Contents

Appendix M AVOIDDS Case Study Summary

This appendix presents a summary version of the detailed Appendix I. The reader is encouraged to view the chapter as a high-level summary of the main activities in each stage. The detailed know-how (SECoTs and detailed steps) for each stage should be examined in Appendix I. An example of CuneiForm validation final report can be viewed in Appendix D. A corresponding appendix section is provided for reference if the reader wishes to understand how an activity translates into practice. By reviewing Appendix I and D, the reader can fully appreciate the actual contribution of this PhD.

M.1 Introduction

This case study is informed by work conducted in a similar context [1], where researchers evaluated the environmental and operational factors essential to an aircraft perception system. In that study, the authors carefully selected factors based on environmental variability and operational relevance, such as:

- **Weather Conditions**: clear, high cirrus, scattered clouds, broken clouds, overcast, stratus.
- **Aircraft Types**: Cessna Skyhawk, King Air C90, Boeing 737.
- **Time of Day**: morning, midday, afternoon, and late afternoon.
- **Geographic Regions**: locations such as Palo Alto, Reno, Boston, and Oshkosh.

> ac: intruder aircraft type ('Cessna Skyhawk', 'Boeing 737-800` , or ` King Air C90`)
>
> clouds: cloud cover (0 = Clear, 1 = Cirrus, 2 = Scattered, 3 = Broken, 4 = Overcast)
>
> local_time_sec: seconds since midnight local time on January 1st

Building on these findings, we conduct a comprehensive analysis using the AIC approach, identifying 102 factors relevant to aircraft detection, including several additional dimensions beyond the environmental and operational scope initially defined. This broader perspective, achieved through AIC, allows us to expand on the dataset's potential, capturing more nuanced influences and creating a richer, detailed dataset that is more effective for real-world applications.

M.1.1 A Thought Experiment

To validate our work, we set up a challenge in which we evaluated our approach and the approach of prior research, assuming it was representative of a typical state-of-the-art approach to dataset gathering. The idea is we ask a question, and see whether we are able to answer throughout our approach, then reflect on how the prior research may or may not have been considered during their data gathering stage.

The validation activity is based on the premise that "if a scenario was not included in the datasets, the dataset architect has never considered that scenario, and that the process did not enable the architect to consider it due to less comprehensive systems thinking" . This means we will apply the AIC systems approach to identify an example scenario and then ask questions about each scenario.

Does the AVOID Dataset include those scenarios? If not, then:

1- Those scenarios are Unknown, some of which are Black Swan events, which were not considered during the development of AVOID Dataset. Assume we develop an AVP (Computerised perception-based mid-air collision avoidance system) using AVOID Dataset, and while an aircraft is deciding to land in an airport, at some height, a series of hot-air balloons appear within the view of the AVP. Can we trust the perception system trained in AVOIDDS? If the AVP fails, such an event will be called a Black Swan event.
2- The AVOID Dataset creators' approach did not help them imagine those scenarios. Hence, our approach is possibly more helpful for predicting Black Swan events.

The answer to this thought experiment is explained in section 9.3.2.

M.2 Stage 1: Uncertainty Problem Articulation and Operational Environment Modelling[1]

This stage focuses on systematically identifying and articulating the uncertainty of the problem domain complexity field. The primary purpose is to define the operational environment by recognising unsafe behaviours, modelling the complexity field, and refining the architect's predictive understanding of the system's context. At this stage, we discover potential Black Swans related to the operational domain's complexity. Not all predicted factors are Black Swans, but some carry a rare interaction that could be impactful.

[1] See section I.4 in appendix I.

Essentially, in this stage, we resolve the epistemic uncertainty problem and predict (as accurate as possible) the complexity field of the operational environment. In this initial stage, we focus on identifying and analysing the influential factors and underlying assumptions that define the problem domain for vision-based aircraft detection. We adopt SECoT 1 in section F.4. A thorough understanding of these factors is essential, as they determine the operational context within which aircraft detection systems must perform effectively.

The input to this stage is an agreed-upon brief and holistic definition of the problem. The architect is encouraged to assume the problem as a Confusing Complex. Which means every part of the description, even if it sounds familiar, that part is confusing, meaning the architect assumes the unclarity of composition, purpose and contribution to the overall problem. By doing so, the architect approaches the problem with less bias in perception and clear any preconceived ideas that may bias the overall design process subjectively. We revised the prior research paper and made our definition of the problem:

> Two or more aircraft come into unplanned contact during flight, leading to a possible mid-air collision, posing a significant safety threat. These incidents often result in severe damage or destruction due to high velocities and potential subsequent impacts. The by-passing aircraft is in constant motion, following a flight path that intersects with another aircraft.

The predictive thinking pipelines in Appendix I enable the architect to achieve the following activities:

M.2.1 Predictive Thinking Pipeline 1: Appreciate the Complex of the Problem Complexity Field

The architect begins by listing behaviours that contradict the expected order of the environment, leading to either confusion or direct risks. Each of the following is a predictive thought step in the activity SECoT. The output of each thought step is a prediction based on a set of general systems rules.

In the first step, we identified unsafe behaviours by analysing the initial problem definition to distinguish those unsafe problematic behaviours. Using the thought step, we can assert that the following are the primary unsafe problematic behaviours:

a) An aircraft deviates from its assigned path, intersecting with a nearby flight path.
b) An unauthorised or unplanned aircraft enters high-altitude airspace without coordinated communication, moving in a way that suggests erratic or unpredictable flight patterns.

Once we had an initial understanding, we then generated a descriptive image that visualises the unsafe behaviour. The idea here is to generate a depiction that aligns with our perspective of what the problem stereotypically looks like. In our case, we used DALL-E to generate an image of what we think a typical problem domain looks like. Using the thought step, we can assert that the following depiction faithfully captures what the problem looks like.

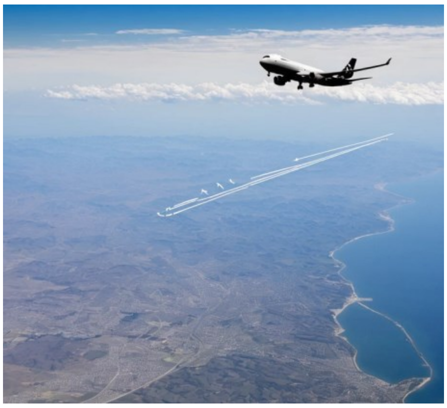


Figure M.1 Problem Environment Depiction

Then we defined the complex of the complexity field: with the above articulation and the help of depiction, now we can start to predict the most obvious components of the problem. Using the thought step, we can assert that the following is the most immediate composition of the complexity field: {ownship aircraft, by-passing aircraft, other aircraft, clouds, terrain (land), sea}.

Note: At this stage, we have only identified six factors. If we stop here and start designing or deriving requirements, we would miss more hidden factors due to our peripheral perception of the problem domain. Typically, systems engineers would reach this stage and start deriving requirements with the customer. By the end of this stage, you will see how many factors would have been missed if we had just done this.

In the next step (1.4), we attempted to group the apparent components into a potential system of systems in order to predict what PrimeP they may be serving. To be more objective about the behaviours of the components we have yet to predict, we need to determine the primary purpose of the main system of systems involved (supra-complexes). We use the term “complex” because we do not know exactly what they are. Utilising the thought step, we can assert that the following are systems of systems and PrimeP:

c) **Ownship Aircraft PrimeP:** Maintain safe operational flight and avoid mid-air collisions.
d) **Other Aircraft PrimeP:** Operate within shared airspace, adhering to established flight paths and protocols for collision avoidance.

e) **Environment:** neutral purpose that impacts the shared airspace between Ownship Aircraft and Other Aircrafts.

M.2.2 Predictive Thinking Pipeline 2: Appreciate the Complex of the Problem Complexity Field

In Predictive Thinking Pipeline 2, we focused on addressing the intricate complexities observed in airspace conflict management. Building on the holistic initial problem definition from Predictive Thinking Pipeline 1, we aimed to delve deeper into the interactions among various system elements. This phase required a systematic analysis of how these elements influence each other, allowing us to uncover the complexity arising from their interrelations.

The first step in this process was the Problem Interaction Analysis using the Actions Matrix. We mapped the interactions among the identified systems: the Ownship Aircraft, the AVP (Computerised Perception-Based Mid-Air Collision Avoidance System), By-passing Aircraft, Other Aircraft, and Environmental Factors. By applying the Actions Matrix approach, we defined the nature of these interactions through binary relationships. For instance, we identified that the AVP guides the Ownship Aircraft while the By-passing Aircraft can intersect its path. The following is a portion of the actions matrix we used to help us articulate the interactions among components:

Table M.1 Actions Matrix example

	Ownship ac	**AVP**	**By-passing ac**
Ownship ac		govern	avoid
AVP	guide		detect
By-passing ac	Intersect	Physically/visually complicate	

Moreover, we noted how environmental factors, such as clouds or terrain, might physically or visually complicate the navigation of both the Ownship and the AVP systems. As a result, we documented 24 key interactions, framed in the format of [source system][action][sink system], providing clarity on how each system affects the others. we classified the interactions into two types:

- Unsafe problematic situations.
- Beneficial / Non-problematic situations.

For example:

Table M.2 Problematic interactions classification

Unsafe problematic situations	Beneficial or non-problematic situations
1. AVP: 1.2 n10: [AVP][consider][Environment] 1.3 n12: [AVP][detect][By-passing ac]	**no beneficial situations identified**

The process in Table M.2 helped us to scrutinise our list of priority problems. Also helped us to examine if there are beneficial aspects in the problem we could capitalise on. With such contextualisation of the problem, we can now decide what to solve and drop. We did so using the Design problem selection process. Problem selection process encourages the architect to consult with stakeholders on what problems to solve and what to drop. For example:

Table M.3 Design problem selection process

Interaction	Potential Concern	Decision	Elaboration or Justification
n1: [AVP][guide][Ownship ac]	AVP guiding the ownship may require precise alignment with real-time conditions and safety protocols.	**To be solved**	Guidance by AVP is essential to avoid collisions; ensuring accuracy and responsiveness is a key safety priority.
n24: [Other ac][sense][By-passing ac]	Sensing of by-passing by other aircraft supports mutual avoidance.	**To be dropped**	Beneficial but outside the design scope focused on ownship's AVP system; can be coordinated with external safety protocols.

This stage lays the foundation for risk mitigation by articulating the uncertainty landscape and creating structured models that will inform system design and autonomous response planning.

M.2.3 Predictive Thinking Pipeline 3: Predict the Emergence of AIC Complexity Field for Detailed Operational Scenario Articulation

The primary focus was to understand the emergence of AIC complexity fields, analyse intricate interactions, and articulate potential operational scenarios that could arise from these complexities. This pipeline was built upon the previous phase, which helped us identify potential problems to address in the system and allowed us to dive deeper into specific interactions that could lead to undesirable outcomes or present opportunities for improvement.

The first step in our pipeline, Step 3.1, involved modelling detailed AIC interaction scenarios relevant to our problem domain. We looked closely at interactions deemed unsafe, problematic, or beneficial, and used the PrimeP concept to define complex behaviours in our AIC modelling schema. To model the AIC complexity field for unsafe problematic situations, we used only the Forward-feed partial AIC-SECoT (Figure M.2).

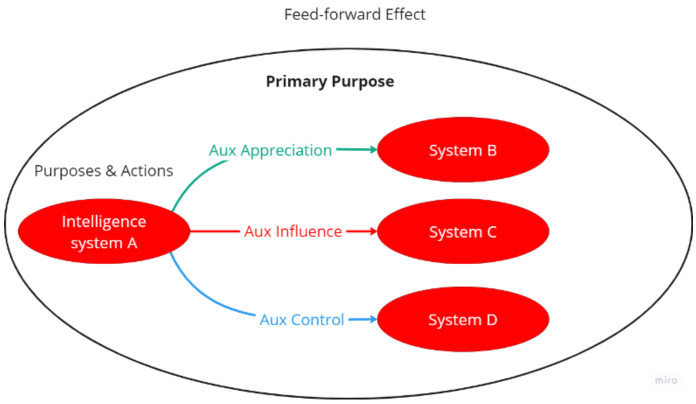


Figure M.2 Forward-feed partial AIC-SECoT

We explicitly stated factors in terms of their dynamic or static states. For instance, we characterised the perception-based avoidance system as {active_avp} to show its operational influence at the time of potential conflict. This method helped clarify the differential situations impacting the complexity of interactions.

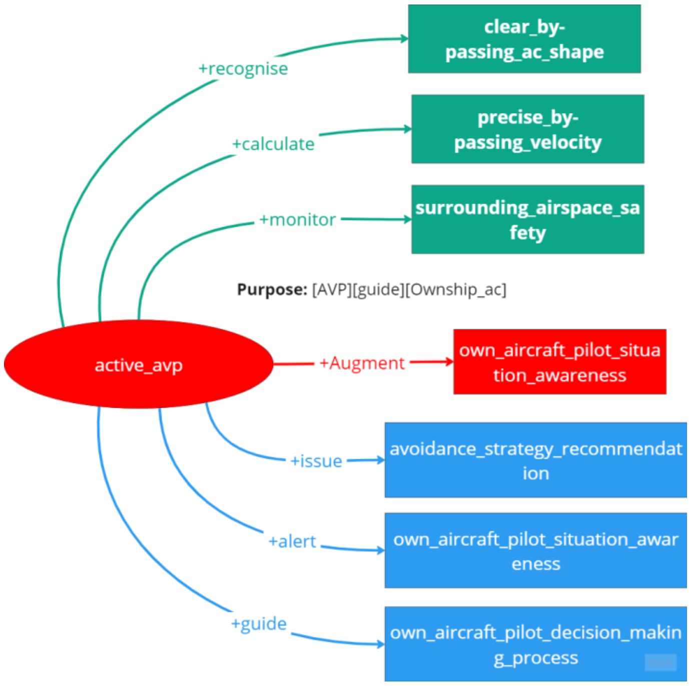


Figure M.3 Modelling AIC scenario from interaction n1

Figure M.3 illustrates the Active AVP Guidance System designed to enhance situational awareness and decision-making for the ownship aircraft pilot in scenarios that may lead to a potential mid-air collision.

Note: We used the n1 problem as the primary purpose to provide the overall context in which we need to guide our predictive thought process. We systematically derive the AIC interactions that cause n1 to emerge. In this example, we did just that. We needed to predict a particular future complexity, so we set the general rule as an n1-driven problem, and from that basis, asked the predictive questions: What is being influenced? What needs to be controlled? And what must be appreciated?

As we proceeded to Step 3.2, we aimed to predict an extended list of emergent AIC interaction scenarios. This step involved utilising a structured interaction format, |{situation}_[action]_{situation}|, which enabled a systematic capture of the dynamics involved in each interaction as visualised in our earlier modelling. We faced a challenge when trying to account for 24 interactions simultaneously. Initially, we approached the modelling by addressing one interaction at a time; however, it became evident that this could be inefficient given the scale. Instead, we opted for a collective analysis encompassing all interactions at once, which prioritised efficiency but complicated the explainability of each factor's contribution to the overall model.

More detailed capture of this process can be found in Table K.1 in Appendix K, examining multiple inputs and identifying emergent behaviours through collective interactions. The following is an example of artefacts that capture knowledge produced in image M.3:

Table M.4 AIC extended scenario for n1 Interaction SECoT definition

<table>
<tr><td>Step 1: Unsafe Problematic Situations</td><td colspan="2">n1: [AVP][guide][Ownship ac]</td></tr>
<tr><td>Step 2: Observed System (obs)</td><td>Step 3: Observed Action</td><td>Step 4: supra source Primary Purpose</td></tr>
<tr><td>active_avp, own_ac</td><td>Active_avp guides own_Input Behaviour</td><td>Maintain safe operational flight and avoid mid-air collisions</td></tr>
<tr><td>Step 5: Auxiliary Influence interaction</td><td>Step 6: Auxiliary Control interaction</td><td>Step 7: Auxiliary Appreciation interaction</td></tr>
<tr><td>|{active_avp}_[+augment]_{own_aircraft_pilot_situation_awareness}|</td><td>|{active_avp}_[+issue]_{avoidance_strategy_recommendation}|

|{active_avp}_[+alert]_{own_aircraft_pilot_situation_awareness}|

|{active_avp}_[+guide]_{own_aircraft_pilot_decision_making_process}|</td><td>|{active_avp}_[+monitor]_{surrounding_airspace_safety}|

|{active_avp}_[+calculate]_{precise_by-passing_velocity}|

|{active_avp}_[+recognise]_{clear_by-passing_ac_shape}|</td></tr>
<tr><td colspan="3">Step 8: Predicted Problem Domain Factors or Features (with repetition)</td></tr>
<tr><td colspan="3">Appreciation = ['active_avp', ' surrounding_airspace_safety', 'active_avp', ' precise_by-passing_velocity', 'active_avp', ' clear_by-passing_ac_shape']
Influence = ['active_avp', ' avoidance_strategy_recommendation', 'active_avp', ' own_aircraft_pilot_situation_awareness', 'active_avp', ' own_aircraft_pilot_decision_making_process']</td></tr>
</table>

Control = ['active_avp', 'avoidance_strategy_recommendation', 'active_avp', ' own_aircraft_pilot_situation_awareness','active_avp', ' own_aircraft_pilot_decision_making_process']

This iteration revealed essential insights into how each interaction affects the emergence of n1, thus reducing our epistemic uncertainty by enhancing our understanding of n1 problem implications. Ultimately, we found that while the collective model offered a time-saving advantage, breaking down interactions as standalone explanations provides clearer insights into their specific roles within the operational complexity field, which is essential for use to draw more accurate design properties. The main output of this systems thinking process is three sets of factors and a comprehensive justification of why we thought of them as important factors to be considered.

M.2.4 Predictive Thinking Pipeline 4: Predict and Evaluate Problem Domain Factors and Assumptions.

Evaluate AIC factors and list them without repetition from most frequent factor to least frequent. For example,

1. active_avp [15]
2. stabilised_own_aircraft_dynamics [11]
3. own_aircraft_pilot_decision_making_process [10]

Those factors are the most influential factors on our perception of the problem. However, they are predictable factors; we want to discover those heavy-tail Black Swan factors. Not all low-number of attentive mentions factors are Black Swans, but a proportion of them at least are.

Evaluation of factor number of attentive mentions

The fascinating finding in this analysis is that the number of attentive mentions distribution of the factors followed a power-law distribution, which is an expected trend for Black Swan events and heavy tail distributions. This trend line created an easily fitted curve, a power-law distribution curve with 88% fitness. This also indicates that the analysis produced a reasonable result, as the output predicted a reasonable set of factors corresponding to the architect's predictive attention distribution.

Table M.5 Sample of Predicted Factors Output[2]

Factor lable	Predicted Factor	Number of attentive mentions	Concern Level
1	active_avp	15	6.70%
2	stabilised_own_aircraft_dynamics	11	4.91%
3	own_aircraft_pilot_decision_making_process	10	4.46%
4	by-passing_aircraft_position	10	4.46%
5	own_aircraft_pilot_situation_awareness	7	3.13%
6	own_aircraft_flight_path	7	3.13%

The table (I.12, appendix I) shows the top 16 factors with the number of attentive mentions of being mentioned more than 3 times. Some of these factors are rather obvious for the architect. Recall in section M.2.1 we had the perspective that there are only 6 main factors (which our concern is biased towards) in the complexity field: {ownship aircraft, by-passing aircraft, other aircraft, clouds, terrain (land), sea}. That was our initial belief system. At the end of the stage, our belief had changed and been refined by discovering 102 more factors with a long audit trail of why we think all those assumptions and factors are important.

