based on pilot's gaze using detection of fixated instruments on the highest level only. Meaning, only whole AOIs representing individual instruments were considered, without any further subdivision of individual instrument AOIs into another sub-clusters. This showed to be a difficult task after some trials and analyses. It has been observed, that if a pilot performs the maneuver well enough while using a stable scanning pattern, at the end of the maneuver, there is a low diversion between utilization of individual instruments during individual maneuvers.

An example of how experienced pilot scans instrument panel during individual phases can be seen in figure 5.6, which clearly shows, that all instruments are used similarly across all flight maneuvers. As a small exception, we can mention the vertical speed indicator, where we can see a hint of a distinction between vertical and horizontal maneuvers. Note, that the AI plays a major role in the pilot's scanning strategy during each maneuver. Based on the knowledge presented in chapter 2, this outcome is understandable, since the AI is considered a master instrument in all commonly used scanning strategies. Another explanation of this phenomena is, that the AI as a single instrument provides two key information to the flight control, pitch and bank angle.

Results of analyzes presented above led to an idea of the division of some AOIs into individual sub-clusters, as some of them are capable of distinguishing individual maneuvers if they are divided correctly.

Division of the Attitude Indicator

It has been already mentioned and presented in the previous section, that the AI, as a master instrument, plays a major role in the pilot's scanning strategy in all flight maneuvers. Also, it combines two instruments together - pitch and bank indicators. Thus, the AI is naturally

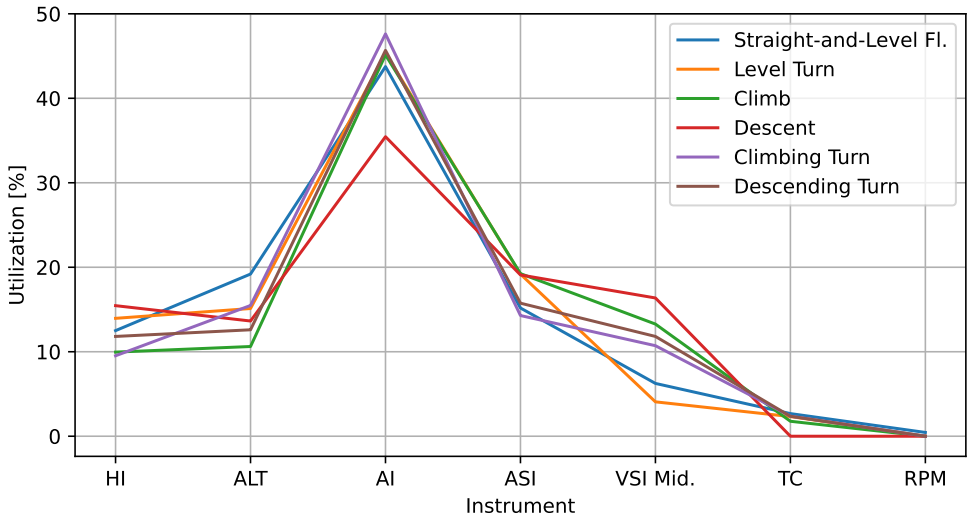


Figure 5.6: An example of utilization of individual instruments expressed in percentage for each of 6 maneuvers in a single flight.

chosen as the first candidate for further logical partitioning of the original AOI into multiple sub-sections.

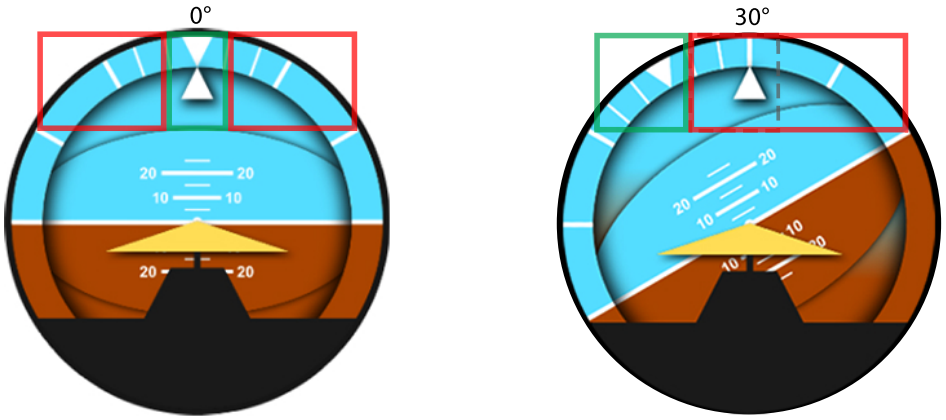
At first, the original AOI was divided into two new regions representing pitch and roll parts of the AI. The division was made with the use of the common clustering method K-Means. Inputs of the K-Means algorithm consisted only of fixations mapped into the original attitude indicator AOI.

Green rectangle indicates `Roll_mid` AOI, while the red one indicates `Roll_side` AOI. Dashed rectangle indicates the original width of the middle section.

Newly defined AOIs however, would still not have enough power to distinguish between at least longitudinal and lateral maneuvers. If we think about the principles of the AI, we know that while the scale of the bank angle rotates as the angle changes, the triangle indicating current angle is actually static. Thus, pilot has to look all the time into the middle section of the roll indicator, in order to obtain current angle of bank. If pilot maintains same heading during the straight-and-level flight, the triangle in the middle indicates 0 °. This situation is depicted in figure 5.7a. Then, when heading changes, the triangle still located in the middle, indicates different bank angle values as the scale is rotated accordingly. As a result, during each fixation into the region of bank angle indicator, pilot obtains a different value while looking into the middle section. Example of the scale rotation with 30 ° bank angle can be seen in figure 5.7b.

Having this in mind, two AOIs were defined — `Roll_mid` and `Roll_side`. `Roll_mid` AOI represents the range of bank angle scale between -10 ° and 10 °, while `Roll_side` groups two extreme ranges of the scale together. These are ranges, where the scale indicates angles greater than 10° and lower than -10 °. Boundaries of new AOIs for 0° bank angle are shown in figure 5.7a. Green rectangle represents `Roll_mid` AOI, whereas `Roll_side` AOI is represented by red rectangles.

Then a middle section region in the planar surface system is defined. It is used to control the remapping process. Width of this section is set to fit between indicators of -10 °



(a) 0° angle of bank

(b) 30° angle of bank.

Figure 5.7: The division of the roll indicator into two sections according to the current angle of bank.

and 10 ° bank angle range on the scale for 0 ° bank angle situation, as it is shown in figure 5.7a. In this figure, the middle section has same proportions as the green rectangle. The middle section is used to decide, where the fixation is located with respect to this section. Indicated location is further used in the process of new AOI assignment to the fixation. Note the middle section and `Roll_mid` AOI each have different meanings. In this case, `Roll_mid` and `Roll_side` represent the information the pilot obtains from the instrument rather than some region with static boundaries.

Having `Roll_mid` and `Roll_side` AOIs, bank angle data from X-plane and fixation location with respect to the middle section are used, to determine to which AOI each fixation will be remapped. New AOI of each fixation is selected according to the fixation position and current bank angle value. If the fixation was detected inside the middle section region boundaries, and the bank angle at the time of the fixation was in range ± 10 ° bank angle, `Roll_mid` AOI was assigned. If the pilot was looking outside of the middle section, fixation was mapped on `Roll_side` AOI. In case the bank angle was greater than 10 ° and the fixation was to the left of the middle region, `Roll_mid` AOI was assigned, otherwise, fixation was mapped on `Roll_side`. Similar logic applies for the angle lower than -10 °.

As the angle exceed ± 20 °, for every degree above 20 ° or below −20 °, border of the middle section is expanded by 5 % against the `Roll_mid` AOI. This expansion can be seen in figure 5.7b, where the dashed rectangle represent the middle section. Green rectangle represent `Roll_mid` AOI, while red rectangle is the boundary of `Roll_side`. However, since pilots were briefed to perform standard-rate turns, which corresponds to approximately 15° bank angle at around 100 kts, for the majority of time, width of the middle section did not change.

Similar logic of the AOI partitioning was applied onto the pitch angle scale, where a single AOI was replaced by two — `Pitch_mid` and `Pitch_vert`. The height of the middle section was set to match the range of pitch angle from -2 ° to 5 °. If the pitch angle exceeded ± 10 °, using same logic as in case of roll indicator, middle section is expanded by 15 % of the original height against the `Pitch_mid`.

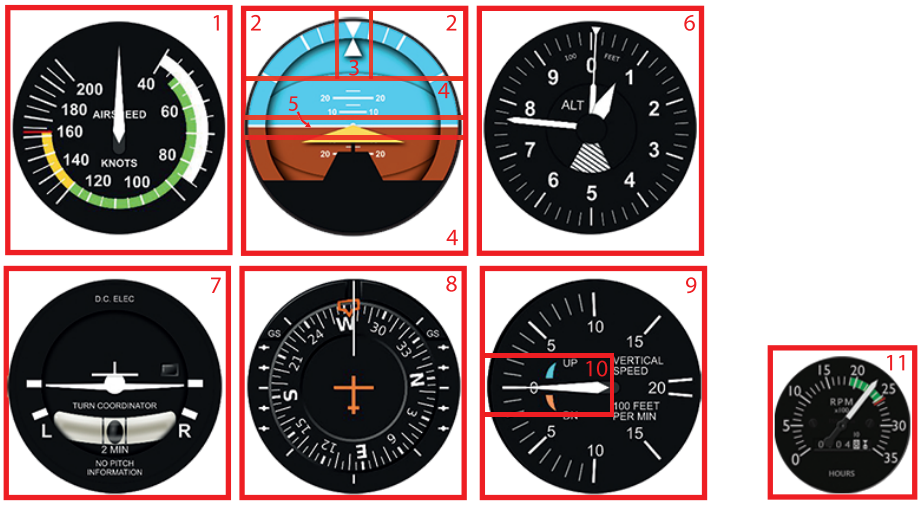


Figure 5.8: Mid-and-Vert layout of AOIs.

Division of the Vertical Speed Indicator

If we consider, that an experienced pilot maintains desired rate of climb and airspeed, the pitch angle does not change in such a scale as roll angle. In a stabilized climb, the pitch angle barely exceeded 10 °, while during the descents the threshold was at around -5 °. As the space on the pitch scale between the range of ± 5 ° is quite narrow, there is a high change of assigning of `Pitch_mid` to most of fixations. This can result in the low sensitivity and low ability to distinguish between some maneuvers, especially between straight-and-level flight and climbs/descents. Furthermore, during the analysis was observed, that some pilots did not use pitch angle scale as much as the roll angle scale.

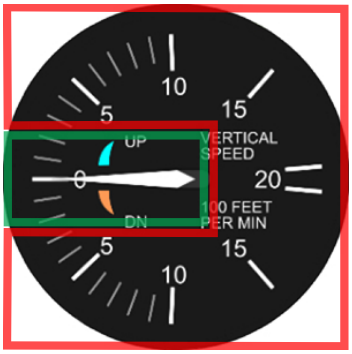


Figure 5.9: Divided VSI AOI into a middle and vertical sections.

