Mapping Parameter Space Using Human Detection Metrics

A Quasi-Monte Carlo Approach to Camouflage Optimization

Master thesis TU Delft Author: Coen B. Duijnhouwer

Supervisors: Maarten A. Hogervorst, Sylvia C. Pont, Maarten W.A. Wijntjes

Date: 04/05/2026

TABLE OF CONTENTS

Abstract

Camouflage development traditionally relies on comparing a small number of handpicked patterns in human detection experiments or, more recently, on automated evaluations using computer vision models. Both approaches come with their own limitations: the former depends on a restricted number of candidate patterns which may suffer from a researcher's bias, while the latter may fail to reflect human perception. These methodological constraints hinder systematic exploration of how camouflage pattern parameters interact to influence detectability by human observers.

This thesis introduces an alternative method that captures human detection performance via a sampling process across a continuous N-dimensional parameter space, where each parameter combination defines a unique camouflage pattern (in this study N = 4). The corresponding human response is then modelled as a noisy observation from an underlying latent detection-difficulty function, allowing the modelling of how individual parameters and parameter interactions shape overall detectability.

Blender's render software and Python API integration were used to generate a large volume of fully synthetic, parametrically defined stimulus samples (18000 unique images in total). Since fully synthetic camouflage visualizations can be generated with precise, granular control and at negligible cost, this approach enables dense, non-repeating, human-in-the-loop sampling across multidimensional parameter variations. This study shows evidence that a machine learning model can map the parameter space and predict camouflage performance. We also provide recommendations which should enable a significant improvement in model accuracy and reduce the number of trials needed to reach saturation, as well as suggestions for other areas where this method and tool can be applied.

Keywords: *camouflage optimization, military, parametric design, synthetic environment, simulation, automation, Blender, Geo-Scatter, human-in-the-loop, visual search, psychophysics, Sobol sampling, Quasi-Monte Carlo*

1 Introduction

Camouflage involves the strategic use of materials, colors, shape, or lighting to diminish the visibility of an object against its immediate surroundings. Employed in both the natural world and military practices, the primary aim of camouflage is to enhance survival prospects by lowering the likelihood of detection and/or identification. Within a military setting, camouflage patterns are used to merge crucial assets seamlessly with their environment. Understanding the effectiveness of camouflage patterns in reducing an object's visibility and discerning the conditions under which they are most effective is crucial.

The study of these principles is not new; Penrose's (1941) *Home Guard Manual of Camouflage* already synthesized lessons from WWI and early WWII, using nature as a guide to principles such as background matching, texture, color, and shadow. The codification of these concepts in wartime field manuals reflects a long-standing and practical understanding of visual concealment.

Today, the same principles underpin modern camouflage development, now supported by computational modeling, controlled detection experiments, and quantitative design-space analysis, enabling a more rigorous development cycle built on foundations established a century ago.

Despite these advances, camouflage assessment studies have typically evaluated a small, handpicked set of candidate patterns in direct or ranked comparisons. This approach can reveal which of the nominated patterns performs best, but it cannot answer the more fundamental question: which pattern, across the full space of possible designs, performs best? The answer depends entirely on the quality of the initial selection. Patterns that were not considered cannot be evaluated and selection is inevitably shaped by intuition, precedent, or bias rather than purely systematic exploration. Systematic exploration is constrained in part by the limited stimulus generation potential of traditional methods.

This thesis takes a different approach. Rather than comparing a fixed set of candidates, it combines the advantages of synthetic simulations and automated stimulus generation with a human detection metric to map a broad parameter space, then models and selects an optimal pattern from that mapping, borrowing from parameter-space exploration logic common in computational optimization while keeping human

perception as the primary evaluation signal. This approach was developed in collaboration with TNO, whose existing capabilities in hybrid simulation and conspicuity measurement defined both the gap this study addresses and the complementary role it is designed to fill.

1.1 Background

Three categories broadly define camouflage research methodology: *stimulus creation*, how images depicting a candidate camouflage pattern within a surrounding environment are created; *evaluation*, how a candidate pattern's performance is quantified; and *optimization*, strategies used to iteratively find higher performing patterns. Optimization strategies, a staple of computer vision approaches, are not covered in depth here, as an overview is not necessary to understand the novelty and complementary positioning of the chosen approach.

Stimulus creation ranges from physical prototypes to various forms of digital simulation. Evaluation methods range from human psychophysical experiments to computational metrics. Understanding these two categories, their respective trade-offs in realism, cost, and scalability is necessary to appreciate why different research contexts call for different combinations, and to situate the approach taken in this thesis among them.

1.1.1 Stimulus Creation Methods

Physical Prototypes

Physical prototypes (Figure 1) represent the gold standard in camouflage pattern validation (van der Sanden et al., 2022), as no method can better simulate the interaction between a camouflaged target and a real environment. The method therefore yields the most accurate understanding of a camouflage pattern's performance. Naturally, this is the most expensive type of stimulus, as it requires a (scale) vehicle to be painted with a chosen camouflage pattern and for it to be photographed or observed in a real landscape (field trial). In many cases, this may be entirely unpractical or even impossible. Consider for instance the development of camouflage for regions of active conflict.

Figure 1. Photograph of STANDCAM, (STANdard Decoy for CAmouflage Materials) a standardized, non-classified, armored vehicle-shaped target developed by Germany's Wehrtechnische Dienststelle 52 to measure and test the effectiveness of camouflage materials. Photograph by van Beem (2012).

2D Composition

The most simple and accessible alternative is 2D composition: the process of photographing or otherwise creating a background scene, then overlaying a flat, two-dimensional image of a camouflaged target onto that scene (Figure 2). While this approach is cheap and fast, it lacks geometric wrapping of the pattern around the target's form, usually contains no physically accurate shading, and no mutual influence between target and scene lighting, all of which are factors known to influence detectability: directional lighting interacts with target geometry to produce shape-from-shading cues that aid detection (Penacchio et al., 2014), and camouflage effectiveness is tied to illumination conditions (Penacchio et al., 2018). Consequently, 2D composition trades physical realism for accessibility, making it suitable for exploratory research but limited in its ability to capture the full complexity of real-world detection conditions.

Figure 2. Example targets with camouflage patterns used in 2D composition stimulus (environment not shown). Adapted from "Evolving Camouflage" by Van der Burg et al., 2023, SPIE Security & Defence.

Hybrid Simulations

A hybrid simulation uses a photograph of a real-world scene in combination with a digital 3D model of the target (see Figure 3). The target is placed in the scene and blended in by applying effects like blur and noise to match the sensor and lens the photograph was taken with. Measurements taken during the photographing of the scene allow for highly accurate color representations to be applied to the digital model to further fit into the scene. All-in-all, this can produce high degrees of realism and thus accurate evaluations of camouflage effectiveness (van der Sanden et al., 2022). This can be done at a much lower cost than a physical prototype, but it shares the downside of needing access to areas to take these measurements and photographs while also being time consuming.

Figure 3. Top: three painted spheres in the same colors as the camouflage pattern, used for lighting information in the hybrid (material capture) method. Bottom: A hybrid scene generated using the material capture method. The middle vehicle is the virtual object. Reproduced from van der Sanden et al. (2023).

Fully Synthetic Simulations

Fully synthetic simulations (figure 4) model both the target and environment digitally, usually by raytracing. These simulations have historically been restricted by computing power as well as the time investment and expertise needed to create a high-fidelity synthetic scene. Additionally, a non-representative virtual environment will inevitably lead to patterns optimized for the virtual scene rather than real-world conditions, carrying little to no relevance to real-world performance.

However, in recent years ray tracing has become a much more affordable rendering technique from a computing perspective, allowing for large volumes of high-fidelity stimuli to be generated overnight. The major advantages of this approach are its flexibility and granular control to simulate any time of day or year, including weather conditions on any scene. Due to its scalability, fully synthetic simulations offer a great opportunity for broadly investigating potential camouflage pattern candidates.

Figure 4. A synthetic scene with target on the horizon created in Blender.

1.1.2 Evaluation Methods

How a stimulus is created and how performance is subsequently measured are independent choices; the same rendered scene can be evaluated by a human observer or a computer vision model, and the strengths and weaknesses of each evaluation approach apply regardless of how the stimulus was produced.

Computer Optimization

The most recent development in camouflage research is the advent of effective automated generation and optimization models. They often employ generative adversarial networks (GANs) to iteratively generate and identify camouflage patterns when given a collection of backgrounds (2D composition stimulus). GAN models such as CamoGAN (Talas et al., 2019) and VSAI (Gulrez et al., 2024) have demonstrated their effectiveness in creating patterns which can hide an object from detection. However, due to their self-evaluation, such methods tend to develop camouflage patterns optimized against a specific computer vision model (Schwegmann, 2023) rather than against human perception. While some studies have validated results using human observers as a post-hoc check (Talas et al., 2020; Nguyen et al., 2025), humans serve as a validator rather than as a direct driver of the optimization process.

A more recent diffusion-based model, CamoX (Nguyen et al., 2025), outperforms GANs in both computer and human vision trials while requiring significantly less training time. Notably, this study relied on 2D composition when placing their generated patterns onto a target in a scene (see section 1.1.1). An example of this application process can be seen in Figure 5. This dependency on 2D composition means that geometric and lighting

interactions between target and environment are absent from the optimization process; whether performance holds when a pattern is applied to a large vehicle, where surface curvature, self-shadowing, and viewing-angle variation substantially alter its appearance, remains untested.

Figure 5. Visualization of detection performance by a computer vision model with camouflage applied to a person's silhouette. The images show the comparison between CamoGAN (left) and CamoX (right), which the computer vision model completely fails to recognize. Reproduced from Nguyen et al. (2025).

Human Detection Metrics

Psychophysical methods provide a direct measure of how difficult it is for humans to spot a target. Human detection metrics are not strictly limited to the two methods below, but these are the ones that are important to understand the background and positioning of this study.

Search Task

In a search task, observers are shown a scene and asked to locate a target as quickly as possible; detection time (or detection rate within a fixed time window) serves as the dependent variable. Because performance is measured on a continuous scale and random factors (e.g. search pattern) add noise to the data, search tasks are best suited to high-volume stimulus sets where noise can be averaged out.

Conspicuity

Conspicuity is a method developed by TNO that directly measures how much a target stands out by having participants first fixate the target, then gradually shift fixation away until it can no longer be perceived in the periphery. They then move fixation back toward the target, recording the (angular) distance at which the target first becomes detectable. This measure correlates strongly with mean visual search time ($r = 0.84–0.89$; Toet et al., 2004). Because no unguided search is involved, the method yields low-noise data. Additionally, stimulus images can be partly reused, placing new camouflage patterns into existing scenes and presenting them to the same participants. A drawback is that the evaluation of each stimulus requires three repeated measurements which takes more time than a traditional search trial usually does. Additionally, the procedure is not as intuitive as free visual search, requiring more supervision and training of participants.

1.2 Direction

The background above makes the choice of methods in this study apparent. Fully synthetic simulations address the scalability bottleneck that constrains both physical prototypes and hybrid approaches: they allow large, parametrically controlled stimulus sets to be generated without field access or manual scene construction. The search task provides a direct human detection measure well suited to the volumes that synthetic generation makes possible, and crucially, it requires little to no training or supervision of participants making it practical to recruit and run the large numbers of trials the method demands.

This pairing also explains why a machine learning model is needed: human search data is inherently noisy, and individual trials carry little signal. At scale, however, that noise averages out and a statistical model can be fitted to the accumulated responses, mapping which regions of the parameter space produce hard-to-detect patterns, and which do not. This is what distinguishes the approach from a simple large-scale comparison: the output is not a ranked list of tested patterns but a model of the parameter space itself.

This positions the method as complementary to TNO's existing expertise in hybrid simulations and conspicuity measurement. Where hybrid simulation combined with conspicuity measurement yields a small number of stimuli evaluated with high scene realism and fine perceptual resolution, the fully synthetic search task approach offers rapid, broad coverage of the parameter space, the two working together as a funnel: broad parameter-space mapping first, high-fidelity validation of the best candidates second. To our knowledge, no prior study has evaluated contextually applied camouflage pattern performance across a parameter space of this scale using human detection data. The following section describes the pipeline developed to generate the stimulus volume this

method requires and the experimental and modelling framework built around it.

2 Method

To realize the direction described in the introduction, three components were developed: an automated rendering pipeline capable of producing stimuli at scale (Section 2.1), a remote, self-guided search experiment designed to collect detection data efficiently across many participants (Section 2.2), and a modelling framework able to extract a reliable signal from inherently noisy trial-level responses (Section 2.3). A validation experiment (Section 2.4) then tested whether model predictions corresponded to observed detection performance across four candidate patterns sampled from the predicted performance map.

2.1 Stimulus generation

Stimulus generation involved two interdependent components: the parametric camouflage pattern applied to the target, and the synthetic scene in which it was embedded. Both are described below, followed by the automated pipeline used to combine them into usable stimuli.

2.1.1 Seeded Synthetic Environment

Synthetic scenes were put together in Blender version 5.0 (Blender Online Community, 2025) using the GeoScatter addon version 5.6.2 (BD3D, 2024). The scenes contain 13 unique assets from Botaniq 7.1.1 (polygoniq, 2024). Some models (including grass and trees) had their hue and saturation altered to increase chromatic variation. All assets were distributed in the scene using GeoScatter, usually in clusters, and had varying transformations in size and orientation applied. A minimum distance and falloff was put in place to assure trees could not appear close to the virtual camera, possibly obstructing the view. The geometry of the ground is consistent throughout all samples with only slight displacement applied to the surface and a hill or ridge at the end to obstruct the artificial horizon. A master seed controlled all scatter-systems. The per-system random states govern asset placement, orientation, and clustering, enabling unlimited scene variations while preserving the same distributional constraints.

A note on terminology:

This thesis distinguishes between two related terms. An *environment* refers to the procedural ruleset governing a scene, specifying which assets are used and how they are scattered. A *scene* is a specific expression of that environment, produced by a particular seed, resulting in a unique spatial arrangement of assets. Critically, within each scene the target location is fixed and remains the same across all trials in which that scene appears. For this study a single environment was used.

Environment anatomy

The foreground and midground consist of an open grassland with continuous patches of grass in a range of dryness and height (see Figure 6). The surface contains small patches of exposed brown soil. Rocks and boulders are scattered in small clusters throughout the field as well as a sparse number of juvenile trees and logs. These form visual distractors which can cast and display internal shading. Additionally, they may partly occlude the target which may interfere with target detection.

The background is defined by a dense row of mature trees forming an irregular but usually continuous tree line. Here targets can be partly occluded or harder to detect due to increased visual clutter from overlapping canopy and ground elements. Additionally, greater cast shadow in this region may further hinder detection.

Figure 6. Example of scene used in the experiment. While each scene was unique all had the same general anatomy.

Environment design

Environment creation and the choice of camera perspective were the result of many iterations and pilot experiments, outlined in Appendix A. Asset selection was motivated by a goal to represent a Dutch operational environment, with De Veluwe serving as the primary reference. However, the available asset pack (Botaniq) constrained what could be faithfully recreated: species that define much of De Veluwe, such as Pinus sylvestris (Scots Pine), were absent from the library. Asset availability thus partly shaped the reference environment rather than the reverse, and the final scene should be understood as a rough approximation.

The tree line assets were comprised of four unique Pinus Ponderosa (Ponderosa Pine) models. The small trees in the

field are Picea Abies (Norwegian Spruce). Grass assets were Muhlenbergia Rigens (deer grass).

A clearing similar in feel to De Veluwe was chosen. Tall grass was included in the shades observed in reference material: yellows, browns, oranges, and greens. Colors were matched by shifting the native hue and luminance of the Botaniq assets in Blender to visually approximate reference photographs.

While the final environment does not faithfully represent De Veluwe, using plants not native to it, it serves as an adequate test-bed for this study's purpose: not to identify the optimal camouflage pattern for an existing environment, but to establish whether the method is valid and has merit over traditional approaches, before optimizing for a chosen setting.

This should not be interpreted as a strict limitation to the potential of this approach, since relevant assets do exist (and could otherwise be created). It was simply a restriction in budget and time for this particular study.

Fidelity

The validity of any camouflage evaluation conducted in a synthetic environment depends on how faithfully that environment reproduces the visual conditions of the real-world scenario being simulated. Because the parameter space mapping approach optimizes camouflage against the scenes used in the experiment, patterns that perform well in low-fidelity or visually unrepresentative scenes may not transfer to real operational conditions. The environment is therefore not merely a backdrop but an active determinant of what the optimization converges on.

Culpepper et al. (2017) found moderate to strong correlations between high-fidelity synthetic scenes and real photographs for both detection probability and search time, but low-fidelity scenes lost the correlation for detection probability while retaining it for search time, showing that fidelity of synthetic environments is important in studies such as this one, but even low-fidelity scenes have some correlation with real world outcome. We believe our synthetic stimulus is closer to the high-fidelity images used by Culpepper et al., primarily due to the amount of visual clutter and fidelity of the 3D assets. While the same 50mm focal length was used, the range and camera angle differ: our camera was positioned closer to the ground and at a shorter range than those used by Culpepper et al.

A deliberate difference in simulation in our study concerned lighting. Pilot testing revealed that strongly directional light produced hard shadows that had a strong effect on detection times in two opposing directions: in open terrain, a sharp shadow cast by the vehicle acted as an immediate giveaway, leading to much lower detection times overall, while targets falling within a cast shadow became effectively undetectable on non-HDR displays, which lack the dynamic range to preserve detail in dark image regions. Diffuse lighting (sun size 45°) was therefore used throughout to avoid this floor and ceiling effect, ensuring that camouflage pattern properties had a chance to influence search results. This has the side-effect of optimization being done for diffuse light situations (e.g., overcast weather), which may have had an effect on optimal pattern outcome (Penacchio et al., 2018).

Render settings

Each stimulus image was rendered at a resolution of 1920×1080 with a modest 125 samples, without denoising, using 8-bit color depth. The output was saved as a png since this is a lossless format. The choice of file format and the absence of denoising was made to prevent any compression or simplification of the camouflage pattern.

Settings were optimized for short render times: each unique scene took approximately 30 seconds to render (a rate of ~2900 scenes per day). For repeated scenes, only the target and its direct surroundings were re-rendered using the render region feature in Blender, further reducing render times. These renders took a little under seven seconds on average (a rate of ~12500 targets per day).

Sample to Stimulus Pipeline

To efficiently generate the large number of unique images required for this study, an automated sample-to-stimulus workflow was developed. Blender's extensive Python API made it possible to control the necessary variables of the rendering program programmatically. By supplying the script with CSV files containing camouflage parameter combinations and selected scene seeds, the system could automatically produce complete stimulus renders without any manual intervention. This automated pipeline also enabled rapid rendering of the second generation, ensuring that the iterative nature of the experiment could proceed with minimal delay.

Sobol Sampling

The parameter space was explored using (quasi-random) Sobol sampling, chosen for its efficiency in uniformly

covering high-dimensional spaces. Unlike purely random or pseudo-random sampling, a Sobol sequence is deterministic and sequential: each newly added point is generated specifically to fill the largest remaining gaps in the space (Sobol, 1967; Kucherenko et al., 2015). This property allows for expanding the sequence for subsequent generations without ever repeating a previously evaluated sample, while still maintaining excellent space-filling characteristics and avoiding both clusters and voids.

Iterative approach

An iterative two-generation approach was taken for sampling and evaluation. Generation 1 sampled the full parameter space broadly, and the resulting human detection data were used to identify high-performing regions. Generation 2 then concentrated additional samples within those regions, allowing the model to refine its predictions where they mattered most. The use of Sobol sampling was critical to making this expansion seamless: because Sobol sequences are deterministic and gap-filling by construction (see Sobol Sampling above), new points added in Generation 2 slotted into the existing coverage without duplicating any previously evaluated combination, creating clusters, or leaving voids, properties that stratified or pseudo-random sampling would not guarantee when expanding an existing dataset.

Each generation consisted of 9000 synthetic stimuli, created by combining quasi-randomly sampled parameter combinations for the camouflage pattern with 600 unique scenes generated from the same single environment. Each participant evaluated one or two batches of 300 images per generation, never evaluating the same scene twice. This resulted in 15 evaluations per scene, allowing the inherent difficulty of the scene/target location to be approximated while assuring participants had roughly equally difficult samples to evaluate.

Selection

Generation 1 sampled the entire parameter space broadly. The sampling region for generation 2 was narrowed to areas of high performance. These areas were selected based on binned search time metrics. Out of 54 bins the top 27 (50%) highest median bins qualified for resampling. The resulting dataset was used to map the parameter space, showing predicted performance for any parameter combination. Lastly, four camouflage patterns were chosen based on the model's prediction and evaluated in a validation experiment where each pattern was shown to each participant 150 times to obtain stable performance estimates. This allowed for comparisons between model predictions and empirically observed performance of specific parameter combinations.

Stimulus filtering

Vehicle Placement

Basic placement constraints were applied directly to prevent intersections with large objects such as rocks or boulders.

However, not all constraints could easily be controlled using GeoScatter. Since the vehicle's position is ultimately determined from the master seed, every scene comes with a predetermined target location. Therefore, some post-render filtering was necessary to determine on a per scene basis if inclusion in the study was possible.

Occlusion Masks and Filtering

Occlusion masks were generated using special render passes, enabling only relevant assets to quantify the proportion of the vehicle hidden by trees and other objects (see Figure 7). Only geometry using the camouflage texture was included in the mask computation, and grass was excluded to enable significantly faster rendering of the occlusion masks while retaining acceptable accuracy. High occlusion rates in pilot experiments showed a strong impact, increasing detection times, lowering detection probability and generally making underlying camouflage parameter performance harder to extract. The value of 25% was chosen because it still shows a good majority of the vehicle. This is important because we are optimizing camouflage parameters. If only a fraction of the vehicle is visible detection metrics will not say much about camouflage parameters.

Scenes were retained only when:

- <25% of the vehicle area was occluded (see Figure 7).
- The vertical location of the target centroid exceeded 780 pixels (measured from the top of the 1920x1080 image), somewhat counteractively, ensuring a minimum target distance (see Figure 8).
- No vehicle pixels touched the image border, guaranteeing a fully in-frame target.