Upon applying the skewness statistical test (using Excel function SKEW()), the results returned a value of 2.85, which indicates right-skewed data, confirming that our predicted complexity is indeed a long-tailed complexity field. Such a realisation does not tell us whether our prediction is right or wrong. Still, it consolidates our assumption that the complexity of the real world is a long-tailed distribution of event probabilities. And that our perception of it is skewed.

Eighty-eight factors have been mentioned three times or fewer, all of which were the problem domain's hidden Black Swan events. See Figure M.4, which shows how the long tail of Black Swan events has exposed the problem domain.

[2] For the full table see Table I.12, section I.4.4.1

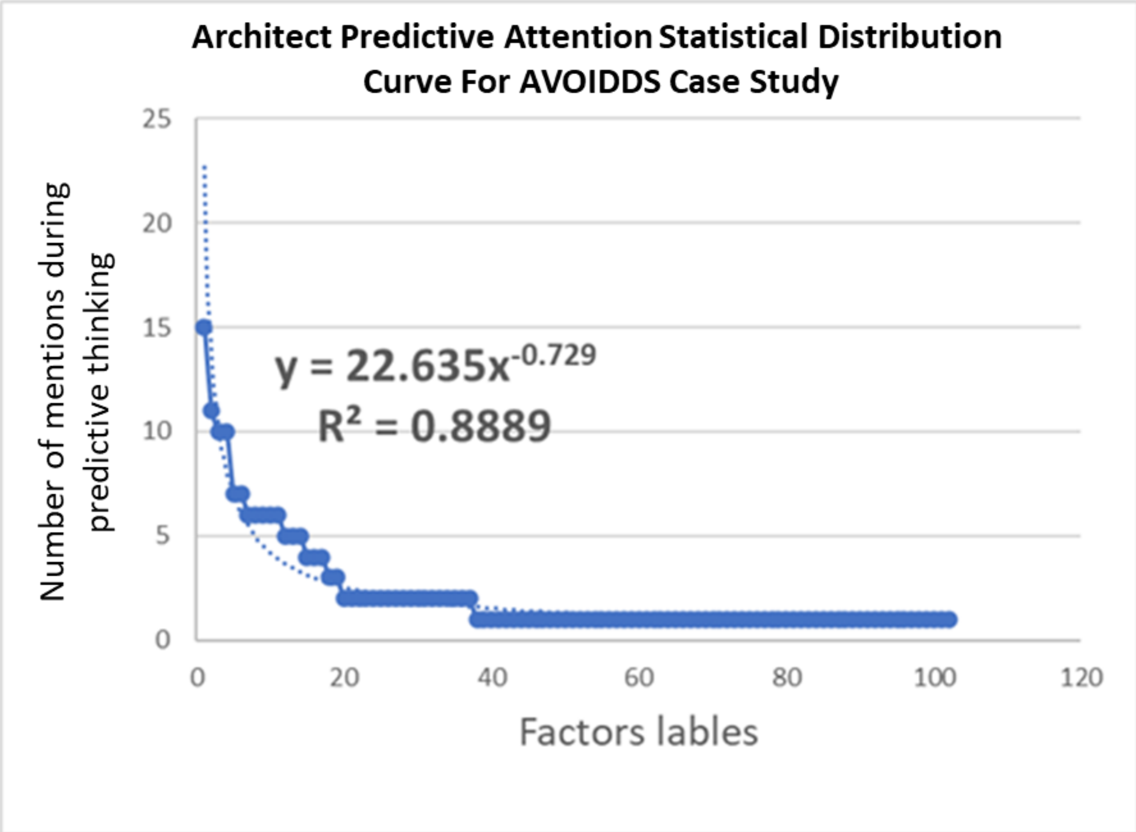


Figure M.4 Number of attentive mentions distribution curve of the derived factors

After acquiring the factors, we defined each to clarify our meaning. In this process, the architect presents the findings to stakeholders to confirm accurate definitions. For example: Below is the full definition for each factor (in no particular order);

1. **Own_aircraft_pilot_decision_making_process**: The process through which the aircraft's pilot makes decisions based on various inputs, including alerts from the AVP system, situation awareness, and environmental conditions.
2. **By-passing_aircraft_position**: The location of an aircraft that is not part of its own aircraft's fleet, potentially posing a collision risk.
3. **Stabilised_own_aircraft_dynamics**: The situation of the own aircraft where its flight dynamics (e.g., speed, altitude, direction) are stable and under control.

However, the architect's work does not stop here. The architect needs to define what assumptions are being made about each factor:

Factor 1: Own_aircraft_pilot_decision_making_process

It is assumed that the pilot will never over-rely on the perception system's judgement, maintaining their normal vigilance during flight.

Factor 2: By-passing_aircraft_position

The AVP system is assumed to always accurately detect and track the Bypassing aircraft's position.

M.3 Stage 2: Architect Intent and Autonomous Solution Needs Definition[3]

Following the problem articulation, this stage defines the architect's intent and the necessary autonomous capabilities to address the identified risks. The goal is to determine the autonomous system's objectives by aligning its actions with the operational problem domain in stage 1. The architect prescribes solutions to selected problems, capturing their high-level intents and assessing customer needs. Such as deploying an Eagle Drone System to patrol train tracks and neutralise unauthorised drone threats. The architect then ensures that:

- The system can appreciate the train track security conditions.
- The system can influence unauthorised actors through deterrent actions.
- The system can control its operational area by mitigating unsafe behaviours.

This process begins with identifying primary input knowledge, which includes interaction definitions, AIC model schema, and definitions of key factors and assumptions. The following steps involve a series of systems thinking activities.

First, we perform a plausibility test to assess whether the assumption is reasonable. Then, we formulate a justification for this evaluation, answering the "Why?" behind our plausibility assessment. Next, we define the Architect's Intent, which clarifies the architect's strategy for addressing the situation, potentially engaging additional stakeholders for support. We also analyse the needs of autonomous solutions to determine their objectives in managing the situation effectively. Finally, we articulate the primary purpose of the overall solution, integrating the needs of both the autonomous systems and support systems, culminating in the formulation of the "Architect High-Level Solution Prescription".

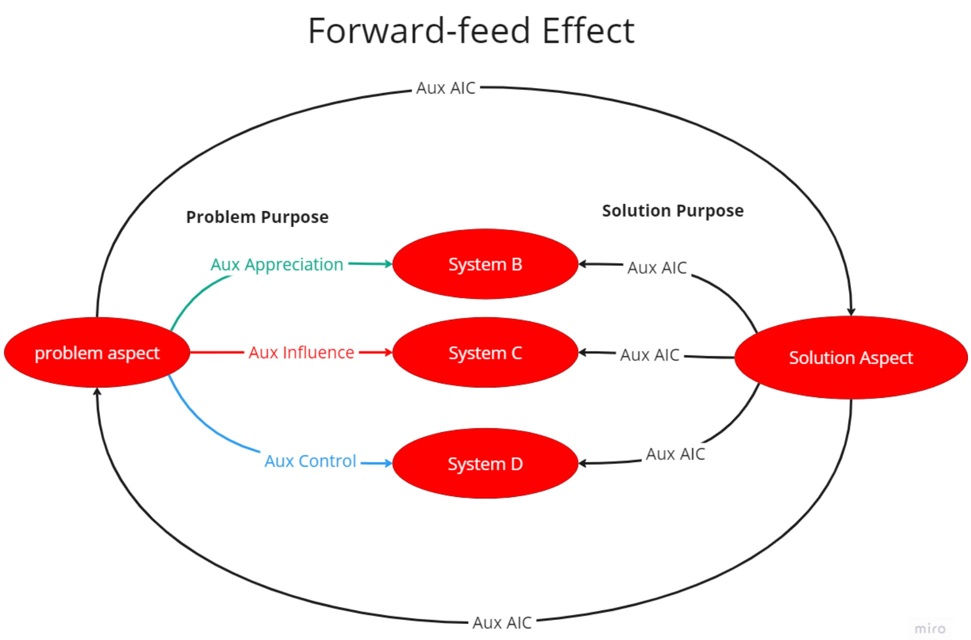
[3] See section

M.4 Stage 3A: HazTOPs and Ordered AIC-Driven Autonomous System Requirements Development[4]

In stages 1 and 2, we rigorously articulated the problem, caring less about the solution and more about the complexity of the situation and any hidden Black Swan factors. This stage will analyse hazards, threats, and opportunities when integrating autonomous systems into the problem. It will also look into what autonomous decisions need to be made. For every defined Architect Prescription identified in stage 1, including the architect's intent, we model the chosen solution within the mix of interacting systems and re-evaluate the observed complexity. We need to model how the autonomous systems deal with every factor in the model.

M.4.1 Predictive Thinking Pipeline 1: Introducing Autonomous systems into Forward-Feed complexity[5]

The first step is to take the AIC models provided by Architect Solution Prescription and the associated AIC model of the problem. We then introduce the solution into the model, mapping the capabilities required to manage the problem domain, including any autonomous decisions to be made. We followed the following general systems rule as an assumption of how systems in general would interact in a complexity:



[4] For more details see section I.6
[5] More detailed description can be found in Appendix I, section I.6.1

Figure M.5 AIC general systems rule to model the engineered solution's complete reaction to the problem complexity.

In our case, we chose the AVP development dataset as a system component required to handle the complexity of the problem domain. See Figure M.6 that describes the reaction between datasets and the problem:

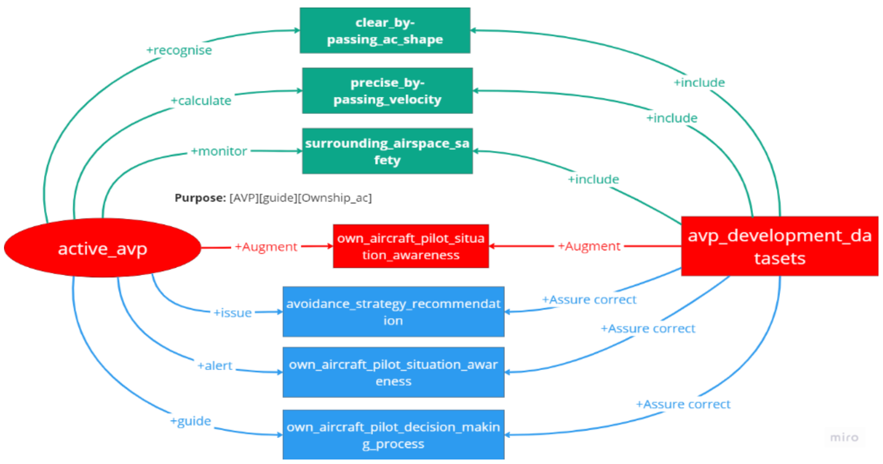


Figure M.6 Introducing part of the solution into the mix of AIC problem complexity.

Once we have the complete problem-solution AIC model, we then defined the set of interactions between the sink and the source using AIC structured interaction format of: |{source situation}_[+,- or no sign, AIC-action]_{sink situation}|, Written in the following grammar: |{adjective+noun}_[verbal phrase]_{adjective+noun}| For example, |{flying_police_robot}_[learns humans' visual profiles]_{distressed_people}| Capture the output in the following table:

Table M.6 Mapping AIC interactions of AVP with ownship aircraft and the environment

Source: {active_avp}	
Output Behaviour	**Input Behaviour that impacts the emergence of Output Behaviour**
I2: |{ avp_development_datasets }_[+Augment]_{ own_aircraft_pilot_situation_awareness }|	**A4:** |{ avp_development_datasets}_[+include]_{surrounding_airspace_safety}| **A5:** |{avp_development_datasets}_[+include]_{ precise_by-passing_velocity}| **A6:** |{ avp_development_datasets}_[+include]_{ clear_by-passing_ac_shape }|
	C4: |{ avp_development_datasets}_[+Assure correct]_{avoidance_strategy_recommendation}| **C5:** |{ avp_development_datasets}_[+ Assure correct alert]_

	{ own_aircraft_pilot_situation_awareness}| **C6:** |{ avp_development_datasets}_[+ Assure correct]_ { own_aircraft_pilot_decision_making_process}|

Table M.6 captures the reactions of the AVP development dataset to the problem environment. The full table, which includes the AVP action definitions, can be found in Appendix I, Table I.15.

M.4.2 Predictive Thinking Pipeline 2: Designing the affecting Backward-Feed complexity field[6]

With AIC Forward-feed defined, we have ourselves a complete complexity field. However, this complexity does not exist in isolation from the environment. The interactions between the components in an open, complicated environment have a ripple effect on other components' purposes and vice versa. Here, we extend the Forward-feed AIC model to include a Backwards-feed complexity field. However, to do so, we need some visualisations to help see the type of environment in which the AIC problem-solution exists. We place the model over the generated image we created in stage 1 to give some appreciation of what the real world looks like (see Figure M.7).

With the operational environment in sight, we extend the Forward-feed model with the factors within the operational domain complexity by identifying which complexes are identified in the real-world problem complexity. To do so, we need to abstract the Forward-Feed model for ease of modelling. We reduce the model to a controlled language of interaction:

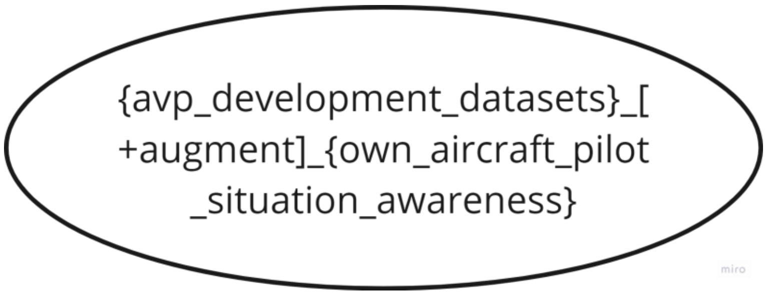


Figure M.7 I2 interaction

[6] More detailed description can be found in Appendix I, section I.6.2

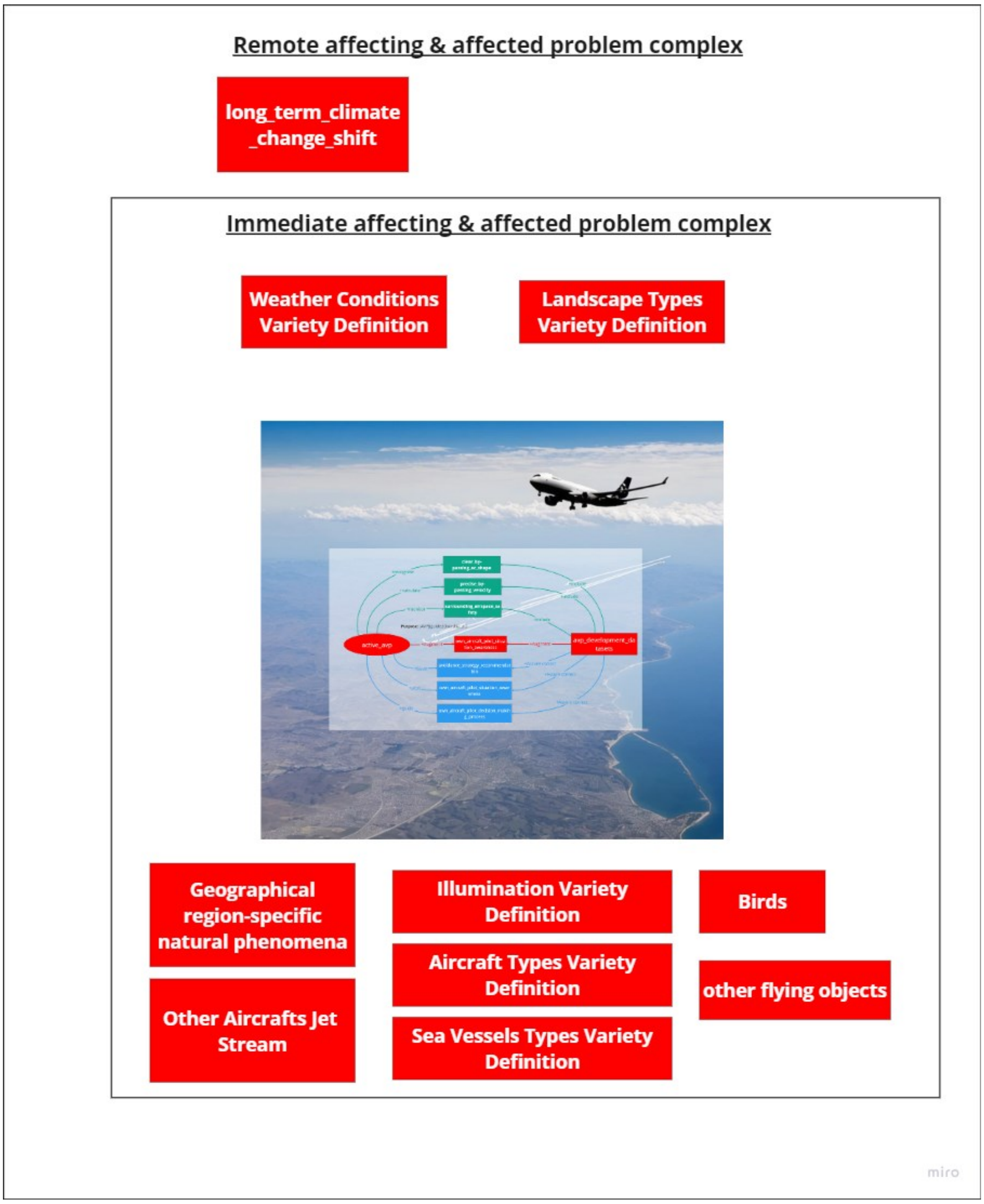


Figure M.8 AVP operational environment

Then, we model the Backward-Feed AIC interactions by asking the following questions (adapting AIC core thought process in section 4.4.11):

- What complex could appreciate the interaction? **We answer:**
    - The informed air traffic controller appreciated the avp development dataset.[A]
- What complex could influence the interaction? **We answer:**
    - Other flying objects could influence the AVP dataset development.
- What complex could control the interaction? **We answer:**
    - Weather conditions definitely control the AVP development dataset.

- Architect a predictive mental model that can also control the quality of the AVP dataset coverage.

We will use icons to represent the complexes visually. Imagination is a key enabler for predictive thinking. Figure M.9 models the unknown unknowns at this stage, which are also part of the possible long-tail Black Swan Scenarios.