To enhance ability to distinguish vertical maneuvers from horizontal, and to support low sensitivity of pitch angle scale, the VSI was also divided, as it provides an instant response to an altitude change. In fact, it is the most sensitive instrument on the SimStar instrument panel. The advantage of this instrument is, that its dial is divided into two parts. Each indicates individual trend and the needle also does not spin around the whole dial as in the case of the ALT. Similarly to the pitch scale, the VSI was divided into two sections

— `VSI_mid` and `VSI_vert`. Since the target climb/descent rate was set to $500\,\text{ft} \cdot \text{min}^{-1}$, middle section boundaries should be set somewhere between $\pm 500\,\text{ft} \cdot \text{min}^{-1}$. For the higher sensitivity, vertical boundaries of the middle section were set on the level, where the tip of the needle indicates vertical speed of $\pm 200\,\text{ft} \cdot \text{min}^{-1}$. Final look of the VSI partitioning into two respective sections is shown in figure 5.9. Final layout of the AOIs, in this thesis referred to as Mid-and-Vert layout, is introduced in figure 5.8. Respective AOIs are: (1) ASI, (2) Roll side, (3) Roll middle, (4) Pitch vertical, (5) Pitch middle, (6) ALT, (7) TC, (8) HI, (9) VSI vertical (10) VSI middle and (11) RPM.

Separation of Ascending and Descending Maneuvers

In order to distinguish between ascending and descending maneuvers, vertical regions of both VSI and Pitch AOIs were further divided into respective clusters, each corresponding to one of the two vertical directions — `VSI_down`, `Pitch_down`, `VSI_up` and `Pitch_up`. Final layout of all AOIs after the partitioning process, referred to as Up-and-Down layout, is depicted in figure 5.10. The layout is defined by 13 AOIs in total: (1) ASI, (2) Roll side, (3) Roll middle, (4) Pitch up, (5) Pitch down, (6) Pitch middle, (7) ALT, (8) TC, (9) HI, (10) VSI up, (11) VSI down, (12) VSI middle and (13) RPM.

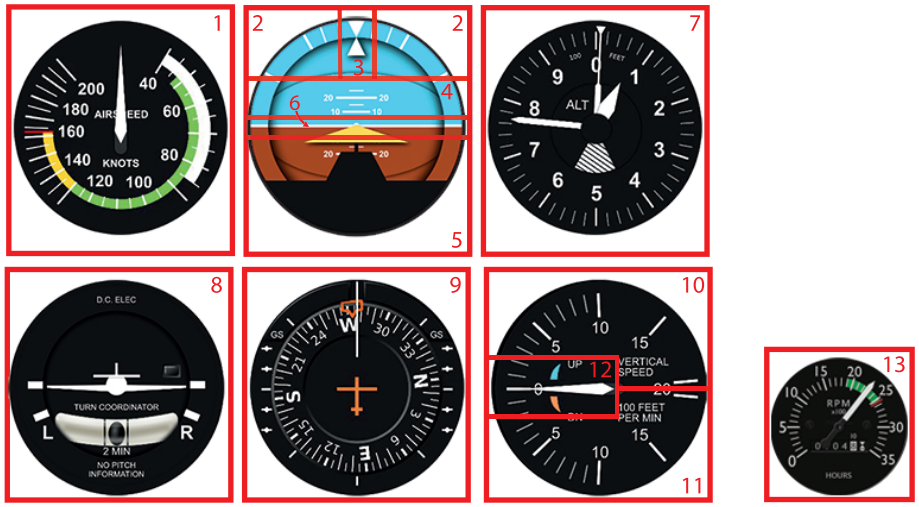


Figure 5.10: Up-and-Down layout of AOIs

5.5.3 Dataset Preparation

For the machine learning purposes, a dataset had to be prepared in order to train and evaluate respective models. In total, three classification cases were implemented and evaluated. Individual dataset for each case was prepared.

All three datasets consisted of samples representing single phase of each flight of each pilot. Every sample consists of series of AOI IDs, each representing a single fixation. Fixations inferior to 100 ms were discarded, while consecutive fixations were maintained. Thus, individual fixations instead of dwells were considered. As two different layouts of AOIs were defined, two of three datasets shared the same layout of AOIs.

Data of each sample were stored in a `*.csv` file with a single column `AOI`. Each row of the column stores a single AOI ID. Labels of respective maneuver and proficiency level were included in the filename of each sample in the following format: `<pilot_ID>___<f_number>___<maneuver_ID>___<prof_ID>___<phase_name>.csv`, where `maneuver_ID` represent the maneuver label and `prof_ID` is the proficiency label.

5.6 Proficiency and Maneuver Classifier

Overall, designed proficiency and maneuver classifier utilizes two different classification techniques, one for each classification problem. At first, pilot's proficiency is inferred using the SVM model. Then, based on the result of the inference, an appropriate maneuver classification model is selected ("amateur" or "experienced" class model) to infer flight maneuver type. Each class's classification model consists of respective number of HMM models, one for each basic flight maneuver. Result of the inference is a tuple of inferred proficiency level and executed maneuver. Scheme of the implemented classifier can be seen in figure 5.11. In this figure, y_L denotes classified proficiency level and y_C is the classified maneuver class.

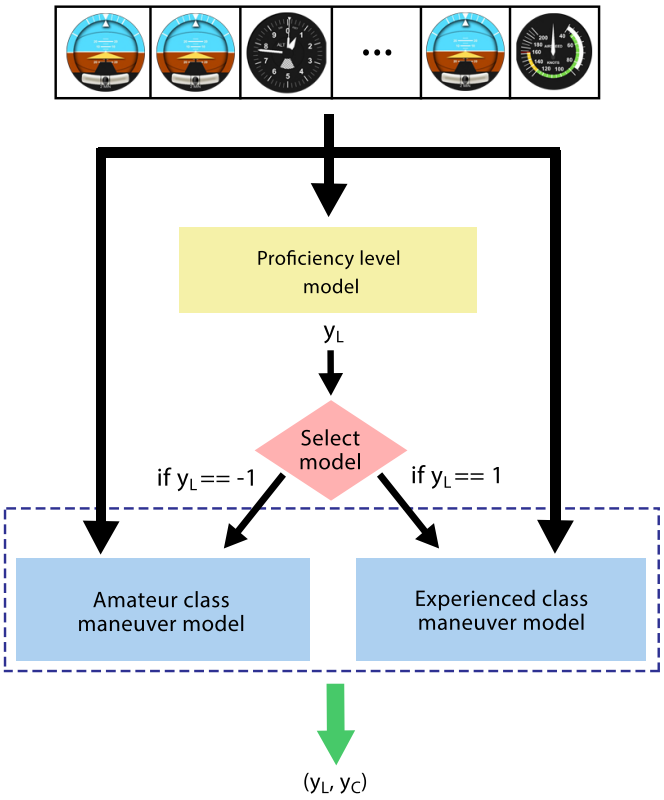


Figure 5.11: Designed classification framework scheme.

5.6.1 Technologies Used

The whole classification framework was implemented in programming language Python 3.8 in combination with numerous libraries. Regarding the data processing, the most utilized libraries are NumPy and Pandas. Machine learning related implementation parts utilized mainly Scikit-learn and Hmmlearn libraries.

NumPy

NumPy[5] is an open source library providing wide range of tools for scientific computations. It's functionality ranges from simple and basic mathematical functions to high dimensional data processing. Due to an indexing and vectorization techniques, processing of high dimensional data is very fast and effective, which makes of NumPy today's standard for this purposes. In addition, NumPy is not restricted to only numerical values, but it is capable of processing data various types [16].

Pandas

Pandas[6] is another powerful, easy to use, open source library for an easy and fast data processing and analysis. It introduces an efficient object called `DataFrame` for a fast data indexation and manipulation. Pandas is capable of high performance merging and joining multiple data sets. Missing data values can be efficiently handled according to developer needs. It allows to read and write data from multiple data formats such as `*.csv`, `*.json`, Microsoft Excel or SQL databases. Data stored in the DataFrame object can also be converted into NumPy arrays. Similarly to the NumPy, Pandas provides a wide range of functions for data manipulation [39].

Scikit-learn

Scikit-learn[7] is library featuring wide range of algorithms and models for machine learning purposes. In particular, it provides algorithms for classification, regression and clustering problems. In addition, another tools to facilitate work with machine learning related problems are provided such as data preprocessing, tools for models and parameters selection and validation, and also algorithms for dimensionality reduction. It is an easy to use library, which can be utilized for a fast and efficient development of machine learning applications [30].

Hmmlearn

Hmmlearn[8] is open source library, dedicated to the unsupervised learning of HMMs. It is an easy to use library build on Scikit-learn library and tries to follow its API as close as possible. Overall, 3 different HMM types are provided, each utilizing different emission distributions [23]:

- Gaussian emissions,
- Gaussian mixture emissions,

[5] NumPy: https://numpy.org/
[6] Pandas: https://pandas.pydata.org/
[7] Scikit-learn: https://scikit-learn.org/stable/index.html
[8] https://hmmlearn.readthedocs.io/en/latest/

- multinomial (discrete) emissions.

Additionally, base HMM model for creation of models with custom emission distribution is provided.

5.6.2 Level of Proficiency Classification

First level of the classifier represents the proficiency classification process. As has been mentioned, for the purpose of proficiency classification, SVM machine learning algorithm is utilized, as it is an optimal choice for binary classification. Another advantage of the selected algorithm is effectiveness in high dimensions, simplicity and its robustness against the high dimensional data, i.e. when the number of features exceed the number of observations. This was not the case of this thesis in particular. However, if we wanted to use KNN, we would need much more samples in order to overcome "curse of dimensionality" problem, since final feature vector has over 20 features and the total number of samples is around 580.

Feature Selection

The class, representing experienced pilots, was given the label value 1, while -1 was assigned for amateurs class. Out of all available metrics of gaze tracking, only a transition matrix was selected for the feature vector construction. A two-way probability transition matrix was used in particular. It proved to be a flexible metric independent on the maneuver type. For the computation of two-way transition matrix, consecutive fixations were grouped together, so dwells were created. Because this transition matrix is symmetric, values on and below the diagonal were discarded. If consecutive values would be maintained and diagonal values would be considered, other transitions would not stand out due to significantly high occurrences of self-transitions. Remaining values were concatenated into a single feature vector. To decrease a high dimensionality of the feature vector, Principal Component Analysis (PCA) analysis was used to reduce the number of features. As a result, size of the feature vector was reduced down to 24 feature for Mid-and-Vert AOI layout, and 30 features for Up-and-Down AOI layout respectively, while 95 % of the variance is explained by respective principal components.