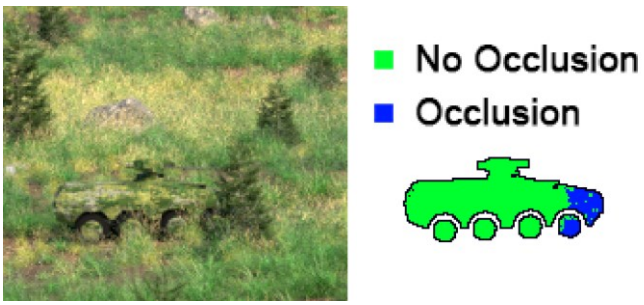


Figure 7. Target in a scene (left) with the corresponding occlusion masks (right), from which an occlusion rate of 11.5% was computed.

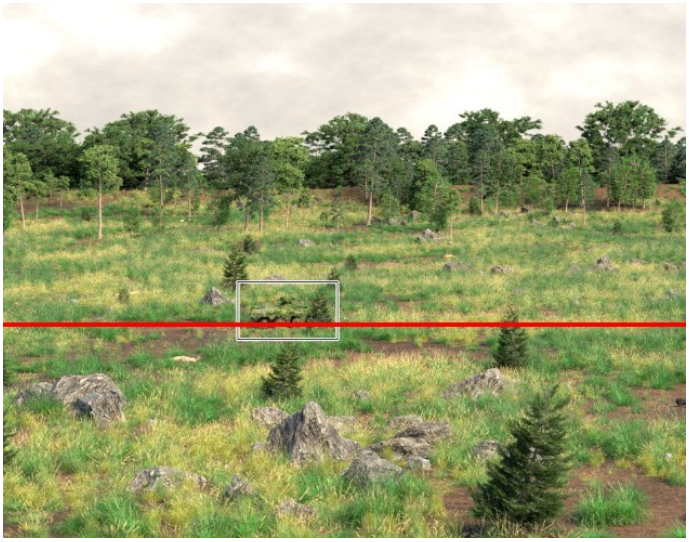


Figure 8. Cropped stimulus image showing the minimum-distance threshold (red line). The threshold indicates the minimum vertical position the average target pixel must exceed to qualify for inclusion in the experiment. In this example, the wheels fall slightly below the threshold, but target centroid lies above it; therefore, this image met all requirements.

2.1.2 Parametric Camouflage Pattern

Color Selection

A set of 8 representative colors was hand-picked for the type of environment that was used. Color selection was centered around terrain features including grass, both dry and fresh, trees as well as rocks and dirt. The exact colors and sampling areas can be found in Figure 9.

Figure 9. Colors used on target and their general sampling areas. From left to right: #141b10, #717b49, #9ca05e, #584f3a, #223c12, #517848, #476b2c, #909588.

This approach was chosen over KNN color clustering because relevant colors such as that of the rocks would not be included, as they make up only a small percentage of pixels, yet they are important visual distractions present in the field.

Pattern Creation

To create different camouflage patterns, a node-based material was used in Blender version 5.0. A Voronoi texture was applied to the vehicle surface, dividing it into discrete cells that each received a color from the predefined palette. Cells were then grouped into coherent patches by using layered noise masks: each noise mask is a procedurally generated grayscale texture whose output thresholds determine which cells belong to the same patch at a given scale. Multiple noise masks were combined in a hierarchical mix-node structure, at each scale level one mask acts as a binary factor that routes cells into one of two sub-patterns, and this procedure is repeated at finer scales so that the final pattern emerges from nested sub-pattern assignments (see Figure 10). The scale, frequency, and distortion of each noise mask, as well as the thresholds governing which sub-pattern a cell inherits, are exposed as node inputs. These inputs were wired to four parameter controls, each spanning a continuous range from 0.0 to 1.0. Together they form the parameter space and will be referred to as P1–P4.

The four parameters control the pattern globally, but they do so by modulating a large set of underlying seed values rather than directly dictating the appearance of individual patches. Each layer in the nested noise-mask structure has its own collection of seed values, abstract numerical constants that set the base state of that layer's texture before the parameters act on it. P1–P4 influence the pattern by scaling, shifting, or otherwise transforming these seeds: a parameter value is effectively multiplied through or added to the seeds at each layer, nudging the frequency,

phase, and distortion of each noise mask by an amount proportional to that value. Because the seeds are numerous and low-level, no single seed corresponds to a recognizable property like patch size or border roughness, those emergent qualities arise from many seeds across all layers being pushed in the same direction by a parameter. The practical result is that patches retain individual character (one patch may be slightly larger or grainier than its neighbor) even at identical parameter values, because each patch is governed by a different subset of these underlying seeds.

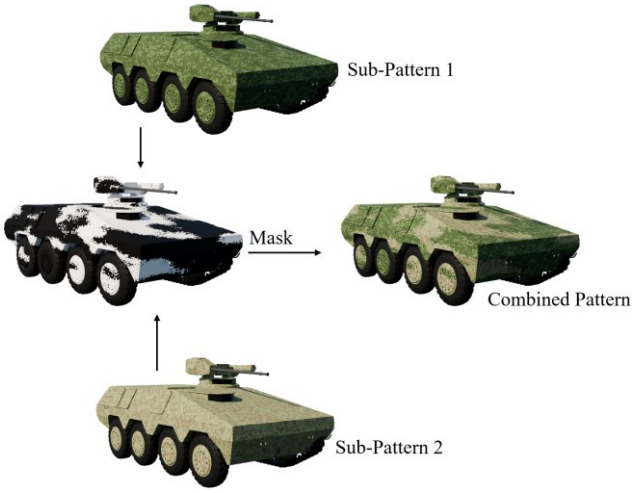


Figure 10. Construction of the camouflage pattern in Blender. Two sub-patterns are combined using a noise-texture which is used as the 'factor' (read: mask) in a mix node. Effectively, all of sub-pattern 1 is assigned to the black areas and all of sub-pattern 2 to the white areas of the mask. This procedure is applied consistently across all scales, with unique noise-masks at each level.

P1 – Patch Size

This parameter influences the size of clusters present in the pattern. A low value leads to many adjacent Voronoi cells having the same sub-pattern applied (figure 11), whereas a high value leads to an un-structured or scattered look with each cell having a seemingly random color assigned.

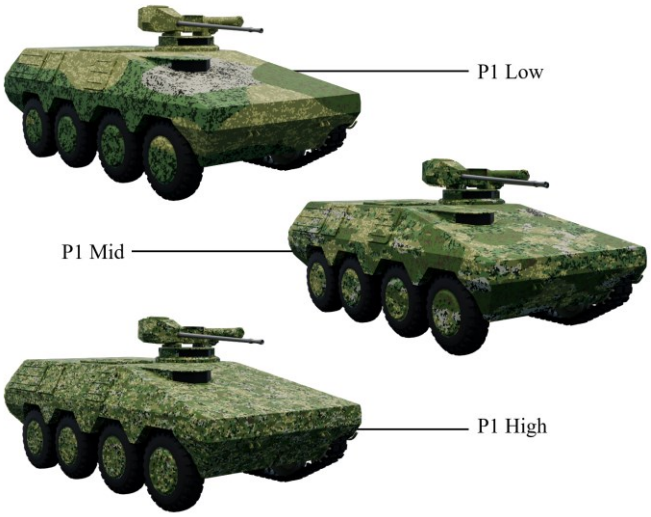


Figure 11. Different expressions of P1.

P2 – Horizontal Stretch

This parameter stretches patches out over the horizontal plane, creating wider shapes. Low values leave the pattern (mostly) unaffected, and high values increase the degree of stretching (Figure 12).

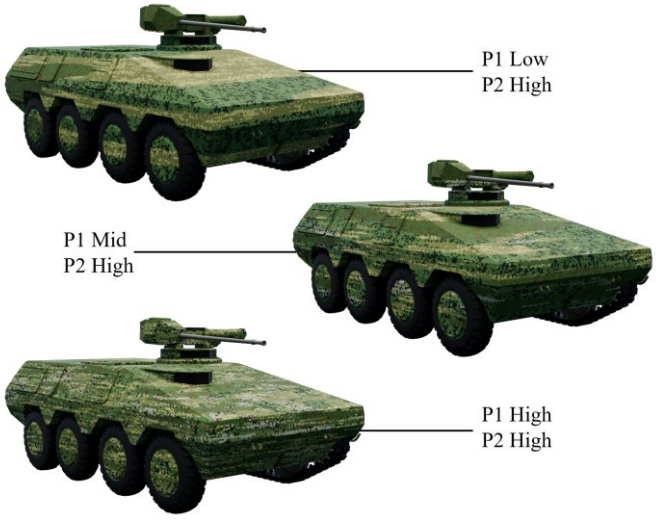


Figure 12. Different expressions of P2.

P3 – Grain

The third parameter has an effect on borders between patches. At the low-end borders are easy to identify with clearly smooth but defined borders (Figure 13). Toward the middle of this parameter range the borders become less smooth and more jagged until the higher end causes borders to become discontinuous and cause patches to scatter in pieces.

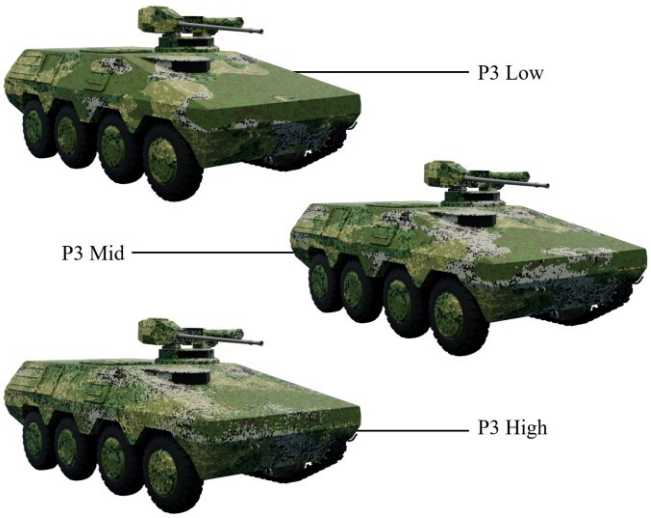


Figure 13. Different expressions of P3.

P4 – Feature Mask

This is an experimental parameter, using the blender geometry nodes system to extract information about the proximity of the camouflage pattern to uncamouflaged parts of the vehicle. This should help hide clearly defined lines which can make the vehicle recognizable even when only a small part of it is visible. Applying the feature mask near the tires was also considered but was decided against for keeping results interpretable. The effects from this parameter are independent and cannot be influenced by the other three. Under a value of 0.5 the parameter had a negligible visual effect, while higher values progressively grew the size of the noisy stain near the selected objects (Figure 14).

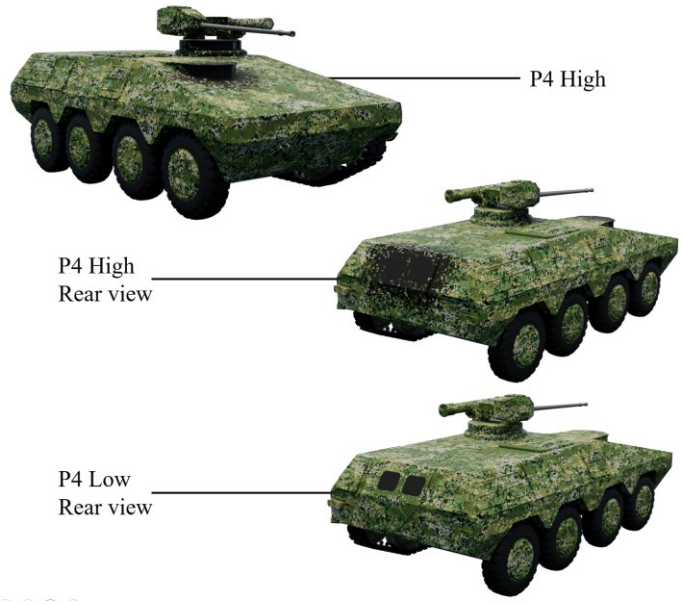


Figure 14. Different expressions of P4.

2.2 Human search trials

Human search trials were conducted in three phases: two iteratively sampled main generations and a subsequent validation experiment. All three shared the same general experimental setup, described in the following sections, differing only in the stimulus sets presented to participants.

Participants

Participants were 32 adults (62.5% male; median age: 23, range: 20-58). None of the participants had a military background. All participants had normal or corrected vision. None of them reported color deficiencies.

Ethical procedure and approval

All participants volunteered for the study, were provided with the participant information form (appendix E) and signed the informed consent form (appendix F). The experimental protocol was reviewed and approved by the TNO Internal Review Board (TNO, The Netherlands: reference 2026-881), and it was in agreement with the Helsinki Declaration of 1975, as revised in 2013 (World Medical Association, 2013)

Setting

For the first eight participants, the experiment was conducted in a lab setting with supervision present to check for any issues in the evaluator program or instructions. These participants had no experience with previously conducted pilot studies.

The majority of the experiment was conducted remotely. Participants were asked to follow instructions on how to move a directory containing their assigned stimuli into the right place for the evaluator program to be able to read them. This evaluator at its core is a Python 3.10 / PsychoPy (2025.2.4; Peirce et al., 2019) script which was converted using PyInstaller (6.18.0; PyInstaller Development Team, 2024) to pack all necessary dependencies into one easy to use windows executable.

Apparatus

Participants were seated (lab setting) or instructed to sit (remote) in a dimly lit room at a comfortable distance from their monitor. The majority of datapoints (64.2%) were collected using 1920x1080 monitors. Refresh rates were similarly distributed with 65.9% of datapoints coming from 60 Hz panels. The exact breakdown of configurations can be found in table 1 below.

Resolution	Refresh rate	Portion of datapoints
1920x1080	60 Hz	62.5%
2560x1440	144 Hz	17.0%
1920x1200	144 Hz	6.8%
2560x1440	180 Hz	3.4%
2560x1600	240 Hz	3.4%
1920x1080	240 Hz	1.7%
2736x1824	60 Hz	1.7%
3440x1440	60 Hz	1.7%
5120x1440	120 Hz	1.7%

Table 1. Distribution of monitors used

For the lab setting, 24-inch screens were used. From the remotely recorded data, it is not possible to tell what physical size monitors were used, only resolution. It is likely that some participants used monitors of similar sizes, while some used laptops, which generally have smaller screens. Important to note is that all monitors used had a resolution equal to or higher than the source images. All stimuli were rendered at a resolution of 1920x1080 and scaled to the monitor's resolution while retaining the rendered native 16:9 aspect ratio to prevent stretching. The range of monitor configurations in the remote setting, including variation in physical screen size and display quality, could not be controlled for and may have introduced some variability in stimulus appearance across participants.

Stimulus allocation

Each generation consisted of 9,000 stimuli, combining 600 unique scenes with 15 different camouflage patterns each. Scenes were divided into two batch types of 300, with each batch containing one render per scene. To accommodate varying participant availability while maximizing data yield, participants evaluated one or two batches per generation, drawn from either or both batch types. This ensured no participant encountered the same scene twice. Across generations, participants completed between one and four batches in total (see Table 2). Since each scene was evaluated by 15 participants, this ensured no single participant contributed more than one fifteenth (6.7%) of evaluations (see Table 2), consistent with the minimum observer count recommended for camouflage detection experiments in NATO guidelines (Peak et al., 2006).

Batches evaluated	Number of participants	Portion of total datapoints	Contribution per participant
1	18	30.0%	1.7%
2	6	18.4%	3.3%
3	1	5.0%	5%
4	7	46.7%	6.7%

Table 2. Batch distribution among participants.

Search Task

After reading the participant information document and signing the informed consent form participants opened the program and were shown an information screen with controls and a visual impression of the type of scenes they would be looking at and the vehicle (STANDCAM) they were looking for, shown in Figure 15.

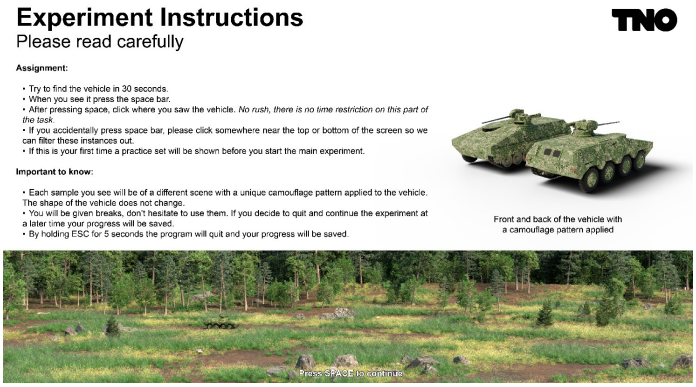


Figure 15. Information screen

After continuing, each participant evaluated a training set of 25 images to become familiar with the task and shape of the target. No data was recorded during the training sets.

Upon completion of the training set, they were notified they would be starting the main experiment. This screen also showed how many images they were about to evaluate over how many batches.

The evaluation of every image followed the same order, seen in Figure 16. From the moment the image appeared on screen, they had a time limit of 30 seconds to spot the vehicle and press the spacebar. The task was to detect the target as quickly as possible. If the time limit was exceeded, a screen would be displayed informing them the time limit had been reached and to prepare for the next image which was automatically displayed three seconds later.

If they did press spacebar in time, a backwards mask was displayed for 100ms, an effective way of limiting post stimulus visual processing (Bacon-Macé et al., 2005), after which a grid was displayed, with the task to click where the vehicle was seen. The grid served solely as a visual guide. There was no time restriction on when to click on

the target location. During this task, a text was displayed at the top of the screen for participants to see how many samples they had evaluated.

A brief loading screen was shown after the click task to make the transition from a black screen to the bright stimulus less immediate, which was found to be more pleasant in pilot testing. It is worth noting that a message saying "loading next image" was present, but no formal fixation point (e.g., a cross) was provided on this screen.

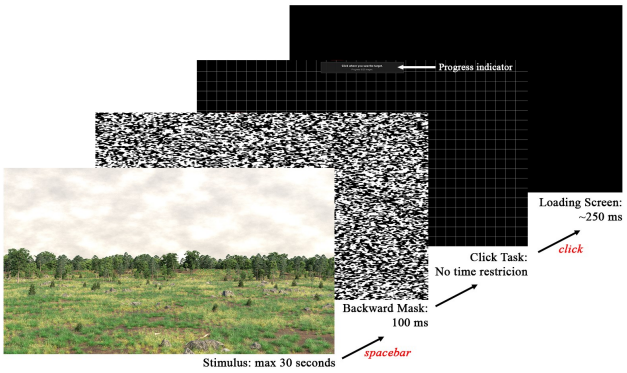


Figure 16. Stimulus evaluation sequence with participant actions in red.

After every 50 samples a screen was showing the participant's progress and suggesting a break, reminding the participant of the possibility to close the program and continue the experiment where they left off later. After ending a break, a three second countdown allowed the participants to ready themselves for the next stimulus to appear on screen. At the end of the experiment, participants were informed of their completion and given instructions on where to send their results.

Higher resolution copies of Figure 15 and 16 can be found in appendix B.

Data integrity

False positives

False positives were automatically detected by measuring the distance between the target's centroid and the click coordinate. A threshold of 150 pixels was applied to distinguish false positives from valid clicks based on pilot testing. This threshold was deliberately lenient to avoid discarding outlying but valid clicks. Click distances in pilot testing plateaued at approximately 97% with a distance of 100 pixels, leaving a 50-pixel margin for outliers.

Target separation

Stimulus order was randomized within each batch to prevent any unknown order effects from Sobol sampling or GeoScatter's seeding to influence results. Additionally, a system was put in place to prevent sequential targets from appearing in nearly the exact location. A radius of 350 pixels was enforced within which stimuli with target centroids in that range would not be allowed. This underlying mechanism was not communicated with participants.

2.3 Model training

The goal of the study was to build a prediction model mapping the camouflage design parameter space to detection performance, identifying regions associated with high and low detectability. Raw detection times, however, followed a heavily right-skewed distribution with a hard ceiling at 30 seconds imposed by the trial time limit, making them unsuitable as a direct model outcome. This skew will be further discussed in results. To reduce sensitivity to the ceiling and to the high noise inherent in single-evaluation trials, arising from factors such as target distance, background elements, shadows, and individual scan paths, detection performance was converted into a binary outcome suitable for model fitting. Trials in which the target was detected within 5 seconds were classified as short detections; trials exceeding 5 seconds and non-detections were grouped into a second category. False detections were excluded prior to classification. The 5-second threshold was selected based on the shape of the detection time distribution, which showed a natural concentration of fast detections in the sub 5-second range; full selection details are reported in Appendix D. The resulting binary outcome variable is referred to throughout as P(>5s).

Both models were fitted to P(>5s) using Python 3.13 with scikit-learn 1.8.0 (Pedregosa et al., 2011) and GPyTorch 1.15.2 (Gardner et al., 2018). All nine predictors (P1–P4 and five covariates) were standardized to zero mean and unit variance prior to fitting.

The logistic regression model takes the form:

$$P(y = 1 \mid \mathbf{x}) = \sigma(\beta_0 + \boldsymbol{\beta}^T \mathbf{x})$$

where $\sigma(\cdot)$ is the logistic sigmoid, $\mathbf{x}$ is the standardized feature vector, β_0 is the intercept, and $\boldsymbol{\beta}$ is the coefficient vector. The model was fitted with an L2 penalty ($C = 0.01$) using the lbfgs solver (max. 1000 iterations). Standardized coefficients, standard errors, 95% confidence intervals, and p-values were obtained from a separate unregularized fit using statsmodels (Seabold & Perktold, 2010), as

regularized logistic regression does not produce valid significance estimates.

The GP classifier was implemented as a Sparse Variational GP (SVGP) with a Bernoulli likelihood. The covariance function was a scaled Matérn-5/2 kernel with Automatic Relevance Determination:

$$k(\mathbf{x}, \mathbf{x}') = \sigma^2 \cdot k\_\text{Matérn}(\nu=2.5)(\mathbf{x}, \mathbf{x}' \mid \mathbf{l})$$

where σ^2 is the learned output scale and $\mathbf{l} = (l_1, \ldots, l_9)$ is a vector of per-feature length-scales, each fitted independently. A Gamma(3.0, 6.0) prior was placed on the length-scales. The model used 500 inducing points, initialized by random selection from the training data and updated during optimization. Training minimized the negative variational ELBO using Adam (learning rate = 0.02, 500 epochs) on an NVIDIA RTX 3080 GPU. ARD feature importance was derived as $1/l_j$ for each feature j, normalized to sum to one; shorter length-scales indicate greater sensitivity of the kernel to variation in that feature, and thus greater predictive relevance.