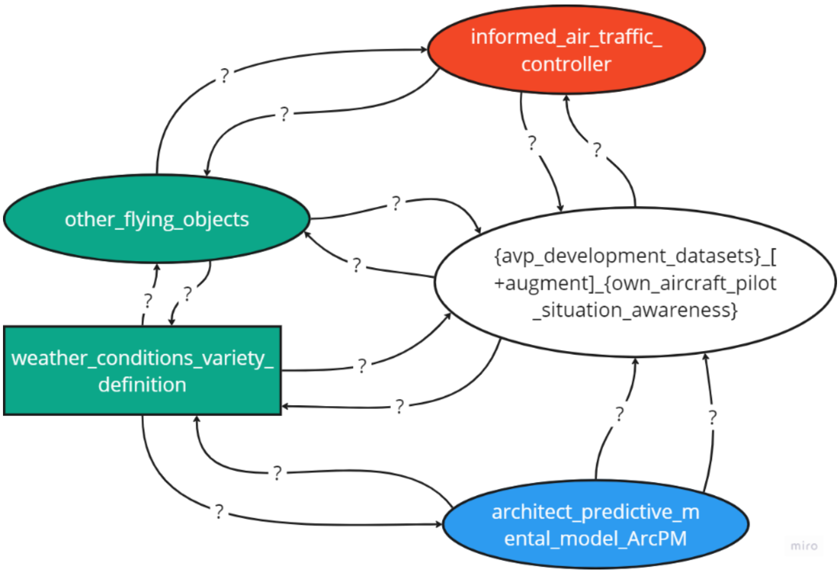


Figure M.9 Backward-feed model for I2 unknown unknown interactions. Black Swan scenarios.

Figure M.9 allows us to predict the potential Black Swans we missed when our belief of the problem was set in the list of factors in stage 1. Question marks indicate missing interactions we missed in stage 1. The next activity would be instantiating to resolve the complexity presented in Figure M.9 through predictions. We implemented the Action Matrix method to help make the predictions needed. We also coloured the background of the matrix to indicate the AIC nature of interaction; green from A, red for I, and blue for C. See the following snippet of the AIC-coloured Action Matrix:

Table M.7 AIC coloured action matrix to model all possible interactions of the AVP dataset influence over

	 **data**	 **flying objects**	 **Weather**	 **Architect**
		+capture	+capture	+assure
	-visually complicate		visually complicate	visually complicate

	-visually complicate	operationally complicate		visually complicate
	+predict	+predict	predict	

The Actions Matrix above defines how the complexity of the scenario is resolved. Now, we will transfer all the knowledge to the final output of the step, which is a comprehensive AIC mental model of the problem. Note how we started and how we ended. Every step in this process carefully explains how we produced the scenario. Note that the interactions between the Forward-Feed bubble and the Backward-Feed operational environment concern the source. We are currently interested in the AVP development datasets in the design. Later, if the design team wishes to evaluate the complexity from the sink perspective, the complexity will change. The approach is flexible and allows for investigating and modelling the whole from any node, thus creating a variety of complexities. Figure M.10 captures all the resolved interactions in Table7.7.

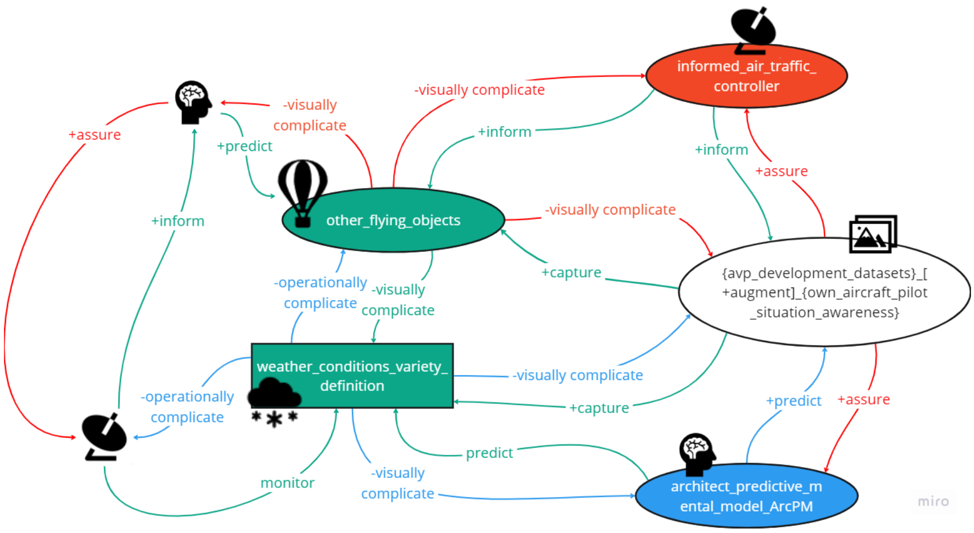


Figure M.10 Resolved complicatedness of AVP dataset complexity field

Now, we need to capture all influence interactions and resolve them into their respective appreciation and control interactions. Although there are six influence interactions, for the sake of simplicity, we will choose only three. We do so by capturing AIC interactions and further extending our belief in the problem domain. The Figure M.11 is an extended version of Figure7.1.

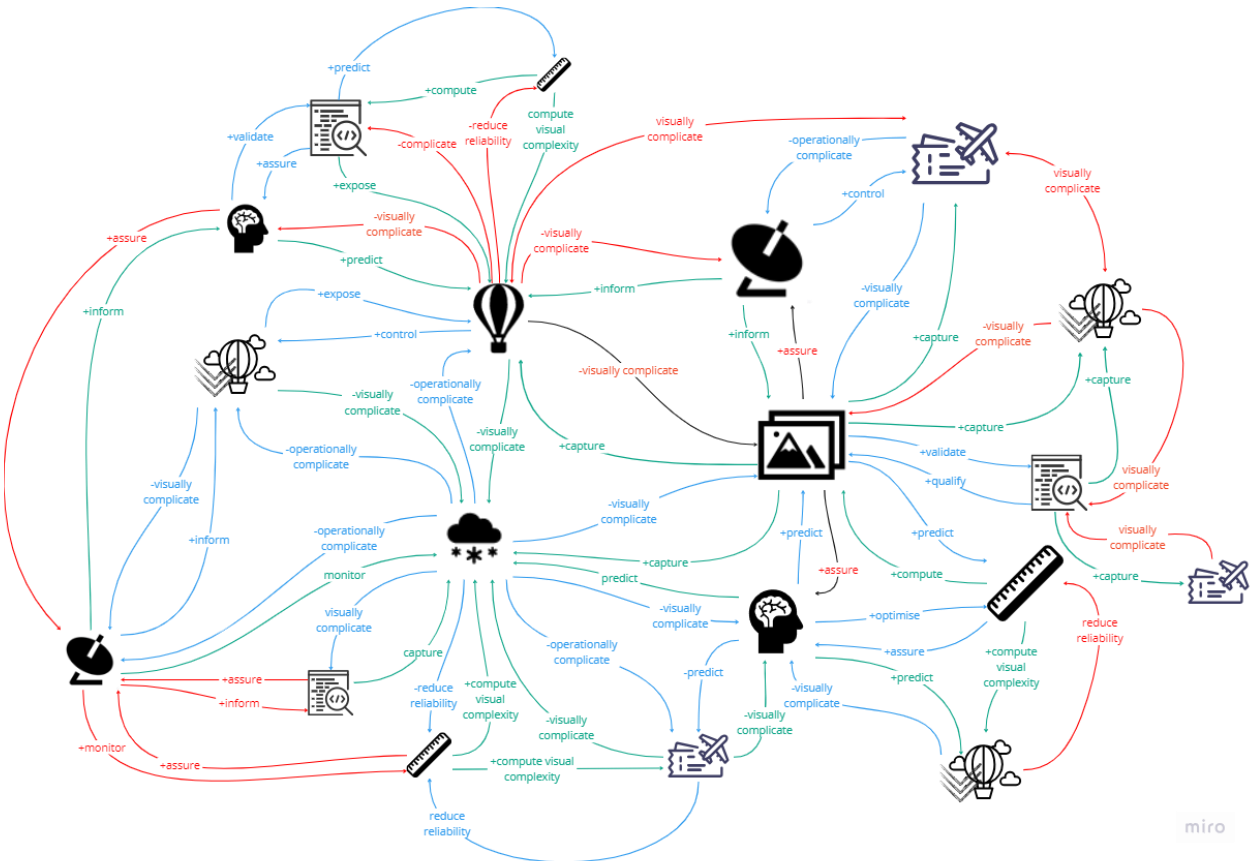


Figure M.11 Extended AIC complexity field for aircraft avoidance problem

Note that the red influence interactions have been changed to black (keeping the action in red). This signifies that those influence interactions have been resolved to their appreciation and control components[7]With the whole problem laid out in front of us, we can start to predict sources of hazards, threats, and opportunities.

M.4.3 Predictive Thinking Pipeline 3: Hazards, Threats and Opportunities Scenarios (HazTOPS) Analysis[8]

We apply the HazTOPS SECoT outlined in section F.4. The following is the application of the process:

[7] For full justification and design behind the complexity field in Figure7.11, see section I.6.2.5
[8] More detailed description can be found in Appendix I, section I.6.3

(1) Step 1) Scope the HazTOPS context domain

Scope the potential safety and security challenges on the AIC schema of the problem domain.

Add the following icon  for safety hazards,  for opportunities and  for potential security threats (cyber-attacks) to every interaction on the mode.

We will mainly apply the HazTOPS analysis to I.9 Schema in order to simplify the process.

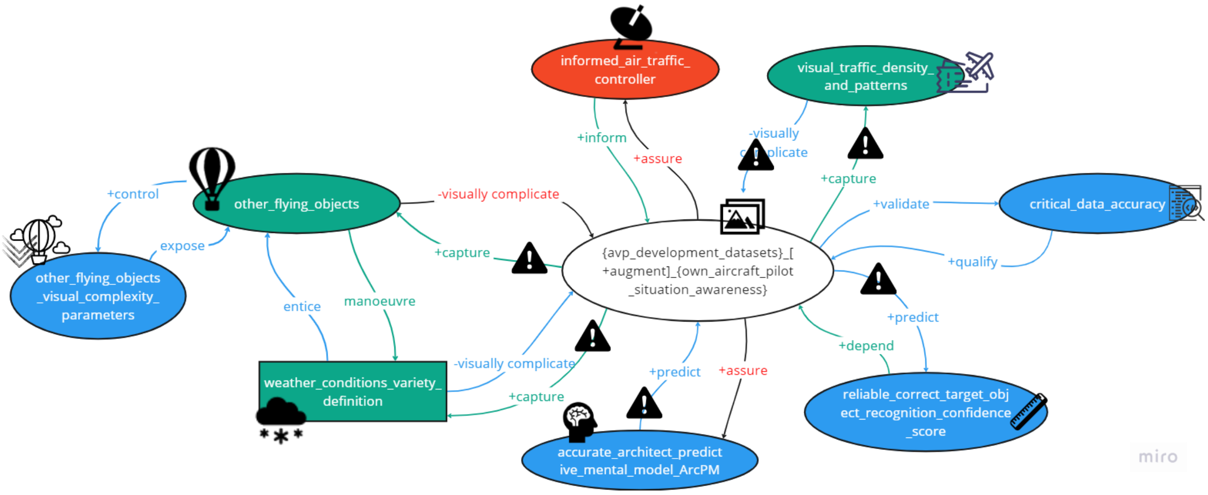


Figure M.12 Sources of Hazards AIC Complexity Field

Figure M.12 presents a HazTOPS analysis applied to the AIC Complexity Field to identify safety hazards and operational challenges. The diagram visualises the interactions between AVP development datasets, environmental factors, air traffic control systems, and data reliability, highlighting areas where hazards may emerge.

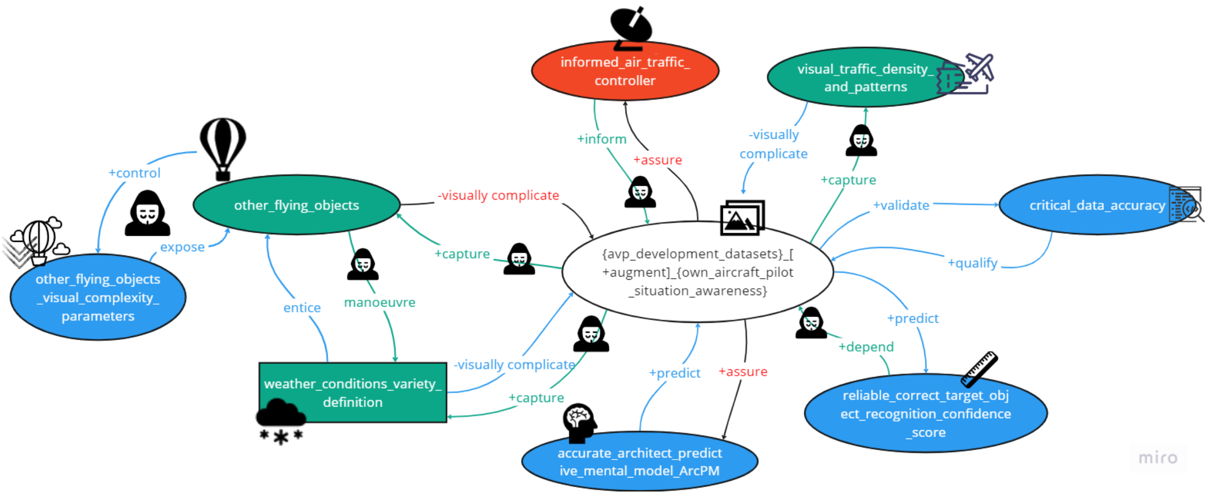


Figure M.13 Sources of Threats AIC Complexity Field

Figure M.13 presents a HazTOPS analysis applied to the AIC Complexity Field for identifying potential threats and vulnerabilities within the operational environment. The diagram visualises the interactions between AVP development datasets, environmental conditions, air traffic control systems, and critical data accuracy, highlighting areas where threats may emerge. Key concerns include visual complexity, misclassification risks, and data reliability issues, which could compromise the architect's situational awareness and AVP dataset development processes. This structured representation aids in assessing systemic threats and enhancing resilience against potential security and operational challenges in complex aviation systems.

(2) Step 2) Characterise the scoped interactions.

In this step, we will take only consider the hazards of the complexity field:

Table M.8 Considering Hazards related to I1 interaction

Source or Sink: {avp_development_datasets}	
Output Behaviour	**Input Behaviour that impacts the emergence of Output Behaviour**
I1: |{ avp_development_datasets}_[+assure]_ { architect_predictive_mental_ model_ArcPM}|	**A1:** |{ avp_development_datasets}_[+capture]_ { other_flying_objects}| **A2:** |{ avp_development_datasets}_[+capture]_ { weather_conditions_variety_definition}|
	C1: |{avp_development_datasets}_[+predict]_{reliable_correct_target_object_recognition_confidence_scores}|

(3) Step 3) Apply potential complications predictive guide words.

Then, identify potential complexity by utilising the following modified keywords: More, Part of, Less, Early, and Late. Follow SECoT_2 (section F.4) to derive the variety of possible deviations. Table7.9 captures the hazards, threats or opportunities predicted:

Table M.9 Example "More" guide word complication

Title	Multiple flying objects (other than aircrafts)
AIC interaction	**A1:** |{ avp_development_datasets}_[+capture]_ { other_flying_objects}|
HazTOPS Aspect	**Definition**
Hazards, Threats or Opportunities Scenario: Guide word	More: multiple balloons involved.

Operating Scenario Context	Clear day, ownship aircraft (with AVP) at 10000ft high.
Hazard, Threat or Opportunity definition (consider used-systems)	More than 1 hot air balloon is visible. At 3000 ft.
Foreseeable Sequence of Events	• Planned a recreational or observational event that involves multiple balloons. • A group of hot air balloons or other balloons (e.g., weather or recreational balloons) are launched in the vicinity of the aircraft's flight path. • The AVP system identifies multiple objects (balloons). • The system classifies these objects based on altitude, size, and motion patterns. • Ownship pilot or automated system receives an alert of potential objects within or near the flight path.
Potential harm or benefit	• The shape of hot air balloons may startle the AVP, leading to a false negative. • The number and location of the hot air balloons may startle the AVP, leading to a false negative.

Table M.9 explores the application of the "More" guide word within a HazTOPS predictive complexity analysis, focusing on the potential impact of multiple flying objects—specifically, the presence of more than one hot air balloon in the operational environment of an ownship aircraft equipped with AVP at 10,000 feet altitude.

The AIC interaction under analysis is:

A1: |{avp_development_datasets}_[+capture]_{other_flying_objects}|

Where the AVP system captures and processes data related to airborne objects in its vicinity. The HazTOPS aspect introduces the "More" guide word, indicating an increase in the number of detected airborne objects—particularly multiple hot air balloons at lower altitudes (3,000 feet).

M.4.4 Predictive Thinking Pipeline 4: Elicitate ordered AIC System-Level Requirements and training requirements[9]

We modelled every derived HazTOPS in this step and defined mitigating system-level requirements. We will perform the process in the following 2 examples:

(1) Step 1) Model the Complex of Interest Operating Scenario Context

At a high level, we model the AVP as a source and the AIC relationships with the required used-systems to achieve the emergent capability. First, we identify the unresolved influence relationship problem between the agent and the target object of interest in Figure M.14:

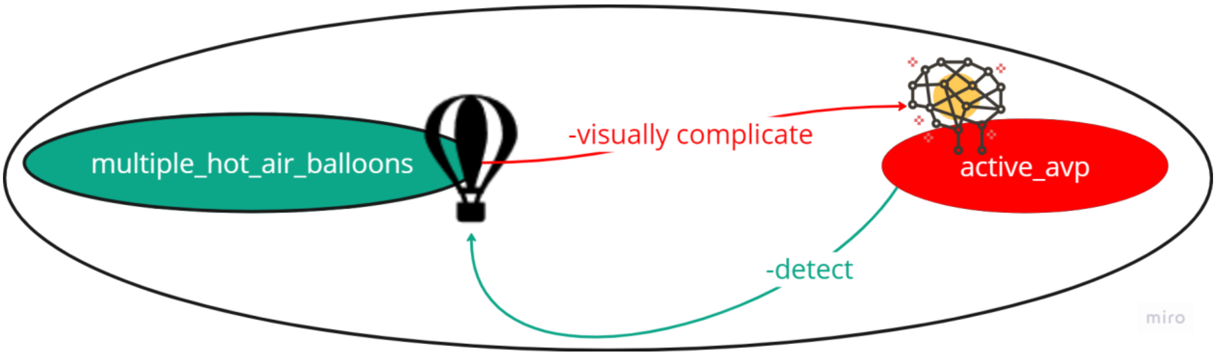


Figure M.14 Initial abstraction of Eagle Robot (ER) Agent emergent capability boundary

The model "multiple_hot_air_balloons" negatively complicates the perception system of active Aircraft Collision Avoidance Perception (AVP). It describes how the presence of hot air balloons can interfere with or degrade the performance of an AVP system, which is specifically designed to detect and avoid aircraft. Below, we have a number of unknown unknowns that we need to predict in Table M.10.