Model and Hyperparameters Selection

For the purpose of classifier implementation, SVM model from the Scikit-learn library was used. Specifically, SVC model for C-support vector classification was used. Prior to the hyperparameters tuning, the whole dataset was randomly shuffled. Then, the dataset was divided into train and test sets in the 80:20 ratio. Test set was used for the hyperparameters selection. Hyperparameters of the model were obtained by using ten-fold cross-validation technique, individually for different AOI layouts. Obtained parameters giving best results are presented in the table 5.2.

Training process

As during the process of hyperparameters tuning using a ten-fold cross-validation was observed the model is capable of generalization on training set, the model was retrained solely on the training set without the use of a validation set. Final model was evaluated using the test set.

Table 5.2: Parameters of the SVM model and number of principal components for different number of maneuver classes. *PC* stands for principal components.

AOI Layout	**PC**	**SVM Parameters**		
		C	gamma	kernel
Mid-and-Vert	24	4	scale	rbf
Up-and-Down	30	50	scale	rbf

5.6.3 Flight Maneuver Classification

In the second step of the classification process, maneuver class is inferred. Overall, two classification models were trained, each for one proficiency class. Each of two models were trained solely on data of the respective proficiency class. In the inference process, the model is selected according to the result of the proficiency classification.

For the maneuver classification, HMMs were utilized. They were selected due to their ability of processing data series, which are the form of fixation sequences. Specifically, the first computational problem of the HMM is utilized, that is, the determination of the probability the model generates given series of observations.

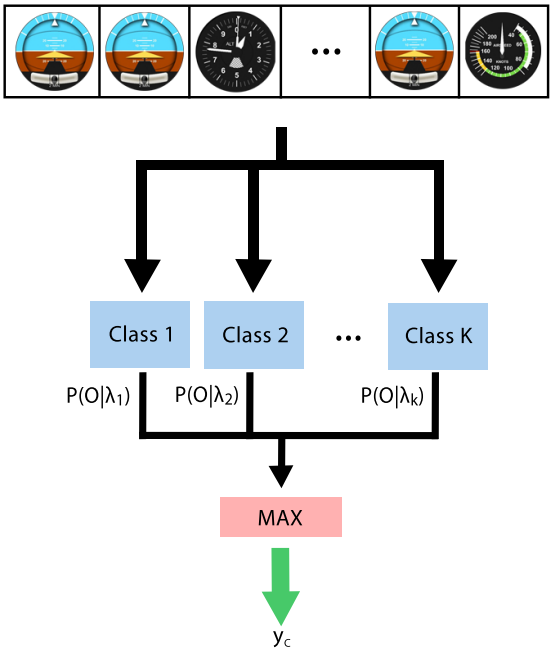


Figure 5.12: Scheme of a single maneuver classification model for K maneuver classes.

As has been mentioned, two independent maneuver classification models were implemented. One for each level of proficiency. These two models further consist of multiple HMM models, each representing a respective maneuver class. Scheme of a single classification model with respective HMM models for K maneuver classes is depicted in figure 5.12.

Given an unknown series of fixation observations O, every HMM model λ_k outputs the log-likelihood $P(O|\lambda_k)$ the model generates the observation sequence. These probabilities are compared, and maneuver represented by the model giving the highest log-likelihood is considered as correct classification. In this thesis, it is considered that every unknown input series was generated by one of the HMM models, i.e. no other maneuver classes than those on which the maneuver classifier was trained are considered. Otherwise, appropriate techniques validating classifier confidence of the inference would need to be implemented.

Model and Hyperparameters Selection

Since the set of possible observations is finite, when each of the observation represents one of defined AOIs, a `MultinomialHMM()` model from the Hmmlearn library is selected in the implementation. Every AOI in the input observation sequence is represented by an unique numerical ID. Every HMM model is from the implementation perspective, considered as a "black box", where the number of hidden states does not correspond to the number of maneuver classes. Thus, unsupervised learning is utilized, where no prior knowledge about the emission, transition and initial distributions are known. All these parameters are subject of learning procedure.

Regarding the model hyperparameters, the most important which had to be set prior to the training process was the number of hidden states. Since every HMM model represents a single maneuver model, the number of hidden states is not related to the number of maneuver classes. Thus, some optimal number had to be found and selected. Based on the experiments, 7 hidden states were selected as an optimal trade-off between the low and high complexity of the model. Regarding other relevant hyperparameters, maximum number of iterations of the EM (Baum-Welsch) algorithm was set to 15 and convergence threshold to 0.01.

Training process

Each HMM model was trained solely on the series related to the respective maneuver class. For each maneuver, 70 % of samples were used for training and 30 % was reserved for the evaluation process. Due to a limited size of the dataset, validation dataset consited of randomly picked samples from the training set. Prior to learning phase of each model, datasets of each maneuver class had been randomly shuffled. Then, during the training phase of each maneuver model, multiple HMM models were trained, while the one with the best score on the validation data was selected as the final model. Overall, 50 models were trained and validated, on the hold-out validation set. Score was represented by the log-likelihood with which the model generates the series of observations from the validation set.

5.6.4 Classification Framework Interface

Simple command line interface of the classifier was implemented, to provide access for learning, validation and inference procedures. Entry point of the classifier is `classifier.py` Python script. It requires two positional arguments:

- `N` — Number of maneuvers to be classified. Three possible values are supported: 2, 4 or 6 maneuver classes.
- `M` — Operating mode. Three modes are available: training (`T`), evaluation (`E`) and inference (`I`).

Training Mode

In training mode, individual classifiers are trained. To select training mode, positional argument `T` should be set. Multiple flags are available for training mode:

- `lp DATASET_PATH [M_EXPORT_PATH] [E_EXPORT_PATH]` — Trains proficiency classifier model. It requires dataset directory path (`DATASET_PATH`) to be set. If model (`M_EXPORT_PATH`) and PCA (`E_EXPORT_PATH`) export paths are set, it stores model in the respective files using `pickle` module for serialization.
- `lm E_DATASET_PATH A_DATASET_PATH [EXPORT_PATH]` — Trains maneuver classifier models. It requires two dataset — "experienced" class maneuvers (`E_DATASET_PATH`) and "amateur" class maneuvers (`A_DATASET_PATH`). If one of the paths is set to '_', respective classifier is not trained. If export path (`EXPORT_PATH`) is set, it stores both models into the .JSON file format. It contains all parameters and hyperparameters of all HMM models from both classifiers.
- `e` — Provides evaluation of individual classifiers at the end of the training processes. In this case, training datasets are split into train and test sets. Otherwise, loaded datasets are considered as training sets.

Evaluation Mode

In this mode, classifiers are evaluated. Script performs individual evaluation of selected classifiers. Evaluation provides following information:

- Precision,
- Recall,
- F1 Score,
- Confusion matrix

Confusion matrix is printed on the standard output and also plotted in the matrix graph. Available flags are:

- `ep DATASET_PATH MODEL_PATH PCA_PATH` — Evaluates proficiency classifier. Besides the test dataset (`DATASET_PATH`), paths to classifier (`MODEL_PATH`) and PCA (`PCA_PATH`) models need to be specified.

- `em E_DATASET_PATH A_DATASET_PATH MODEL_PATH` — Evaluates maneuver classifiers. Paths to test datasets (`E_DATASET_PATH`, `A_DATASET_PATH`) and file containing parameters of classifier models (`MODEL_PATH`) are required. If one of the dataset paths are set to '_', respective classifier is not evaluated.

Inference Mode

This mode provides inference of proficiency and maneuver for a single input sequence. Output of the inference is a tuple in the form of (y_L, y_M), where y_L denotes proficiency level and y_M indicates inferred maneuver from inferred proficiency class.

Required flags are:

- `i INPUT` — Input file containing sequence to be inferred.

- `m PROF_PATH PCA_PATH MAN_PATH` — Loads trained models, required for inference. `PROF_PATH` is path to proficiency classifier model, `PCA_PATH` is path to fitted PCA model and `MAN_PATH` is path to file containing parameters of maneuver classifier models.

Chapter 6

Evaluation

In this chapter, results of evaluation of proficiency and maneuver classifiers under various conditions are provided. Overall, three different cases were tested and evaluated. At first, classification of two classes of maneuvers, longitudinal and lateral, were tested. Proficiency level classifier using Mid-and-Vert layout was also evaluated. Then, two maneuver classes were split into classes corresponding to 4 basic flight maneuvers including straight-and-level flight, level turn, descent and climb. This time, proficiency level was evaluated on Up-and-Down layout. Finally, all six flight maneuvers performed by all participants were classified including vertical turns.

Results of individual cases include confusion matrices and four common evaluation metrics utilized to evaluate classifier models — precision, recall, F1 score and accuracy in tabular form.

6.1 Longitudinal and Lateral Maneuvers

In the first evaluation experiment, the simplest classification case was tested. Maneuvers were divided into two classes — longitudinal and lateral maneuvers. Longitudinal class includes straight-and-level flights, climbs and descents, whereas level turns were representatives of lateral maneuvers. In this evaluation setup, Mid-and-Vert AOIs layout was used.

At first, proficiency level classifier based on the SVM was tested. Confusion matrix in figure 6.1 presents ratios of correct and incorrect classifications for each class. Over 90 % of both classes were correctly assigned. Overall, 91.2 % accuracy was reached. Table 6.1 hows results of precision, recall and F1 score metrics.

Table 6.1: Evaluation statistics of proficiency classification with Mid-and-Vert layout of AOIs used.

Proficiency Level	**Precision**	**Recall**	**F1 Score**	**Accuracy**
Experienced	0.958	0.862	0.908	0.912
Amateur	0.875	0.962	0.917	

Looking at two-way transition matrices samples (figures B.3, B.4) from initial climb phase for both classes, different distribution of individual AOI pairs can be observed. In pilot case, transition matrix is sparse and more structured.

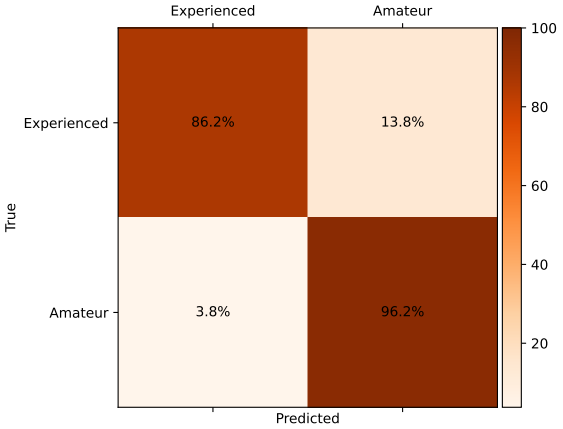
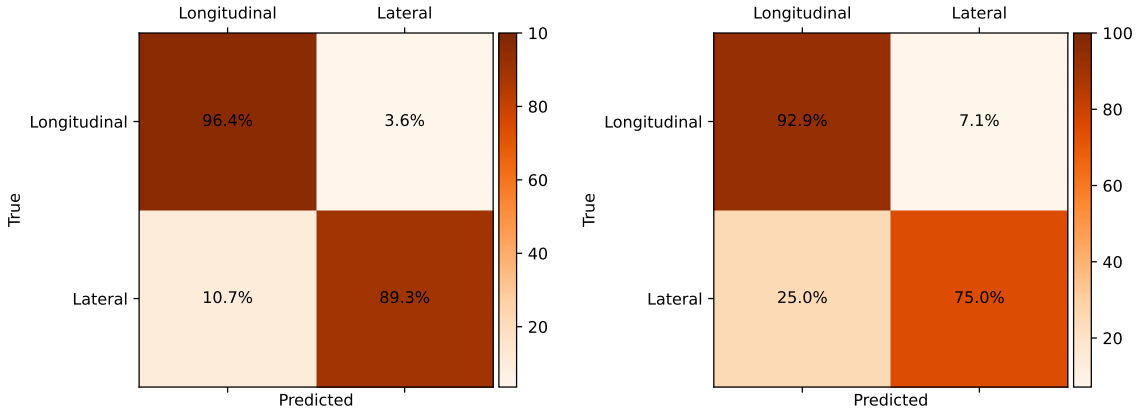


Figure 6.1: Confusion matrix for proficiency classifier with Mid-and-Vert layout of AOIs used.