Both models were evaluated using 5-fold stratified cross-validation, with stratification applied to preserve the class balance of P(>5s) across folds. The GP was retrained from scratch on each training fold. Performance was assessed by mean ROC-AUC and its standard deviation across folds. The final models reported throughout were refitted on the full training dataset after cross-validation. Hyperparameter selection details are reported in Appendix D.

Together the two models offer complementary perspectives on both the direction and structure of parameter effects. Both models included covariates alongside the four design parameters, allowing scene and observer effects to be partitioned from pattern-specific effects.

2.4 Model validation experiment

The validation experiment followed the same procedure as described in section 2.2, with the exception that only four fixed camouflage patterns were used rather than the broad parameter space sample of the main experiment.

To select validation candidates, the GP model generated predicted P(>5s) values across a uniform grid spanning the full four-dimensional parameter space (parameter space mapping), with 21 evenly spaced points per parameter dimension (step size = 0.05), yielding 21^4 = 194,481 combinations from which candidate patterns at specific percentile ranks could be identified.

Four candidate parameter combinations were selected from the GP-predicted performance distribution to serve as stimuli in the validation experiment: the 0th, 25th, 50th, and 100th percentiles of predicted detection difficulty, where lower percentile rank corresponds to a higher predicted probability of a detection time exceeding 5 seconds. This selection strategy allows the validation to assess model accuracy at levels of predicted detection difficulty simultaneously. At the extremes, the 0th and 100th percentile candidates test whether the model correctly orders the easiest and hardest patterns to detect, a basic but necessary check of ordinal validity. The intermediate candidates at the 25th, and 50th percentiles extend this by asking whether the model's rank ordering holds. A 75^{th} percentile candidate was omitted because optimization is focused on areas of high camouflage performance. Together, the four candidates make it possible to evaluate both ordinal accuracy and the resolution of predicted performance differences, while keeping the number of patterns to a practical minimum for a head-to-head experiment.

Each of the four patterns was paired with the same 150 scenes, yielding 600 trials per participant. Unlike the main experiment where unique scene allocation ensured no participant encountered any scene more than once, each scene appeared up to four times per participant here, once per camouflage pattern. Selected scenes for the validation experiment did not appear in the main experiment.

Stimulus order was fully randomized, with the sole constraint that the same scene could not appear in directly consecutive trials. To assess memorization effects, detection performance on first-exposure trials was compared against the full four-exposure dataset. 12 participants completed the validation experiment.

3 Results

Results are organized across seven sections. Section 3.1 describes the narrowing of the parameter space, detailing how first-generation outcomes were used to refine the sampling region for the second generation. Section 3.2 reports descriptive statistics on search time and trial outcomes. Sections 3.3 and 3.4 cover model fit and feature importance, establishing the predictive validity of the classifiers and the relative influence of each predictor.

Section 3.5 examines covariates: scene geometry and observer characteristics that independently shape detection difficulty and must be accounted for before parameter effects can be meaningfully interpreted. Section 3.6 reports the marginal effects of each camouflage parameter. Finally, Section 3.7 presents the results of the validation experiment, testing whether the model's rank ordering of candidate patterns is reflected in observed human detection performance.

3.1 Narrowing the parameter space

The first generation was reviewed after 7700 samples were returned. The samples were binned (see Figure 17) and a subsequent selection was made in the remaining parameter space using Sobol sampling from the same sequence until a new group of 9000 unique samples fit the restricted parameter space.

All parameter ranges were divided into three with the exception of P4, which was divided into two bins (0.0–0.5 and 0.5–1.0), as the parameter produced negligible visual change below 0.5. This yields 3 × 3 × 3 × 2 = 54 bins in total.

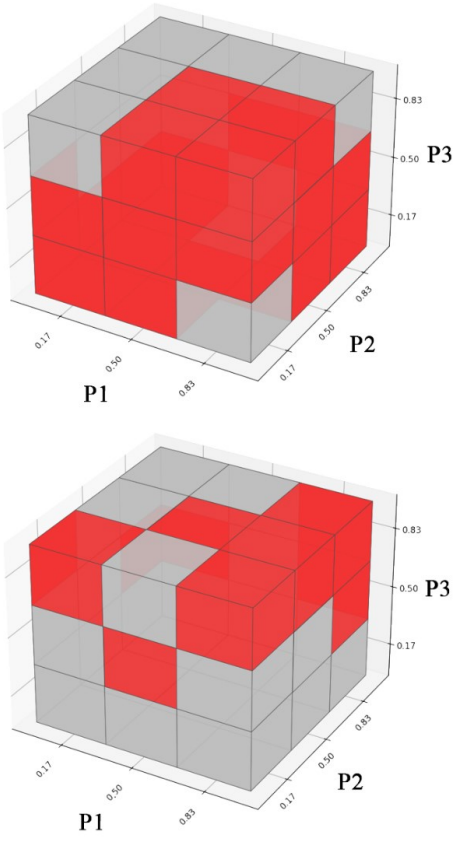


Figure 17. Binned first generation parameter space (P1 × P2 × P3) for P4 = 0.0–0.5 (top) and P4 = 0.5–1.0 (bottom). Red bins correspond to the top 50% of performance; grey bins to the bottom 50%.

A surface level comparison between the two generations can be found in appendix C.

3.2 General statistics

This section reports descriptive statistics on search time and trial outcomes (detection status), both overall and on a per participant basis. All statistics and figures from this point forward are derived from the full dataset i.e. both generations. Of the 18000 generated stimuli, 100 were not evaluated due to a participant not completing their batch, and one image could not be evaluated due to file corruption. Because of this the full dataset comprises 17899 trials.

Search time distribution

As expected in search time experiments, detection times followed a highly right-tailed skew, see Figure 18. The average detection time was 3.20 seconds and the median detection time was 1.67 seconds.

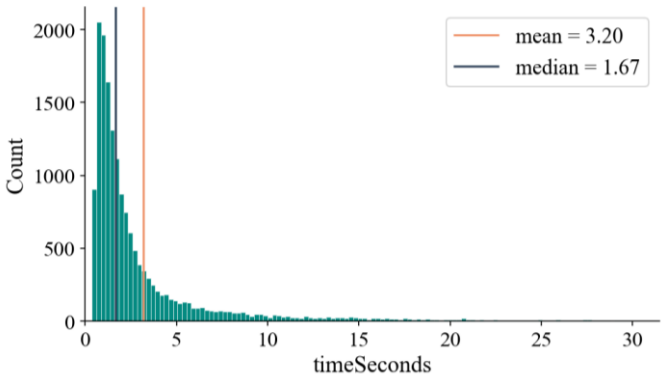


Figure 18. Distribution of detection times: search times exclusively in trials with a valid detection as outcome.

In Figure 19 the cumulative detection times are shown. 25% of detections took only 1 second or less and 75% of targets were found within 3.22 seconds.

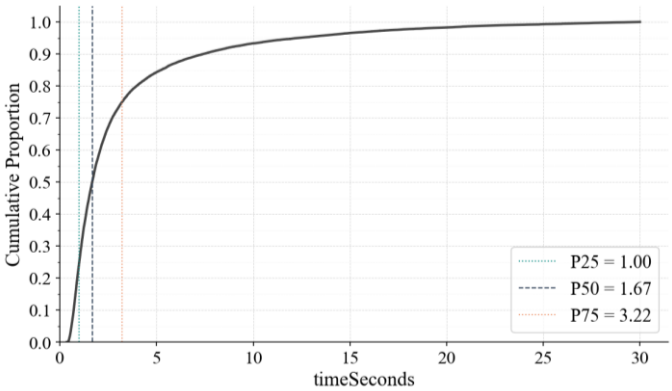


Figure 19. Empirical cumulative distribution function of detection times in trials where the target was found.

Search time varied considerably across participants (Figure 20). Participant 26 stands out with a high median and wide spread, while participants at the opposite end (e.g., 17, 18, 19) show fast, consistent detection times. All participants display right-skewed distributions, though the spread in median detection times points to meaningful individual differences in search efficiency.

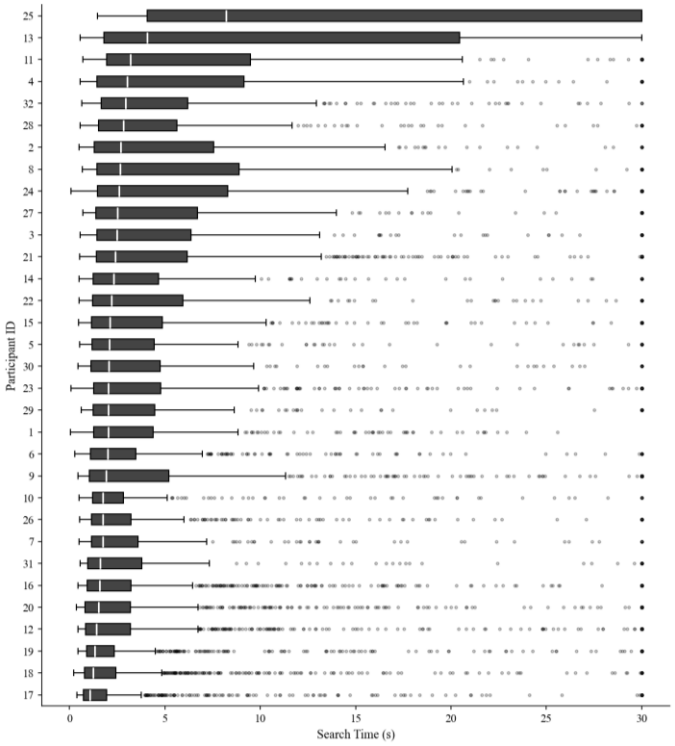


Figure 20. Search time distribution per participant, sorted by median detection time. Each box displays the interquartile range (IQR), with the central line indicating the median. Whiskers extend to 1.5× IQR; individual points beyond this range are plotted as outliers. Non-detects were treated as 30-second search times for this figure.

Search trial outcomes

In 16367 cases (91.4%) the target was found. While in 1045 cases (5.8%) the target could not be detected. In 2566 cases the target was detected after 5 seconds or more (14.3%). The proportion of P(>5s) (the main metric used in parameter space mapping, defined as detection time exceeding 5 seconds or a non-detection) was therefore 14.3% + 5.8% = 20.2% of trials (3611 cases). 487 (2.7%) trials were flagged as false detections due to the participant selecting a spot outside the 150-pixel radius surrounding the target centroid as described in section 2.2.

In Figure 21 trial outcomes are shown on a per participant basis showing a wide range of individual results. With participant 1 showing 0% non-detections but notably around 15% false detections. A profile which resembles that of a few other participants and will be discussed in section 4.2 on search strategy.

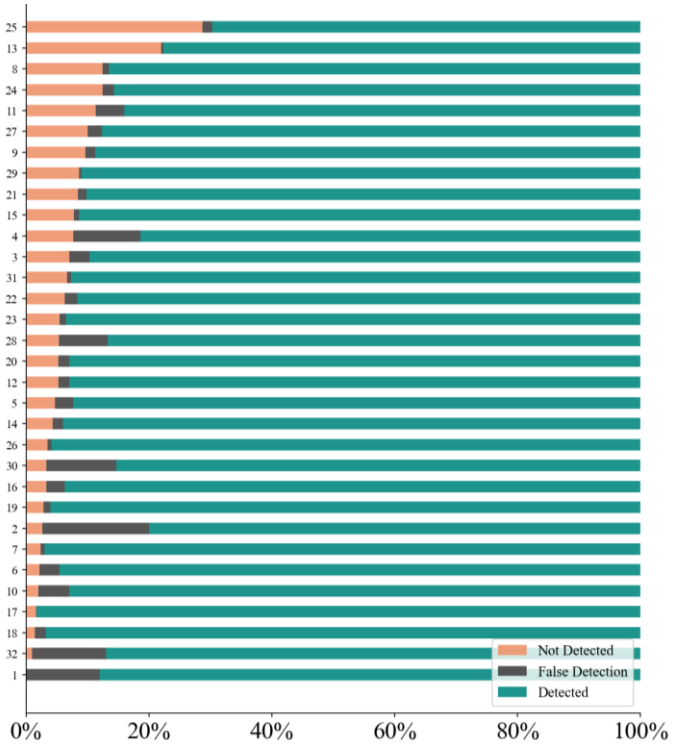


Figure 21. Search trial outcomes on a per participant basis, sorted by percentage not detected.

3.3 Model fit

Both the GP classifier and logistic regression model achieved moderate discriminative performance, with AUC scores of 0.79 and 0.77, respectively, with standard deviations of only 0.01 (Figure 22).

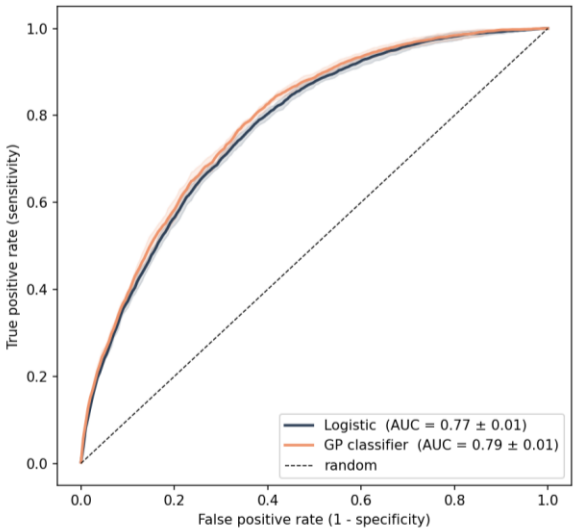


Figure 22. ROC curves for the GP classifier and logistic regression model, predicting P(>5s).

The calibration plot (Figure 23) shows that both models produce well-calibrated probability estimates.

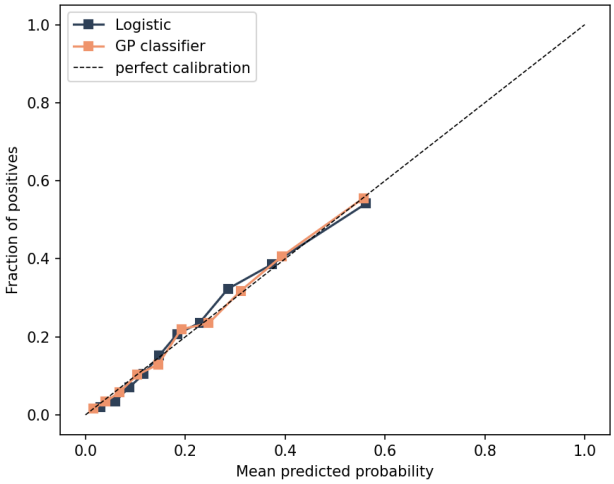


Figure 23. Calibration plot for the GP classifier and logistic regression model, predicting P(>5s).

3.4 Feature importance

Figure 24 shows the standardized logistic regression coefficients and the GP classifier ARD feature importance scores for all predictors, including covariates and design parameters.

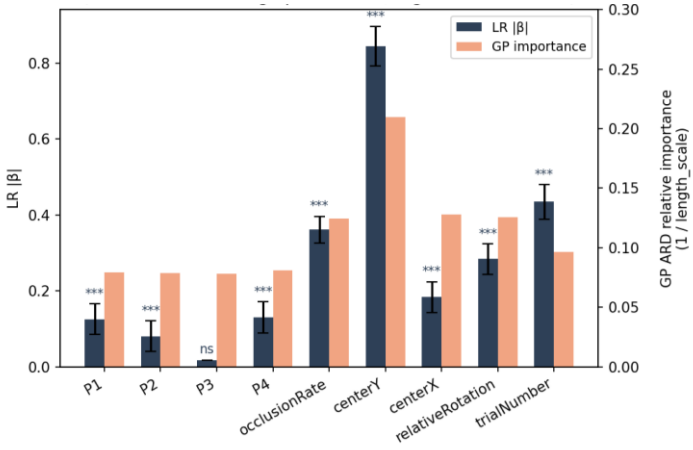


Figure 24. Logistic regression absolute standardized coefficients |β| (dark blue, left axis) and Gaussian Process ARD feature importance scores (orange, right axis) for all predictors. Note: these models use different underlying mechanisms to quantify 'importance' and should be interpreted based on their own scales.

Both models agree on the broad ordering of feature relevance: covariates account for substantially more variance in detection outcome than the camouflage pattern parameters. Out of all variables, *centerY* clearly shows the strongest effect in both models. *occlusionRate* shows a consistent positive effect in the logistic model.

Among design parameters, P1 shows the clearest signal in the logistic regression model, with a positive coefficient (not visible in plot; $p < .001$) indicating that higher values of P1 are associated with increased P(>5s), consistent with the marginal effect plots discussed in the following sections. P2 and P4 carry (as we will later see) negative coefficients (both $p < .001$), indicating that higher values are associated with significantly easier detection. Between those two P4 coefficient size is similar to that of P1, indicating an important negative effect. P3 shows a small and non-significant effect.

3.5 Covariates

The feature importance analysis above established that covariates account for substantially more variance in detection outcome than the camouflage pattern parameters. The following sections describe each covariate in detail, examining the nature and magnitude of these effects.

Target distance

The vertical pixel position of the target centroid (centerY, measured from the top of the image) was the strongest predictor of detection difficulty in both models. Larger centerY values correspond to targets closer to the camera; smaller values place the target deeper in the scene, near the tree line.

The binned P(>5s) rate (seen in Figure 25) declines with centerY, but not uniformly. The descent is steep between approximately 600 and 700 pixels and more gradual from 700 to 775.

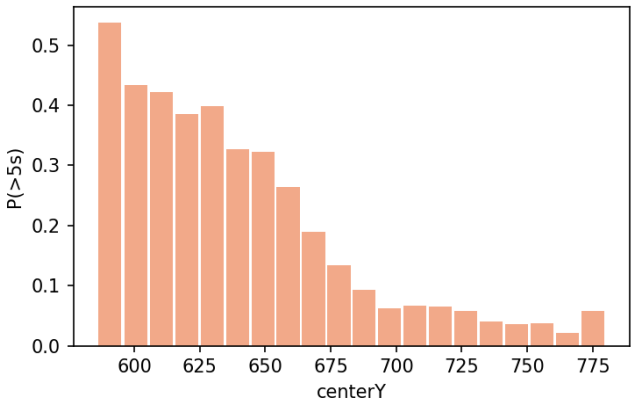


Figure 25. Binned P(>5s) of centerY.

Figure 26 shows how the relatively abstract value of centerY relates to distance visually. The increased detection difficulty threshold around centerY of 675 is clearly found near the tree line.

Figure 26. centerY (target centroid distance) visualized.

Figure 27 shows how both classifiers model the performance impact of target distance with an asymptote at values over ~675.

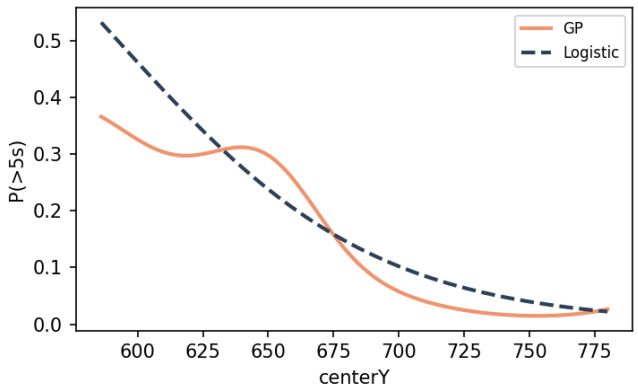


Figure 27. top: centerY visualized, bottom: marginal effect of centerY

Horizontal position

The GP in Figure 28a shows a clear U-shape, although not entirely centered, with the lowest P(>5s) at the horizontal centre of the screen, meaning centrally positioned targets were the easiest to detect. This partially lines up with binned failure rate in Figure 28b, specifically the left side, but notably not on the right side.

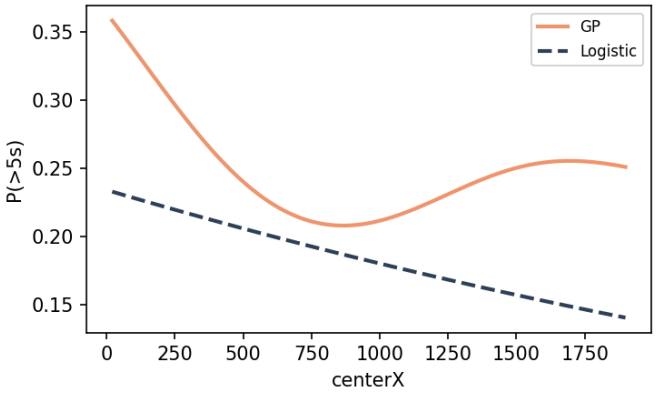


Figure 28a. Marginal effect of centerX.

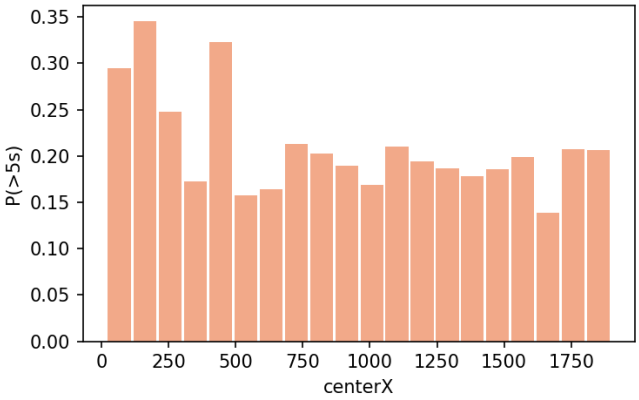


Figure 28b. Binned P(>5s) of centerX.

Figure 29 shows a plot of all 1200 unique target locations and the proportion of trials resulting in P(>5s). It clearly shows nearly all targets in a range of less than 175 meters from the camera have rates of detections in under 5 seconds of less than 20%. Around 250 meters (roughly matching with the tree line) and beyond P(>5s) results clearly tend to be colored yellow-red, indicating over 50%, long detection times / non detections.

Within this long-distance regime there appears to be a cluster of particularly difficult targets at ~Y 270 and X -80. Dots straight down the middle at long distances appear proportionally more blue (nearly all fast detections) than other markers at the same range.