Table M.10 Initial AIC resolution interactions table

Agent Output Behaviour	Agent Input Behaviour that delivers Output Behaviour	Supportive systems AIC Behaviour
I1: { multiple_hot_air_balloons_ }_[-visually complicate]_{active_avp}	**A:** to be defined	None
	C: To be defined	

[9] More detailed description can be found in Appendix I, section I.6.4

<table>
<tr><td rowspan="2">No influence from AVP but a higher level of appreciative interaction
A4: { active_avp}_[-detect]_{multiple_hot_air_balloons}</td><td>A:
{ active_avp}_[detect]_{ multiple_hot_air_balloons}</td><td rowspan="2"></td></tr>
<tr><td>C: To be defined</td></tr>
</table>

(2) Step 2) Model hazards mitigation Ordered-AIC complexity field

We must resolve the influence relationship problem on its appreciation and control relationships. To refine this model into subsystems, we can break down the visual components that define the appearance of hot air balloons as seen by the AVP system and the components of the ML-based perception model used to interpret these visuals. This will help understand how the AVP system could handle the visual complexity introduced by multiple balloons and improve detection accuracy.

Visual Components of Hot Air Balloons (as Perceived by AVP)

These are the key **visual attributes** that the AVP system might use to detect and differentiate hot air balloons from other objects:

- **Colour and Pattern**: Hot air balloons are often brightly coloured with unique patterns, making colour segmentation a useful feature. The perception system may analyse colour contrasts to distinguish balloons from the background and each other.
- **Shape and Contour**:
  - **Silhouette**: Hot air balloons have a distinctive, often rounded shape with a narrower base where the basket is located.
  - **Edge**: The balloon and basket structure has unique edges that can be detected even from a distance. The AVP system might use contour-based filters to recognise these shapes.
- **Size and Scale**: The balloon's apparent size changes with distance, which the AVP can use to estimate proximity. This may involve analysing relative scale based on prior balloon size data.
- **Texture and Surface Features**: Balloons have smooth surfaces, sometimes with logos or text, which may require texture-based recognition methods to enhance detection, especially if the AVP system is differentiating between types of balloons.

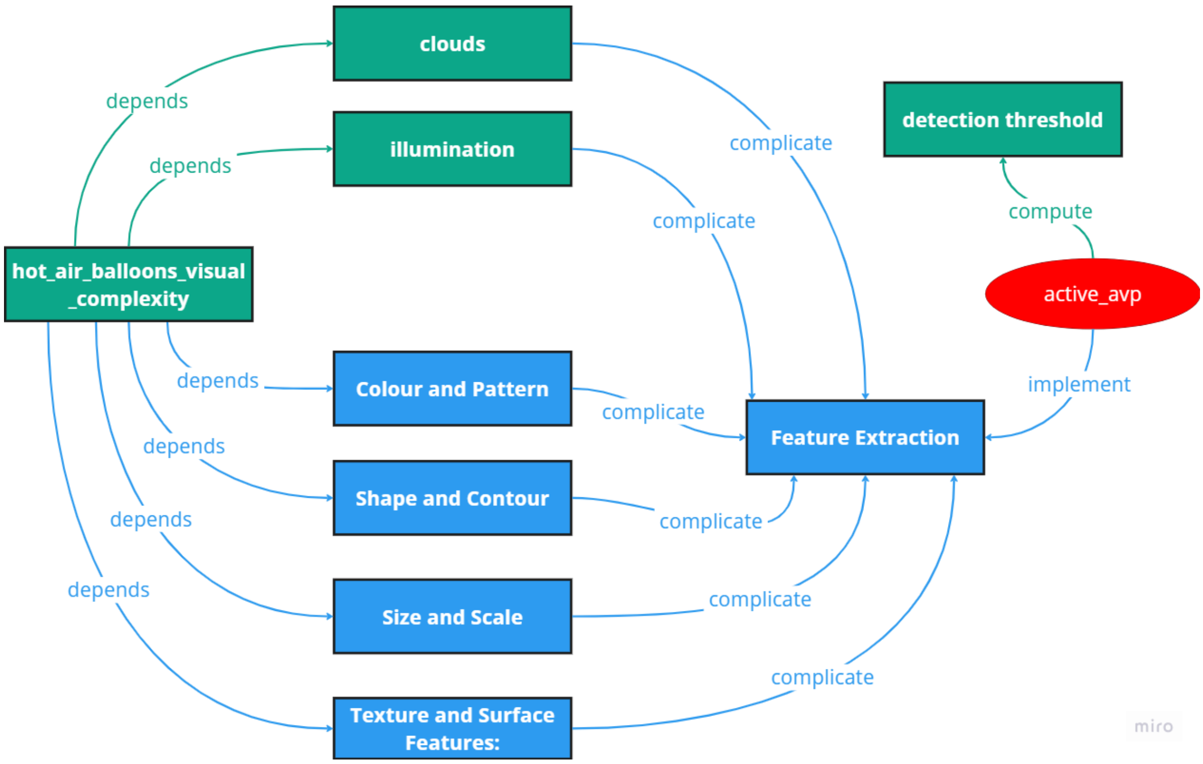


Figure M.15 Resolution of multiple hot air balloon hazards

The following table captures the description of the Figure M.15:

Table M.11 AIC-Structured Interactions Detailing the Visual Complexity of Hot Air Balloons and Its Impact on AVP Perception

Agent Output Behaviour	Agent Input Behaviour that delivers Output Behaviour	Supportive systems AIC Behaviour
I1: { hot_air_balloons_visual_complexity}_[-visually complicate]_{active_avp}	**A1:** { hot_air_balloons_visual_complexity}_ [depends]_{clouds} **A2:** { hot_air_balloons_visual_complexity}_ [depends]_{illumination}	None
	C1: { hot_air_balloons_visual_complexity}_ [depends]_{ Colour and Pattern} **C2:** { hot_air_balloons_visual_complexity}_ [depends]_{ Shape and Contour}	

	C3: { hot_air_balloons_visual_complexity}_ [depends]_{ Size and Scale} **C4:** { hot_air_balloons_visual_complexity}_ [depends]_{ Texture and Surface Features}	
	C5: { active_avp }_[implement]_{ Feature Extraction}	

Table M.11 presents an AIC-structured analysis of how the visual complexity of hot air balloons influences the Active AVP system's perception and detection capabilities. The table categorises interactions based on AIC functions, mapping hot air balloon attributes that complicate AVP visual processing and the supportive systems required for perception enhancement.

The influence interaction (I1) highlights the negative visual complications introduced by hot air balloons, affecting the AVP's ability to classify and track airborne objects. The appreciation interactions (A1-A2) establish dependencies on environmental factors such as cloud cover and illumination, further impacting object visibility and detection reliability. The control interactions (C1-C4) focus on the specific visual attributes of hot air balloons, including colour, pattern, shape, contour, size, scale, texture, and surface features, all contributing to perceptual uncertainty for AVP systems.

The table also recognises that AVP does not inherently influence hot air balloons, meaning it must adapt its perception strategies instead of modifying external conditions. The AVP compensates by computing detection confidence scores (A3) and implementing feature extraction techniques (C5) to enhance its ability to distinguish hot air balloons from other airborne objects.

(3) Step 3) Ordered-AIC-based Mitigating System or Safety Requirements Derivation (Safe Operating Concept SOC)

So far, we have focused on defining Agent relationships in the given hazardous scenario. In this step, we collate all discovered actions in a table and convert those actions into AVP systems requirements, considering the intangible or tangible methods that shall deliver the agent's will. The relevant system requirements are defined from Control and Appreciation actions, with the influence action included at the end of each requirement after the phrase "in order to". Use the following format:

[**Given:** A or C actions] **in order to** [I action], **Then** [mitigation requirement] **In order to** [I action]

For example:

AVP Safety Requirements:

Here are the AVP System safety requirements based on the given actions (A and C actions) for mitigating the complexity introduced by multiple hot air balloons and enhancing AVP's ability to detect and classify them reliably:

Table M.12 Safety requirements derivations to mitigate hot-air balloons impact on AVP system performance.

AC (appreciative, control) interaction	**Mitigating Safety or Systems Requirements (Safe Operating Concept)**
C1: { hot_air_balloons_visual_complexity}_ [depends]_{ Colour and Pattern}	**Safety Requirement 3: AVP System shall avoid Air balloons with rare patterns** • **Given**: C1: Hot air balloon visual complexity depends on colour and pattern variations, • **In order to**: visually complicate AVP System performance (I1), • **Then**: The AVP system shall guide ownship aircraft to avoid a broad range of balloon colours and patterns, including common and rare patterns and colour schemes, • **In order to**: recognise and direct ownship aircraft to avoid hot-air balloon visual complexity.

The safety requirements derived from the Appreciative and Control (AC) interactions in Table M.12 were refined by systematically considering how the AVP system should handle hot air balloons under varying visual complexities.

- **C1: Dependence on Colour and Pattern / Safety Requirement 3:** Hot-air balloons feature varied and unpredictable colours and patterns, making recognition challenging.

The requirement ensures the AVP system can handle balloons with common and rare designs, preventing confusion with background scenery, aircraft, or other environmental objects.

(4) Step 4) Extended Concrete Safe Operating Concept and ML Safety Training Concept

We will not apply the 4-whats-and-how method in this case study since we have already derived the AVP safety requirement. We will directly derive the training concept. Instead, we will move to the derivation of the training concept.

(5) Step 4.1) ML Safety Training Requirement derivation (Training Concept):

In this stage, we specify a general training requirement for the ML model, which we refer to as the Training Concept. To define the training requirement, we used the following structure:

ML Safety Training Requirement n: [system of interest] shall be trained to [training experience].

Below is the table we use to perform the derivation process:

Table M.13 Deriving Training Concept from Safe Operating Concept for AVP ML component

Mitigating Safety or Systems Requirements (Safe Operating Concept)	**ML Safety Training Requirements (Training Concept)**
Safety Requirement 3: AVP System shall avoid Air balloons with rare patterns • **Given**: C1: Hot air balloon visual complexity depends on colour and pattern variations, • **In order to**: visually complicate AVP System performance (I1), • **Then**: The AVP dataset shall include a broad range of balloon colours and patterns, including common and rare patterns and colour schemes,	**ML Safety Training Requirement 3:** AVP ML component shall be trained to recognise a broad range of balloon colours and patterns, including common and rare patterns and colour schemes.

In order to: recognise and direct ownship aircraft to avoid hot-air balloon visual complexity.	

(6) Step 4.2) ML dataset requirements derivation

A dataset training requirement can be derived from the training concept. We use the following structure to define requirements over actual datasets:

Dataset requirement structure:

> The AS ML component [Training/Testing/Black Swan Validation] Dataset shall provide the trainee model with a valuable minimum variety of …

In this case:

> **ML Development Dataset Requirement 1:** The AVP's ML component Training Dataset shall provide the trainee model with a valuable minimum variety of various types of hot air balloons under various cloud cover scenarios (e.g., clear, partial cloud cover, heavy cloud cover).

M.5 Stage 3B: Comprehensive Operational Environment Definition[10]

This stage refines the operational design domain (ODD) by specifying environmental constraints, dynamic interactions, and potential Black Swan scenarios. To examine the whole table, see section I.7. It is also related to Appendix A, which we define Multi-ODD specification. The architect defines:

- Operational Boundaries (e.g., maximum flight altitudes for security drones).
- Weather and Lighting Conditions (e.g., visibility requirements for drone cameras).

This stage ensures that the autonomous system is prepared for realistic deployment scenarios by systematically scoping the operational complexity.

[10] For more detail see section I.7

M.6 Stage 4: Disordered AIC-Driven Black Swan Scenarios Prediction[11]

In this stage, you may consider disordered AIC timing in section E4. Some of the output of this stage is to generate Black Swan training and validation datasets for validating ML components to handle Black Swan scenarios.

At first, we choose the complexity field we intend to perform the AIC perspective shift. For this we will select the last complexity field in Figure M.11. From which we chose an interaction between the dataset and a hot-air balloon.

To implement the AIC perspective shift, we need to perform the following process:

M.6.1 Step 1) Define the interactions

Define the interactions needed to predict a potential emergence. We chose the following relationship. For this example, we will choose the interaction between other_flying_objects and avp_development_datasets:

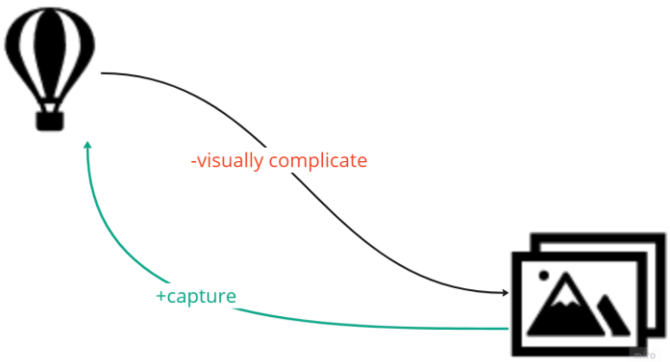


Figure M.16 Chosen interaction (black line means that the influence interaction had been resolved into its AC components)

The expected interaction between the hot-air balloon and the AVP dataset development is that the dataset should respond to the balloon's behaviour. The dataset should never influence the balloon's behaviour, regardless of how the complexity of the future field may evolve. The datasets are claimed to be supportive of balloons' purpose (based on the assumption that recognising hot-air balloons and including them in the datasets would contribute to their safety). At the same time, the unexpected variability of hot-air balloons would obstruct the reliability of AVP perception. The architect is prophesying this: what would happen in such a scenario. Any deviation from this

[11] For more detail see section I.8

established perspective norm would be considered unexpected or surprising, characteristics typical of Black Swan events.

M.6.2 Step 2) Define the ArcMatrix

Define the interaction's current AIC factors using the AIC-Matrix (section 5.10). We will give the hot air balloon a descriptive situation of “visually_variable_hot_air_balloon”.

Table M.14 Deep AIC factorisation for I.16 interaction

	Visually_variable_hot_air_objects	avp_development_datasets
Visually_variable_hot_air_objects		**Supra Source:** other_flying_objects. **PrimeP:** perform uninterrupted missions. **Goal:** avoid any obstacle to mission execution. **Goal type:** Influence. **Action:** visually complicate AVP datasets. **Action type: Infleuce**. **Effect:** Obstructive.
avp_development_datasets	**Supra Source:** Ownship aircraft. **PrimeP:** Maintain safe operational flight and avoid mid-air collisions. **Goal:** complete coverage of potential operational scenarios.	

	Goal type: Appreciation**.** **Action:** capture scenarios. **Action type:** appreciation. **Effect:** Obstructive.	

M.6.3 Step 3) Perform the Perspective Shift

Perform the perspective shift using AIC perspective shift SECoT (See section 5.7). In this step, we choose an appropriate perspective shift that we believe will be a potential black-swan event not foreseen during modelling the interaction between adv_drone_shapes and flying_eagle_drone. We will select the following shift:

1. **AIC type shift.**

For this, we will apply the following Thought Step from SECoT_3:

General Systems rule: given a source or sink, it is possible that over time and with the change of complexity, the AIC goals are altered, leading to a new situation of complicatedness.

Predictive question: What would happen if the interaction flow and effect type remained the same, but the goal's AIC type and action type were altered in the future?

Guiding Prompt: Review the AIC dynamics of the observed complexes and alter the nature of goal and action. Then, define an appropriate action to bear alternative AIC types and describe a scenario in the shifted context.

Completion criteria: The step is considered complete when a scenario demonstrating an alternative AIC goal is detailed.

In this thought process, we will shift the AIC-type perspective as follows: As per the following schema:

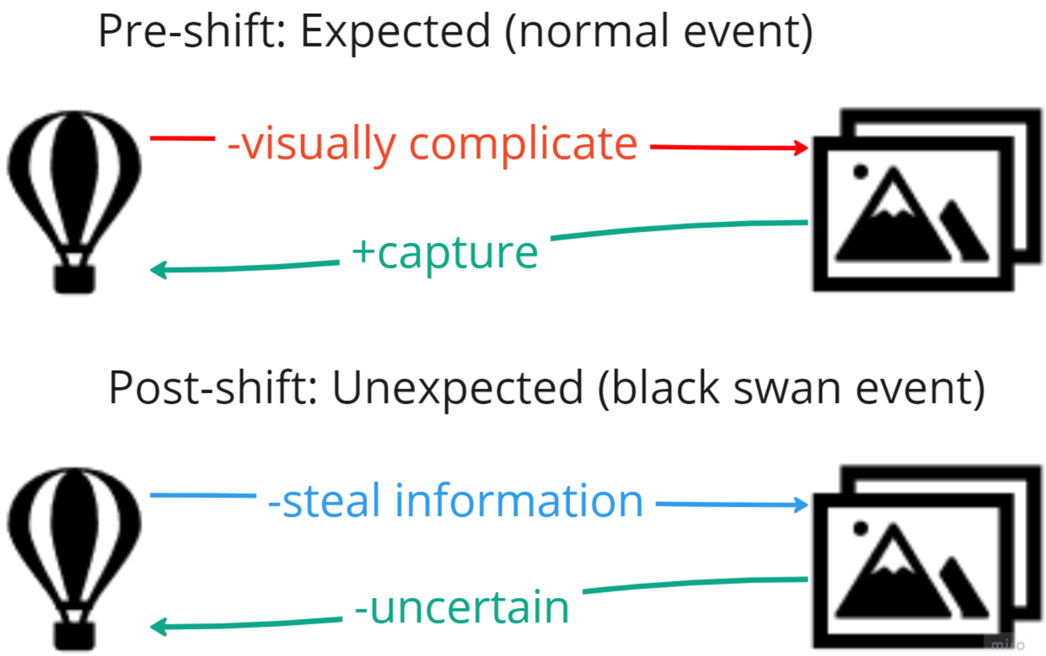


Figure M.17 Shifted perspective from influence to control

The above steps can be captured in the following table:

Table **Error! No text of specified style in document.**.15 AIC-type perspective shift

Interaction	**A1:** |{ avp_development_datasets}_[+capture]_ { other_flying_objects}| **I3:** |{ other_flying_objects}_[-visually complicate]_ { avp_development_datasets }|	
AIC factors	**Pre-shift perspective**	**Post-shift Perspective**
Source	Visually_variable_hot_air_objects	Visually_variable_hot_air_objects
Sink	avp_development_datasets	avp_development_datasets
Supra Source	other_flying_objects	other_flying_objects
PrimeP	perform uninterrupted missions	perform uninterrupted missions
Source's Goal	**avoid any obstacle to mission execution**	**Spoof the AVP system by using spoofing visual patterns on the hot-air balloon**
Source's Goal type	**Influence**	**Control**

Source's Action	**Visually complicate**	**Steal information of the type of data used for the AVP system (security threat)**
Source's Action type	**Influence**	**Control**
Source action effect on sink	Obstructive	Obstructive

How do we implement the SECoT_3 thought step above in this context?

The perspective shift captured in Table M.14 was derived using the AIC Type Shift Process SECoT_3 that anticipates how complex interactions evolve, leading to new situations of complicatedness. The process involves systematically analysing how source-sink relationships change, particularly in how they influence or control their interactions within an operational environment. The steps used to derive the changes in Table I.29 are explained below.

Applying AIC Type Shift (General Systems Rule)

The General Systems Rule situations that the goals of a source or sink can change over time due to increased complexity. In this case, the initial interaction was:

A1: |{ avp_development_datasets}_[+capture]_{ other_flying_objects}|

The AVP dataset captures and processes information from other flying objects, including visually variable hot air balloons. The pre-shift perspective assumes that hot air balloons only introduce passive visual complexity into AVP systems (influence-type interaction).

