On the use of simulated data for target recognition and mission planning

Friso G. Heslinga[a], Ella P. Fokkinga[a], Thijs A. Eker[a], Alma M. Liezenga[a], Richard J.M. den Hollander[a], Victor O. Oppeneer[a], A. M. van Heteren[a], Robbert van Vossen[a], Hugo J. Kuijf[a], Jeroen J.M. van de Sande[a], Dennis W. van der Burg[a], Louis F. Weyland[a], Henk C. Henderson[a], Maarten P.D. Schadd[a], and Klamer Schutte[a]

[a]TNO - Defence, Security and Safety, Oude Waalsdorperweg 63, the Hague, the Netherlands

ABSTRACT

Artificial intelligence (AI) models are at the core of improving computer-assisted tasks such as object detection, target recognition, and mission planning. The development of AI models typically requires a large set of representative data, which can be difficult to acquire in the military domain. Challenges include uncertain and incomplete data, complex scenarios, and scarcity of historical or threat data. A promising alternative to real-world data is the use of simulated data for AI model training, but the gap between real and simulated data can impede effective transfer from synthetic to real-world scenarios. In this study, we provide an overview of the state-of-the-art methods for exploiting simulation data to train AI models for military applications. We identify specific simulation considerations and their effects on AI model performance, such as simulation variation and simulation fidelity. We investigate the importance of these aspects by showcasing three studies where simulated data is used to train AI models for military applications, namely vehicle detection, target classification and course of action support. In the first study, we focus on military vehicle detection in RGB images and study the effect of simulation variation and the combination of a large set of simulated data with few real samples. Subsequently, we address the topic of target classification in sonar imagery, investigating how to effectively integrate a small set of simulated objects into a large set of low-frequency synthetic aperture sonar data. We conclude with a study on mission planning, where we experiment with the fidelities of different aspects in our simulation environment, such as the level of realism in movement patterns. Our findings highlight the potential of using simulated data to train AI models, but also illustrate the need for further research on this topic in the military domain.

Keywords: Simulation, Synthetic data, Deep learning, Object detection, Target classification, Mission planning

1. INTRODUCTION

The rapid rise of artificial intelligence (AI) is supporting the improvement of computer-assisted tasks across numerous fields. In the military context, AI models are anticipated to be integrated into operational systems for a wide range of applications, such as detection of military vehicles,[1] detection of mines,[2] operation of autonomous systems,[3] and mission planning.[4] Development of AI models typically requires large sets of representative data, which can be challenging to obtain in the military domain.

Data accessibility is typically restricted due to its sensitive nature, leading to a scarcity of historical and threat data. Additionally, the time-critical and sensitive dynamic of military operations leads to uncertain and incomplete data, especially when dealing with complex scenarios. Target signatures are subject to change, either due to technological developments or change in doctrine, meaning that even when a large dataset is collected, it might not be representative for new missions.

An alternative to acquiring data is using simulation to generate data, for example by physics-based modeling of the sensor and scenario of interest. Data simulation can be used to generate diverse and extensive datasets, allowing for controlled acquisition variables and creation of novel and complex scenarios.[5,6] Recent work has shown the potential of using simulated data to train AI models in several modalities, including natural photo

Send correspondence to Friso G. Heslinga. E-mail: friso.heslinga@tno.nl

Artificial Intelligence for Security and Defence Applications II, edited by Henri Bouma, Radhakrishna Prabhu, Yitzhak Yitzhaky, Hugo J. Kuijf, Proc. of SPIE Vol. 13206, 132060M · © 2024 SPIE
0277-786X · doi: 10.1117/12.3031638

and video,[1,7–9] infrared (IR),[10–13] radar,[14,15] and sonar.[16–18] Furthermore, simulations can be used to develop AI-based tools for mission planning.[4,19]

Despite this potential, creating simulations that accurately represent real-world conditions remains challenging. One of the primary issues is the "reality gap" - the disparity between simulated and real-world data - which can hinder the effective transfer of a model trained on simulated data to real-world applications.[5] To address this, researchers have focused on investigating the reality gap, by studying the fidelity of simulators,[20] or by directly comparing data from both domains. For instance, studies have evaluated the appearance of raw real and simulated data[21] or the variations in feature space embeddings learned by the AI models during training.[21,22]

Nevertheless, the challenge of simulating datasets that enable AI models to perform well on real-world data remains an open challenge. The key topics addressed in this paper are related to simulation variation and simulation fidelity. With simulation variation, we aim to include all of the variations that occur in the distribution of measured data. In contrast, simulation fidelity refers to the distance between the simulated and measured data distribution. High-fidelity simulated data has a small distance to measured data, and the challenge is how to measure the fidelity.

In addition, we investigate the value of combining simulated data with measured data for AI model training. While simulated data provides a source that is plentiful and allows for all kind of variations and modification of threat signatures, measured data can offer details that support AI model development where simulation fidelity is lacking. Combinations of both sources are implemented by (1) injecting simulated threats into measured environments and (2) mixing few real samples with a large amount of simulated data.

In this paper, we study the use of simulated data to train AI for applications and possible integration into the military operational domain. First, an overview is provided of the relevant literature on applications and methods (Section 2). In Section 3, we discuss how to bridge the reality gap through simulation variation and fidelity, explaining the reasoning behind the selected experiments. Then, we present use cases on military vehicle detection (Section 4), threat recognition in sonar (Section 5), and decision support for mission planning (Section 6). Here, we experimentally investigate the effect of simulation variation and fidelity. We conclude in Section 7, with a general discussion and an overview of the lessons learned.