Results of flight maneuvers classification can be seen in figure 6.2. Regarding class "experienced", majority of maneuvers were correctly inferred, reaching a total accuracy of 93 %. In case of the class "amateur", only 84 % longitudinal and lateral maneuvers were correctly assigned. Samples of emission matrices for both classes can be seen in figures B.1 and B.2. In both cases, effects of roll section division can be clearly observed. Much stronger utilization of roll indicator is present within class "experienced", whereas amateurs used mainly the HI for lateral control.

Table 6.2: Evaluation statistics of longitudinal and lateral maneuvers classification with Mid-and-Vert layout of AOIs used.

Maneuver	**Experience** (Acc.: 0.928)			**Amateur** (Acc.: 0.839)		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Longitudinal	0.900	0.964	0.931	0.788	0.929	0.852
Lateral	0.962	0.893	0.926	0.913	0.750	0.824



(a) Class "experienced"

(b) Class "amateur"

Figure 6.2: Confusion matrices for longitudinal and lateral maneuvers with Mid-and-Vert layout of AOIs used.

6.2 Classification of 4 Basic Flight Maneuvers

In another evaluation case, maneuvers were divided into four classes — straight-and-level flight, level turns, climb and descents. In this case, vertical turns were not included in the dataset. For proficiency classification, Up-and-Down AOI layout was used. Looking at confusion matrix in figure 6.3 and evaluation metrics in the table 6.4, we can see similar accuracy to the previous case. Figures B.7 and B.8 show samples of utilization of instruments during initial climb phase for both classes.

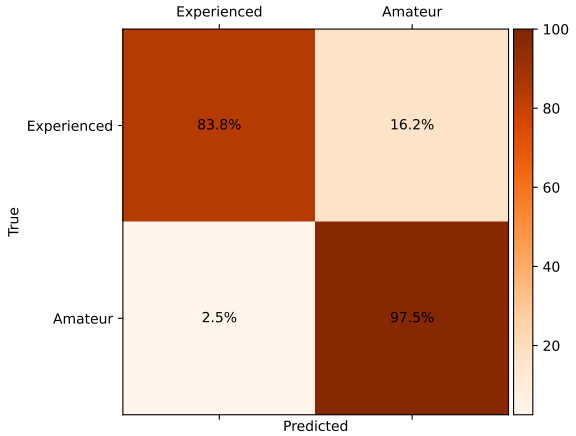


Figure 6.3: Confusion matrix for proficiency classification with Up-and-Down layout of AOIs used.

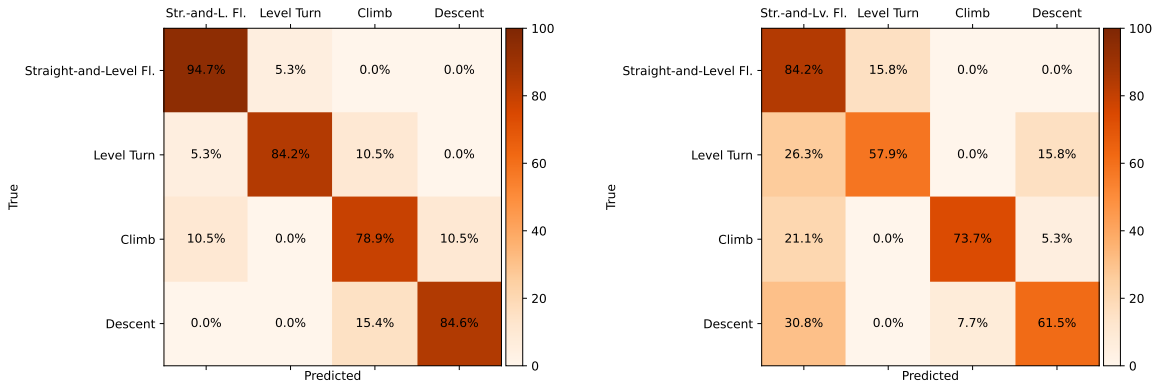
Regarding the maneuver classification, classifier for class "experienced" correctly classified 86 % of tested samples. As can be seen in figure 6.4a, confusion matrix shows, that

Table 6.3: Evaluation statistics of proficiency classification with Up-and-Down layout of AOIs used.

Proficiency Level	Precision	Recall	F1 Score	Accuracy
Experienced	0.985	0.838	0.905	0.906
Amateur	0.848	0.975	0.907	

longitudinal maneuvers were mainly misclassified to another longitudinal maneuvers. Few samples of level turns were also incorrectly inferred.

Significantly low accuracy was reached on the "amateur" class classifier. Due to probably more unstructured and more variable scanning strategies across individual participants, the classifier were able to reach only 70 % of accuracy. We can see in figure 6.4b, that as in the case of experienced pilots, longitudinal maneuvers namely climb and descent, were mainly misclassified to another two longitudinal maneuvers. This may be caused by the fact, that amateur pilots struggled to maintain the altitude during the straight-and-level flight, so there were intensive cross-checks of instruments indicating climbing/descending trends and values. In case of level turns, higher number of misclassifications could be caused by higher utilization of heading indicator rather than roll indicator. Utilization of individual AOIs during individual maneuvers for both classes are presented in figures B.5 and B.6.



(a) Class "experienced"

(b) Class "amateur"

Figure 6.4: Confusion matrix for 4 basic flight maneuvers with Up-and-Down layout of AOIs used.

6.3 Classification of 6 Basic Flight Maneuvers

Last evaluation case was dedicated to test ability of classifiers to distinct between 6 basic flight maneuvers including vertical turns divided into two classes — climbing turn and descending turn. No more validation of proficiency classifier was performed, as the same layout as in the previous case was used. Results of the evaluation can be observed in figures B.9 and 6.5 .

Table 6.4: Evaluation statistics of 4 basic flight maneuvers classification with Up-and-Down layout of AOIs used.

Maneuver	**Experienced** (Acc.: 0.857)			**Amateur** (Acc.: 0.700)		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Straight-and-Level Fl.	0.857	0.947	0.900	0.552	0.842	0.667
Level Turn	0.941	0.842	0.889	0.786	0.579	0.667
Climb	0.789	0.789	0.789	0.933	0.737	0.824
Descent	0.846	0.846	0.846	0.667	0.615	0.640

As you can see, introduction of vertical turns brought a high uncertainty to the classification of turn maneuvers. Almost half of level turn samples were missclassified to vertical turns. Also some climb/descent samples were incorrectly assigned to newly introduced vertical turns. Climbs were mistaken for climbing turns, while descents for descending turns.

These misclassifications occurred probably due to a fact, that vertical turn as a maneuver is a difficult for a coordination. Some pilots tends to divide the maneuver into two separate maneuvers. This may lead to a higher similarity between level and vertical turns, or climbs/descents and vertical turns. Thus, it is more difficult to distinguish between these maneuvers. Also because some level turns were not perfectly horizontal, so vertical trend indicators were more utilized, leading to biased classification towards vertical turns. Similar problem can be expressed for climbs/descents and vertical turns.

In case of the class "amateur", accuracy dropped down to 58 %. This may be the result of higher utilization of performance instruments rather than the AI as a main source of information. For this reason, almost half of the samples was misclassified. Figures B.9 and B.10 shows utilization of individual AOIs.

Table 6.5: Evaluation statistics of 6 basic flight maneuvers classification with Up-and-Down layout of AOIs used.

Maneuver	**Experienced** (Acc.: 0.760)			**Amateur** (Acc.: 0.583)		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Straight-and-Level Fl.	0.857	0.947	0.900	0.619	0.684	0.650
Level Turn	0.733	0.579	0.647	0.474	0.474	0.474
Climb	0.875	0.737	0.800	0.800	0.632	0.706
Descent	0.769	0.769	0.769	0.750	0.462	0.571
Climbing Turn	0.625	0.769	0.690	0.500	0.615	0.552
Descending Turn	0.667	0.769	0.714	0.400	0.615	0.485

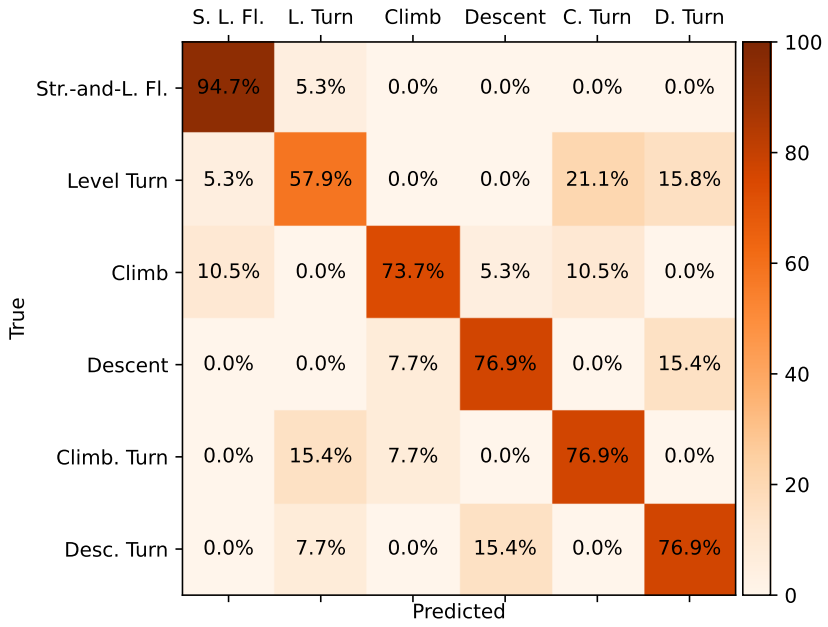


Figure 6.5: Confusion matrix for 6 basic flight maneuvers of class "experienced" with Up-and-Down layout used.