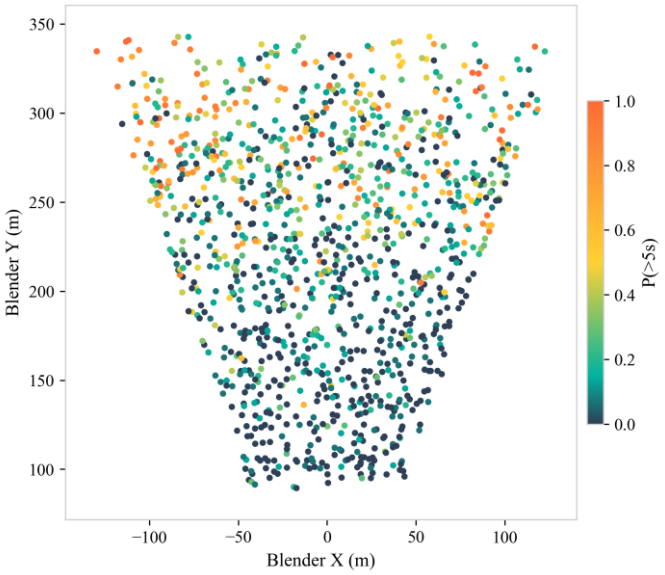


Figure 29. Plot of unique target locations and their average P(>5s) performance.

Rotation

The rotation compass (Figure 30) shows P(>5s) and median detection time results per orientation bin. The 0- and 180-degree headings show the rear and front facing sides respectively (see Figure 31 for reference), showing noticeably higher hiding performance compared to neighboring bins. Another stand out pattern observed is the asymmetry between left and right facing targets. With right facing targets having overall noticeably lower odds of remaining undetected for over 5 seconds.

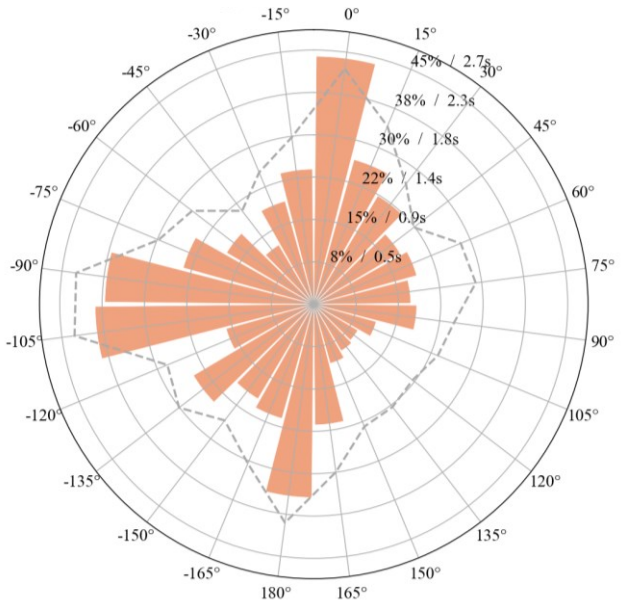


Figure 30. Rotation Compass showing vehicle orientation relative to the camera where 0 degrees is the vehicle facing away from the camera. Bars show P(>5s) and dashed line shows median detection time.

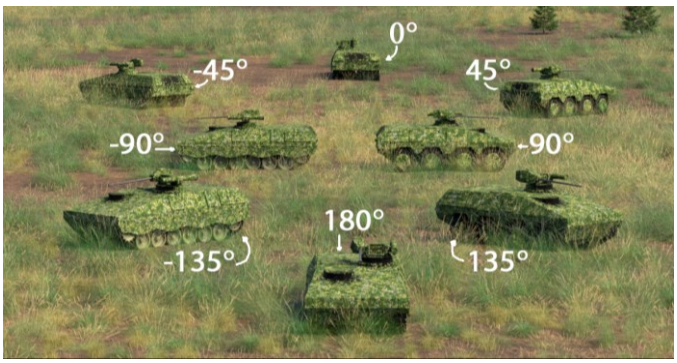


Figure 31. The target at 45-degree rotation intervals. The grass was made translucent in this image to improve the visibility of the shading effect.

The effect of rotation relative to the camera as modeled by the classifiers can be seen in Figure 32.

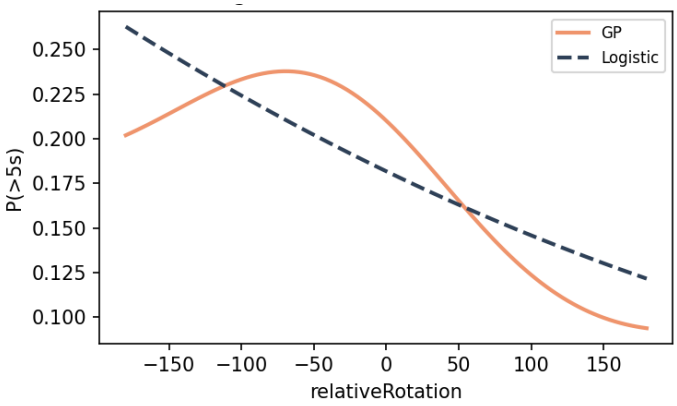


Figure 32. Marginal effect of target rotation relative to camera.

Learning effect

A learning effect is visible in Figure 33, with detection time declining most steeply across the first 300 trials and largely plateauing around trial 900, falling from a group mean search time of 4.30 seconds on the first trial to 2.28 seconds on the last trial.

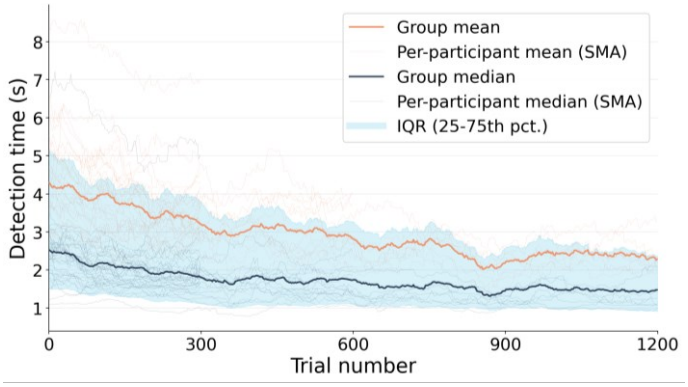


Figure 33. Spaghetti plot of simple moving average (SMA) of mean and median search times per participant as well as group (in trials with a successful detection as outcome).

Figure 34 shows that search trial outcomes follow a similar pattern, with the proportion of non-detections and false

detections making up about 12% of outcomes in early trials. The portion of false detections makes up about half that of non-detects in the first 300 trials. After 300 trials the fraction of false detection appears to have halved and stays fairly consistent from then forward.

From trial 750 onward the non-detection rate shrinks further to a similar proportion as at the start but half as large overall at only about 5% of trials not resulting in a successful detection.

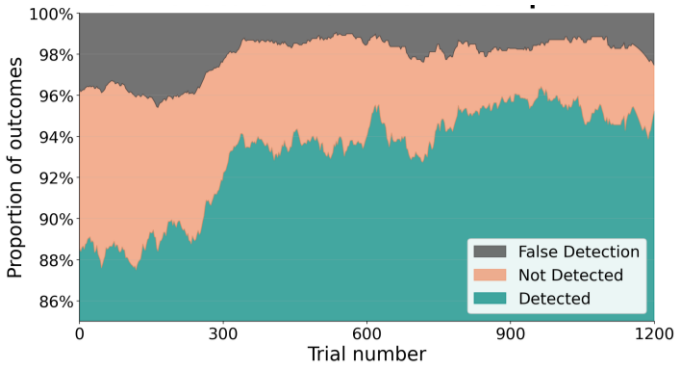


Figure 34. Stacked area chart of search trial outcome composition with SMA applied.

Figure 35 shows the spaghetti plot of mean and median detection times for exclusively the 7 participants who participated in 4 batches (the maximum amount of stimulus available per participant). While other participants were exposed to more than 300 trials, these are the only ones that participated in the full 1200, they also had the most prior experience (varying per participant) in pilot testing.

A much lower starting average detection time of 2.96 seconds, which unlike the full group increases slightly over the first 200 trials. Over that same period median detection time does continue to decrease and it does so fairly consistently till it levels out between trial 600 and 900.

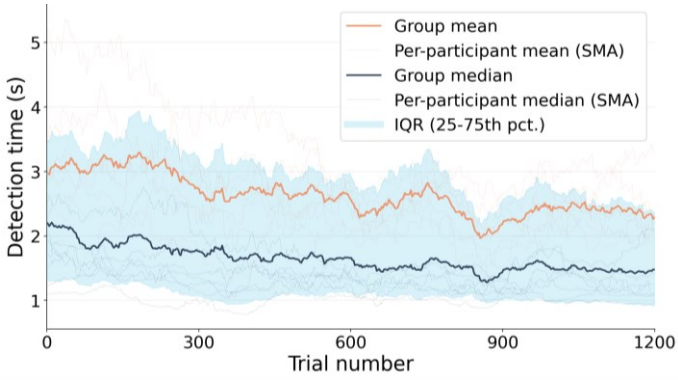


Figure 35. Spaghetti plot with simple moving average (SMA) of mean and median search times per participant and as a group of the four-batch subsection of participants (in trials with a successful detection as outcome).

Trial outcome composition (Figure 36) shows false detections for 4-batch completers staying consistent at around 2%, across all trials. While non-detections halved from around 6% to a little less than 3%.

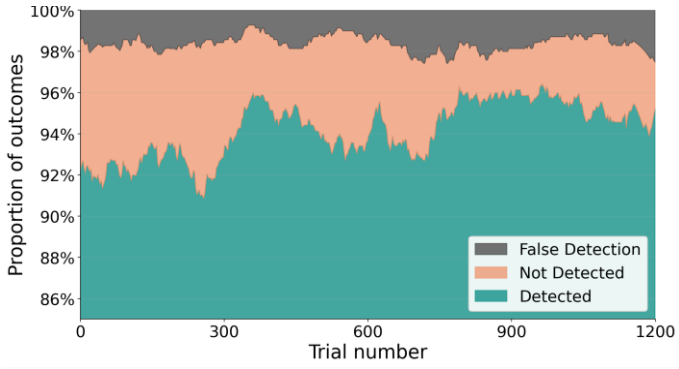


Figure 36. Stacked area chart of search trial outcome composition with SMA applied for four-batch participants only.

Returning to the P(>5s) shown for trial number in Figure 37, we see a clear decrease in number of trials exceeding 5 second search time from 0.28 at the start to what looks like a plateau of around 0.14 at the end.

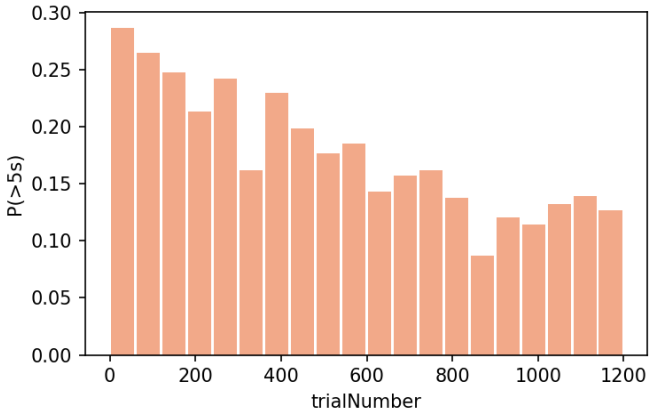


Figure 37. Binned failure rate of P(>5s) based on trial number.

The learning effect was modeled by the LR and GP classifiers as can be seen in Figure 38.

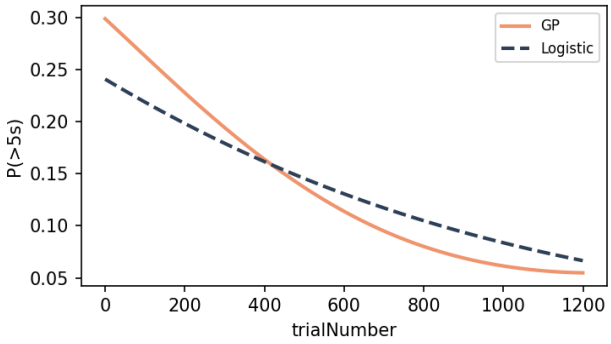


Figure 38. Marginal effect of trial number on P(>5s).

3.6 Parameters

Both classifier models attribute substantially more variance in detection outcome to scene and observer factors than to the camouflage pattern parameters, as shown in the feature importance analysis and covariate detail above.

Parameter 1 – Patch Size

P1 shows the clearest and most consistent effect of the four parameters. Higher values are associated with increased detection difficulty in both models (Figure 39), with the logistic coefficient significant at $p < .001$, as presented in section 3.4

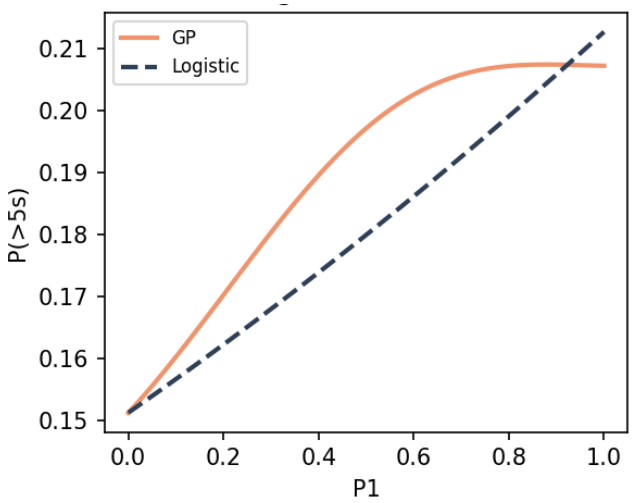


Figure 39. Marginal effect of P1 as modeled by GP and LR.

Parameter 2 – Horizontal Stretch

P2 shows a consistent negative effect: higher values are associated with easier detection in both models (LR: $p < .001$); see Figure 40.

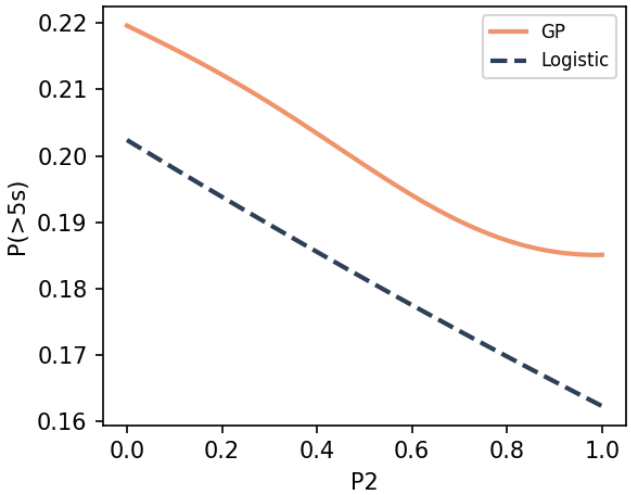


Figure 40. Marginal effect of P2 as modeled by GP and LR.

Parameter 3 - Grain

P3 presents a discrepancy between the two models (Figure 41): the GP suggests a positive effect of some magnitude on detection difficulty while logistic regression marks the coefficient as non-significant.

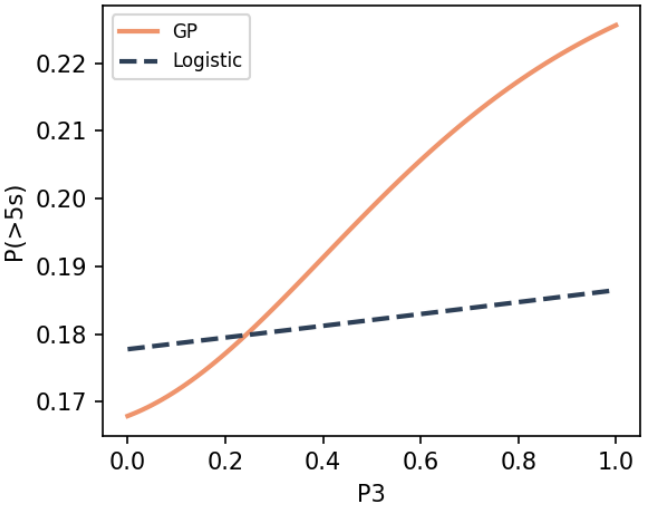


Figure 41. Marginal effect of P3 as modeled by GP and LR, showing a strong difference in perceived performance impact through the range.

Parameter 4 – Feature Mask

P4 displays somewhat surprising performance. Overall, the parameter seems highly detrimental. Detection chance and search time are strongly influenced by high values of the parameter. However, the GP model shows a local maximum in parameter performance between 0.2 and 0.4. This local maximum is where the feature masking effect is barely visible and will be extensively analyzed in the discussion section.

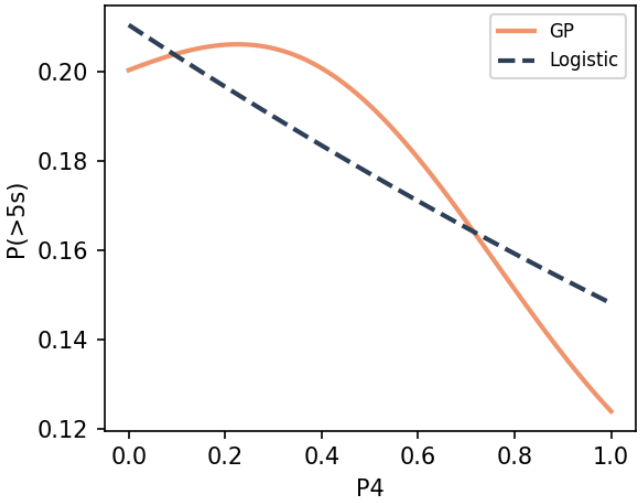


Figure 42. Marginal effect of P4 as modeled by GP and LR.

Asymmetric performance

Figure 43 shows the rotation compass with high and low bins of P4 applied, visualizing the differences in performance for these bins. Higher bins of P4 do not appear to affect the vehicle symmetrically, with better performance for low values of P4 when facing the front and better performance using higher values of P4 when facing away from the camera. A blip of improved performance with high P4 also stands out at 90 degrees.

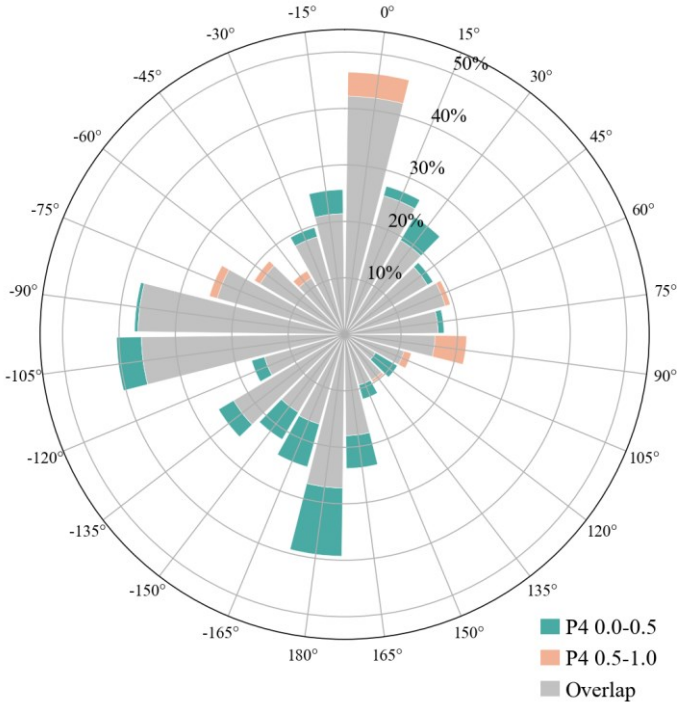


Figure 43. Rotation compass showing the asymmetric influence of high and low P4 values on different orientations of the target relative to the camera.

High performance participants

The ten participants with the lowest median detection times were analyzed separately. This subgroup was responsible for 8700 (48.6%) of evaluated samples. For P1, P3, and all covariates, trends are consistent with the full dataset. P2 and P4 diverge notably: the GP model shows a U-shape for P2 with a minimum near 0.3, and an inverted U-shape for P4 with a peak near 0.55, trajectories that contrast with the full-dataset marginal effect.

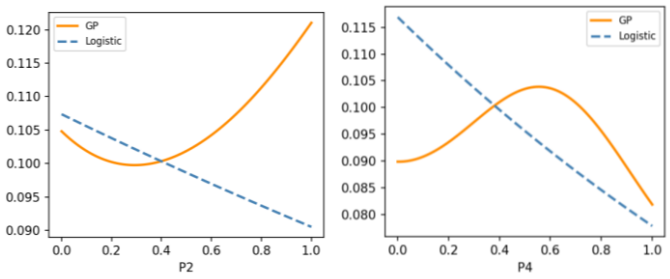


Figure 44. Marginal effects of P2 (left) and P4 (right) for the 10 top performing participants (lowest medians).

3.7 Validation

This section presents the results of the validation experiment, testing whether the GP model's rank ordering of candidate patterns is reflected in observed human detection performance. The four candidates selected from the GP performance mapping and their corresponding parameter values are shown in Table 3. Predicted probabilities of P(>5s) are reported for both the GP classifier and the logistic regression model. The GP predictions were used over LR because it had slightly higher performance (AUC of 0.79 vs 0.77) and because it handles non-monotonic relationships.

Candidate	P1	P2	P3	P4	GP P(>5s)	LR P(>5s)
0th percentile (ID = C1)	0.65	0.00	1.00	0.35	**0.253**	**0.220**
25th percentile (ID = C2)	0.75	0.75	1.00	0.20	0.201	0.178
50th percentile (ID = C3)	0.85	0.45	0.60	0.75	0.177	0.183
100th percentile (ID = C4)	0.00	0.75	0.50	1.00	0.102	0.115

Table 3. Candidate patterns with exact parameter combination values and classifier predicted performance. Highest performance (best camouflage) predictions for both GP and LR models marked in bold and underlined.

Note that throughout, lower percentile rank denotes a pattern predicted to be harder to detect (better camouflage): C1 at the 0th percentile is the predicted best performer and C4 at the 100th percentile the predicted worst.