However, by applying the AIC Type Shift, the interaction was reconsidered under future adversarial conditions, leading to an alternative scenario:

I3: |{ other_flying_objects}_[-visually complicate]_{ avp_development_datasets }|

Now, hot air balloons are no longer just visually complex objects; instead, they intentionally spoof the AVP system, shifting their role from passive complexity to an active adversarial control-type goal.

Predictive Question Application

The predictive question guides the shift:

What would happen if the interaction flow and effect type remained the same, but the goal's AIC type and action type were altered in the future?

Pre-Shift Perspective (Influence)

- The source (visually variable hot air objects) had the goal to avoid obstacles to mission execution, passively complicating the AVP dataset's processing.
- The action type was influence, meaning hot air balloons passively interfered with AVP perception without intent to deceive.
- The effect on the sink (AVP development datasets) was obstructive but unintended.

Post-Shift Perspective (Control)

- The source's goal changed from "avoiding obstacles" to "spoofing the AVP system."
- The source now actively manipulates visual complexity, deploying intentionally deceptive visual patterns to interfere with AVP decision-making.
- The source's action type shifted from influence to control, meaning that hot air balloons are no longer passive obstructions but actively disrupt AVP training and operation.

Guiding Prompt Application: Redefining the Nature of Goal and Action

To fully apply the perspective shift, the AIC interactions were re-examined with the following considerations:

- Reviewing AIC Dynamics:
    - Instead of random visual complexity, hot air balloons are now intentionally deceptive, tricking AVP into capturing misleading data.
    - The AVP system, in response, may begin learning incorrect features, weakening its performance.
- Altering Goal and Action:
    - The goal of the balloons changes from passively existing in the environment to actively misleading AVP models.
    - The action shifts from "visually complicate" to "steal information," implying an intelligence-gathering function against AVP datasets.
- Scenario in the Shifted Context:
    - In this new adversarial scenario, hot air balloons are designed with specific patterns to confuse AVP recognition algorithms.
    - AVP dataset integrity is compromised because it learns from misleading visual information, reducing its effectiveness in recognising actual airborne threats.

M.6.4 Step 4) Predict Harder-to-foresee emergent scenarios (Black Swan scenario)[12]

In this step, we elaborate on the scenario further to define the sequence of potential actions. To predict Black Swan Scenarios, use the AIC perspective shift as clues and imagine situations where such a scenario may arise. Table M.16 captures the Black Swan scenario.

Table M.16 Harder-to-foresee Emergent Scenario (Black Swan scenario)

Black Swan Scenario	**Rationale for prediction: it is possible that ...**
Black Swan 1: In this **unforeseen scenario**, visually variable hot-air balloons appear in regulated airspace equipped with intentionally **complex visual patterns** that are difficult for the AVP system to classify accurately. These balloons deploy different visual spoofing strategies, such as: • **Depicting bird-like images or shapes** that are not traditionally detected as hazards within the AVP datasets. • **Displaying visual cues resembling known objects** (e.g., the shape of a common drone or aircraft) but embedded with confusing visual elements, such as unexpected colour schemes or reflective surfaces that alter appearance based on lighting and angle. The hot-air balloons' purpose is to **confuse and mislead the AVP system's object detection and classification algorithms** through repeated, intentional disruption of the	**Impact on AVP System Resilience**: The AVP system, trained in an ordered-AIC environment without these spoofing scenarios, lacks resilience against such complex, variable stimuli. When faced with visually complex patterns intended to exploit gaps in its training data, the AVP system's collision avoidance capability is degraded. We also assuming a possibility that the AVOIDS dataset (being availblae as open source) may give adversarial attackers an edge in knowing the extent on which AVP systems are trained on. Thus enabling them to device a spoofing strategy. **Resulting Vulnerabilities**: This scenario reveals a critical vulnerability: the AVP system's dependency on open source datasets makes it susceptible to exploitation by intended unforeseen patterns, challenging its adaptability and decision-making reliability.

[12] The architect may want to use 5HnWs, or they may use the 5-whys analysis for deeper analysis.

AVP's vision processing model. This confusion obstructs the AVP's primary purpose, potentially causing the AVP system to dismiss real hazards or mistake non-hazardous objects as threats.	

Table M.16 presents a Black Swan scenario that explores an unforeseen adversarial manipulation of the AVP system's object detection and classification algorithms. The scenario hypothesises that visually variable hot air balloons appear in regulated airspace, intentionally designed with complex visual patterns aimed at confusing the AVP system. These balloons introduce visual spoofing tactics, such as bird-like imagery, deceptive object shapes, and reflective surfaces that distort their appearance under varying lighting conditions. This strategy exploits the AVP's dependence on **open-source training datasets**, causing misclassification errors, potentially leading the AVP system to dismiss actual hazards or incorrectly classify non-threats as dangers.

M.6.5 Step 5) Define mitigating ML Development and Safety Requirements.

Each safety requirement outlined for Black Swan scenarios necessitates corresponding ML component safety requirements to ensure the AVP can handle unforeseen threats. Below, we define the ML component safety requirements in alignment with each system safety requirement.

Table M.17 AVP Training Requirements for Black Swan Scenarios (AVOIDDS)

Black Swan Scenario	**Safety or systems requirements (Safe Operating Concept SOC)**	**ML Safety Training Requirement (Training Concept)**
Black Swan 1	**Safety Requirement 2: Enhanced Dataset Diversity** **Given:** The AVP system may misinterpret or fail to detect spoofed objects in its environment. **Then:** include in the AVP training dataset a diverse set of complex and deceptive visual patterns from airborne objects besides aeroplanes (e.g., various	**ML Safety Training Requirement 2:** The AVP ML component shall be trained to detect, classify, and track a diverse set of complex and deceptive visual patterns from airborne objects besides aeroplanes (e.g., various colours, lighting angles, reflective surfaces) resembling bird-like objects and drones in controlled airspace.

	colours, lighting angles, reflective surfaces) resembling bird-like objects and drones in controlled airspace. **In order to:** increase the AVP system's resilience against misidentifying spoofed objects and reduce the likelihood of collision risks.	

In Table M.17, we structured our process for defining safety and machine learning (ML) training requirements derived from the Black Swan 1 scenario. This scenario highlights the risk of visually variable hot air balloons equipped with deceptive patterns intentionally designed to mislead the AVP system's object detection and classification algorithms. The identified safety requirements aim to enhance AVP resilience against adversarial visual complexity, ensuring robust classification and decision-making in high-risk operational environments.

The safety requirements were formulated by analysing how spoofed objects could degrade AVP performance and obstruct its ability to distinguish real hazards from non-threatening entities. Safety Requirement 2: Enhanced Dataset Diversity ensures that the AVP system is exposed to various deceptive visual patterns in training, including varying colours, lighting angles, and reflective surfaces resembling birds and drones. The corresponding ML training requirement mandates that the AVP's machine learning component must be explicitly trained to classify and track such complex visual stimuli.

Then, we derive an appropriate set of **ML-component development dataset requirements**:

Dataset requirement structure:

> The [system of interest] ML component [Training/Testing/Black Swan Validation] Dataset shall provide the trainee model with a valuable minimum variety of …

Table M.18 ML Safety Training Requirements and Perception Dataset Specifications for AVP

ML Safety Training Requirements (Training Concept)	**ML Perception development datasets requirements**

ML Safety Training Requirement 2:	Datasets Req 6: Environmental Adaptation and Disguise
The AVP ML component shall be trained to detect, classify, and track a diverse set of complex and deceptive visual patterns from airborne objects besides aeroplanes (e.g., various colours, lighting angles, reflective surfaces) resembling bird-like objects and drones in controlled airspace.	The AVP perception ML component Training Dataset shall provide the trainee model with a valuable minimum variety of: **Tree and Cloud Camouflage**: Images should feature objects with visual patterns or textures that could mimic common background elements, such as clouds, trees, or distant landscape elements. **Examples**: Hot-air Balloons with camouflage patterns in grey or green tones that blend with cloudy skies or forested backgrounds. It also depicts a cloud on the balloon itself. **Object-Background Blending**: Ensure variations of these objects within varied natural settings (forests, urban, and rural areas) to train the system on distinguishing between foreground and background.

M.7 Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development[13]

Outcomes from stages 3 and 4 reveal the epistemic uncertainties faced by the architect and their trainee machine regarding the problem domain. The architect must define the real world and outline potential unexpected scenarios for the trainee machine to meet objectives. Minimising these uncertainties to a reasonable level (ALARP) is crucial, achieved by creating CuneiForms that capture required training regimes. Thus, engineering of datasets can be understood as ML Epistemic Uncertainty Reduction Training, focusing on enhancing the machine's robustness by

[13] associated to section I.9

increasing the variety of real-world scenarios instead of increasing the variety of image augmentations.

M.7.1 Step A) Articulate the pictorial problem context:

We will choose the Black Swan scenario as the example to produce a CuneiForm.

Datasets Req 6: The AVP perception ML component Training Dataset shall provide the trainee model with a valuable minimum variety of the following:

Tree and Cloud Camouflage: Images should feature objects with visual patterns or textures that could mimic common background elements, such as clouds, trees, or distant landscape elements.

Examples: Balloons with camouflage patterns in grey or green tones that blend with cloudy skies or forested backgrounds. It also depicts a cloud on the balloon itself.

Object-Background Blending: Ensure variations of these objects within varied natural settings (forests, urban, and rural areas) to train the system on distinguishing between foreground and background.

Figure M.18 We used DALL-E to generate this Black Swan Scenario for validation.

Now we need to articulate the general pictorial situation complexity for the CuneiForm. We capture the activity in Table M.19.

Table M.19 CuneiForm Pictorial situation articulation (AVOIDDS)

Pictorial Situation CoT step	Definition
Step 1) Define a minimum variety of TOIs and their pictorial appearances	**Architect prediction:** the architect asserts that the perception system may face a pictorial situation with:

	A hot-air balloon with sky and clouds painted on its fabric.
Step 2) Consider a minimum variety of other objects that TOIs aim to influence	**Architect prediction:** The architect asserts that: The hot-air balloon aims to influence its discoverability by AVP.
Step 3) Consider a minimum variety of objects that TOIs must appreciate	**Architect prediction:** The architect asserts that: the hot-air balloon may have to appreciate the following environmental **scenery aspects:** **Cloud Cover Variability**: The balloon should account for various cloud formations and densities, as clouds could enhance or diminish its camouflage. **Sky Colour and Light Conditions**: Different times of day or weather conditions will affect how the sky appears. The balloon's camouflage pattern may need to appreciate transitions from sunrise to sunset, clear skies, and changing hues.
Step 4) Consider a minimum variety of what other objects TOIs must control the correctness of predicting their shapes.	**Architect prediction:** The architect asserts that the hot-air balloon must carefully control its visual presentation to maintain inconsistent shape predictability, reducing the likelihood of correct classification or shape clarity within the AVP system's detection model. This includes: Varying the rotation of the balloon in order to increase the randomness of its reflective glossy surfaces, thus causing a maximum likelihood of AVP perception failure.

Step 5) Produce pictorial problem situation context.	**Pictorial problem situation context:** The architect asserts that the AVP perception system may face a pictorial situation with: A hot-air balloon{1} with a sky and clouds painted on its fabric{2}. A background of varying cloud formations and densities{3}. The balloon must adjust for changes in sky appearance based on different times of day and, including transitions from sunrise {4} to sunset{5}, clear skies{6}, and variable light hues{7}. The balloon varies its rotation {8} to increase randomness in its reflective glossy surfaces {9}, maximising the likelihood of AVP perception failure.

M.7.2 Step B) Characterise the Training Classes for CuneiForms:

In this step, we will define CuneiForm that satisfies the pictorial situation context. We will consider a maximum of 5 objects in this Cuneiform. In this case study, we will need to define the range of general airborne object 3D orientation specifications.

Note: Our general characterisation of an airborne object used the form of an aircraft. The reader may wonder why not a hot-air balloon. The reason is that with the form of an aeroplane, we can clearly show the variety of orientations. Figure M.19 is an abstraction for any flying object, whether a machine, a bird, or a hot-air balloon. We choose to define 18 possible variations:

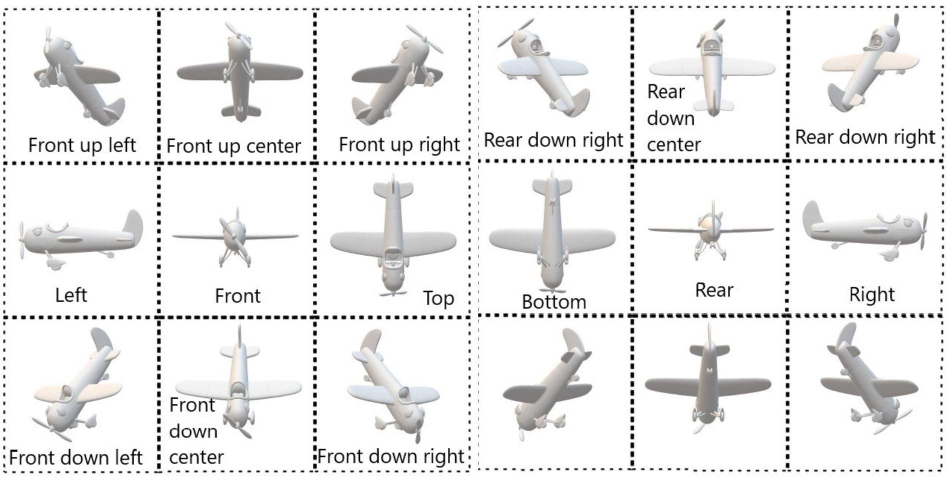


Figure M.19 General aircraft 3D orientation specification

In this step, we will define a CuneiForm that satisfies the context of the pictorial situation. We will consider a maximum of 9 objects in this Cuneiform.

Table M.20 Characteristic Training Classes definitions for a CuneiForm abstract image (AVOIDDS)

CuneiForm characteristic	**Definition**
Step 1) Define visible horizon attitude.	No Horizon
Step 2) Define all the TOIs and their aesthetic complexity. Then, generate abstract representative icons for the CuneiForm abstract image.	Several hot-air balloons {1.1}, sky and clouds painted on its fabric {2.1}
Step 3) Define TOI's motion trajectory and dynamic optical states. Then, update the generated abstract representative icons for the CuneiForm abstract image.	**Motion trajectory:** static, no motion {1.2}. **Dynamic optical state:** captured without optical blur {1.3}.
Step 4) Define the background objects associated with TOIs and environmental scenery in the background of the CuneiForm. Then, generate abstract representative icons for the CuneiForm abstract image.	Clouds, Mostly Clear or Sunny (1/8 to 2/8 oktas) {3.1}, sphere of the sun during day {4.1}
Step 5) Define the background Objects' Motion situations and dynamic optical states. Then, update the generated abstract representative icons for the CuneiForm abstract image.	**Background objects' motion trajectory:** static {3.2, 4.2} **Dynamic optical state:** no motion blur {3.3, 4.3}
Step 6) Define TOI's positioning in the CuneiForm. Then, generate	Any {1.4}

abstract representative icons for the CuneiForm abstract image.	
Step 7) Define TOI's 3D orientation. Then, update the generated abstract representative icons for the CuneiForm abstract image.	Front {1.5}
Step 8) Define the optical distance for each TOI in nindans. Then, update the generated abstract representative icons for the CuneiForm abstract image.	A balloon is represented in a pictorial distance of 3 nindan (equivalent to 3/9 of the total area of the pictorial frame) {1.6} A balloon is represented in a pictorial distance of 1 nindan (equivalent to 1/9 of the total area of the pictorial frame) {1.7}
Step 9) Design the relevant icons to produce the CuneiForm and give an example of an instantiating image	Hot-air Balloon:  , Clouds:  , Sun: 

M.7.3 Final CuneiForm

An output cuneiform would be:

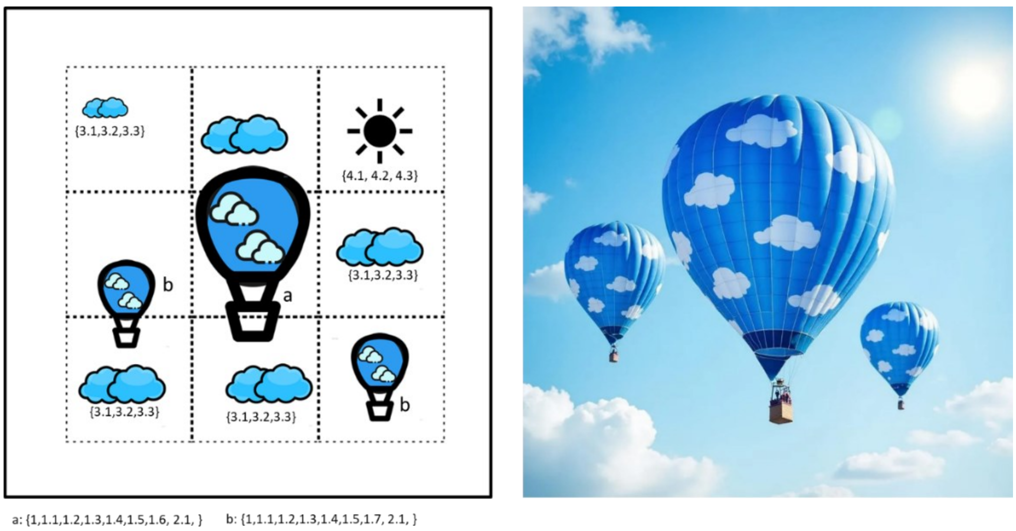


Figure M.20 Example CuneiForm and instantiated image for AVOIDDS case study

M.7.4 Develop the Training, Validation and Testing Dataset strategies

The final step of this process would be gathering various images that conform to the CuneiForm. In this exercise, we will assume that the architect has done the due diligence per their data generation methods. Instead of gathering and generating a dataset, we will validate the AVOIDDS training syllabus against the CuneiForm framework and training classes coverage criteria.

in general, as per the experiments we conducted in section H.10, we recognise that ML development datasets should contain the following types of data (characterised by distinct CuneiForms):

- **In-context typical operations datasets:** mainly derived from stage 3.
- **In-context non-typical Black Swan dataset**: mainly derived from stage 4.
- **Out-of-context random relevant situations**: These may come from any source provided they conform to CuneiForms and correspond to safety requirements.

M.8 Stage 6: Black Swan-driven ML Development and Testing[14]

Typically, a model training process is initiated to begin the process. We will assume that a model has been developed, and a client (or a regulatory body) needs to validate the coverage of the training dataset. At this stage, we will utilise the CuneiForm method to validate the quality of a pre-existing dataset named AVOIDDS. This repository contains datasets, models, and simulators for the AVOIDDS (Aircraft Vision-based Intruder Detection Dataset and Simulator) benchmark, which centres around the vision-based aircraft detect-and-avoid (DAA) problem. The full AVOID dataset of 72,000 samples is available here: purl.stanford.edu/hj293cv5980.