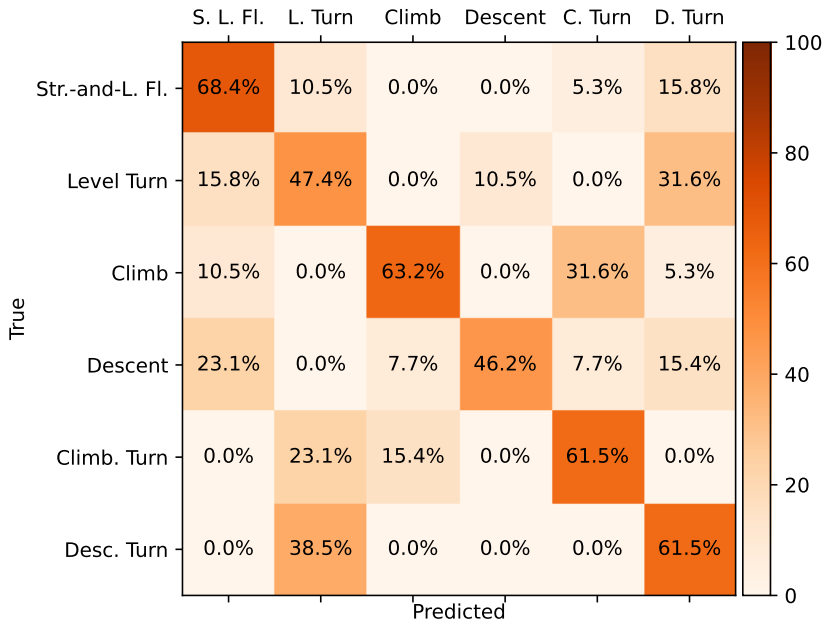


Figure 6.6: Confusion matrix for 6 basic flight maneuvers of class "amateur" with Up-and-Down layout of AOIs used.

Chapter 7

Conclusion

The goal of this thesis was to design and implement pilot proficiency classification framework, which would be able based on the pilot's gaze to (1) classify pilot's proficiency level, and (2) infer the maneuver which was performed by the pilot. Overall two proficiency levels — "experienced" and "amateur" were classified. Concerning flight maneuvers, three different cases were tested.

At first, state of the art study was made on possible techniques for analyzing visual scanning strategies, and machine learning approaches for classification. Based on the acquired knowledge and data analysis, design of the classification framework was made. Unlike other related studies, a different approach of selecting AOI layouts was used. Instead of considering individual instruments as a whole, some instrument AOIs were further divided into sub-parts to highlight certain properties, which can be observed from these instruments. In some cases, dynamic changes of these instruments also had to be taken into account.

The designed framework consists of two parts. First part infers pilot's proficiency level. Proficiency level classifier is based on the SVM algorithm, known as a simple yet powerful machine learning algorithm widely used for binary classification. Another advantage is its ability to deal with low datasets and higher number of features. Feature vector for the classification consists of values of two-way transition matrix of individual AOIs. Based on the inference result, appropriate maneuver classification model, for either "experienced" or "amateur" class, is selected. Each model consists of multiple HMM models, each representing one class of maneuvers. The HMMs for maneuvers classification were selected for their efficiency in processing sequential data, which was the case in this thesis.

Finally, multiple evaluation cases were performed to evaluate both classifiers and investigate, if there are any significant differences between both classes. Results confirmed, that differences between visual strategies of two tested classes exist. Proficiency level classifier reached high classification accuracy. Concerning maneuver classifiers, three different cases were evaluated. In first two cases (2 and 4 flight maneuver classes), classifier trained and evaluated on data of experienced pilots performed well. As vertical turns were introduced and became subject of classification, the classifier did have problem mainly to distinguish between level and vertical turns. However, this was probably due to a high complexity of vertical turns, so some pilots divided these flight maneuvers into two individual maneuvers — lateral and longitudinal. Overall, classification results of this class satisfied initial expectations. These results were reached mainly due to the logical division of original AI and VSI AOIs into individual sub-sections, each uncovering originally hidden information relevant for maneuver determination process.

Unlike the classifier of level class “experienced”, accuracy of the second class classifier was inferior. As the number of maneuver classes increased, classifier accuracy decreased. This was caused probably due to a high utilization of performance instruments rather than the AI, which was one of the subjects to AOI division. Thanks to this, the classifier were not able to distinct between individual flight maneuvers more accurately.

7.0.1 Potential Further Development

Subject of further development on the basis of achieved results could be the investigation of existence of differences in visual scanning strategies between pilots with different qualifications. Because an efficient scanning strategy is a sign of experience, this could lead to its utilization in pilot training. Instructor would have another tool of evaluation of pilot's skill in performance of individual maneuvers. Another utilization could be in the determination, if the pilot is in condition, in which he is able of performing desired tasks according to some standards and if the is not safety risk.

Bibliography

[1] Awad, M. and Khanna, R. Support Vector Machines for Classification. In:. January 2015, p. 39–66. ISBN 978-1-4302-5989-3.

[2] Baum, L. E. An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov Processes. In: *Inequalities III: Proceedings of the Third Symposium on Inequalities*. University of California, Los Angeles: Academic Press, 1972, p. 1–8.

[3] Bellenkes, A. H., Wickens, C. D. and Kramer, A. F. Visual scanning and pilot expertise: the role of attentional flexibility and mental model development. *Aviation, space, and environmental medicine*. 1997, 68 7, p. 569–79.

[4] Bishop, CH. M.. *Pattern Recognintion and Machine Learning*. Springer, 2006. ISBN 978-0387-31073-2.

[5] Boussemart, Y., Cummings, M. L., Fargeas, J. C. L. and Roy, N. Supervised vs. Unsupervised Learning for Operator State Modeling in Unmanned Vehicle Settings. *J. Aerosp. Comput. Inf. Commun.* 2011, vol. 8, p. 71–85.

[6] Brams, S., Hooge, I. T. C., Ziv, G., Dauwe, S., Evens, K. et al. Does effective gaze behavior lead to enhanced performance in a complex error-detection cockpit task? *PLoS ONE*. 2018, vol. 13.

[7] Brown, D. L., Vitense, H. S., Wetzel, P. A. and Anderson, G. M. Instrument scan strategies of F-117A pilots. *Aviation, space, and environmental medicine*. 2002, 73 10, p. 1007–13.

[8] Burkov, A. *The Hunder-Page Machine Learning Book*. Andryi Burkov, 2019. ISBN 978-1999579500.

[9] Federal Aviation Administration. *Instrument Flying Handbook*. Aviation Supplies & Academics, Inc., 2013. ISBN 978-1490414508. Available at: https://www.faa.gov/regulations_policies/handbooks_manuals/aviation/media/FAA-H-8083-15B.pdf.

[10] Federal Aviation Administration. *Pilot's Handbook of Aeronautical Knowledge*. Aviation Supplies & Academics, Inc., 2016. ISBN 978-1619544734. Available at: https://www.faa.gov/regulations_policies/handbooks_manuals/aviation/phak/media/pilot_handbook.pdf.

[11] Federal Aviation Administration. *Airplane Flying Handbook*. Aviation Supplies & Academics, Inc., 2021. ISBN 978-1619545120. Available at:

https://www.faa.gov/regulations_policies/handbooks_manuals/aviation/airplane_handbook/media/00_afh_full.pdf.

[12] Fitts, P. M., Milton, J. L. and Jones, R. E. *Eye Fixations of Aircraft Pilots. III. Frequency, Duration, and Sequence Fixations When Flying Air Force Ground-Controlled Approach System.* Air Materiel Command, november 1949.

[13] Gainer, C. A. and Obermayer, R. Pilot Eye Fixations While Flying Selected Maneuvers Using Two Instrument Panels. *Human Factors: The Journal of Human Factors and Ergonomics Society.* 1964, vol. 6, p. 485 – 501.

[14] Glaholt, M. G. Eye Tracking in the Cockpit: a Review of the Relationships between Eye Movements and the Aviators Cognitive State. In:. 2014.

[15] Goldberg, J. H. and Kotval, X. P. Computer interface evaluation using eye movements: methods and constructs. *International Journal of Industrial Ergonomics.* 1999, vol. 24, p. 631–645.

[16] Harris, C. R., Millman, K. J., Walt, S. J. van der, Gommers, R., Virtanen, P. et al. Array programming with NumPy. *Nature.* Springer Science and Business Media LLC. september 2020, vol. 585, no. 7825, p. 357–362.

[17] Harris, R. L., Glover, B. J. and Spady, A. A. Analytical techniques of pilot scanning behavior and their application. In:. 1986.

[18] Harris, R. L., Tole, J. R., Stephens, A. T. and Ephrath, A. R. Visual scanning behavior and pilot workload. *Aviation, space, and environmental medicine.* 1982, 53 11, p. 1067–72.

[19] Harris, D.. *Flight Instruments & Automatic Flight Control Systems* . 6th ed. Blackwell Science, 2004. Ground Studies for Pilots. ISBN 0-632-05951-6.

[20] Haslbeck, A., Schubert, E., Gontar, P. and Bengler, K. The relationship between pilots' manual flying skills and their visual behavior: a flight simulator study using eye tracking. In: *Advances in Human Aspects of Aviation.* 2012, p. 561–568.

[21] Haslbeck, A. and Zhang, B. I spy with my little eye: Analysis of airline pilots' gaze patterns in a manual instrument flight scenario. *Applied Ergonomics.* 2017, vol. 63, p. 62–71. ISSN 0003-6870. Available at: https://www.sciencedirect.com/science/article/pii/S000368701730073X.

[22] Hayashi, M. Hidden Markov Models for Analysis of Pilot Instrument Scanning and Attention Switching. In:. 2004.

[23] hmmlearn. *Tutorial* [online; last visited May. 13, 2022]. Available at: https://hmmlearn.readthedocs.io/en/latest/.

[24] Jakkula, V. R. Tutorial on Support Vector Machine (SVM). In:. 2011.

[25] Jukes, M.. *Aircraft display systems.* . AIAA, 2004. ISBN 1-56347-657-6.

[26] Kasarskis, P., Stehwien, J., Hickox, J., Aretz, A. and Wickens, C. Comparison of Expert and Novice Scan Behaviors During VFR Flight. *Proceedings of the 11th International Symposium on Aviation Psychology.* january 2001.

[27] Katoh, Z. Saccade amplitude as a discriminator of flight types. *Aviation, space, and environmental medicine*. March 1997, vol. 68, no. 3, p. 205—208. ISSN 0095-6562.

[28] Khodabandelou, G., Hug, C., Deneckère, R. and Salinesi, C. Supervised vs. Unsupervised Learning for Intentional Process Model Discovery. In: *BMMDS/EMMSAD*. 2014.

[29] Krejtz, K., Duchowski, A. T., Krejtz, I., Szarkowska, A. and Kopacz, A. Discerning Ambient/Focal Attention with Coefficient K. *ACM Transactions on Applied Perception (TAP)*. 2016, vol. 13, p. 1 – 20.