Visually a few things stand out from the candidate patterns. C1-C3 (Figure 45a–45c) clearly have a lot in common; C1 and C2 in particular are very similar, differing only slightly in P1 and P4 and moderately in P2, differences that are difficult to distinguish at range. Their

high values of P1 are the most defining feature yielding small patch sizes. Visually, C3 stands out the most among the three due to its high P4 value (0.75), recognizable by the masking around the windows. C4 (Figure 45d) is a clear outlier, a low P1 value of 0.00 combined with a high P2 of 0.75 has expressed itself as large, elongated patches, stretching the length of the vehicle. It also has a maximum P4 value of 1.00, creating a large mask around the windows.

Figure 45a. Candidate pattern 1 (C1) representing the 0th percentile of performance according to the GP model.

Figure 45b. Candidate pattern 2 (C2) representing the 25th percentile of performance according to the GP model.

Figure 45c. Candidate pattern 3 (C3) representing the 50th percentile of performance according to the GP model.

Figure 45d. Candidate pattern 4 (C4) representing the 100th percentile of performance according to the GP model.

Validation outcome

The P(>5s) results show the expected ordinal structure holding in all but one case (Figure 46). For first exposures, the rank ordering matches predictions; however, the full dataset, which includes repeated exposures to the same scenes, shows performance averaging out, with C1–C3 showing reduced apparent effectiveness and C4's performance increasing slightly.

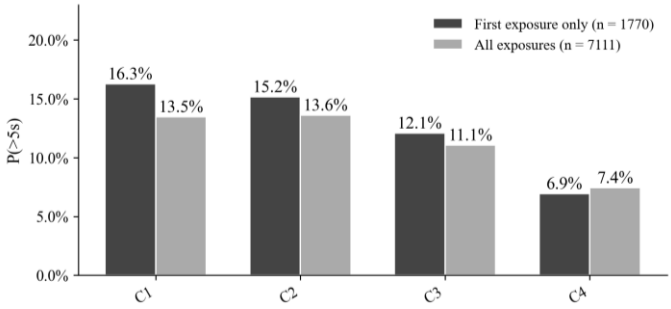


Figure 46. P(>5s) results of all candidate patterns for all exposures as well as the first exposure subset.

This possible memorization effect is visualized in Figures 47, 48 and 49. The first figure shows a significant decrease in detection time across repeated exposures to the same scene. The second shows how exposure number affected each pattern individually: the expected ordinal structure holds for the first two exposures but loses consistency afterward.

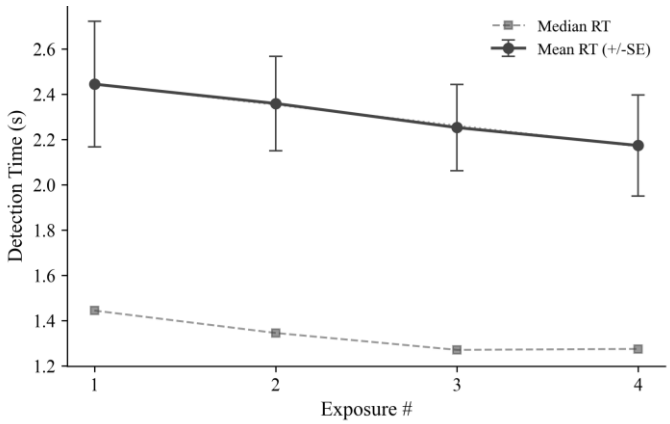


Figure 47. Mean detection time per exposure.

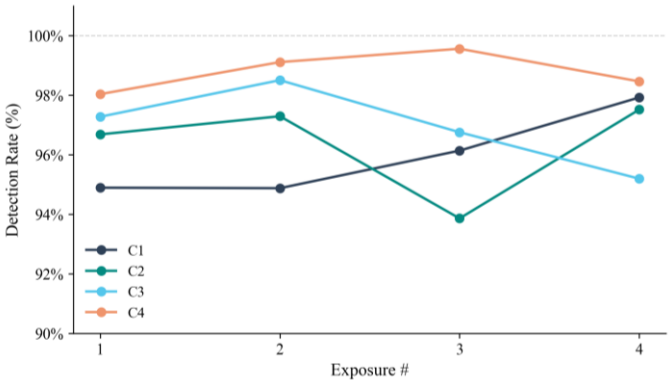


Figure 48. Detection rate per exposure.

Figure 49 shows the same ordinal structure as Figure 48 for P(>5s) holding for first two exposures and then losing structure before largely showing the same outcomes for all patterns in the last exposure.

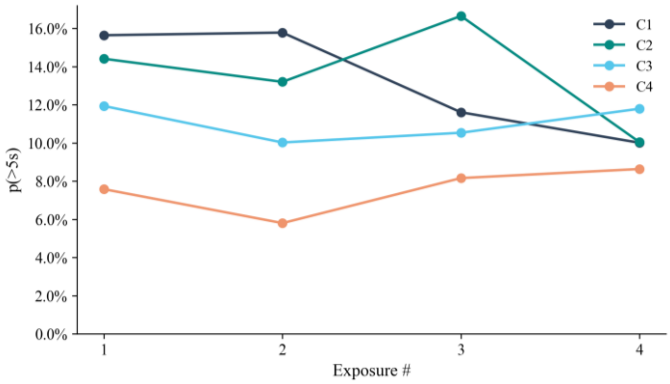


Figure 49. P(>5s) per exposure.

Figure 50 shows the non-detection rate and median detection time per candidate pattern. Although these metrics were not directly predicted by the GP model, both show ordinal results consistent with the expected ranking, even on the full dataset.

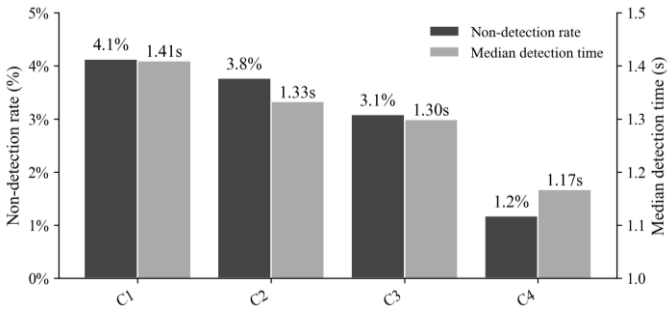


Figure 50. Non-detection rate and median detection time per candidate pattern.

To test for significance in performance differences, a repeated measures analysis of variance (RM-ANOVA) was conducted with camouflage pattern as the within-subjects factor, treating scenes as the unit of analysis. The full dataset across all exposures was used, because not every scene had all four patterns present on the first exposure (due to randomized stimulus order and n=12), making a first-exposure only RM-ANOVA incomplete. This means results are based on exposure-averaged statistics and are likely conservative estimates of true between-pattern differences. Despite this, significant differences were found between most pattern pairs, all of which were in the expected rank order. The notable exception was C1 and C2, which showed equivalent mean P(>5s) across scenes of 14.2%. The effect of camouflage pattern on P(>5s) was significant ($F(3, 447) = 16.62$, $p < .001$) with a small effect size of ($\eta^2 = 0.015$); pairwise comparisons are shown in Figure 51.

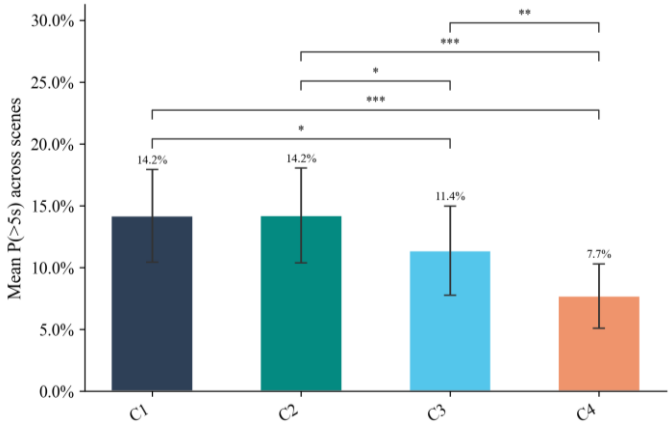


Figure 51. RM-ANOVA of P(>5s) per pattern for all exposures numbers combined across scenes.

Finally, predicted and observed performance differences are compared using first-exposure data only, as this is the least contaminated part of the dataset. An overall offset was observed between predicted and measured P(>5s) values, observed outcomes were approximately one-third lower across all candidates. This is likely attributable to participant selection: seven of the twelve validation

participants were the same individuals who completed four batches in the main experiment, and only one had no prior task experience, a much higher experienced fraction than in the main experiment pool. Given the learning effects reported earlier, this composition plausibly accounts for the elevated baseline performance. To isolate the rank-ordering accuracy of the models from this absolute offset, results were normalized relative to C1, yielding a comparison of relative performance across candidates.

Figure 52 shows observed performance tracking the GP rank ordering more closely than the LR ordering (mean absolute error of pattern-level detection rates: GP = 0.054, LR = 0.104). The one point of disagreement between the two models: LR placing C3 above C2, which the GP did not, is resolved by the validation data: C2 showed significantly higher performance than C3 ($p < .05$, Figure 51). The one case where predictions overestimated was the gap between C1 and C2: a difference is present in the first-exposure data, but it is considerably smaller than both models predicted, while the other pairwise differences align more closely with predicted magnitudes.

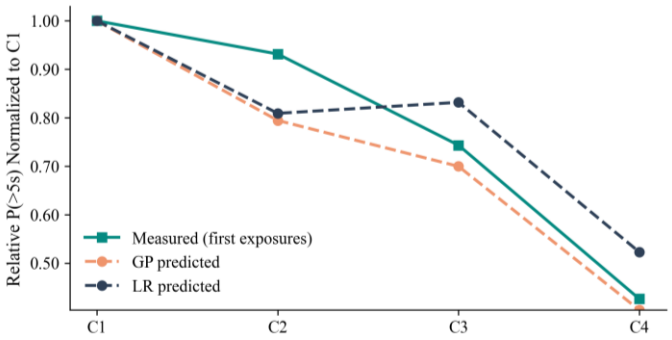


Figure 52. Predicted LR and GP performance vs measured performance relative to C1.

4 Discussion

This study examined whether camouflage performance can be mapped across a continuous, multidimensional parameter space using human search trial data, and whether a model fitted to that data can identify pattern combinations that perform well. The discussion that follows is organized around four questions: the validity of the method, the pattern design implications of the parameter map within the environment tested, the conditions under which those implications hold, and the methodological tensions the design imposed on the analysis.

Section 4.1 examines the validation experiment as the primary evidence on the method's efficacy and addresses the memorization effect and predicted-versus-observed offset that complicate its interpretation. Section 4.2 considers the range of context surrounding the experiment parameter performance model outcomes, as well as inherently interesting findings from covariate condition effects. Section 4.3 synthesizes what the parameter map revealed about pattern design in this environment, including the divergence observed in high-performing participants.

4.1 Validity of prediction models

Rank ordering

On first-exposure data, the GP rank ordering held in full: C1 > C2 > C3 > C4 in P(>5s). Logistic regression diverged at one point, placing C3 above C2, a reversal the validation data refuted (C2 > C3, $p < .05$). The better performance of the GP here is consistent with it capturing nonlinear structure the linear model cannot. Across all metrics and data subsets, C1 and C2 scored considerably closer together than either model predicted. As the two patterns share high P1 values and differ mainly in P2, this compression is consistent with P2 having a weaker marginal effect in high-P1 regions than the global model assumes, a potential interaction the stationary ARD kernel cannot resolve. This is an important reason to consider other prediction models in the future which can map these interaction effects. Certainly, this will be important if the goal is to find the absolute best patterns in a parameter space.

Predicted vs observed offset

Observed P(>5s) was approximately one-third lower than predicted across all four candidates, a broad downward shift rather than a difference in ordering. The most likely explanation is participant composition: seven of twelve validation participants were four-batch completers from the main experiment, the most experienced subgroup in the study. The learning effect reported in Section 3 shows non-detections declining from about 8% in early trials to a plateau near 2.5% by trial 750. The values observed sit close to that plateau, 2 to 5% for C1–C4 on first exposures. This appears consistent with a participant pool that had largely saturated prior experience. This justifies the comparison between model prediction performance and measured performance of first exposures normalized to C1. The predictions and measured performance relative from pattern to pattern are what is most important, not the

exact values associated with them. This is because the model's predictions reflect the broader, more naïve population from the main experiment.

Memorization and within-participant scene reuse

The RM-ANOVA found significant differences between patterns across scenes, and every significant pairwise comparison fell in the predicted order. C1 and C2 were not significantly different from each other, which is itself meaningful: the two patterns are visually almost indistinguishable at range, and the model's separation of them reflects a very fine performance gradient. Decomposing results by exposure number reveals a more important complication: the rank ordering holds for exposures one and two, then degrades. C1–C3 show declining apparent effectiveness in exposures three and four and ordering changes for each of the last exposures.

This was not anticipated. Given the visual similarity of scenes it seemed unlikely that participants would recognize individual ones. However, full scene recognition is not required for memorization to shift aggregate statistics. A likely explanation is that participants simply recalled approximate target locations. Since P(>5s) was around 12.5% (for first exposures), only about 19 out of 150 scenes were difficult enough to require longer search times on average. Given that around a third of those were non-detects, that leaves around 13 targets which were found between 5 and 30 seconds on average. Further taking into account the long right tail of search trial results one can quickly see that only a good handful of targets were found after a long search. These are plausible numbers of locations to roughly recall, particularly after repeated exposures. This would explain overall participant performance of increasing somewhat at the start before the memorization effect compounds (on certain scenes) and causes ordering to become chaotic. By the third and fourth exposure, participants may not think "I recognize this scene" but rather "I saw a few very hard scenes, and I remember roughly where the target was" and check those coordinates early. Eye tracking studies could substantiate this theory.

First-exposure data therefore provides the cleanest comparisons; the full-dataset RM-ANOVA is a conservative estimate of true between-pattern differences.

Taken together, the validation confirms ordinal consistency with GP model predictions and that C1/C2 at least occupy the same high-performance region; resolving the fine gradient between them would require a first-exposure-only follow-up at higher trial counts.

4.2 Non-camouflage effects

Beyond the validity of the prediction models themselves, a range of factors unrelated to camouflage pattern design substantially shaped detection outcomes in this study. This section addresses these non-camouflage influences in turn: participant search strategy, model calibration, the consequences of scene repetition for model fitting, and the relative importance of predictors. It then discusses the covariate effects.

Search strategy

Upon closer inspection of the search trial outcomes on a per participant basis (Figure 21), it becomes clear that there are significant differences between participants in detection rate which in part may be explained by search strategy. Participants with high non-detection rates appear to have lower false detection rates, while participants with low non-detection rates appear to have comparatively high false detection rates. Participant 1 does not have a single non-detect while their false detection rate is one of the highest.

This suggests that some participants may have simply clicked somewhere when a target could not be found, whereas others were (perhaps overly) hesitant to make any mistakes and instead elected only to press spacebar when they were certain they saw the target.

This raises questions about whether false detections should be included in analysis under certain circumstances. Especially since P(>5s) has a relatively small signal of only 20.2% of data, when a significant chunk of trials with what should be signal are excluded, this doesn’t mean signal is missed, but signal is suppressed.

Model calibration

On the lower end predictions of low likelihood appear extremely accurate in Figure 23, which is to be expected since only 5.8% of trials resulted in non-detects and >80% of detections were made within 5 seconds as previously seen in Figure 19. The positive class P(>5S) was comprised of 20.7% of datapoints. The higher range of the calibration plot are scenarios where detection is much less likely. A very small subset of trials falls into this category

which explains why the models show less accuracy in this area.

Scene repetitions and overfitting

The decision to evaluate each scene 15 times, intended to approximate and filter the inherent difficulty of each scene, had an unintended consequence for model fitting. Because identical scenes produce identical covariate combinations, cross-validation folds inevitably contained the same scenes in both training and test sets. This meant that overfitting on covariate structure was not penalized during validation, forcing a more conservative model configuration than would otherwise be warranted. As a result, the GP's capacity to resolve fine-grained parameter interactions, its main edge over LR, was limited. A more detailed account of hyper parameter selection can be found in appendix D.

The choice of including covariates into prediction models however, is very promising. Not only does it yield a much more substantial understanding of the impact of covariate conditions, which in itself is useful, but it also has much more potential for isolating the true effect of design parameters across varying conditions than knowing nothing more than the fact that a *particular scene simply had higher average detection times*. While that may be sufficient in a traditional direct comparison study, when each parameter combination only has a single exposure, understanding covariates becomes just as important as understanding parameters.

Predictor importance

Across both models, scene and observer effects dominate over pattern parameters, a finding reinforced by earlier covariate-free attempts that yielded AUCs of only ~0.55 for both classifiers. The low variance in parameter effects follows directly from this, underscoring that covariates must be understood before parameter-level differences can be meaningfully interpreted.

While GP and LR models broadly agree on feature importance, the trialNumber parameter presents a notable discrepancy: despite a sizeable logistic coefficient, its ARD importance is among the lowest of the covariates.

The ARD kernel assigns each predictor its own length scale, which controls how sensitive the model is to changes in that variable when distinguishing between detected and undetected trials. A predictor that changes gradually and approximately linearly across trials, as trialNumber does, can be captured with a large length scale, meaning the model treats nearby values as nearly interchangeable. This results in low ARD importance, even when the overall association with detection outcome is substantial. Logistic regression, by contrast, is sensitive to the magnitude of a predictor's linear association with the outcome and therefore yields a sizeable coefficient for trialNumber regardless of whether its effect is locally discriminating.

Covariates

Target location

Target location clearly had a huge impact on detection results. The primary driver was the target distance, reflected by centerY. But the horizontal position also had an impact.

Target distance

The sudden change in performance around 650-700 for centerY coincides with the approximate tree line position in the rendered scenes, where targets are small, viewed against a visually complex background, and subject to partial occlusion by trees. The GP and logistic marginals agree closely across the full range, with the GP showing slight additional curvature in the 600–675 region.

At large centerY values, P(>5s) approaches its minimum (see Figure 27). Participants reported concentrating their search on the mid-to-far field, possibly they got a feeling for where targets usually show up, so a near-field target that was not immediately salient could occasionally accumulate a disproportionately long detection time. This reported phenomenon appears to be visible in binned rates as well as the GP marginal (Figure 25 and Figure 27).

Horizontal position

The horizontal location of the target showed a U-shape effect on detection results, with the highest odds of detection when targets were located in the center. This mostly lines up with expected search times and lateral target location (Toet & Bijl, 2003). Since this pattern is not visible in the bin chart it is likely that the bins contain amplified noise from repeated scenes containing tough covariate combinations which the GP is able to filter out. This explanation is in part corroborated by Figure 29, showing a cluster of hard to find targets on the left side of the environment. It is also worth noting that the left appears to be slightly more populated with targets at this distance while the right appears to have slightly more targets at close range. Quantifying density within coordinate combination ranges may confirm this. If so, it

would explain how the expected U-shape emerges despite binned P(>5s) not showing it, and be a good illustration of the advantages of classifiers such as GP over LR.

Rotation

The results from the rotation compass (Figure 30) play an important role in not just understanding the effects of certain parameters, as will later be described, but also in understanding how the geometry of STANDCAM in particular effects detectability. Despite the sun being at a constant angle, the compass shows signs of solar angle influence on results.

1 Visible pixels

Cross section varies with rotation: a head-on or tail-on presentation exposes a narrower profile, reducing the number of pixels the vehicle occupies and thus making detection inherently harder. We see this in Figure 30 with 0- and 180-degree headings yielding particularly high performance.

2 Asymmetry

Lateral asymmetry appears to play a substantial role. The right-hand side of the vehicle has four large, visually distinctive tires that appear to increase target saliency dramatically (Figure 31), a pattern reflected in the GP classifier's learned non-detect probability surface (Figure 32). This raises practical questions about where signature reduction gains are best achieved: through improved camouflage patterns, or through modifications to the vehicle exterior itself, such as side skirts, netting, or opting for tracks over wheels to reduce visual signature at the source.

3 Solar angle

Solar angle determines when hard shadows are cast along the vehicle's flanks, which can significantly increase signature (Penacchio et al., 2017). Notably, the rotation compass (Figure 30) shows two distinct gaps in P(>5s) around -45 and 135 degrees. The fact that these gaps occur 180 degrees apart highly suggests it is due to the shadow cast by the sun at that angle as seen in Figure 31. Furthermore, the fact that solar orientation showed up in rotation results, despite the environment being selected specifically to minimize these effects underlines the impact of cast shadows on detection, and puts into perspective what impact camouflage has compared to weather conditions the target is at the mercy of.

Learning effect

As shown in Figure 33, detection times declined steeply over the first 300 trials before largely plateauing around trial 900, suggesting that participants quickly internalized the search task. The early reduction in false detections, from roughly half the rate of non-detects in the first 300 trials to a near-stable ~2% thereafter (Figure 34), supports this interpretation: as participants grew more confident in identifying the target, they appear to have shifted from guessing to sustained searching, which would account for why non-detections remained relatively more common than false detections throughout the early phase.

These trends should be interpreted with caution, however. Because 56% of participants completed only a single batch (300 trials), the majority of the data falls within the steepest part of the learning curve and does not extend into the plateau region. This skews the group-level picture. Furthermore, the seven participants who completed all four batches (1200 trials) had varying levels of prior experience from pilot testing, likely compressing the apparent early learning curve for the full group. The true rate of skill acquisition may therefore plateau later than Figure 33 suggests.

Examining the four-batch subgroup in isolation (Figure 35) clarifies this. Their mean starting detection time was notably lower at 2.96 seconds, reflecting their prior exposure and, unlike the full group, rose slightly over the first 200 trials, possibly indicating early fatigue rather than learning. Median detection time did continue to decline for this group, leveling out between trials 600 and 900. Individual trajectories (faint lines in Figure 35) reveal that some participants showed very little improvement across all 1200 trials, with median detection times close to or below one second throughout. This likely represents a performance ceiling for the search task itself, rather than continued learning.

The outcome composition for four-batch completers (Figure 36) supports this ceiling interpretation: false detections held steady at around 2% across all 1200 trials. This consistency suggests the false-detection rate reflects a fixed background of accidental responses, participants reported occasional accidental spacebar presses rather than a strategic behavior that changes with experience. A stable ~2% false-detection rate may therefore serve as a proxy for task familiarity. That said, the distinct non-detection and false-detection profiles observed across individual participants (see Figure 21) suggest these response

tendencies are partly trait-like rather than purely experience-driven.