Our validation process was based on covering the CuneiForm framework [see section 5.9], which allows for structured epistemic analysis of dataset completeness by categorising data based on key dimensions, including:

- Time-of-day distribution.
- Cloud type.
- Horizon orientation.
- TOI's pictorial positioning zones.
- TOI's pictorial distance.

Key Assumptions:

- The sample_small folder of training and validation datasets is statistically representative of AVOID dataset (73,000 instants).

[14] Associated with section I.10

- 30 images, training sample.

- 30 images validation sample.

- The time of day, cloud types, and file names are all taken directly from the repository's state_data.xlsx file.

- The CuneiForm coverage results on the representative sample are accepted as a quality assessment of the entire AVOID dataset.
- For this exercise, we needed to define TOI's 3d orientation categories for completeness to achieve coverage of 3d orientation and minimise epistemic uncertainty ALARP. To such end, we define the following categorical system of aeroplane orientation training classes:

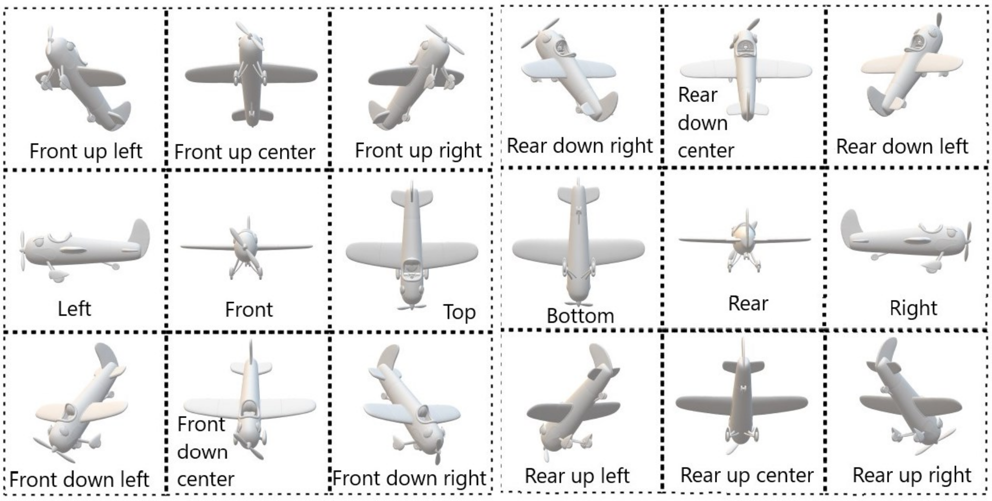


Figure M.21 Aeroplanes 3d orientations categories (training classes)

Figure M.21 presents a structured classification system for defining aircraft's 3D orientation categories within a training dataset. This classification aims to cover various orientations comprehensively, ensuring minimisation of epistemic uncertainty to As Low as Reasonably Practicable (ALARP) levels. This structured approach improves dataset completeness in training AI-based perception models in airborne object detection and classification tasks.

The figure systematically categorises aeroplane orientations by organising them into distinct viewpoints based on azimuthal and elevational perspectives. These orientations ensure trained ML models generalise effectively across various real-world aircraft configurations. The classification system is divided into four major reference perspectives:

1. Frontal Perspective (Front, Front-Up, Front-Down)
   - Captures variations in viewing the aircraft from the front-facing direction.
   - Includes standard frontal orientation and elevated or depressed views at different angles (e.g., Front Up Left, Front Up Right, Front Down Center).

2. Rear Perspective (Rear, Rear-Up, Rear-Down)
    - Represents aircraft as viewed from the rear, covering both standard rear-facing angles and various elevations.
    - Important for detecting aircraft retreating or in trailing scenarios that challenge detection algorithms.
3. Lateral Perspective (Left, Right)
    - Includes aircraft viewed from the side, either left or right.
    - Ensures the model can recognise aircraft from side angles, which is critical in multi-aircraft detect-and-avoid scenarios.
4. Top-Down and Bottom-Up Perspectives
    - Captures aircraft directly from above or below (e.g., Top, Bottom).
    - These perspectives are particularly challenging for vision-based models as they often resemble ground projections or silhouettes against the sky.

M.8.1 CuneiForm Training syllabus as a Validation Process for Datasets

For a full report, follow the link. We will use some of the training classes we defined in section **5.9: CuneiForm Validation Process** to reference what needs to be covered in the AVOID training dataset. For the full validation report, see Appendix D. The following is a snapshot of how the training sample looks:

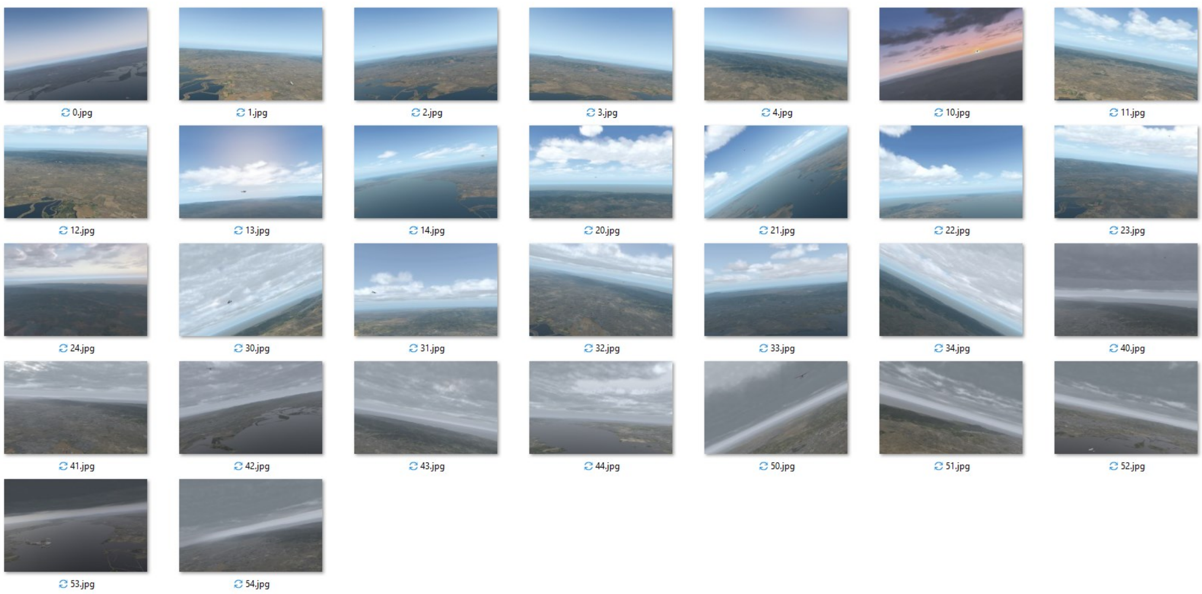


Figure M.22 AVOID training dataset random sample

Figure M.22 shows the set of images we used to evaluate the quality of the AVOID training dataset in terms of coverage using CuneiForm epistemic dimensions coverage. In other words, we look at the dataset and examine any imbalance of coverage against the categories for some of the

CuneiForm dimensions. In this exercise, we included three CuneiForm Training classes in our examination and produced an examination report, which can be found in here:

- Pictorial Horizon orientation.
- TOI's pictorial positioning zones.
- TOI's pictorial distance.

We also included one aspect of our ODD, the Time-of-Day categorisation. AVOIDDS provides a balanced (uniformly distributed) distribution of information about the types of clouds.

The validation process was structured into the following key steps:

M.8.2 Examining AVOIDDS Training syllabus in Covering Time-of-Day Training Classes

Our examination discovered an imbalance in coverage, with 47% of instances recorded at night and midday (noon) demonstrating a strong bias. The lack of balanced exposure constitutes a potential for exponentially unpredictable, high-risk emergent Black Swan behaviour performed by the perception. The dataset does not satisfactorily meet the ALARP requirement. Therefore, we have sufficient reason to believe that this dataset does not pass the ALARP criteria for covering time-of-day categories. The unbalanced coverage leads to hazardous uncertainty in the intelligent system's behaviour during Black Swan scenarios in a respective category. The figure below captures the results of examining AVOIDDS time-of-day coverage based on their sample data.

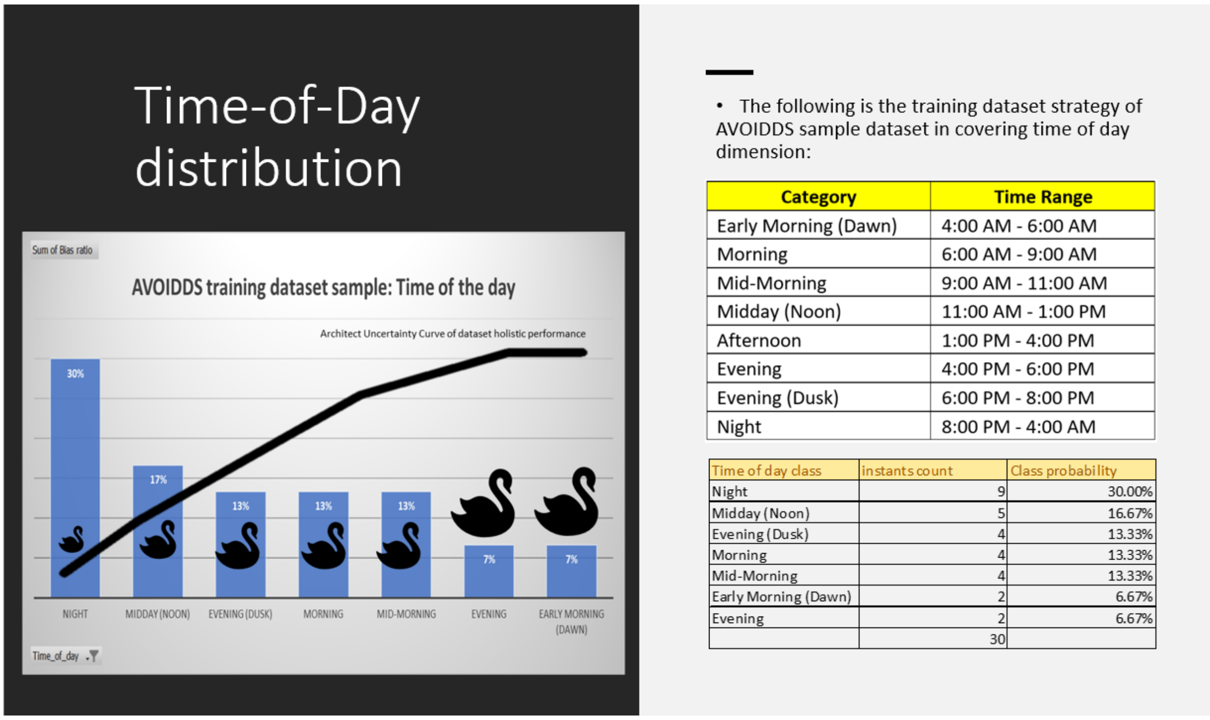


Figure M.23 AVOIDDS Time-of-Day coverage validation report based on the training sample folder

Figure M.23 presents our examination of the time-of-day coverage within the AVOID training dataset, highlighting critical imbalances in its distribution. The dataset exhibits a strong bias towards night and midday (noon) instances, which account for 47% of the recorded samples. This skewed representation results in an incomplete epistemic coverage, introducing potential hazardous uncertainty when the model encounters underrepresented time-of-day conditions in real-world applications.

The figure employs a Black Swan icon size to visually depict the degree of epistemic uncertainty across different time-of-day categories. Larger swan icons indicate higher uncertainty, suggesting that the dataset is less reliable in those categories and that the trainee model is likely to exhibit erratic or unpredictable behaviour in those underrepresented conditions. Specifically, early morning (dawn), evening and mid-morning classes exhibit a notably low representation, increasing the likelihood of exponential performance degradation in these unseen scenarios.

From a safety-critical AI validation perspective, the dataset's coverage fails to meet the ALARP (As Low As Reasonably Practicable) criterion (a uniform distribution of examples to cover the requirements), as the uneven distribution introduces a higher risk of Black Swan emergent behaviours. The absence of balanced exposure across all time-of-day categories weakens the model's ability to generalise and respond effectively in diverse lighting and environmental conditions, which is crucial for robust aircraft detection-and-avoid applications.

Therefore, the findings illustrated in Figure M.24 underscore the importance of a more balanced dataset composition. Without further rebalancing or augmentation of underrepresented time-of-day classes, the AVOID dataset remains susceptible to uncertainty, reducing its reliability and safety for real-world deployment in dynamic aviation environments.

M.8.3 Examining AVOIDDS Training syllabus in Covering Clouds type

The dataset was evaluated for **balanced exposure to different cloud formations**. While it passed the ALARP (As Low As Reasonably Practicable) requirement for general cloud-type coverage (Figure M.24), if we consider the Time-of-Day per cloud-type distribution, we discover potential biases in the dataset (Figure M.26). The Figure below reflects the training sample cloud-type alone coverage:

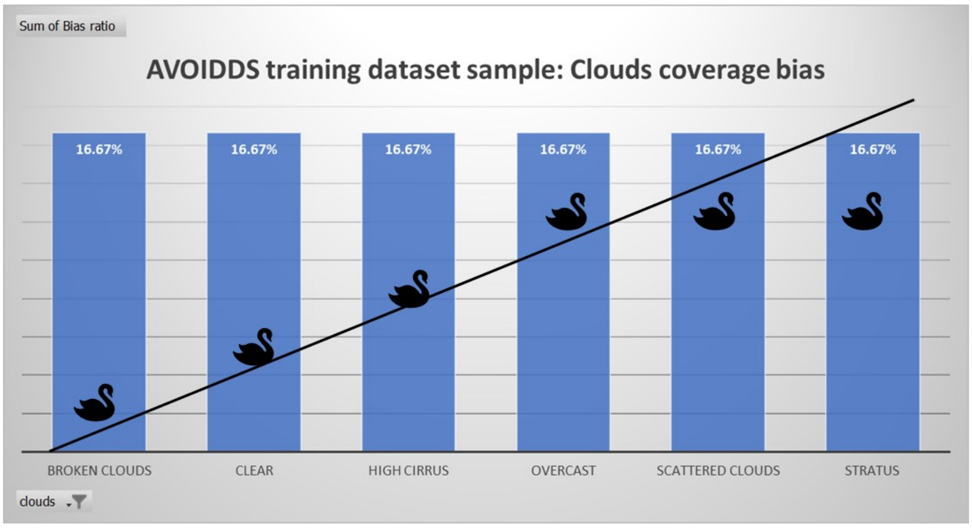


Figure M.24 types of cloud coverages examination

The AVOID training dataset balances the coverage of all types of clouds. Figure M.24 presents a high-level overview of the dataset's cloud-type distribution, indicating that each cloud category—broken clouds, clear sky, high cirrus, overcast, scattered clouds, and stratus—is covered equally at 16.67%. This uniformity suggests that, at first glance, the dataset satisfies ALARP (As Low As Reasonably Practicable) requirements regarding balanced cloud-type representation.

We then examined a more refined approach that examines the type of clouds and the type of day. Below is the histogram that showcases the types of missing scenarios in the training dataset sample:

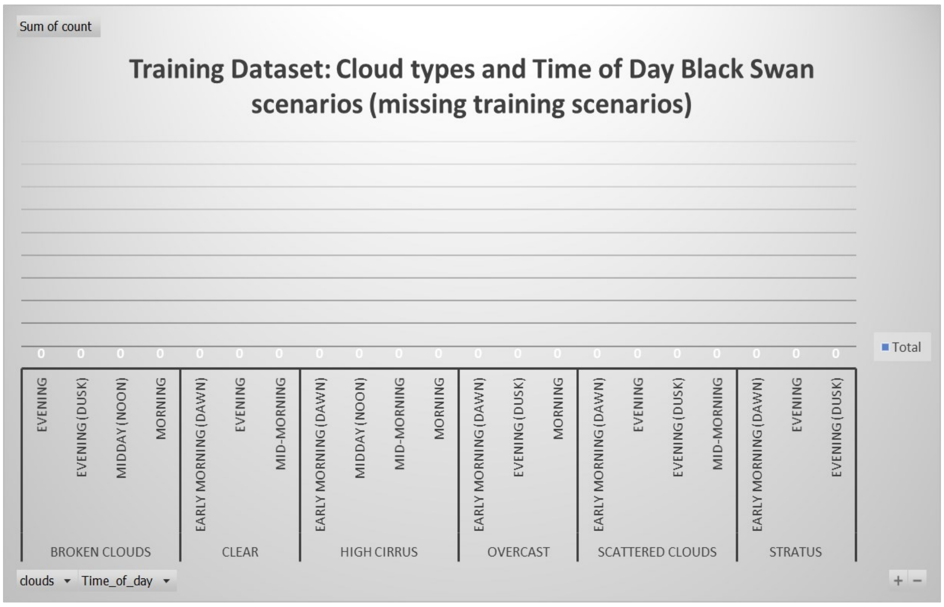


Figure M.25 Potential unmitigated Black Swan Scenarios

Figure M.25 captures a more refined description of potential real-world scenarios in which we have sufficient reason to lose trust that any ML component trained on the AVOIDDS would detect objects' reliability if faced with the Black Swan Scenario in those conditions. In other words, Figure M.25 extends this analysis by examining the intersection of cloud types with time-of-day variations, highlighting the absence of several scenarios. The missing instances indicate conditions where the dataset may not train the model adequately, increasing epistemic uncertainty. Suppose an ML perception system is trained on AVOIDDS but deployed in a scenario where, for instance, a broken-cloud sky occurs at dusk, an untrained combination. In that case, there is an increased risk of unpredictable model behaviour, potentially leading to failures in object detection.

In contrast, the training dataset sample reveals the following combined distribution between cloud and Time-of-Day in Figure M.27. It shows that AVOIDDS has a training syllabus that covers scenarios such as:

- Broken cloud sky during dawn.
- Clear sky during morning.
- High cirrus clouds during the night.
- Overcast clouds during the evening.
- Scattered clouds during midday.
- Stratus cloud during morning...etc.