[30] learn.org scikit. *Developer's Guide* [online; last visited May. 13, 2022]. Available at: https://scikit-learn.org/stable/.

[31] Li, W.-C., Chiu, F.-C., Kuo, Y. shin and Wu, K.-J. The Investigation of Visual Attention and Workload by Experts and Novices in the Cockpit. In: *HCI*. 2013.

[32] Lounis, C., Peysakhovich, V. and Causse, M. Visual scanning strategies in the cockpit are modulated by pilots' expertise: A flight simulator study. 2021.

[33] MLMath.io. *Math behind SVM(Support Vector Machine)*. Available at: https://ankitnitjsr13.medium.com/math-behind-svm-support-vector-machine-864e58977fdb.

[34] Mumaw, R. J., Nikolic, M. I., Sarter, N. B. and Wickens, C. D. A Simulator Study of Pilots' Monitoring Strategies and Performance on Modern Glass Cockpit Aircraft. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. 2001, vol. 45, p. 73 – 77.

[35] mycfibook.com. *Basic Instrument Maneuvers* [online; last visited Dec. 13, 2021]. Available at: https://mycfibook.com/lesson-plans/basic-instrument-maneuvers/.

[36] Nibbeling, N., Oudejans, R. R. D. and Daanen, H. A. M. Effects of anxiety, a cognitive secondary task, and expertise on gaze behavior and performance in a far aiming task. *Psychology of Sport and Exercise*. 2012, vol. 13, p. 427–435.

[37] Norris, J. *Markov chains (No. 2)*. Cambridge university press, 1998.

[38] Pallet, E. H. J.. *Aircraft Instruments - Principles and Applications*. 2nd ed. Pearson, 2009. Ground Studies for Pilots. ISBN 978-8 1-317-28 13-0.

[39] pandas.pydata.org. *About Pandas* [online; last visited May. 13, 2022]. Available at: https://pandas.pydata.org/about/.

[40] Pennington, J. E. Single pilot scanning behavior in simulated instrument flight. In:. 1979.

[41] psychopy.org. *Pupil Labs - Core* [online; last visited May. 13, 2022]. Available at: https://psychopy.org/api/iohub/device/eyetracker_interface/PupilLabs_Core_Implementation_Notes.html.

[42] Rabiner, L. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*. 1989, vol. 77, no. 2, p. 257–286.

[43] Rayner, K. Eye movements in reading and information processing: 20 years of research. *Psychological bulletin*. 1998, 124 3, p. 372–422.

[44] Reingold, E. M. and Sheridan, H. Eye movements and visual expertise in chess and medicine. In: *Oxford handbook on eye movements*. Oxford, UK: Oxford University Press, 2011, vol. 528.

[45] Ren, Q., Li, M. and Han, S. Tectonic discrimination of olivine in basalt using data mining techniques based on major elements: a comparative study from multiple perspectives. *Big Earth Data*. vol. 3.

[46] Salvucci, D. and Goldberg, J. Identifying fixations and saccades in eye-tracking protocols. In:. January 2000, p. 71–78.

[47] Schrum, J. *Cessna 172 cockpit*. Available at: https://pinterest.com/pin/145311525454878283/.

[48] Shapiro, K. L. and Raymond, J. E. Training of efficient oculomotor strategies enhances skill acquisition. *Acta Psychologica*. 1989, vol. 71, no. 1, p. 217–242. ISSN 0001-6918. Available at: https://www.sciencedirect.com/science/article/pii/0001691889900103.

[49] Sharma, S. *Kernel Trick in SVM*. Available at: https://medium.com/analytics-vidhya/how-to-classify-non-linear-data-to-linear-data-bb2df1a6b781.

[50] Suthaharan, S. *Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning*. 1stth ed. Springer Publishing Company, Incorporated, 2015. ISBN 148997640X.

[51] Vickers, J. N. and Lewinski, W. Performing under pressure: Gaze control, decision making and shooting performance of elite and rookie police officers. *Human Movement Science*. 2012, vol. 31, no. 1, p. 101–117. ISSN 0167-9457. Available at: https://www.sciencedirect.com/science/article/pii/S0167945711000571.

[52] Wickens, C. D. and Alexander, A. L. Attentional Tunneling and Task Management in Synthetic Vision Displays. *The International Journal of Aviation Psychology*. Taylor & Francis. 2009, vol. 19, no. 2, p. 182–199.

[53] Yang, J. H., Kennedy, Q., Sullivan, J. and Fricker, R. D. Pilot performance: assessing how scan patterns & navigational assessments vary by flight expertise. *Aviation, space, and environmental medicine*. 2013, 84 2, p. 116–24.

Appendix A

Content of the Enclosed Medium

- `/doc` — documentation source codes
- `/src` — classifier source codes
- `/models` — trained models
- `xrutad00_DP.pdf`
- `README.md`

Appendix B

Evaluation Charts

Longitudinal and Lateral Maneuvers

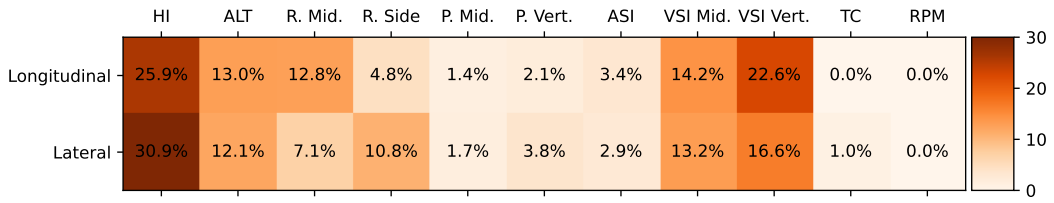


Figure B.1: Emission matrix of 2 maneuver classes from a sample flight performed by an amateur pilot representative. Mid-and-Vert layout of AOIs is used. High utilization of HI, and performance instruments overall, in both maneuver classes can be observed.

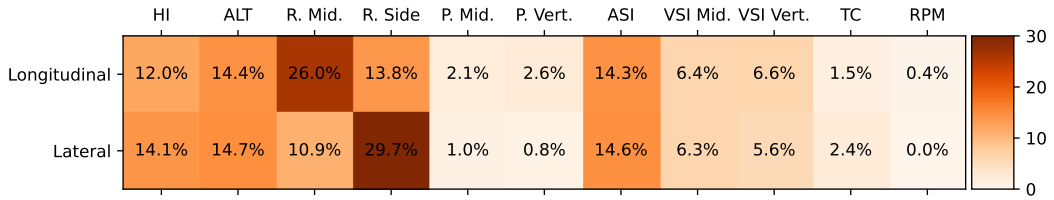


Figure B.2: Emission matrix of 2 maneuver classes from a sample flight performed by an experienced pilot representative. Mid-and-Vert layout of AOIs is used. Both classes can be easily distinguished based on the utilization of individual roll indicator sections.

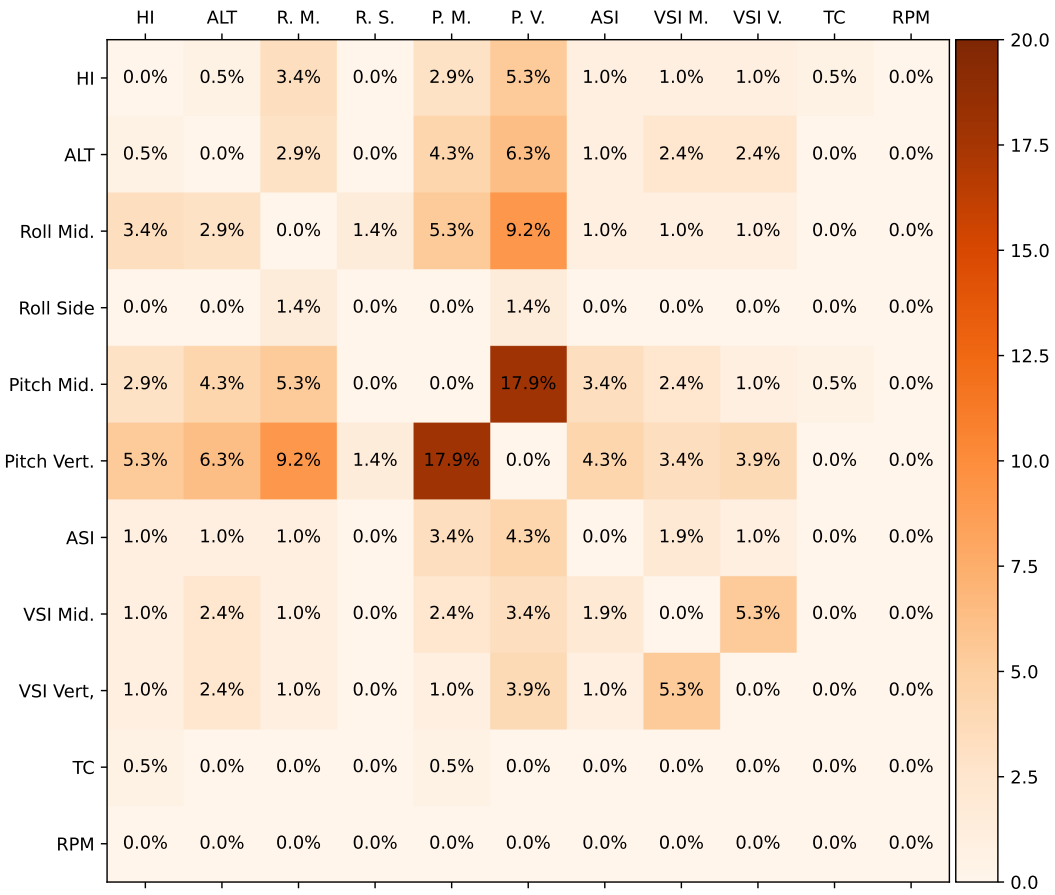


Figure B.3: Transition matrix between individual AOIs from a sample of initial climb phase performed by an amateur pilot representative. Mid-and-Vert layout of AOIs is used.