4.3 Parameter effects

With the covariate conditions discussed and general model validity addressed, we can finally discuss the perceived parameter effects, what they signal is important for these particular environments and how these findings may translate to other environments.

P1 and P2 – Patch size

The parameter with the largest LR coefficient was P1, and with good reason: it had the most dominant effect on patch size in the camouflage pattern. Results clearly point to smaller patches performing better, with GP identifying a plateauing region past P1 = 0.6. Levels of P1 this high are barely distinguishable from one another at range, as can be seen in Figure 53.

Figure 53. Target near the tree line with P1 0.1 (left), 0.6 (middle) and 1.0 (right), all other parameters set to 0.

At closer ranges differences can still be identified between 0.6 and 1.0 (Figure 54), but the spatial frequency between them is much more similar compared to 0.1 and 0.6.

Figure 54. Target at close range with P1 0.1 (left), 0.6 (middle) and 1.0 (right), all other parameters set to 0.

This particular environment appears to favor these high frequency patterns since they visually match the appearance of the grass, as can be seen in Figure 55.

Figure 55. Target in grass at medium range with P1 0.1 (left), 0.8 (right), all other parameters set to 0.

The effect of P2 had less influence according to reported feature importance in section 3.4, but showed the same general direction. Larger patches lead to worse performance. While P2 increases the patch size like P1 but horizontally, introducing anisotropy, its effect is still overshadowed by the base patch size as set by P1. Figure 56 shows this in practice. Where values of P1 determine what starting point P2 has to interact with the patches.

Figure 56. Targets showing P1 at 0.1 (left pairs) and 1.0 (right pairs). Where the left of each pair has P2 of 0.0 and the right a P2 of 1.0.

The performance implications of this become more clear when considering the candidate patterns from the validation experiment. C1 and C2's main difference lies in their values of P1 (0.65 vs 0.75 respectively) and P2 (0.0 vs 0.75). Figure 45a and 45b show slightly noticeable differences in spatial frequency, but it is limited. This is reflected in their similar performance with C1 arguably showing indicators of outperforming C2 in some metrics, but not to a significant degree given limited trial counts. While P1 (strongest predictor) was slightly higher for C2, it's combination with a high P2 resulted in overall slightly larger patches and thus, probably, slightly worse performance.

Another reason this optimum might exist and thus a limiting factor in transferability is the general image grain from the low sample count and 1920x1080 resolution. In Figure 57 we clearly see how noise in the field is resolved with higher resolution and sample counts. While small patch size may still work reasonably well, the grainy surroundings clearly contain straight lines at the highest shown render settings, making individual blades of grass visible, a noticeable departure from the stimulus material used. Whether this increased detail is sufficiently noticeable and changes optimization outcome, since Figure 57 shows cropped zooms of the full stimulus, is an aspect that warrants more studying.

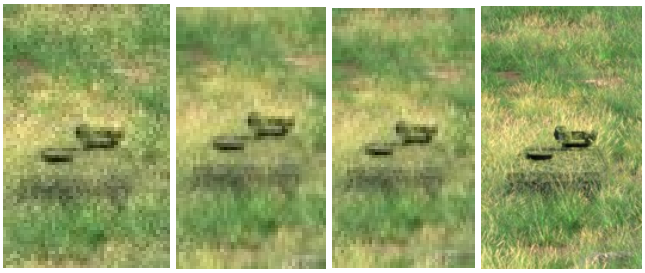


Figure 57. Target in grassy field with 125 sample render, matching the settings used for stimulus creation (left), 2000 sample render (left-middle) and 125 samples at 2.5 times higher resolution (right-middle) and 2000 samples at 2.5 times higher resolution (right).

The finding of smaller patches performing better in this environment type may not hold to the same extent in conditions with direct lighting. Because at range, high spatial frequency patterns perceptually blend into a uniform, low-contrast surface as adjacent colors are summed below the observer's resolution limit (Troscianko et al., 2017). Under direct lighting, however, cast shadows increase scene contrast, meaning that a low-contrast blended surface provides less silhouette disruption than a pattern retaining high-contrast edges at the target boundary. High-contrast edge markings have been shown to break up perceived outlines by overriding true body contour signals, and are more effective for concealment than background matching alone under such conditions (Hall et al., 2025; Stevens & Merilaita, 2009). In such light conditions low P1 high P2 may become beneficial.

P3 – Grain

While grain did have a visually noticeable effect at short range on targets with low P1 as seen in Figure 13, it was unnoticeable for most of the parameter space. Particularly after parameter space narrowing was done and fewer low P1 combinations were included. An example of this lack of expression is visible in Figure 58, where both examples show no visual distinction despite P1 being only 0.5.

Figure 58. Targets with P1 0.5, P3 0 (left), and P3 1 (right).

This particular feature thus had very little chance of expressing itself. However, marginal effect as analyzed by the GP model does show performance increasing with higher values of P3 Figure 41. This is likely the GP model being able to identify these sub-cases where high values of the parameter do help, even if it is not often. It however has no way of modeling that this is almost certainly an effect that only holds in specific conditions bound by the value of P1 and to a lesser extent P2. The LR classifier is not as easily swayed by a data pattern that only exists in some sub-cases and thus shows disagreement with GP.

P4 – Feature masking

P4 displays somewhat surprising performance. Overall, the parameter seems highly detrimental. Detection chance and search time are strongly negatively influenced by high values of P4 (see Figure 42). This even motivated the decision to bin P4 differently. The drop-off in performance was that obvious even in the noisy dataset. However, the GP model identifies a local maximum in parameter performance between 0.2 and 0.4, which is where the feature masking effect is barely visible. Only some small specks near the windows can be seen in Figure 59 and 60. Note that for these images specific parameter combinations with near 0 P1 had to be chosen as to not confuse the feature mask with noise of the rest of the camouflage pattern.

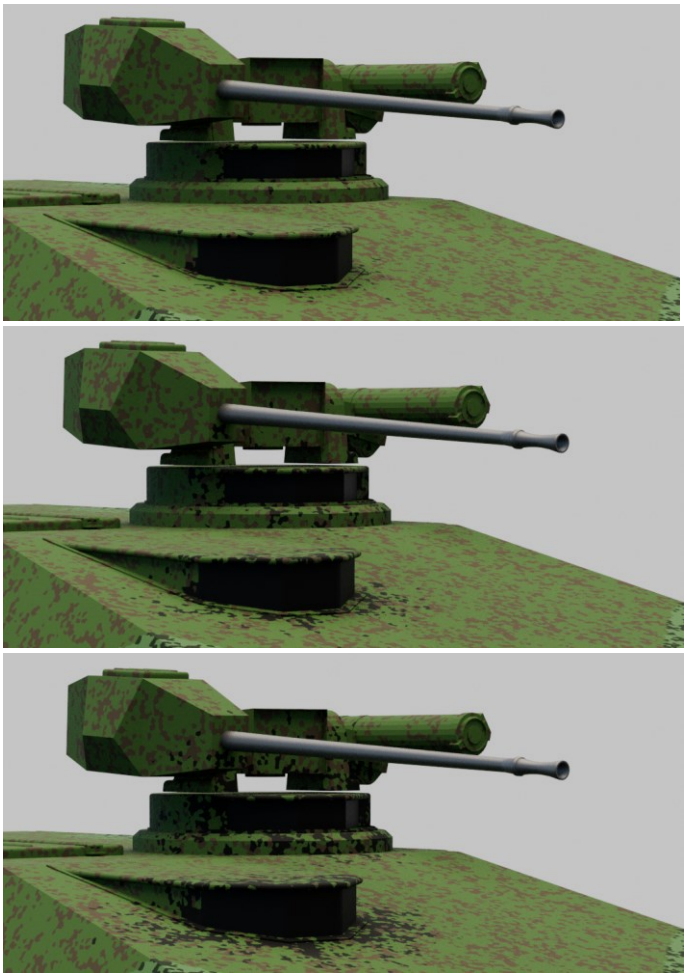


Figure 59. Levels of P4 on the front of the target, 0.2 (top), 0.4 (middle) and 0.6 (bottom).

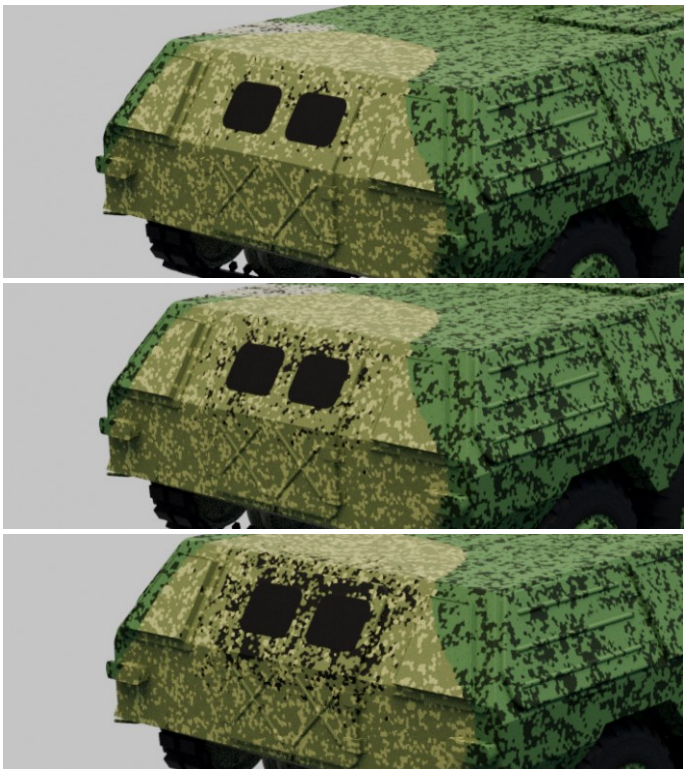


Figure 60. Levels of P4 on the rear of the target, 0.2 (top), 0.4 (middle) and 0.6 (bottom).

The model appears to suggest that there is a slight benefit to having a small amount of noise there while high values of P4 results in a clear identification marker (as seen in Figure 45d), since the stain becomes massive and thus hugely conspicuous in an environment with little low frequency noise and little dark shading. This appears like an artifact of the model possibly fitting to noise in the data since these values are barely perceivable, even up close. Only for values above 0.4 (such as 0.6, as shown) does the feature become noticeable, and this is exactly where we see performance plunge according to the GP classifier. An alternative explanation for the local maximum, one that does not require invoking model noise, emerges from the subgroup analysis of high-performing participants.

Participant performance

The subgroup analysis of high-performing participants (top 10 participants with lowest median detection time) revealed parameter effects that diverged notably from the full-dataset models for P2 and P4 (Figure 44). This subset of participants largely follow the same trend for P2 and P4, as indicated by LR but show diverging performance in particular ranges which is picked up by the GP model. Notably, we see a peak, around P4 values of 0.6. The magnitude is small but it may show how in a subset of instances mid-high values of P4 is beneficial in hiding (for over 5 seconds) from high performing participants. The location of this bump may explain how in aggregate a bump shows up for lower values of P4 in the complete dataset.

High-performing observers only approach the detection threshold in genuinely difficult scenes, meaning their data captures parameter effects in the regime where camouflage actually matters, considering camouflage should be effective against professional observers and camouflaging performance in an open field is likely less important than in places where crews can actually hope to remain hidden (e.g. a tree line). In Figure 61 we see a scenario in which a higher P4 may be effective, with the left target showing two salient black squares, whereas the right appears more like an oval black stain which matches more closely with spots of shade in trees and rocks in the field.

Figure 61, showing the target from the rear perspective in the tree line with P4 at 0.25 (left) and 0.75 (right).

Average participants often don't stand a chance when tough covariate conditions are at play, making their responses less informative about pattern-specific effects in the scenarios that may be most relevant for optimization. This raises a practical question for future studies: optimizing camouflage against an average observer may underestimate the capability of an adversary. The choice of participant population is therefore not trivial but determines what the resulting parameter map represents. Future work should consider explicitly targeting professional participants, training participants (e.g. by giving feedback on non-detections), or reporting parameter effects separately for high- and low-performing subgroups.

Asymmetry

Figure 43 (P4 rotation compass) shows signs of P4 having an asymmetric effect on P(>5s), with the front of the vehicle and most of the tracked side showing higher camouflage performance when P4 is under 0.5, while the rear of the vehicle shows slightly higher performance when P4 is over 0.5. Interestingly this is the side which the mask was conceived for, as the mask was supposed to obscure the sharp edges of the square rear windows, which was reported as a salient feature when the vehicle was facing away and stood (at a distance) in the tree line.

While the improvement (given the bounds of P4 used) is fairly small, it is peculiar that the opposite is true for the other side directly facing the camera. In any case this raises a methodological point of improvement, features with visually asymmetric application should be modeled as separate parameters, since grouping them, as done here makes it hard to pinpoint where (reduced) performance is coming from.

There is also an interesting blip in Figure 43 at 90 degrees. The difference between 90 and −90 degrees is not as large, but it is worth exploring. In Figure 62 we see the side profile of the front facing windows on the STANDCAM. Given the light was always coming from the same direction we can see that the left pointing side always had some amount of natural shading, whereas the right pointing side did not. At these angles it may have been beneficial to mask the windows somewhat, where on the former P4 visually made little impact. This assessment, however, should be taken with a grain of salt since the lack of signal at this range (relatively few P(>5s)) and repetition of scenes means a few unique scenes with disproportionally difficult covariate conditions could already influence results.

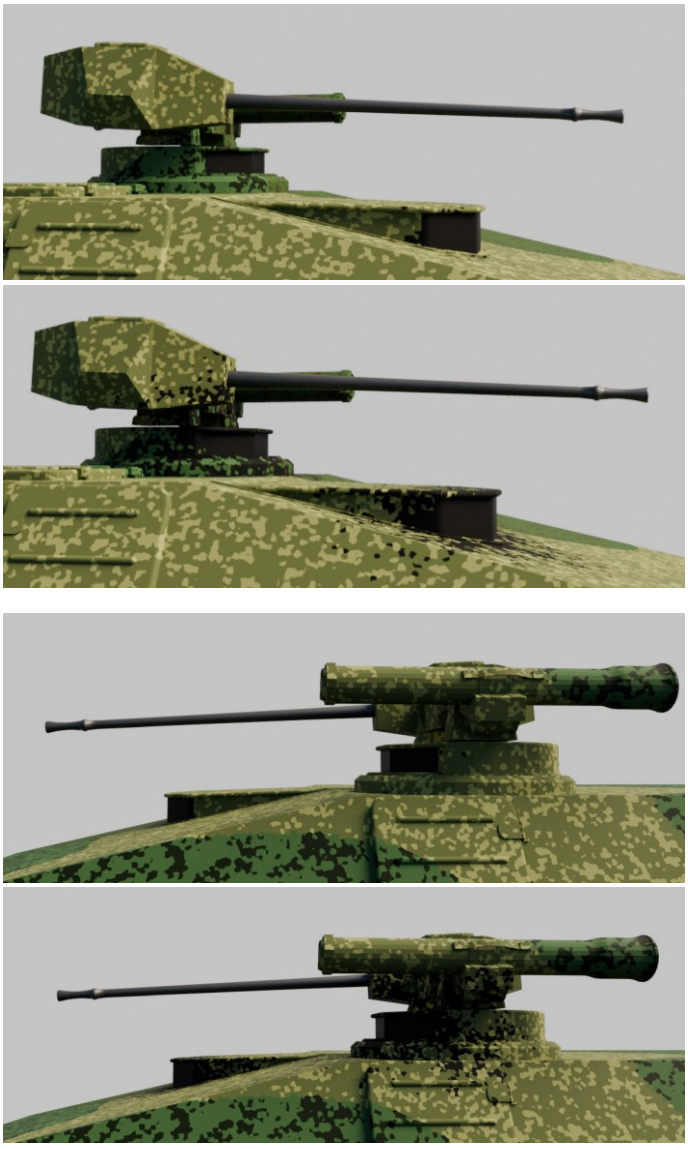


Figure 62. Effect of high and low P4 on front facing windows as seen from the right side (top images) and the left side (bottom images). The top of either set has a P4 of 0.2 while the bottom has a P4 of 0.8.

From analysis of P4 performance one cannot definitively conclude it has a blanket negative influence on performance, as a first impression of the data would appear to indicate. Its performance impact, however, appears to be more strongly dependent on the position/rotation of the vehicle than other tested parameters. Broadly speaking, only saving the vehicle from detection in cases where it was well hidden and would only be found by a keen observer.

More research can be done on alternative shapes of the feature mask, currently it is mostly a shapeless noisy blob but, it could take on more organic shapes, opening the door to more parameter combination testing. Analysis that could be done with the exact tools and method this thesis is about. It also raises questions about dimensionality in such studies. Should all optimizations be done at once, where every interaction between parameters is taken into account. Or is it more feasible to, for instance, find a good general camouflage pattern for an environment and then do a secondary experiment where a feature mask is applied. A tradeoff in dimensionality (trials/participants required, ultimately cost) and maximizing performance discernability.

P(>5s) as a metric

The merit of P(>5s) as a metric warrants a brief reflection. The measure itself was born out of necessity when direct modelling of detection time proved challenging. The value of the threshold was set based on a range of modeling attempts briefly described in appendix D, and it appears to be effective in doing that. Significant effects can and have been identified in this study using it, therefore it clearly captures an important signal. The value of 5.0 seconds, however, should not be considered infallible and may actually warrant reconsideration. Given the real possibility that high-performing participants have different optimal camouflage pattern outcomes, this has problematic implications for the metric used. In practice, participants do not provide the same amount of signal to the metric used in evaluation; this is because high-performing participants simply exceed the threshold less often. This also explains why charts specific to these high performers (Figure 44) show very small magnitude.

Simply moving the threshold to a lower value does not solve the issue, and in fact may be exactly why 5.0 seconds was such a good setting. When the threshold is lowered this will primarily include more signal from lower performers, up to a point where the majority of trials become positive signal making it impossible to discern anything meaningful at all. This asks for a modified version of the metric, which is participant specific. A threshold relative to participant search time distribution: the nth percentile of search times for instance. Additionally, a weighting may be applied to push that threshold further up or down based on participant performance relative to the whole participant population, emphasizing signal from high performers while muting that of low performers, the opposite of what happened during this study. While differences in optimal camouflage pattern are likely restricted to specific predictors and sub-scenarios (like P4 and tough scenes), this could be a meaningful improvement to parameter mapping results.

5 Limitations

This section identifies the key constraints of the study's design, methodology, and execution. The limitations are grouped by theme and are intended to contextualize the findings presented above, supporting a more accurate interpretation of the results and informing the recommendations that follow.

5.1 GP classifier

The GP classifier assumes a stationary ARD kernel, meaning a single lengthscale is learned per feature across the entire input space. This limits the model's ability to resolve localised high-performance regions or capture interaction effects where the influence of one parameter reverses depending on the value of another. It also cannot be ruled out that covariate masks subtler parameter interactions. Additionally, the sampling design introduces a density confound: the adaptive second-stage Sobol sampling concentrates approximately three times as many observations in 27 selected bins covering 50% of the parameter space. While this reflects a legitimate adaptive sampling strategy that improves local resolution in promising regions, the GP's hyperparameter optimization is disproportionately influenced by those bins, and posterior uncertainty is lower there relative to the remainder of the space. Conclusions about global parameter importance should therefore be interpreted with this asymmetry in mind.

5.2 Light and orientation

A deliberate choice was made to use diffuse lighting throughout the experiment to avoid hard shadows that are

difficult to render faithfully on non-HDR displays. This means findings from this study should not be generalized to direct lighting scenarios.

5.3 Apparatus inconsistency

One remote participant reported not using the same monitor across both generations, meaning that observed performance differences between generations may partly reflect changes in display conditions rather than genuine learning or camouflage effects for some participants.

5.4 Unsupervised execution of the experiment

Participants completed the experiment remotely and without supervision, meaning protocol adherence could not be verified. Factors such as ambient lighting, viewing distance, and distraction levels likely varied across participants and sessions. This introduces unmeasured noise into the detection time data that cannot be disentangled from camouflage or covariate effects.

5.5 Uneven batch distribution

A substantial learning effect was observed across trials, discussed in detail in §4.2 The uneven batch distribution, where a small subset of participants completed four batches while most completed only one, means this effect is unevenly represented across the dataset.

That said, the imbalance was a practical consequence of allowing participants to return for additional batches, which significantly increased the total number of stimuli evaluated. Higher-performing and more experienced participants are arguably more informative for camouflage assessment purposes, so the skew is not without merit. Therefore, such an approach need not be strictly avoided in the future; however, increasing the minimum trials per participant may be worth aiming for.

6 Recommendations

The following recommendations are directed at researchers looking to replicate or extend the parameter space mapping methodology used in this study. Each recommendation is grounded in a limitation or tension identified during this study and is intended to improve information yield in future iterations.

6.1 Scene selection and covariate coverage

Scene selection determines what the experiment actually optimizes for, so researchers should first define the operational context (target distances, occlusion levels, lighting conditions) and use that to guide scene filtering criteria. Beyond relevance, coverage matters: scene selection should be verified to provide balanced coverage across key covariate dimensions. Stratified guardrails during selection, for example, enforcing minimum counts per distance bin or occlusion level would prevent systematic underrepresentation and reduce confounds between covariates that complicate model fitting.

6.2 Scene repetition

Future studies should use fully unique scenes. Repeated scenes introduce covariate interdependence that confounds model fitting and forces conservative hyper parameter settings, limiting the ability to resolve fine-grained parameter interactions. Unique scenes decouple covariate bins, allow models to train for more epochs without overfitting, and can yield higher-fidelity parameter space mapping even with a smaller total dataset.

6.3 Parameter space narrowing

Rather than removing bins to narrow the parameter space, a machine learning model could guide resampling decisions at a finer resolution than bins allow, since large bins cannot discriminate between performance areas at a fine level, and small bins are vulnerable to local noise clusters. Regardless of method, narrowing should not be applied too aggressively or too early. Analyzing underlying patterns with sufficient data before making resampling decisions reduces the risk of biasing samples toward particular parameter combinations. If unequal density is unavoidable, a density-robust model such as gradient boosting may be better suited for subsequent fitting than a GP, as tree-based methods do not assume stationarity and are largely unaffected by regional concentration of observations.