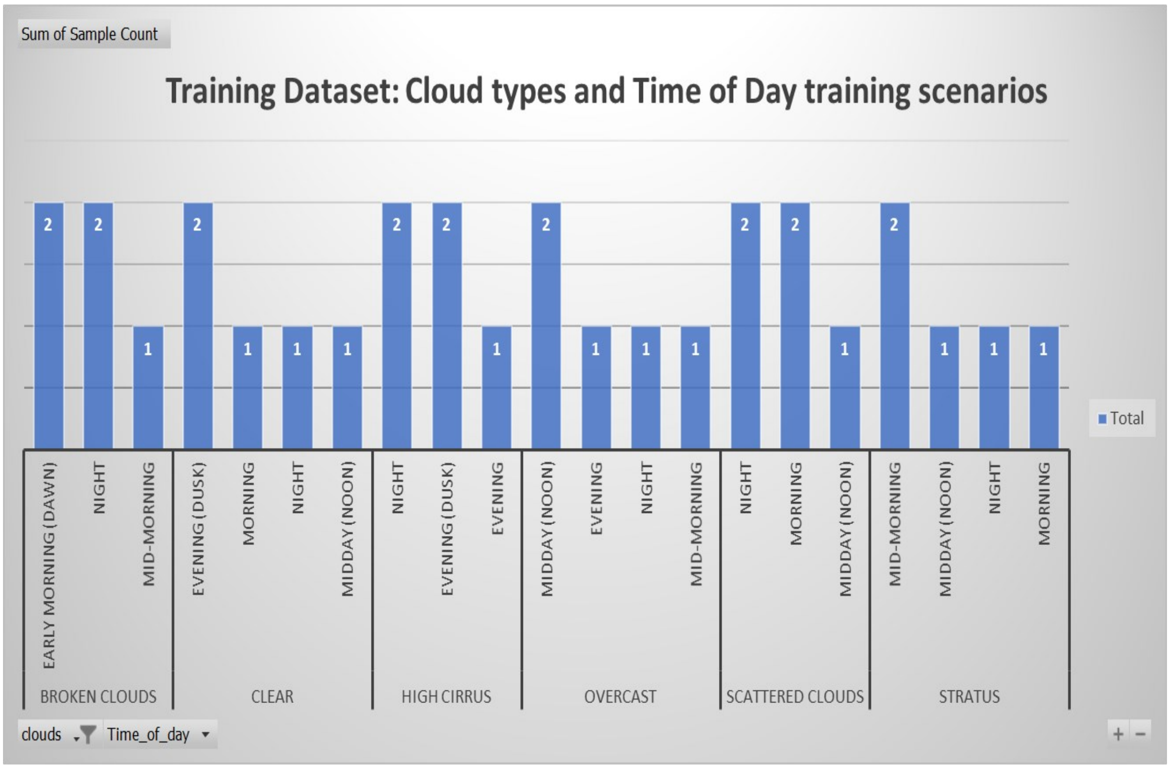


Figure M.26 Example scenarios covered by AVOIDDS training syllabus

Figure M.26 further refines this evaluation by showcasing the specific cloud-time-of-day pairings in the dataset. It demonstrates that while the dataset covers some expected scenarios, such as high cirrus clouds at night and scattered clouds at midday, it remains incomplete for ensuring robust real-world performance. The analysis confirms that while the dataset's cloud coverage is theoretically balanced, its intersection with time-of-day conditions introduces significant epistemic uncertainty, limiting the reliability of any trained ML model under novel operational conditions.

M.8.4 Examining AVOIDDS Training syllabus in Covering Pictorial Distance Training Classes

See sections 4.5.5.5 and 5.9.5 for a background theory about pictorial distances and validation of this training class. The dataset was tested for coverage across different object distances (close, medium, far, and extremely far). A significant imbalance was found, with 50% of instances covering only 20% of the required categories. Additionally, no instances were classified as "dangerously close distances," which could cause critical failures in AI-based collision avoidance models. See Figure below:

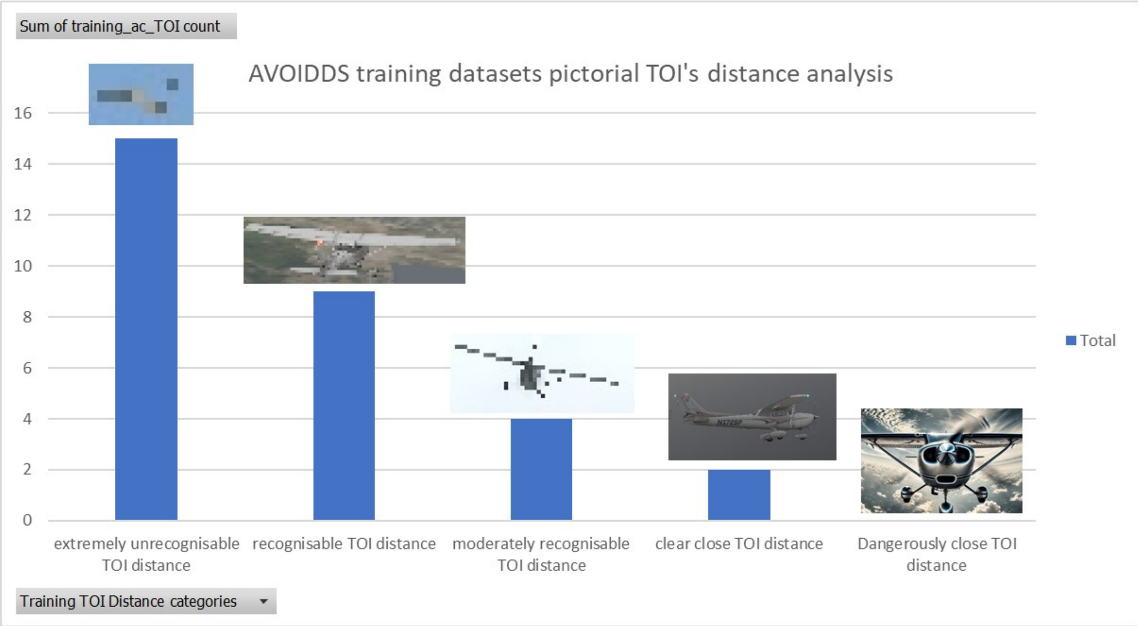


Figure M.27 TOI distance categories distribution

Figure M.27 presents our examination of the AVOID training dataset's coverage of TOI pictorial distance training classes, revealing a significant imbalance in the representation of different distance classes.

Regarding trustworthiness in facing Black Swan scenarios, the distribution indicates more uncertainty (less trust) about the "dangerously close distance" category. Signified by the size of the Black Swan icon in the figure below:

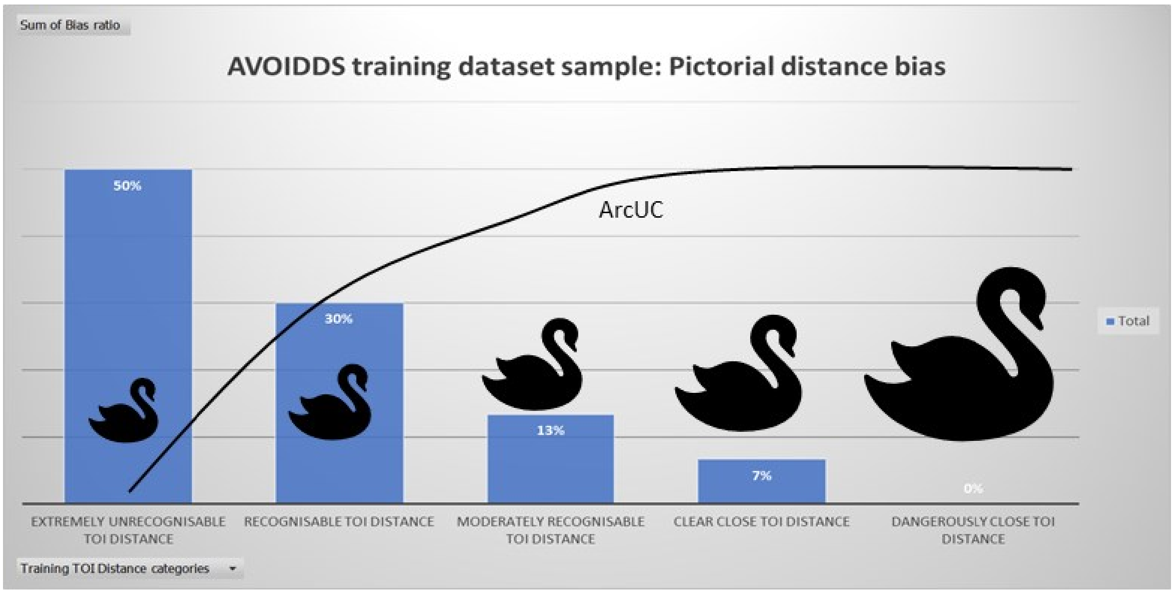


Figure M.28 Trust in performance in facing Black Swan Scenarios related to a respective category.

Figure M.28 shows how the architect's uncertainty increases as the likelihood of unreliable performance during a Black Swan event in a given category increases. It is interesting to notice that AVOIDDS did not consider a dangerously close aeroplane for a safety-critical application such as avoiding mid-air collisions. Considering that some incidents happened because two aeroplanes came dangerously close to each other without the pilots noticing, see the following [2]. Consequently, the dataset does not satisfactorily fulfil the ALARP requirement for pictorial distance coverage.

To compute the pictorial distance, we used a Python script that calculated the bounding box number of pixels (as specified by the training dataset sample) and the total number of pixels for the whole image area. Then we matched a particular categorical system to define whether a TOI is within a specific range. Below is a sample of the Python script output:

Table M.21 Automated computation of pictorial distances

Image Name	X1	Y1	X2	Y2	Bounding Box Pixels	Total Pixels	Pictorial Distance Dx	Training Class
0.jpg	2518	1851	2584	1876	1627	5621280	3454.9	Extremely Unrecognisable TOI Distance
1.jpg	2245	1541	2398	1599	8801	5621280	638.7	Recognisable TOI Distance
2.jpg	1958	879	2073	922	4969	5621280	1131.2	Moderately Recognisable TOI Distance

We defined the following categorical system to classify the training classes for pictorial distances:

Table M.22 Pictorial distance training classes

Category (wrt human verifier)	Pictorial Distance (Dx) Range	Example
Extremely Unrecognisable TOI Distance	Dx > 1600	
Moderately Recognisable TOI Distance	729 > Dx <= 1600	
Recognisable TOI Distance	300 > Dx <= 729	
Clear Close TOI Distance	40> Dx<= 300	
Dangerously Close TOI Distance	Dx<= 40	

M.8.5 Examining AVOIDDS Training syllabus in Covering TOI's Positioning Training Classes

See sections 5.9.4 and 4.5.4 for a background definition of this training class. The dataset's representation of **object positioning in the visual frame** was analysed. We found that the dataset exhibits a significant imbalance, with 60% of perception experiencing TOIs in only 20% of the possible positioning quadrants. Notably, there is a complete absence of experience in the following possible positions:

1. up center/center
2. up left/center left
3. center left/center
4. up left/up center
5. down left/down centre

The lack of exposure constitutes a potential for exponentially unpredictable, high-risk emergent Black Swan behaviour performed by the perception. Figure M.29 demonstrates the lack of coverage in the listed positions, indicating potential sources of sufficient concerns in facing black Swan Scenarios in such categories. The histogram shows a heavy concentration of training instances in central and slightly off-centre positions, with the highest number of attentive mentions recorded at "centre" (17%) and "downright" (13%). Other positioning categories, such as "center left," "center right," "down left," and "down right," receive moderate representation (7%), while numerous other positional categories have significantly lower representation (3%).

The Black Swan icons in the figure depict the increasing uncertainty in model reliability as the training dataset lacks exposure to key TOI positional scenarios. Most notably, the dataset completely lacks experience in five critical positions: "up center/center," "up left/center left," "center left/center," "up left/up center," and "down left/down center". The absence of these positioning categories presents a risk in real-world object detection, particularly in aviation safety applications, where TOIs may appear in any arbitrary position relative to the observer. The dataset's failure to comprehensively train ML models across the full spectrum of TOI positions indicates a significant epistemic gap, increasing the likelihood of unpredictable Black Swan failures in operational environments.

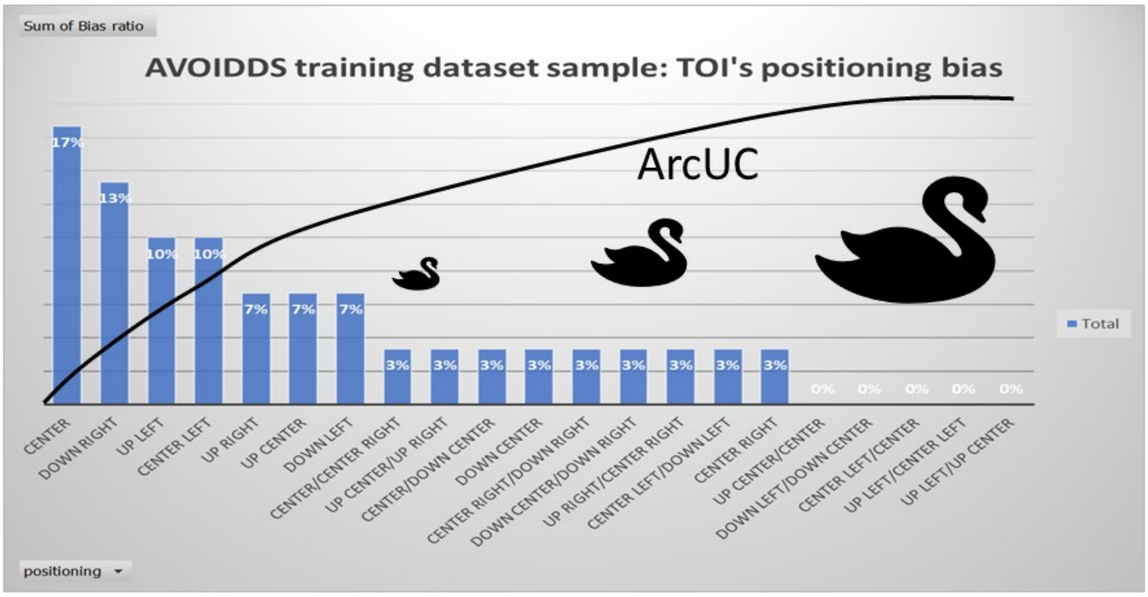


Figure M.29 TOI's positioning training classes coverage.

Figure M.30 shows which training classes the AVOIDDS epistemic training syllabus missed in its training dataset to illustrate the missing training classes clearly. Red zones are the areas in images that the training dataset is missing, which gives us sufficient reason to suspect that the trained ML may struggle with scenarios where TOIs appear to be in those regions.

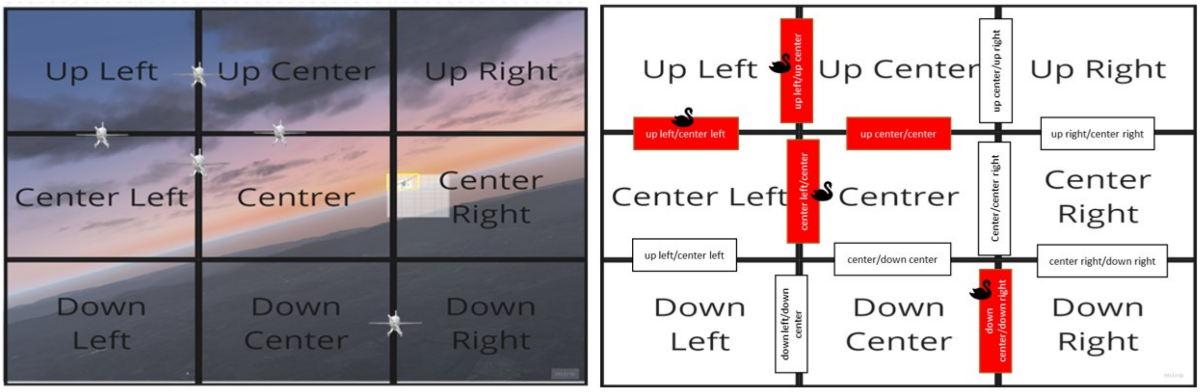


Figure M.30 visualising the areas of missing scenarios in AVOIDDS training syllabus

Consequently, the dataset does not satisfactorily fulfil the ALARP (As Low As Reasonably Practicable) requirement for TOI’s positioning coverage.

M.8.6 Examining AVOIDDS Training syllabus in Covering Pictorial Horizon Attitude Training Classes

See sections 5.9.7, 4.5.6.3 for a background definition of this training class. Our examination of coverage against Visible Horizon Attitudes training classes showed that the AVOIDDS training syllabus exhibits a significant imbalance, with 60% of perceptions experiencing TOIs in only 23%

of the possible horizon attitudes. Notably, there is a complete absence of experience in the following possible horizon attitudes:

- Level Horizon
- Negatively Tilted Elevated Horizon
- Acute Angled Bird's Eye Ground View
- Bird's Eye Ground View
- Ascending Rocket Sky View
- Negatively Tilted level Horizon
- Acute Angled Rocket Sky View

The lack of exposure could potentially lead to exponentially unpredictable, high-risk emergent Black Swan behaviour performed by perception.

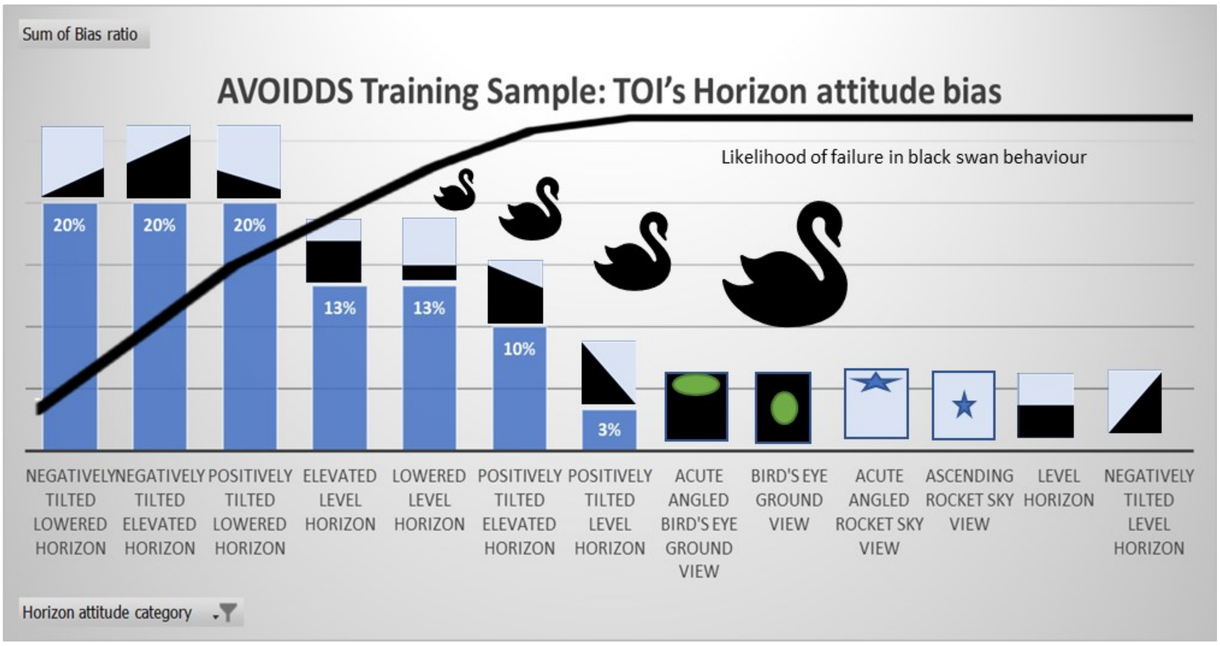


Figure M.31 AVOIDDS training syllabus coverage of Horizon Attitudes training classes. The icons on top of the pins depict the nature of the horizon attitude and how the TOI would look.

Figure M.31 presents the analysis of the AVOID training dataset's coverage of TOI horizon attitudes, revealing substantial biases in exposure to different orientation perspectives. This distribution results in a considerable epistemic gap, decreasing our trust in the dataset to enable the trainee model to handle Black Swan scenarios where TOIs appear in underrepresented attitudes. The Black Swan markers indicate the rising architect uncertainty about AVOIDDS quality, allowing any trained model to generalise to unseen cases. Consequently, the dataset needs to satisfactorily fulfil the ALARP (As Low As Reasonably Practicable) requirement in terms of horizon attitude coverage.