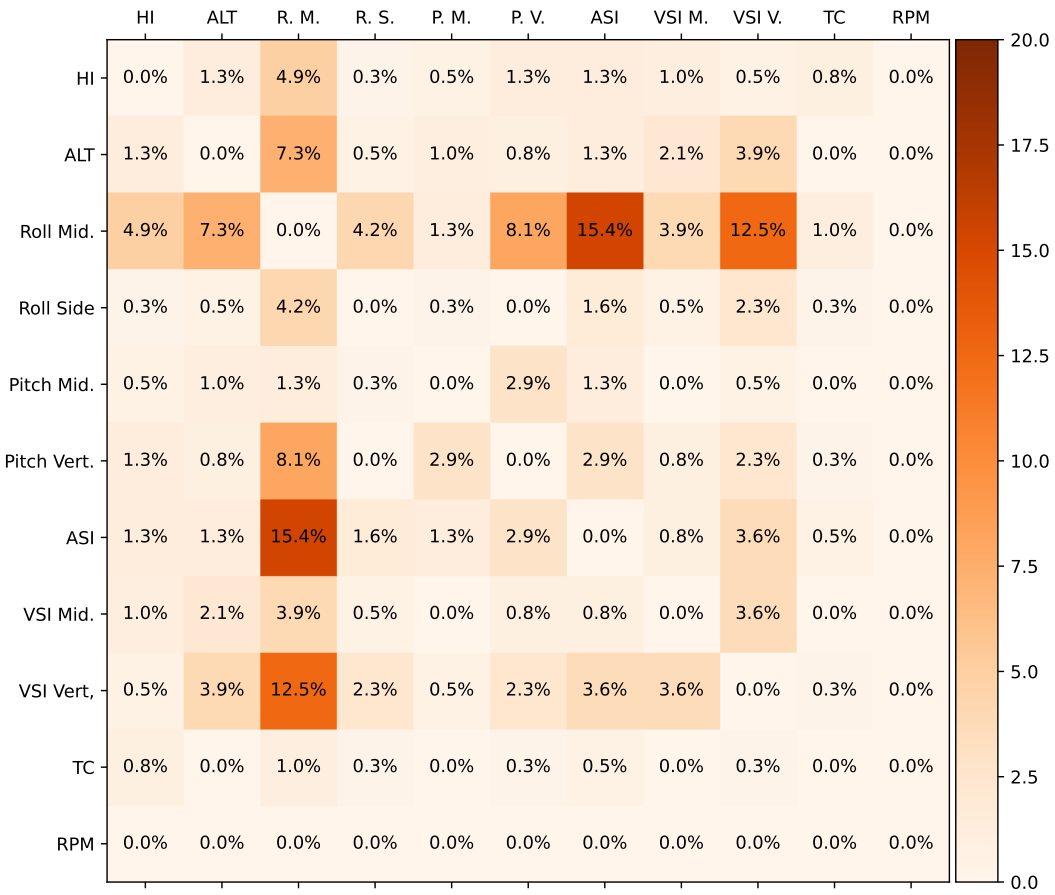


Figure B.4: Transition matrix between individual AOIs from a sample of initial climb phase performed by an experienced pilot representative. Mid-and-Vert layout of AOIs is used.

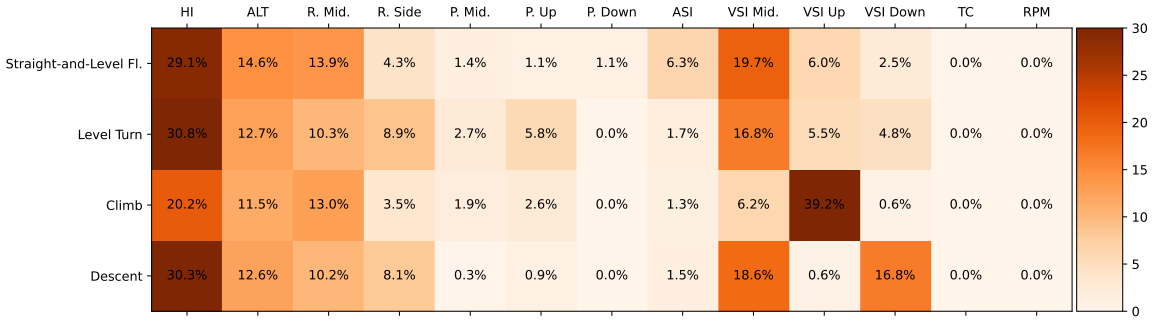


Figure B.5: Emission matrix of 4 maneuver classes from a sample flight performed by an amateur pilot representative. Up-and-Down layout of AOIs is used. There is no significant difference between level turn and other longitudinal maneuvers. Climb and descent classes can be clearly distinguished based on VSI AOIs.

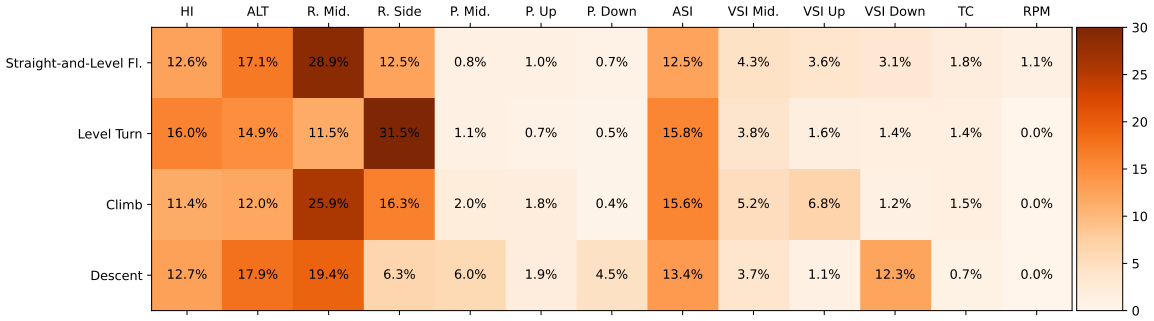


Figure B.6: Emission matrix of 4 maneuver classes from a sample flight performed by an experienced pilot representative. Up-and-Down layout of AOIs is used. Climb and descent classes can be distinguished mainly based on VSI AOIs.

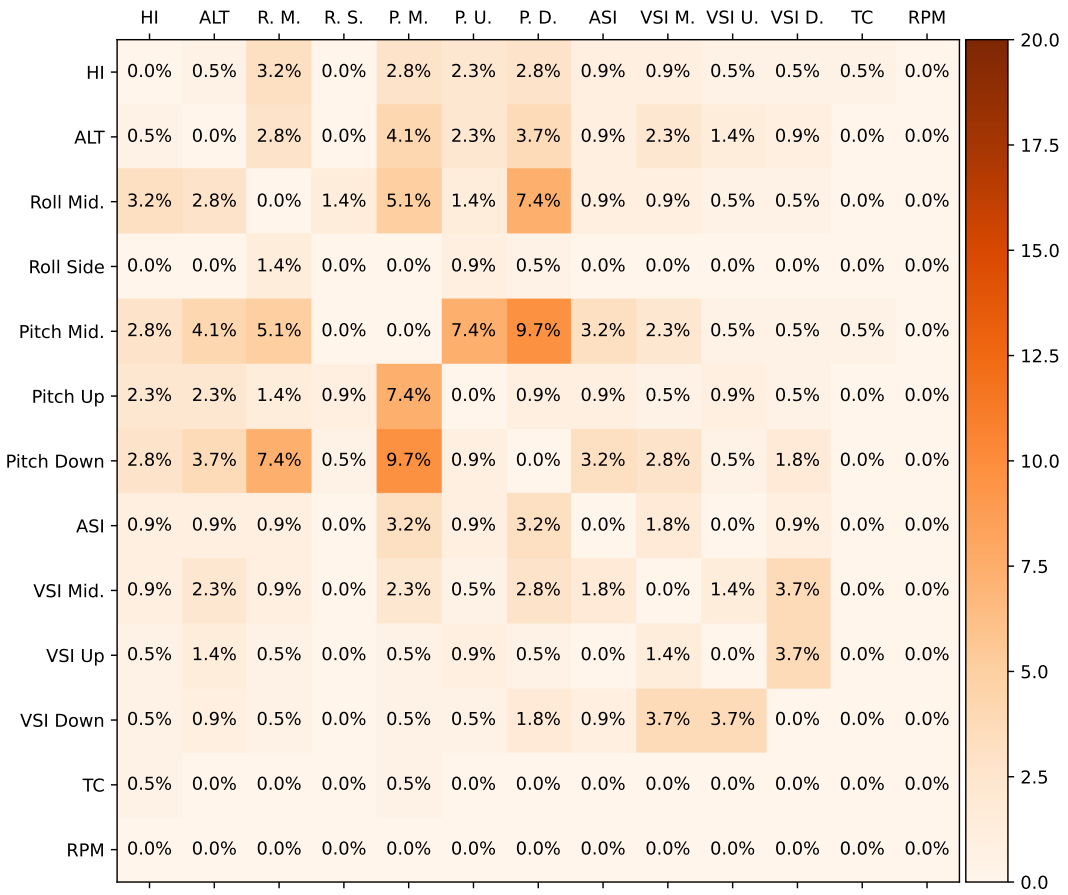


Figure B.7: Transition matrix between individual AOIs from a sample of initial climb phase performed by an amateur pilot representative. Up-and-Down layout of AOIs is used.

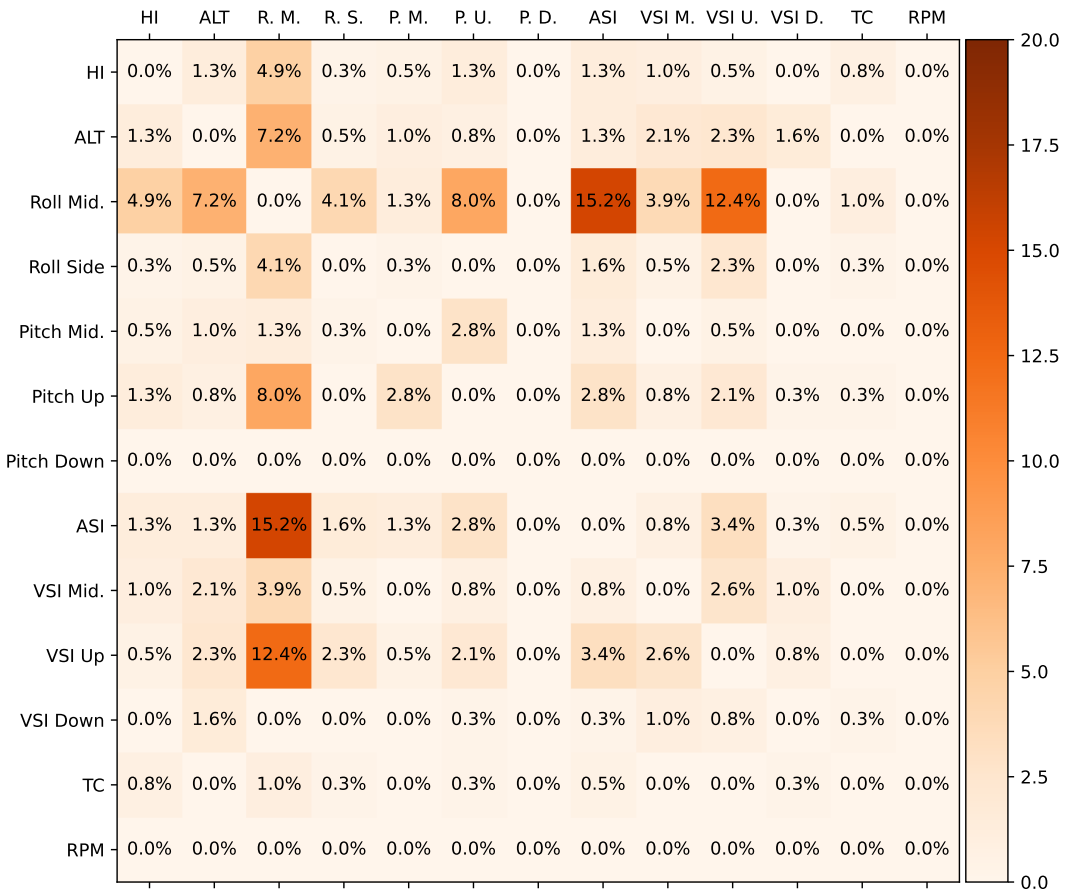


Figure B.8: Transition matrix between individual AOIs from a sample of initial climb phase performed by an experienced pilot representative. Up-and-Down layout of AOIs is used.