6.4 Camouflage model

Future work should assess the perceptual uniformity of the parameter space before sampling, ensuring that the sampling distribution is concentrated in regions where parameter variations produce perceivable differences. In this study, the dominance of P1 meant that large portions of the space were visually indistinguishable, reducing effective information gain. Pilot perceptual tests or coarse

pre-screening renders could identify such low-return regions and guide a more efficient sampling strategy.

6.5 Time limit

The search time data was incredibly noisy, which was to be expected since each parameter combination was evaluated only once and the scene the target was evaluated in clearly had a massive influence on detection time and outcome. While more sophisticated data analysis techniques may be able to retrieve more information from the resolution this data provides, the fact of the matter is that only around 6% of detections took place after 10s, but the cutoff wasn't until 30 seconds. Because of this, participants spent a lot of time (8.7 minutes on average, per batch) looking for targets they would never find. In some cases, due to a well camouflaged target but in many cases, it was the scene that made it nearly unfindable. A lower cutoff (e.g. 10 seconds) would allow more time to be spent on evaluating additional samples, directly improving information gain per session.

6.6 Participants selection and learning effect

Future studies should prioritize recruiting participants with relevant visual search experience, such as trained military scouts, as the operationally relevant population and the one for which the resulting parameter map is most meaningful. Where specialist recruitment is not feasible, structured pre-experiment training can partially close the gap: post-trial feedback showing target location on missed trials, combined with instruction on vehicle-specific signatures (cross section, uncamouflaged features, turret silhouette), can accelerate skill acquisition, reduce the number of trials before participants reach a stable performance level, and limit the degree to which early-trial noise dilutes parameter effect estimates.

6.7 Render settings

The render settings used for stimulus generation in this study were optimized for speed at the cost of some image quality. Notably, grain was visible in the rendered images. Whether this is a concern depends on what is being optimized but render noise should ideally be reduced to prevent it from influencing optimal pattern outcomes.

7 Implications

This study produced both a validated camouflage evaluation method and a generalizable pipeline for human-in-the-loop parameter optimization. The implications extend beyond the specific patterns tested: the tool, the modelling approach, and the underlying framework each open avenues for future application in military signature management and beyond.

7.1 Future use of the tool/method

The automated sample-to-stimulus pipeline developed for this study is already being repurposed at TNO for other detection experiments, which confirms its utility extends beyond the specific camouflage application presented here.

In its current form, the tool supports dense human-in-the-loop sampling across any parameter space that can be expressed in a Blender material or geometry node setup. Its usefulness has been demonstrated and should be considered in the manner described, a pre-selection tool to inform candidate selection for higher accuracy evaluation methods such as hybrid simulations.

The framework is also well suited to scenarios where physical or hybrid simulation is impractical, such as conflict-region environments or mission specific camouflage.

Geometry based parameters

The use of geometry node systems to drive camouflage properties based on vehicle feature proximity is a distinctive and underexplored capability of this pipeline and warrants further investigation. Unlike texture-based parameters, geometry-driven parameters can target specific structural features of the vehicle, opening up a new design space.

Areas like countershading and edge masking but also the possibility of simulating camouflage nets are well within the reach of this tool.

7.2 Broader applicability

Signature optimization

The geometry node system used for P4 points toward vehicle geometry as a variable: features such as side skirts, track covers, or netting could be parameterized and evaluated in the same pipeline. This is particularly relevant given the rotation results, which showed that the lateral tire profile was a stronger determinant of detection than any

pattern parameter tested. Addressing that through exterior modification rather than pattern design may yield greater signature reduction per unit of effort.

Beyond camouflage pattern optimization, the same framework could be applied to signature analysis under varying illumination conditions. Simulating different solar angles, times of day, or weather states would allow the detectability surface to be mapped as a function of environmental factors rather than pattern parameters. Understanding one's own signature plays an important role in estimating detectability in the field and can inform operational decisions on the ground e.g. which route to take from A to B. This is becoming increasingly relevant in modern conflicts driven by a constant stream of data.

Moving away from optimization, the method shown in this study could form the basis of a standardized benchmarking tool to map signature on a per vehicle basis. This may be of great interest for militaries to determine whether a vehicle meets operational demands: e.g. special forces needing to keep a low profile. Thereby offering a methodical approach to guide acquisition and selection of the right tools for the job.

Civil applications

Beyond the domain of signature assessment and optimization these findings could inspire designers to use parameterized experiments to optimize their designs. Digital designs (2D or 3D) lend themselves to this purpose best due to the fine control and automation potential. However, these optimization problems may not encounter the same amount of noise, if the input metric is (for instance) a measure of perceived esthetic quality rather than visual search data, which is inherently noisy. Less noise would mean fewer datapoints are needed.

Bringing this method into the physical domain of design may also be possible using techniques such as 3D printing to evaluate a range of parameterized designs on a chosen metric.

Another area that may meet the qualifications for precise control and automation could be the beverage industry, where optimizing a recipe based on a range of ingredients (parameters) may give more insight than direct comparisons.

8 Conclusion

This thesis set out to answer whether camouflage pattern performance can be mapped across a continuous, multidimensional parameter space using human detection data, and whether a machine learning model fitted to that data can identify ordinal structure in which parameter combinations are harder to detect. The answer to both questions is yes.

A fully synthetic, parametrically defined stimulus pipeline was developed and used to generate 18,000 unique images across a four-dimensional pattern parameter space, sampled quasi-randomly via a Sobol sequence. Human search trial data collected from these stimuli was used to train a logistic regression model and a Gaussian process classifier. Both models successfully recovered ordinal structure in the parameter space. In the validation experiment, the rank ordering predicted by the GP classifier held for first-exposure data across all four candidate patterns, with pairwise performance differences reaching significance in the predicted direction. To our knowledge, this is the first study to map camouflage pattern performance across a parameter space of this scale in operational context using human detection metrics.

The parameter results themselves yielded a coherent account of what makes a pattern difficult to detect in the tested environment. Fine-grained, isotropic noise at high patch density (P1) had the strongest effect on detectability, with horizontal stretch (P2) providing a secondary contribution. Feature masking (P4) showed an asymmetric effect driven by vehicle geometry, pointing to geometry-based modification as a partially separable axis from pattern-level optimization. Grain (P3) showed no reliable effect within the range tested.

Unexpected findings also raised broader questions for future research. Participant performance heterogeneity influenced the apparent optimal pattern outcome, pointing to the value of professional participant selection or explicitly training observers before or during data collection. Separately, vehicle geometry emerged as an independent contributor to detectability, one that may rival pattern-level camouflage in its effect on visual signature.

The pipeline and remote evaluation program created to enable this study are already being extended to other signature assessment experiments at TNO. With the methodological refinements identified in the recommendations, parameter-space mapping with human-

in-the-loop data is a practical route toward evidence-based camouflage design and more broadly, toward any design problem that can be parametrized and evaluated through human response. Ultimately, this work demonstrates that human perception, not just computational proxies, can serve as the guiding signal in a systematic, scalable search through design space.

Acknowledgements

To my friends and family: thank you for always supporting me, believing in me and inspiring me. For sharing the best times and the worst times. I am very lucky to have you all in my life.

I want to thank Neana, Victor, Viktor, Sophie and Robin for always being available on short notice and being excited to participate when I was running my many (pilot) experiments. Without you this thesis would be in a very different shape.

A special thank you goes out to Paul for sharing my passion and excitement for making sense of whatever is in front of me and always testing my reasoning, and to Mees for early proofreading.

Of course, the great guidance of my supervisors Sylvia Pont and Maarten Wijntjes cannot go unmentioned, as well as the invaluable insight from Maarten Hogervorst during our many meetings and interactions at TNO, that played an important role in shaping the explorations, final direction of this thesis and my understanding of the world of camouflage.

I also want to thank Jo Culpepper, who I had the pleasure of having incredibly insightful discussions as well as many laughs with at TNO. It's hard to describe why, but your presence made me feel at home.

Last but certainly not least I **need** to thank my amazing girlfriend Belle, for enduring countless months of a never-ending waterfall and endless meandering of thoughts concerning this project (really, this only barely qualifies as an exaggeration), keeping me on track, and always being more caring than anyone could ever ask for. I really can't thank you enough.

References

Bacon-Macé, N., Macé, M. J.-M., Fabre-Thorpe, M., & Thorpe, S. J. (2005). The time course of visual processing: Backward masking and natural scene categorisation. *Vision Research, 45*(11), 1459–1469. https://doi.org/10.1016/j.visres.2005.01.004

BD3D. (2024). Geo-Scatter: The #1 scattering plugin for Blender (Version 5.6.2) [Software]. https://geoscatter.com/

Blender Online Community. (2025). Blender — A 3D modelling and rendering package (Version 5.0) [Software]. Blender Foundation. https://www.blender.org/

Barnett, J. B., Cuthill, I. C., & Scott-Samuel, N. E. (2017). Distance-dependent pattern blending can camouflage salient aposematic signals. *Proceedings of the Royal Society B: Biological Sciences, 284*(1858), 20170128. https://doi.org/10.1098/rspb.2017.0128

Culpepper, J. B., Richards, N., Madden, C. S., Winter, N., & Wheaton, V. C. (2017). Comparing synthetic imagery with real imagery for visible signature analysis: Human observer results. *Proceedings of SPIE, 10432*, 104320H. https://doi.org/10.1117/12.2277580

Gulrez, T., Culpepper, J. B., Phung, S. L., & Le, H. T. (2024). VSAI: Visible signature reduction with camouflage patterns via generative AI algorithm. In *Proceedings of SPIE: Target and Background Signatures X: Traditional Methods and Artificial Intelligence* (Vol. 13199, Article 1319904). SPIE. https://doi.org/10.1117/12.3034275

Gardner, J. R., Pleiss, G., Bindel, D., Weinberger, K. Q., & Wilson, A. G. (2018). GPyTorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. Advances in Neural Information Processing Systems, 31, 7576–7586. https://proceedings.neurips.cc/paper_files/paper/2018/hash/27e8e17134dd7083b050476733207ea1-Abstract.html

Hancock, G. R. A., Cuthill, I. C., & Troscianko, J. (2026). Shining a light on camouflage evolution: Using genetic algorithms to determine the effects of geometry and lighting on optimal camouflage. *PLOS ONE, 21*(4), e0346231. https://doi.org/10.1371/journal.pone.0346231

Kucherenko, S., Albrecht, D., & Saltelli, A. (2015). *Exploring multi-dimensional spaces: A comparison of Latin Hypercube and Quasi Monte Carlo sampling techniques*. arXiv. https://arxiv.org/abs/1505.02350

Nguyen, T. T. P., Gulrez, T., Culpepper, J. B., Le, H. T., & Phung, S. L. (2025). CamoX: A diffusion-based method with few-shot learning for environment-guided camouflage pattern generation. *IEEE Transactions on Artificial Intelligence*. https://ieeexplore.ieee.org/document/11159253

Peak, J. E., Hepfinger, L., Balma, R., Christopher, G., Fleuriet, J., Honke, T., Huebner, G., Mauer, E., Dotoli, P., Ronconi, P., & Jacobs, P. A. M. (2006). *Guidelines for camouflage assessment using observers* (RTO-AG-SCI-095). NATO Research and Technology Organisation.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830. https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf

Peirce, J. W., Gray, J. R., Simpson, S., MacAskill, M. R., Höchenberger, R., Sogo, H., Kastman, E., & Lindeløv, J. (2019). PsychoPy2: Experiments in behavior made easy. Behavior Research Methods, 51, 195–203. https://doi.org/10.3758/s13428-018-01193-y

Penacchio, O., Lovell, P. G., Ruxton, G. D., Cuthill, I. C., & Harris, J. M. (2014). Countershading camouflage: Exploiting photons to break shape-from-shading inference. *i-Perception, 5*(5), 471–471. https://doi.org/10.1068/ii40

Penacchio, O., Lovell, P. G., & Harris, J. M. (2017). Establishing the behavioural limits for countershaded camouflage. *Scientific Reports, 7*, 13672. https://doi.org/10.1038/s41598-017-13914-y

Penacchio, O., Lovell, P. G., & Harris, J. M. (2018). Is countershading camouflage robust to lighting change due to weather? *Royal Society Open Science, 5*(2), 170801. https://doi.org/10.1098/rsos.170801

Penrose, R. (1941). Home Guard manual of camouflage. Lee Miller Archives Publishing.

polygoniq. (2024). botaniq: Vegetation asset library for Blender (Version 7.1.1) [Software]. https://polygoniq.com/3d/botaniq/

PyInstaller Development Team. (2024). PyInstaller: Freeze (package) Python programs into stand-alone executables (Version 6.18.0) [Software]. https://pyinstaller.org/

Schwegmann, A. (2023). Evaluation of dual attribute adversarial camouflage and counter-AI reconnaissance methods in terms of more realistic spatial alignment. In *Target and Background Signatures IX* (Proc. SPIE Vol. 12736, Paper 1273603). SPIE. https://doi.org/10.1117/12.2679322

Seabold, S., & Perktold, J. (2010). Statsmodels: Econometric and statistical modeling with Python. Proceedings of the 9th Python in Science Conference (SciPy 2010), 92–96. https://doi.org/10.25080/Majora-92bf1922-011

Sobol, I. M. (1967). Distribution of points in a cube and approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, *7*(4), 86–112.

Stevens, M., & Merilaita, S. (2009). Defining disruptive coloration and distinguishing its functions. *Philosophical Transactions of the Royal Society B: Biological Sciences, 364*(1516), 481–488. https://doi.org/10.1098/rstb.2008.0216

Talas, L., Fennell, J. G., Kjernsmo, K., Cuthill, I. C., Scott-Samuel, N. E., & Baddeley, R. J. (2019). CamoGAN: Evolving optimum camouflage with Generative Adversarial Networks. *Methods in Ecology and Evolution*. https://doi.org/10.1111/2041-210X.13334

Toet, A., & Bijl, P. (2003). Visual search. In R. G. Driggers & E. Lichtenstein (Eds.), *Encyclopedia of optical engineering*. Marcel Dekker.

Toet, A., Hogervorst, M. A., & Bijl, P. (2004). Conspicuity and identifiability: Efficient calibration tools for synthetic imagery. *Proceedings of SPIE, 5431*, 1–12. https://doi.org/10.1117/12.539118

Van Beem, A. (2012). *Standcam photo8* [Photograph]. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Standcam_photo8.JPG

Van der Burg, E., Hogervorst, M. A., & Toet, A. (2023). Evolving camouflage. *Proceedings of SPIE, Target and Background Signatures IX*. https://doi.org/10.1117/12.2679515

Van der Sanden, K., Hogervorst, M. A., & Bijl, P. (2022). Hybrid simulation for creating realistic scenes for signature assessment. *Proceedings of SPIE, 12270*, 1227008. https://doi.org/10.1117/12.2638886

Van der Sanden, K., Hogervorst, M. A., & Bijl, P. (2023). Hybrid simulation ray-tracing improvements for creating realistic scenes for signature assessment. *Proceedings of SPIE, 12736*, 1273606. https://doi.org/10.1117/12.2680348

World Medical Association. (2013). World Medical Association declaration of Helsinki: Ethical principles for medical research involving human subjects. *JAMA, 310*(20), 2191–2194. https://doi.org/10.1001/jama.2013.281053

Appendix

Appendix A: Study Development

Determining the angle and range used for the experiment in this thesis was a long process of trial and error. At least 17 mini-pilot experiments were conducted with 1-7 participants, where different combinations of perspective as well as lighting conditions were explored. Additionally, during this time the camouflage pattern system underwent iterations with increasingly more parameters being introduced, redundant parameters being identified and subsequently excluded. The evaluator program also underwent testing and development, going from a simple task, only requiring spacebar to be pressed, to the final version with false positive filtering and all the instructions surrounding the experiment.

All in all, this process of trial and error informed and shaped the final experiment. This section gives insight in what that iteration process was like, although it is not necessarily linear or complete. The examples shown here may paint a picture of why certain approaches were not taken, or give ideas for alternative valid directions to explore in further research.

Texture mapping

Mapping 2D textures onto a three-dimensional object is a common challenge in creating representative stimulus images. This challenge is often solved by simply filling the silhouette of the target with the 2D texture, thereby ignoring the surface geometry of the object and producing a flat appearance as described in Section 1.1.

Planar and other projection methods address this partially by projecting a texture onto the surface, but introduce visible distortion on faces that deviate from the angle of the projection plane and is noticeable at the seams of an object where mappings don’t align and create a noticeable edge (Figure A1).

Figure A1. Planar projection example on STANDCAM. Notice the seams on the edge of the front of the vehicle.

UV unwrapping resolves distortion by explicitly defining a map between surface points and texture coordinates, but requires per-asset attention and is time-consuming, demands familiarity with the unwrapping process to avoid seams, artefacts or inconsistent scaling, and introduces a potential source of error: a poorly mapped texture may compromise the representativeness of the stimulus, interfering with the evaluation of camouflage effectiveness. Time consuming as it may be, it is an often-used method because it can deliver accurate results if done correctly.

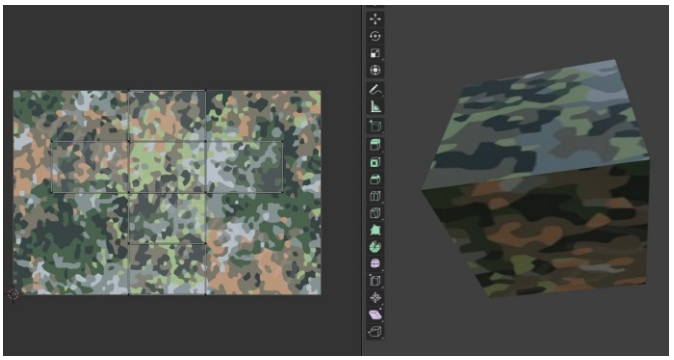


Figure A2. A UV unwrapped cube. An intuitively simple example of a problem that becomes exponentially harder to solve properly as geometric complexity increases. Even in this example it should stand out that edges are not connected seamlessly.

The camouflage patterns in this study were generated instead using object-space coordinate-driven textures in Blender. Rather than mapping a 2D image onto the surface, texture values are evaluated at each surface point based on its position within the object's local coordinate space. This is conceptually analogous to carving a shape from a solid block of material: the texture exists throughout the volume, and the visible surface pattern emerges naturally from wherever the geometry intersects that volume.

Consequently, no manual mapping is required, and the approach generalizes across objects without per-asset adjustment.

Figure A3. Body panels on STANDCAM showing perfect texture mapping using coordinate-driven textures.

Pairwise comparison

After implementing object-space coordinate-driven textures in Blender and using the node tree parameters to create texture variations, the idea of randomizing these parameters arose. A simple Python script within Blender proved to be able to do these randomizations of parameters.

A proof of concept was developed in which two versions of the camouflage pattern were shown side-by-side (see figure A4), each built by different parameter combinations. The user was asked to identify the better camouflaged target, after which the worse candidate would be overwritten with a new camouflage pattern drawn from the winning pattern's parameters with bounded random mutation. This king-of-the-hill style pairwise comparison was the first human-in-the-loop optimization technique explored in this project.

Figure A4. Pairwise comparison proof of concept screenshot.

Ultimately, this approach was deemed too time-consuming. Several iterations were explored, including pairwise comparison optimization algorithms to select the most informative comparisons rather than the king-of-the-hill approach. However, limitations remained: optimization could only be done for one scene at a time and participants had little control over the optimization direction. Giving direct access to parameter values may have been more time-effective, particularly at the start when obvious comparisons have to be shown to establish which regions of parameter space are not worth exploring.

Furthermore, a limitation was identified where participants may be biased toward selecting for what they believe a camouflage pattern should look like since no objective measure of performance is being given. Notably, this bias is not resolved by giving participants direct access to parameter sliders either: steering a continuous parameter space without feedback still relies on the participant's intuition of what effective camouflage looks like. This mirrors the researcher-level selection bias discussed in Section 1.1, where candidate patterns are handpicked based on prior expectations rather than measured performance. Additionally, this approach relied on real-time rendering, requiring adequate hardware and hindering remote participation and thus data collection.

Perspective

Experimentation was done with various perspectives when exploring the capabilities of synthetic simulations. When creating the first search experiment however, a mostly top down perspective (figure A5) was used for a few reasons. First, it meant not needing to create a horizon, which is a relatively hard task both from a modelling perspective as well as computationally. The bigger a scene is the more assets need to be loaded and thus the longer renders take and the more sluggish Blender becomes to work with. Second, this perspective could be used to optimize camouflage against drone perspectives, an increasingly relevant threat.

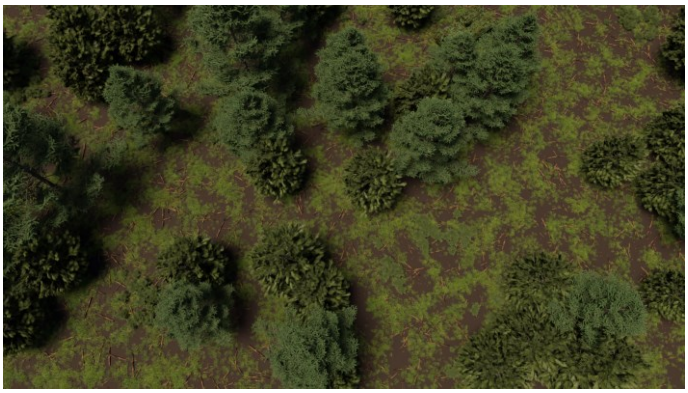


Figure A5. A stimulus image from the first search experiment, showing a compact scene with only a few dozen trees. This image shows a well-hidden target (top, center-right) which showed a clear performance difference compared to the other two camouflage patterns tested in this early pilot.

Results from this early test were encouraging, showing differences in performance between three manually selected parameter combinations, consistent between participants and matching with intuition.

Larger scenes were tested to get a more representative perspective of what a drone operator may encounter when flying over an area. During this stage of testing more light variations were experimented with as will be discussed in the next sub-section. Additionally, a better understanding of GeoScatter was gained with optimizations in the design process thanks to features enabling the dense foliage only during rendering. This made it possible to make these larger scenes as seen in Figure A6.

Figure A6. A larger scene containing many more foliage assets than before.