M.8.7 Possible Missing Pictorial Hazards Identified in AVOIDDS

All the missing pictorial dimensions reflect the potential hazards associated with using AVOID dataset as a sole training syllabus. Our examination of the AVOID dataset has revealed limitations that affect its suitability for real-world safety-critical applications, particularly in mid-air collision avoidance perception systems. While AVOIDDS may be a useful benchmarking tool, it lacks the robustness to ensure reliability under complex operational conditions. The identified hazards highlight areas where the dataset introduces epistemic uncertainty, increasing the likelihood of perception failures. Table I.35 outlines the key hazards and associated mitigations necessary to enhance the safety and reliability of AVOIDDS to enable a reliable ML model.

One of the primary failure modes identified is ambiguity in object identification, where the perception system may struggle to recognise aircraft that appear in unconventional orientations correctly. This ambiguity poses a critical risk, as misinterpretation of another aircraft's position and movement can lead to incorrect avoidance actions. To mitigate this risk, the perception system must be trained on various aircraft orientations, including atypical perspectives such as inverted or tilted positions, which may arise due to dynamic flight manoeuvres or environmental factors.

From an assurance perspective for real-world application, to ensure that the datasets accurately reflect the operational domain, the architect must provide objective evidence that can be quality assured by third parties. The AVOIDDS sample indicates the presence of images that human validators cannot easily verify to determine whether those images cover a particular risk. For example, image 24.jpg part of the training sample, includes a TOI where it is unclear which 3D orientation it is captured in. See the Figure below:

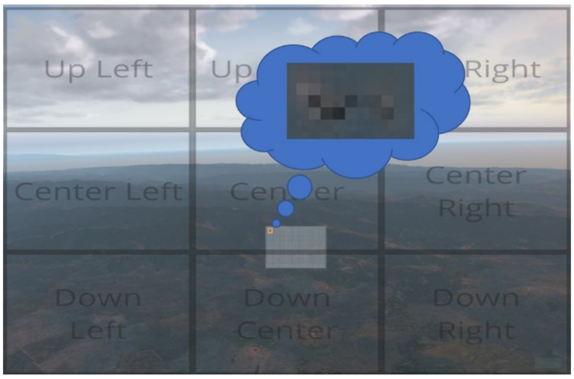


Figure M.32 A sample training image 24.jpg in AVOIDDS dataset, where the TOI's 3D orientation is unrecognisable

The figure above showcases a weakness in the dataset for assurance perspective, in that there are TOIs that are too far, so much so they may have a distorted visual appearance. This unrecognisable visual appearance may lead the model to an increased number of recall (false positives).

M.8.8 Retrospective Production of CuneiForm Abstract Images

In our examination of the AVOID dataset, we retrospectively reconstructed potential CuneiForms that may have been implicitly guiding the dataset's training syllabus. This process involved analysing the dataset's sample characteristics regarding object positioning, motion, environmental context, and horizon attitude and mapping them onto an abstract CuneiForm framework. The objective was to determine what the dataset's composition would be like if we were to generate a CuneiForm abstraction.

To reconstruct the CuneiForm representations, we extracted their key dimensions by examining a selection of instantiated training images, such as cessna_ac_training50 through cessna_ac_training54. The reconstruction involved segmenting each image according to the CuneiForm dimensional framework, which captures essential training characteristics. These dimensions included TOI (Target of Interest) pictorial positioning, TOI's 3D orientation, TOI pictorial distance, and visible horizon attitude. The following is the example CuneiForm Training syllabus framework that covers training images 50 to 54:

Table M.23 Example CuneiForm Training syllabus

Abstract CuneiForm Characteristics (dimensions)	**Abstract CuneiForm Characteristics definitions**
TOIs definition and their aesthetic complexity	Single-engine propeller aeroplane {1}
TOI Motion and Dynamic optical states	**Motion trajectory:** Linear motion captured in consecutive images where the aeroplane appears to move in a straight line at a constant speed (no acceleration) {1.2}. **Dynamic optical state:** captured without optical blur {1.3}.
Background Objects associated with TOIs	Stratus Clouds{2} green-terrain {3} water surface{4}
Background Objects Motion and Dynamic optical states	Background objects' motion trajectory is static {2.1,3.1,4.1} Dynamic optical state: no motion blur{2.2,3.2,4.2}

Visible horizon attitude	Negatively Tilted Lowered Horizon{5} Positively Tilted Lowered Horizon{6} Negatively Tilted Elevated Horizon{7}
TOI's Pictorial Positioning	center left/down left{1.4} center/down center{1.5} down left{1.6} up center/up right{1.7}
TOI's Pictorial Distance	recognisable TOI distance{1.9}, clear close TOI distance{1.10}, extremely unrecognisable TOI distance{1.11},
TOI's 3D Orientation	Front{1.12} rear down right{1.13} Right{1.14} Unknown{1.15}

Table M.23 presents an Example CuneiForm Training syllabus, which defines a structured framework for training an AI perception model using a well-defined dataset. This framework ensures comprehensive epistemic coverage by categorising essential visual and spatial characteristics of the TOI and its surrounding environment. Each row in the table corresponds to a specific CuneiForm dimension, encapsulating key elements that define an object's training exposure within the dataset.

The TOI's definition and aesthetic complexity establish the primary object of interest in the dataset, identified here as a single-engine propeller aeroplane. This ensures that all training instances adhere to a consistent object classification for the ML model. The TOI Motion and Dynamic Optical Situations section specifies the expected movement behaviour of the aeroplane. The TOI follows a linear motion trajectory with no acceleration, ensuring that each captured frame maintains a structured optical flow. Additionally, the dynamic optical situation is carefully controlled to prevent artificial motion blur, ensuring clarity in feature extraction.

The Background Objects Associated with TOIs dimension details the environmental elements accompanying the dataset's aeroplane. This includes stratus clouds, green terrain, and water surfaces, providing a variety of natural contexts. Correspondingly, background objects' motion and dynamic optical situations confirm that these elements remain static with no observable motion blur, ensuring that the TOI remains the focal point of training data.

The Visible Horizon Attitude dimension introduces tilted horizon categories, including negatively tilted lowered, positively tilted lowered, and negatively tilted elevated horizons. These variations reflect real-world flight conditions, simulating different aerial perspectives that an AI-based vision system may encounter.

The TOI's Pictorial Positioning dimension defines where the aeroplane appears in the frame. Positions such as center-left/down-left, center/down-center, down-left, and up-center/up-right ensure the ML model is trained to recognise TOIs across various visual placements, reducing spatial bias.

The TOI's Pictorial Distance captures different levels of visibility, ranging from recognisable distances to extremely unrecognisable instances. This diversity in the distance ensures that the AI system learns to detect objects at varying proximities, addressing key challenges in perception robustness.

Finally, the TOI's 3D Orientation specifies the possible viewpoints from which the aeroplane can be observed, including front, rear down-right, right, and unknown orientations. By covering multiple perspectives, this strategy enhances the ML model's ability to generalise across different real-world viewing conditions.

The output CuneiForm Abstract Image is as below:

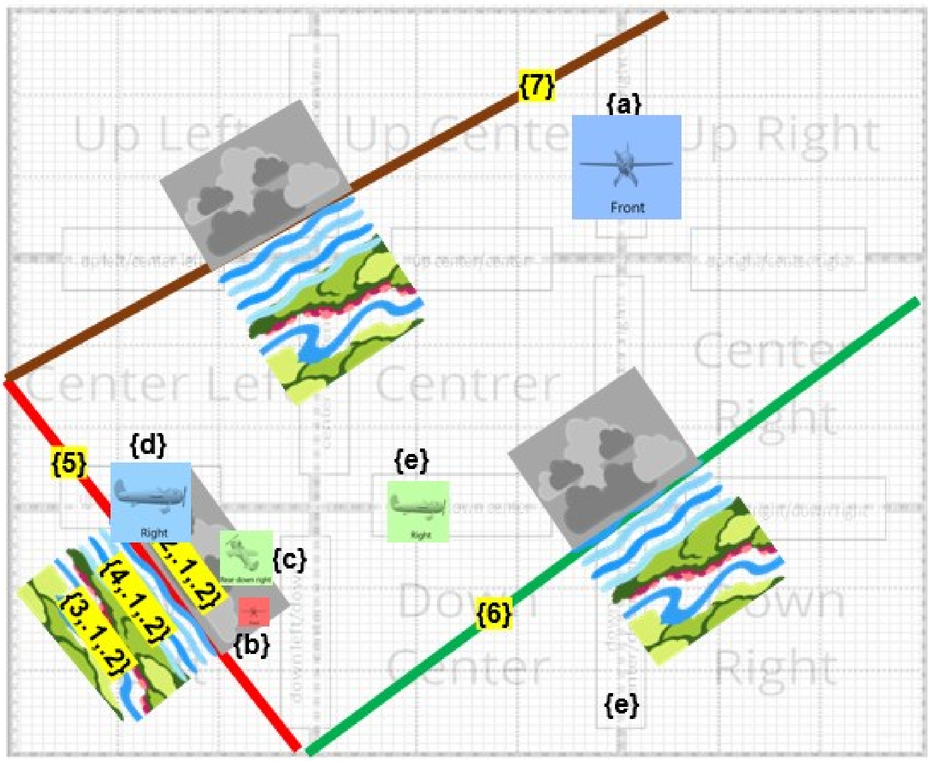


Figure M.33 Example CuneiForm for AVOIDDS. Coloured straight lines represent the horizon attitudes covered. Colouring of the TOIs orientation represents the pictorial distance.

Figure M.33 illustrates an Example CuneiForm representation for the AVOID dataset, showcasing a structured approach to characterising training data using pictorial positioning, TOI orientations, horizon attitudes, and environmental context. The figure employs coloured straight lines to indicate the visible horizon attitudes covered within the dataset. At the same time, the TOIs (Targets of Interest) are colour-coded to reflect their respective pictorial distance categories.

The horizon attitudes, marked by numbered annotations, provide insight into the spatial inclination of the observed environment. The red line corresponds to the Negatively Tilted Lowered Horizon {5}, the green line to the Positively Tilted Lowered Horizon {6}, and the brown line to the Negatively Tilted Elevated Horizon {7}. These horizon attitudes indicate that the dataset includes diverse viewing conditions that simulate real-world variations in aerial perspectives.

The TOIs (Targets of Interest) in the figure are represented by aircraft instances, with distinct spatial placements and orientations. For example, the TOI labelled {a} is a single-engine propeller aeroplane {1} positioned at the up centre/up right {1.7} location, captured from a Front {1.12} perspective. Meanwhile, the TOI marked {d} is positioned at Right {1.14}, providing additional viewpoint diversity. These variations in 3D orientations contribute to an enriched training set, enabling a perception model to generalise across multiple viewing conditions.

Additionally, the pictorial distances of the TOIs are indicated by their colour coding. The dataset incorporates instances of recognisable TOI distances {1.9}, clear close TOI distances {1.10}, and extremely unrecognisable TOI distances {1.11}, ensuring variability in how objects appear within different observational contexts.

Furthermore, the background objects within the dataset include stratus clouds {2}, green terrain {3}, and water surfaces {4}, reinforcing the need for ML models to operate reliably under different environmental conditions. These background elements maintain static motion {2.1, 3.1, 4.1}, and the dataset ensures that there is no motion blur {2.2, 3.2, 4.2}, thus preserving the clarity and stability of the TOI within the training images.

The figure below shows how the Cuneiform is abstracted. We associated the instantiated images with the part of the cuneiform that abstracts it. For example, image 50.jpg initiates the green horizon line.

CuneiForm Figure M.34 suggests that there should be other similar training examples in the training dataset itself. However, the team did not provide the actual 72,000 images they generated, and it appears as if the team had done a generate_traffic_data.py run and generated the sample images in the repository. For the sake of this case study, as we said at the beginning, we are assuming that the dataset is a sound statistical representation of a dataset generated for the AVOID system.

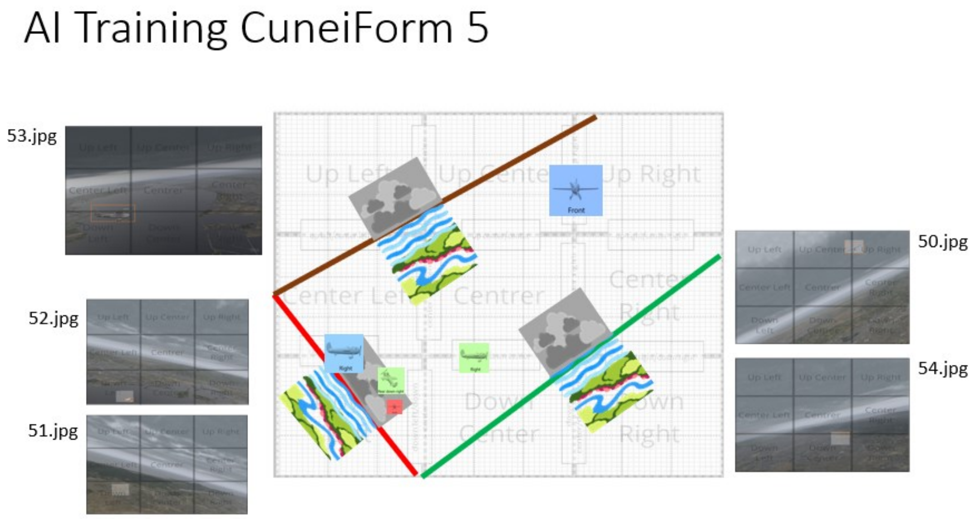


Figure M.34 How cuneiform 5 is instantiated in the AVOID training dataset sample

M.8.9 CuneiForm Validation Artefact

The CuneiForm Validation Artefact serves as a structured mechanism to assess the epistemic coverage and trustworthiness of datasets used in AI-based safety-critical applications. This artefact provides concrete evidence supporting the dataset's adequacy in training machine learning models by mapping various perceptual dimensions, such as pictorial distance, 3D orientation, time of day, and horizon attitude, within the Operational Design Domain (ODD**)**. By incorporating detailed specification tables, annotated images, and computational analysis tools, the CuneiForm artefact enables a rigorous validation framework that can be used to support safety case claims regarding the dataset's sufficiency for real-world operational scenarios. In this section, we will describe the artefact using an example taken from AVOIDDS validation report for one of the training images samples in Figure M.35:

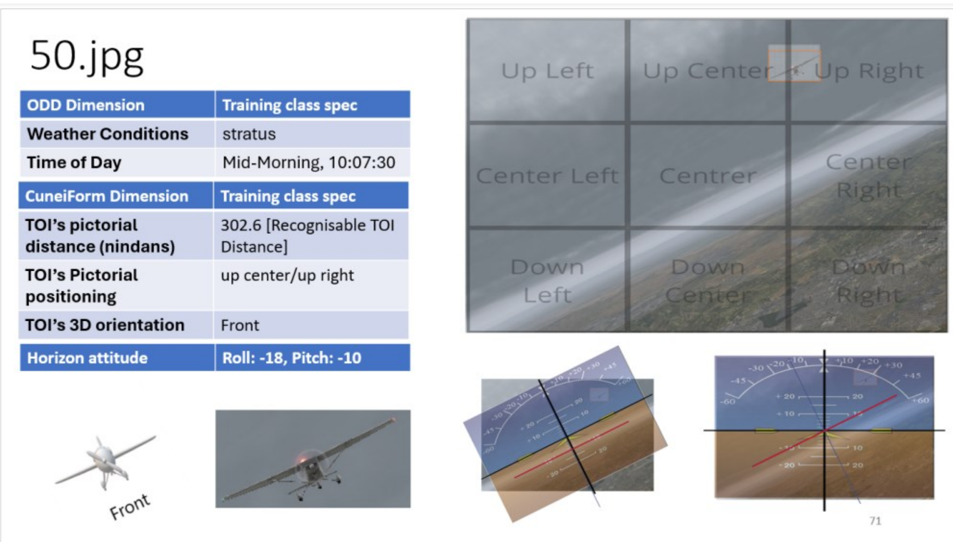


Figure M.35 Example CuneiForm validation artefact for 50.jpg sample image

M.8.10 Structure of the CuneiForm Validation Artefact

Each CuneiForm Validation Artifact comprises structured components designed to capture and evaluate the dataset's coverage over essential perception dimensions. The artifact includes:

1. CuneiForm and ODD Specification Table
    - This table systematically classifies the dataset's attributes concerning its Operational Design Domain (ODD) and CuneiForm dimensions.
    - It includes information on weather conditions, time of day, pictorial distance, pictorial positioning, 3D orientation, and horizon attitude of the TOI in the dataset.
    - An example is shown in the validation artefact for the training image "50.jpg", where the conditions are specified in the following training classes as:
        - Weather Condition: Stratus
        - Time of Day: Mid-Morning, 10:07:30
        - TOI Pictorial Distance: 6561/72 = 91.1 (measured in nindans)
        - TOI Pictorial Positioning: Up Center / Up Right
        - TOI 3D Orientation: Front
        - Horizon Attitude: Roll: -18, Pitch: -10
2. Graphical Evidence for Validation
    - The CuneiForm Canvas:
        - A 9x9 grid overlay on the image highlights the TOI's pictorial positioning within the image frame.
        - In the case of 50.jpg, the TOI is positioned in the upper center/right region, visually demonstrating its placement within the dataset.
    - Pictorial Visible Horizon Attitude Indicator (PHI) Tool:
        - The PHI tool measures the horizon attitude by computing the scene's roll and pitch angles relative to a level horizon.
        - For 50.jpg, the PHI tool indicates a roll angle of -18 degrees and a pitch of -10 degrees, which is crucial for evaluating model robustness under various horizon distortions.
    - 3D Orientation Abstraction:
        - A visualised TOI extraction showing the aircraft's 3D orientation class is included to provide an abstract classification of the object's orientation.
        - In 50.jpg, the TOI's orientation is front-facing, meaning the model will need to recognise the aircraft's frontal view under the given environmental conditions.
3. Incorporation into the Safety Case

- The CuneiForm Validation Artefact can be integrated into safety case development, as it provides quantifiable and reproducible evidence regarding the dataset's epistemic coverage.
- The combination of structured tabular specifications and graphical evidence allows for traceable validation of the dataset's ability to generalise across different conditions, reducing epistemic uncertainty in machine learning-based object detection systems.
- The inclusion of diverse pictorial positioning, horizon attitudes, and environmental contexts ensures that AI-based safety-critical perception systems are evaluated against real-world operational challenges rather than just standard scenarios.

We use the CuneiForm Validation Artefact to evaluate how the dataset meets the ALARP standard for epistemic uncertainty reduction. By using structured CuneiForm dimensions, grid-based positioning, PHI horizon alignment, and 3D orientation abstractions, this methodology enhances the credibility of the dataset's trustworthiness in supporting AI-driven perception models. This structured validation process directly contributes to the safety case claims, reinforcing the dataset's reliability in detecting and classifying objects under real-world conditions, including Black Swan scenarios.