Classification of 6 Flight Maneuvers

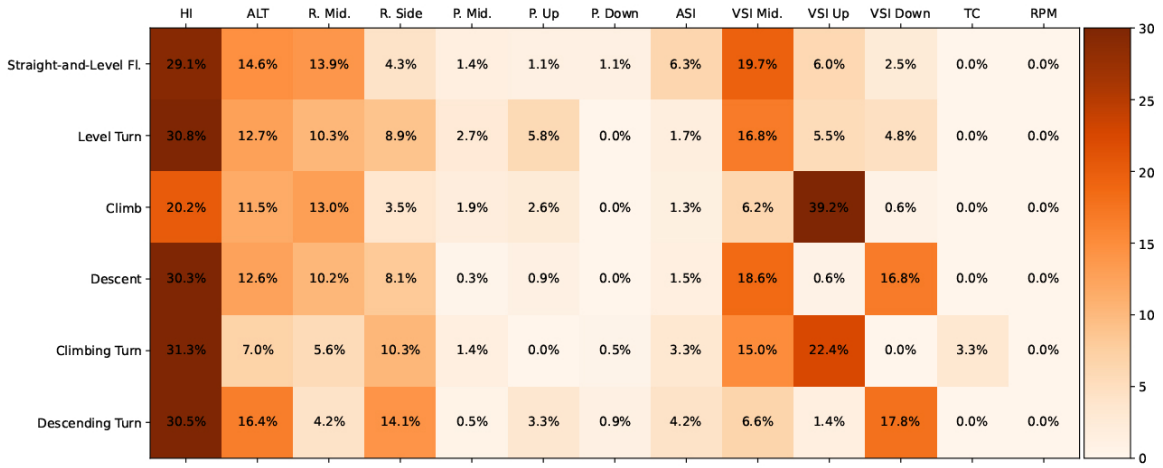


Figure B.9: Emission matrix of 6 maneuver classes from a sample flight performed by an amateur pilot representative. Up-and-Down layout of AOIs is used. VSI AOIs have strongs ability to distinguish between ascending and descending manevers.

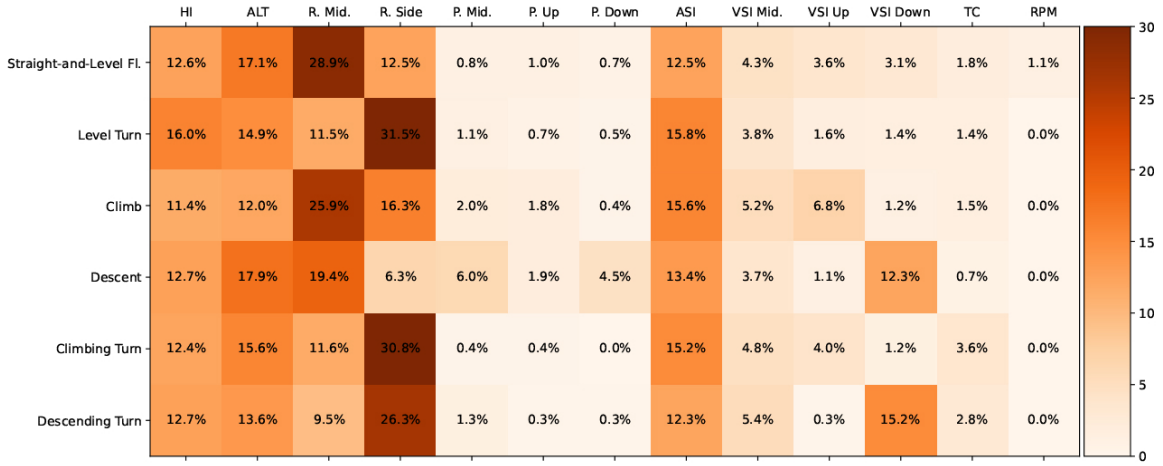
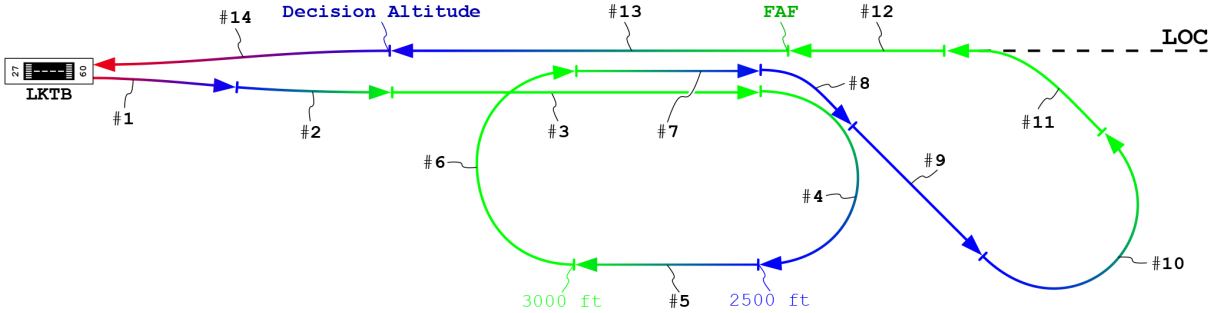


Figure B.10: Emission matrix of 6 maneuver classes from a sample flight performed by an experienced pilot representative. Up-and-Down layout of AOIs is used. VSI AOIs have decent ability to distinguish between ascending and descending manevers.

Appendix C

Designed Flight Pattern

#1 Take-Off & Airborne • Lifting off **45** -> **75** kts • Retracting Flaps • IAS **75** kts • ALT **800** -> **1000** ft • HDG **093°**	**#3 Level Flight** • IAS **100** kts • ALT **3000** ft • HDG **090°** • VS **0** ft/min • Duration **2** min	**#5 Climb** • IAS **75** kts **!** • ALT **2500** -> **3000** ft • HDG **270°** • VS **+500** ft/min **!!**	**#7 Descent** • IAS **75 > 90** kts **!** • ALT **3000** -> **2500** ft • HDG **090°** • VS **-500** ft/min **!!**
#2 Init Climb • IAS **75** kts • ALT **1000** -> **3000** ft • HDG **090°** • VS **+500** ft/min **!**	**#4 Right Descending Turn** • ALT **3000** -> **2500** ft • VS **-500** ft/min **!!** • IAS **90** kts **!** • HDG **090°**-> **270°** • Turn Rate **3** °/sec **!!**	**#6 Right Horizontal Turn** • IAS **100** kts • ALT **3000** ft • HDG **270°**-> **90°** • VS **0** ft/min • Turn Rate **3** °/sec **!!**	**#8 Right Horizontal Turn** • ALT **2500** ft • VS **0** ft/min • IAS **90** kts **!** • HDG **090°**-> **135°** • Turn Rate **3** °/sec **!**



#9 Level Flight • IAS **100** kts • ALT **2500** ft • HDG **135°** • VS **0** ft/min • Duration **1** min	**#11 LOC Capture** **NOT CLASSIFIED**	**#13 ILS Approach** **NOT CLASSIFIED**	**#15 Touchdown** **NOT CLASSIFIED**
#10 Left Climb Turn 180° • IAS **90** kts **!** • ALT **2500** -> **3000** ft • HDG **135°**-> **315°** • VS **500** ft/min **!!** • Turn Rate **3** °/sec **!!**	**#12 GS Capture** **NOT CLASSIFIED**	**#14 Decision Altitude** **NOT CLASSIFIED**	

Figure C.1: Designed flight pattern with respective constraints of individiual maneuvers.

Appendix D

Simulation Framework Photos

Figure D.1: Visualization subsystem of the SimStar simulation framework.

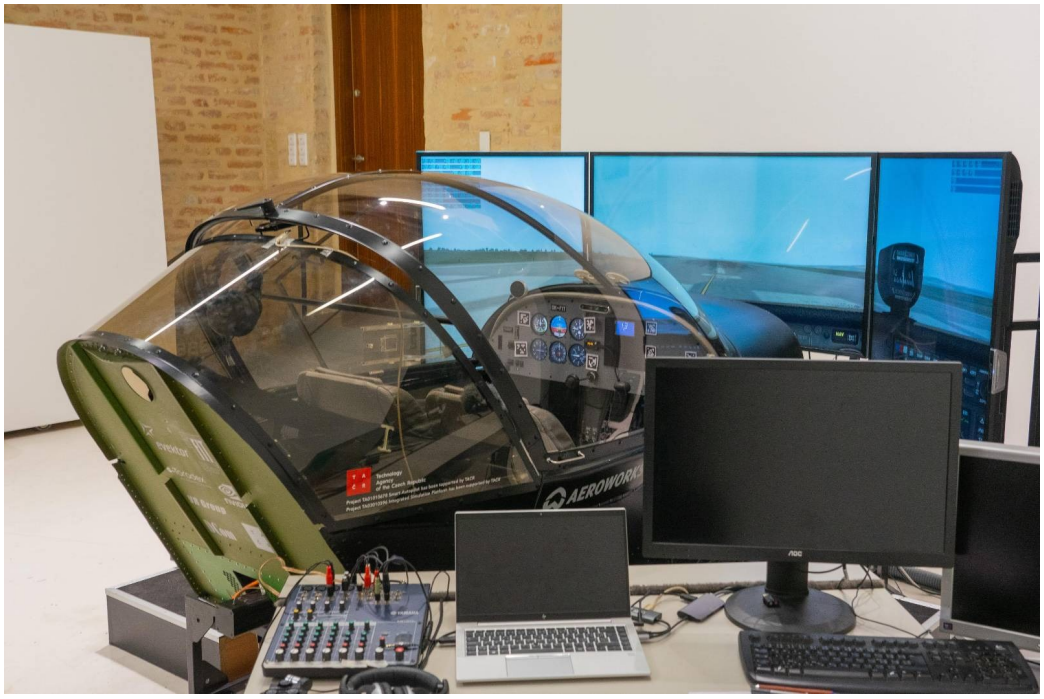


Figure D.2: Control background of the SimStar simulation framework.

Figure D.3: Instrument panel of the SimStar simulation framework.

Figure D.4: Simulator operated by one of the participants.