This building of an understanding of optimization features continued, enabling even larger scenes at angles showing the horizon. In figure A7 this is clearly visible, in order to render thousands of trees at a time a secondary plane was put into place here where the target could not appear. In this case the tree assets in the distance were heavily simplified 2D copies of the tree assets seen in the foreground. They can be thought of as cardboard cut-outs. Using marginal resources compared to the full 3D assets in the front. This allowed even the background to be randomized from scene to scene. A more pragmatic approach could have been to pre-render the background and use that one static image between all scenes.

Testing using these environments however revealed a very limited ability to discern any camouflage pattern characteristics. Detection times were a product of participant performance, the vehicles geometry, its location in the environment and a good amount of chance. Targets became identifiable primarily by their cast shadow. Therefore, this approach was deemed not suitable for search-task camouflage pattern optimization.

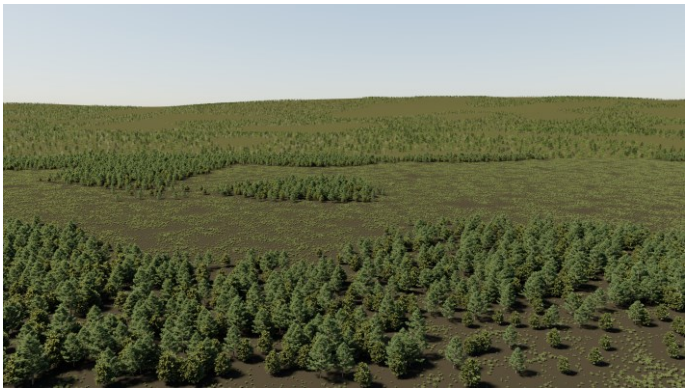


Figure A7. A large synthetic scene showing many trees in both foreground (high fidelity) and background (low fidelity).

After returning to a reduced environment scale, subsequent creations focused on implementing and expanding on lessons learned about scatter systems. Figure A8 shows improvements over Figure A6 in this department. It includes more variety in assets used on multiple levels: there are more types of assets like boulders and logs, there are more asset variations per category (5 unique tree geometries compared to 2), within these categories color variation was applied (2 shades of grass), and assets were used more efficiently, where before clumpy bush assets were present, here grass assets were used and stretched out in the horizontal plane, yielding full scene coverage of 3D grass with a reasonable polygon count.

Figure A8. An improved stimulus image showing more realistic scattering and variety of assets compared to earlier iterations.

This then brought us to the final change in perspective where a ground level camera was finally decided upon since it gives the target more interesting opportunities for blending in. The thought behind this was that the target could blend in with the trees as a background as well as with grassy sections. Creating potential for local optima in the parameter space, these optima were not identified in the final results however. Although it cannot be said with certainty that they are not there. More analysis and perhaps cleaner data is necessary to draw such conclusions.

Figure A9. A stimulus image from the perspective used in the main experiment.

Lighting

Throughout the previously shown environment iterations, different lighting scenarios were tested. Both in lighting direction as well as different levels of light diffusion using the sun-size value in Blender.

It was expected that a smaller sun size (direct/non-diffuse lighting) would result in lower detection times, because a cast/internal shadow with more contrast would stand out more. Interestingly the opposite appeared to be true for the first environment this was tested on. In figure A10, two identical, cropped scenes are shown with the target in diffuse and direct lighting. In this pilot, including only a few participants, average search times were longer for the direct lighting scenario. This may be due to the cast shadow of the target, matching the cast shadow of some of the bushes in length and thus not standing out.

Figure A10. Diffuse lighting (top) and direct lighting (bottom).

Naturally, this meant testing of lighting conditions had to be done on a per-environment basis. This is where further realism improvements were made and tested, including gobos (shadows cast by 2D clouds) pictured in figure A11, as well as volumetric fog.

Figure A11. A stimulus sample showing realistic lighting with a noticeable cloud shadow cast over the center-left of the scene.

Some of these final touches of realism did not make it into the final experiment stimulus images. This was a deliberate design choice, since for these scenes direct light did strongly influence detection times and made it harder to separate camouflage performance from scene influence. The shadows cast by gobos and trees also made detection incredibly difficult when targets were located inside them, further tying outcome of trials to covariate conditions. Therefore, a lighting scenario was ultimately chosen which minimized these influences and maximized the effect the parameters could have on trial outcomes. However, as discussed in section 2.1, this comes at the cost of optimization being biased toward diffuse lighting scenarios.

Beyond the previous reasons, volumetric fog and clouds were not included since they significantly increased render time. For the amount of stimulus necessary in the given timespan of the project and the computation available this was not feasible.

Figure A12. A stimulus scene used in the final experiment.

Appendix B: Evaluator information screens

Experiment Instructions

Please read carefully

TNO

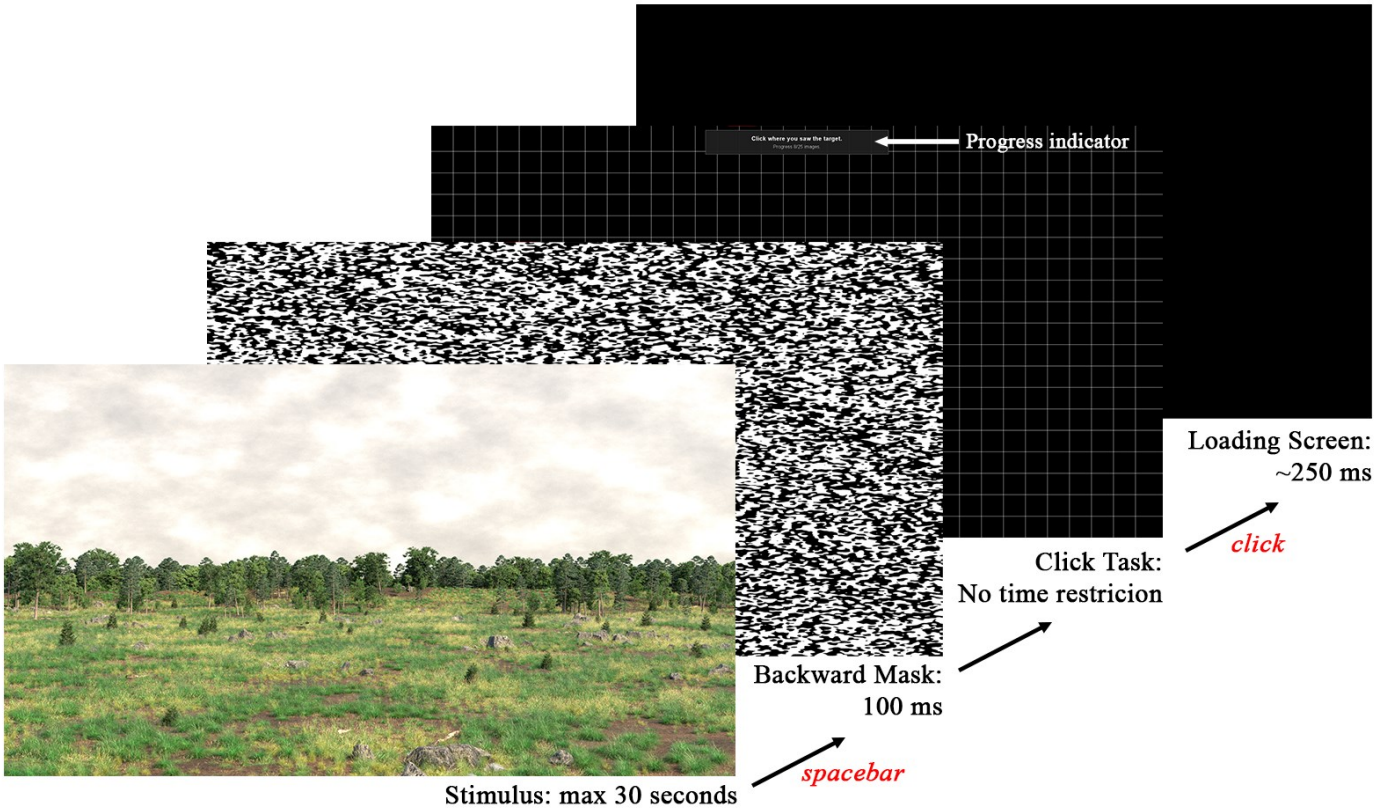
Assignment:

- Try to find the vehicle in 30 seconds.
- When you see it press the space bar.
- After pressing space, click where you saw the vehicle. *No rush, there is no time restriction on this part of the task.*
- If you accidentally press space bar, please click somewhere near the top or bottom of the screen so we can filter these instances out.
- If this is your first time a practice set will be shown before you start the main experiment.

Important to know:

- Each sample you see will be of a different scene with a unique camouflage pattern applied to the vehicle. The shape of the vehicle does not change.
- You will be given breaks, don't hesitate to use them. If you decide to quit and continue the experiment at a later time your progress will be saved.
- By holding ESC for 5 seconds the program will quit and your progress will be saved.

Front and back of the vehicle with a camouflage pattern applied



Appendix C: Generation comparison

At first glance camouflage performance seems to be lower on the second generation as can be seen in figure C1. With all metrics showing worse or equal camouflage performance. However, this may be accounted for by the fact that participants who participated in the first generation were improving at the search task, which are further discussed in section 4.2 Learning effect. Additionally, participants who experienced most difficulty with the task were less inclined to partake in the second generation, introducing a self-selection bias toward better-performing participants.

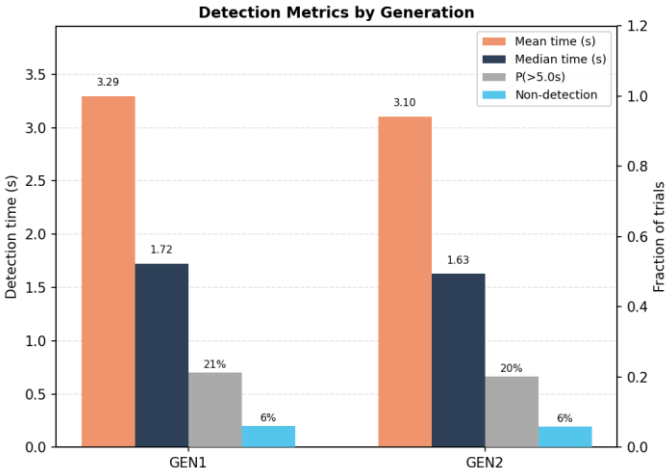
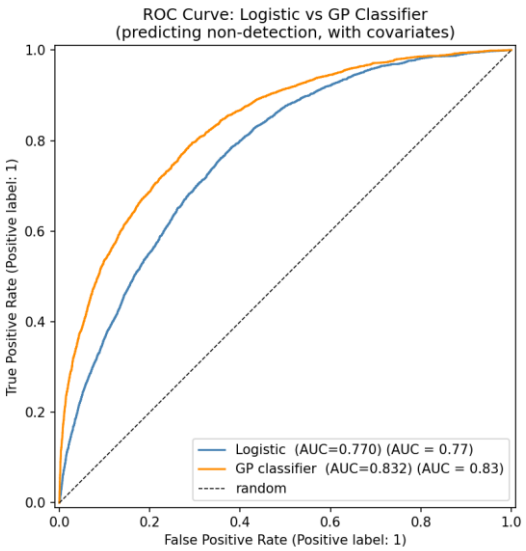


Figure C1. Detection metrics per generation.

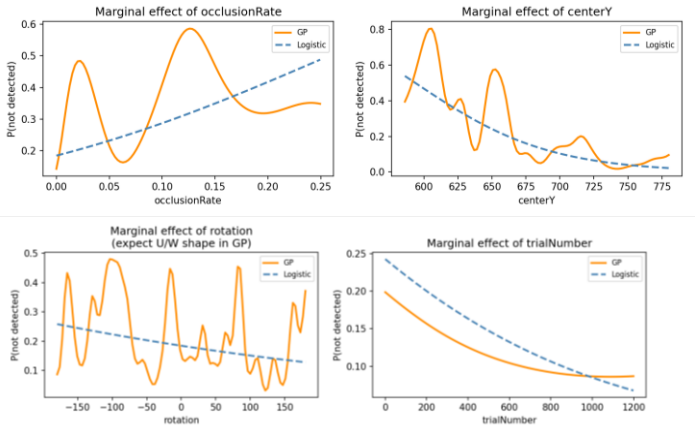
Appendix D: Hyper parameter tuning

A broad combination of hyperparameters (>250) were evaluated, ranging from 100–1600 inducing points, 100–1500 epochs, and learning rates from 0.01–0.1. While AUC's as high as 0.84 were found along with AUC standard deviations for the 5 folds used around 0.008 even for the higher performing models, it was clear that these higher accuracies were only possible because the model was overfitting on the covariates. Usually this would be caught by cross-validation, but since each scene has identical covariates which have a disproportionate effect on detection time and outcome, it was highly effective for the model to fit that part of the data instead of the camouflage parameters, leading to undescriptive design parameter marginals which would occasionally even disagree with the logistic fit entirely.

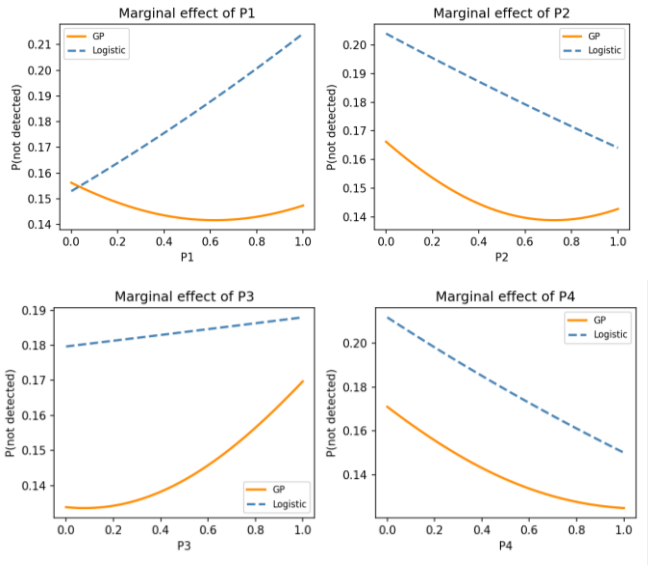


Graphs from 5s_ind1500_ep1600_lr0.02:

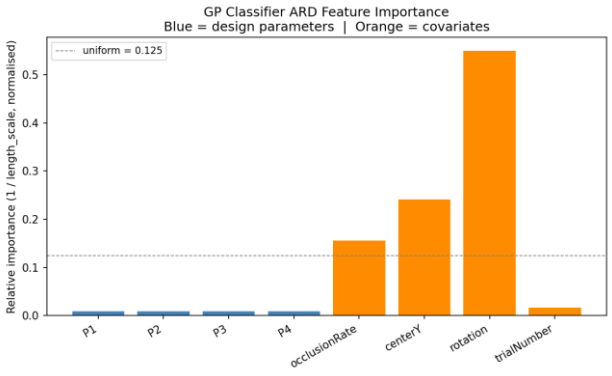
Here we see high classifier performance with low variance, suggesting a good combination of hyper parameters.



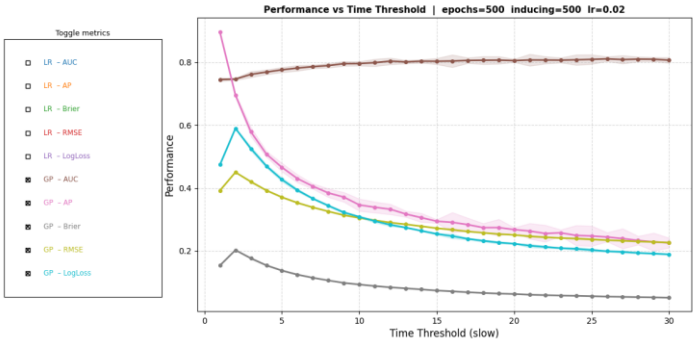
We see a lot of detail in the covariate marginals, particularly for centerY and rotation. Upon closer inspection the occlusion rate is peculiar as well, it looks smooth but doesn't show a logical pattern, more occlusion should always lead to higher detection difficulty. This suggests overfitting, but could be mistaken for genuine accuracy due to the low variance between folds.



The design parameter graphs for this hyperparameter combination show behavior which is inconsistent with that of more conservative hyperparameters. The disagreement between GP and logistic regression on P1 immediately stands out and that of P4 also doesn't match the plateau and subsequent steep drop off that can be seen, even in the binned data representation.



The ARD feature importance graph confirms that nearly all importance has been placed in the scene covariates. Memorizing them proved more effective and was not caught due to duplicate scenes spread through the validation folds.



Appendix E: Participant information

TNO innovation for life

Informatiebrief

Optimalisatie Ontwerp Voertuigcamouflages

Wie zijn wij?

Wij zijn onderzoekers van TNO, een onderzoeksorganisatie voor toegepast onderzoek. Een groot deel van het onderzoek dat bij TNO Soesterberg wordt uitgevoerd, is voor het Ministerie van Defensie en onderzoekt de rol van mensen en de interactie tussen mensen en technische systemen.

Wat onderzoeken wij?

In de huidige studie willen we menselijke doel-detectie van voertuigen met verschillende camouflage patronen beoordelen en deze menselijke zoekprestaties gebruiken om nieuwe camouflage patronen te ontwerpen.

Wat moet je doen? En wat doen wij?

- Je moet een computertaak uitvoeren waarbij een voertuig in een omgeving wordt getoond.
- Bedoeling is om zo snel mogelijk het voertuig te vinden in de omgeving en vervolgens op spatiebalk te klikken. Hiermee meten we de detectietijd.
- Na het meten van de detectietijd verandert het scherm kort naar een witte ruis afbeelding en is het de bedoeling om te klikken waar je het voertuig dacht te zien. Zo kunnen wij controleren of je het voertuig inderdaad gezien hebt.
- Je deelname duurt 30 tot 60 minuten. Elke van de 6 series zoekopdrachten duurt ongeveer 7 minuten, je voortgang is ondertussen te zien. Tussen de series kan je pauzes nemen, zo lang je wilt.
- Specifieke instructies zullen getoond worden in het onderzoeksprogramma.
- We zullen de detectietijden gebruiken om betere camouflage patronen te genereren.

Hoe neem je deel?

- Na het uitvoeren van het zoek experiment stuur je het (csv) bestand met je resultaten naar de naar coen.duijnhouwer@tno.nl .
- Je kunt vrijwillig beslissen of je wilt deelnemen. Zelfs nadat je hebt besloten deel te nemen, kun je je deelname op elk moment stoppen. Je resultaten hoef je dan ook niet op te sturen.

Wat zijn de inclusie-/exclusiecriteria?

- Je moet een normaal of gecorrigeerd normaal gezichtsvermogen hebben.
- Je mag geen kleurenzichtdeficiënties (kleurenblindheid of gestoord kleurenzicht) hebben.
- Je moet minstens 18 jaar oud zijn.
- Je mag geen bekende aandachtsproblemen, zoals aandachtstekortstoornis (ADHD / ADD) hebben.

Wat gebeurt er met de gegevens na afloop van het onderzoek?

- De gegevens worden bewaard op een beveiligde TNO-server.

Waarom zou je deelnemen?

- Je helpt ons beter te begrijpen hoe toegepaste camouflage presteerd in verschillende omstandigheden.
- Door ons te helpen in deze studie help je de overlevingskansen van onze soldaten te verbeteren.

Hoe beschermen wij je privacy?

- We gebruiken persoonlijke informatie zoals naam en e-mailadres alleen voor wervingsdoeleinden en voor het toestemmingsformulier.
- De gegevens worden geanonimiseerd, d.w.z. de persoonsgegevens worden vervangen door een nummer. De verbinding tussen de nummers en de persoonlijke informatie wordt apart bewaard en is alleen toegankelijk voor de TNO-onderzoekers die bij deze studie betrokken zijn.
- Deelnemersinfo w.o. naam en emailadres worden 2 jaar na het onderzoek vernietigd.
- Voor onderdelen niet vermeld in dit document wordt verwezen naar de algemene regels en afspraken die door TNO zijn vastgelegd om je privacy te beschermen (zie deze link).
- Het onderzoek valt onder de TNO verzekering.

Contact: Coen Duijnhouwer coen.duijnhouwer@tno.nl
Supervisor: Maarten Hogervorst maarten.hogervorst@tno.nl

Appendix F. Informed consent form

Informed consent / toestemmingsverklaring

Ondergetekende,

Naam

Geboortedatum

verklaart op vrijwillige basis deel te nemen aan het onderzoek, getiteld "**Optimalisatie Ontwerp Voertuigcamouflages**" bij TNO.

☒ Ik bevestig dat ik de informatie over bovengenoemd onderzoek heb gelezen en de informatie heb begrepen.

☒ De bedoelingen van het experiment en de daarbij gevolgde aanpak zijn tot mijn tevredenheid uitgelegd.

☒ Ik heb de gelegenheid gehad om aanvullende vragen te stellen en deze vragen zijn naar tevredenheid beantwoord.

☒ Ik heb voldoende tijd gehad om over deelname na te denken.

☒ Ik weet dat mijn deelname aan het onderzoek geheel vrijwillig is en dat ik mijn toestemming op ieder moment kan intrekken zonder dat ik daarvoor een reden hoef op te geven.

☒ Ik geef toestemming om mijn persoonsgegevens te verwerken voor de doelen zoals beschreven in de informatie.

☒ Ik geef toestemming om mijn onderzoeksgegevens te hergebruiken voor toekomstig onderzoek op het beschreven onderzoeksgebied op voorwaarde dat deze zo gecodeerd zijn, dat ze niet meer naar mij als persoon terug te leiden zijn.

☒ Ik geef toestemming voor het bewaren van de gegevens en dat bevoegde leden van het onderzoeksteam en bevoegde inspecteurs hier inzage in hebben. Ik heb de mogelijkheid om deze toestemming op een later moment alsnog in te trekken, waarbij al gepubliceerde data niet meer kunnen worden aangepast

Voorts verklaar ik geen mij bekende belemmeringen te hebben om aan het experiment deel te nemen.

Plaats, datum

Handtekening proefpersoon:

TOELATING
Ik heb me ervan vergewist dat ik deze proefpersoon goed geïnformeerd heb over het onderzoek waaraan hij/zij gaat deelnemen. Ik heb mij ervan overtuigd dat deze proefpersoon voldoet aan de selectiecriteria om aan bovengenoemd onderzoek deel te mogen nemen.

Naam proefleider: _____

Plaats, datum _____

Handtekening proefleider: _____