2. RELATED WORKS

Simulated data provides a valuable resource when dealing with a lack of high-quality, annotated training data in the context of developing AI models. The creation and use of simulated data can save time and resources in the acquisition and annotation process, offers representation of a more diverse set of scenarios, and helps reduce bias and privacy concerns.[5,6] Simulated data has proven its value in numerous machine learning tasks,[5] but the use of simulated data also presents challenges, such as reaching photo-realism, data diversity, and bridging the gap between real and simulated data.[5] This section will provide an overview of the increasing use of simulated data in the military domain, the different techniques for generating simulated data, and the reality gap.

2.1 Simulated data in the military domain

Particularly in the military domain, using simulated data can resolve issues regarding data accessibility and confidentiality. Several recent publications have demonstrated the potential of using simulated data to train AI models for tasks in the military domain. Most prominently, models have been developed for relevant object detection tasks, such as military vehicle detection,[1,8,23,24] automatic target recognition (ATR),[11,25,26] navigation of unmanned aerial vehicles (UAV),[27,28] and mission planning.[29–31] This work covers RGB, IR,[10,11,13,26,32] and multi-modal, acoustic data.[27] Other works have focused on the development of simulated data for mission planning[33,34] and the maritime domain.[13,26]

Several studies have demonstrated the viability of training solely on synthetic images and evaluating on real images. In one of the first works on this, Moate et al. (2018) trained a CNN using only IR imagery generated from computer-aided design (CAD) vehicle models using a rendering tool.[10] Westlake et al. (2020) focused on the generation of a synthetic IR dataset of maritime vessels to train for an ATR task and demonstrated that high average precision and recall could be reached on real-world data despite training solely with synthetic imagery.[26] Ruis et al. (2024) demonstrated the feasibility of training solely on simulated images,[1] developed best practices and showed the value of using a Transformer architecture and data augmentation methods for this purpose. Alternatively, Spell et al. (2024) trained their models on a combination of real and simulated IR data to perform vehicle and person detection, object identification, and vehicle parts segmentation.[32] They concluded that training on a combination of real and simulated imagery performs better than training solely on real data.

Lastly, de Melo et al. (2023) add a simulated dataset to the real training data for an ATR task and find that this boosts their performance significantly.[25] McKenzie et al. (2024) focus on acoustic sensing data, using also a combination of real-world and synthetic data and thus demonstrating a proficient capability in ATR for use in UAV.[27] Xian et al. (2024) also experiment with real, synthetic and combined datasets but for the task of real-time human recognition in UAVs and conclude that the combination of both performs best.[28] Finally, Vo et al. (2024) incorporate synthesized TIR data for a IR maritime classification task.[13]

2.2 Methods for generating synthetic data

Several approaches exist for generating simulated and synthetic datasets. Traditionally, many follow a physics-based approach, where laws of physics are used to replicate behavior of systems and physical phenomena in the real world.

Several tailor-made physics-based simulations have been designed and evaluated specifically for the purposes of gathering synthetic datasets for military purposes. As an example, Kenul et al. (2024) introduced a framework that produces synthetic imagery that closely resembles real-world maritime data[35] and Polat and Özer (2022) designed an approach using 3D rendering to simulate radiometric EO/IR data.[36] As another example, Williams et al. (2023) presented a customizable tool to generate large synthetic video datasets for aerial threat detection.[37] Uplinger et al. (2023) presents a solution based on a simulator that supports generation of physically accurate custom synthetic IR training data.[11]

Besides these approaches, gaming engines are also commonly used to support the simulation of datasets. One regularly used one is Unreal Engine (UE), which enables multi-modal imagery generation. Nadell et al. (2023) employ UE to generate a large training set for validation on real IR data.[38] Dabbiru et al. (2024) also use UE in combination with a separate vehicle simulator for the generation of object detection data.[39] Soloviov et al. (2024) design a dedicated headless open-source framework to generate high volumes and variety of synthetic data for AI training, built on UE.[40]

More recently, attention has shifted to novel, AI-based approaches to create synthetic datasets. Generative models, like Generative Adversarial Networks (GAN) and diffusion models are particularity popular for this purpose.[41] These models can either generate datasets from scratch or augment existing datasets. One example is the work by Vo et al. (2024), where a diffusion-based image-to-image translation model was used to translate RGB images into synthetic infrared (TIR) data.[13] Khullar et al. (2023) utilized a diffusion model pretrained on a generic dataset for generation of road scenes from scratch.[42] Yuan et al. (2023) designed an approach that uses diffusion models to inpaint, inserting objects into scenes.[43] Pichler and Hueber (2024)[8] developed a method using Stable Diffusion to overlay vehicle patches onto relevant backgrounds. Trabucco et al. (2023) also employ diffusion models as a data augmentation method, to generalize new visual concepts from a few labelled samples.[44] Clement et al. (2024) implement a synthetic data pipeline for neural style transfer and GAN reskinning to train object detectors.[45] Eilertsen et al. (2021) took a different approach by using an ensemble of GANs to create a diverse dataset.[46] Pulakurthi et al. (2024) employ GANs to generate synthetic datasets from only a small sample of real-world data.[47] As a hybrid approach between aforementioned game engine UE and AI-based generation methods, Kerley et al. (2024) use Large Language Models (LLMs) to generate scenes in UE.[48]

Lastly, shifting the focus from sensor data to mission planning, the generally accepted approach for generating simulated data is to use a simulator. The NATO Modeling and Simulation Group offers vast materials about military simulations. Such military simulations require accurate models for various aspects, such as combat modelling.[29] Creating a precise and accurate simulator is not easy, and holds many theoretical and practical challenges.[30] An overview of military simulation tools can be found in Table 1 of van Oijen and Toubman.[31]

In mission planning, there are two main paradigms. Firstly, data can be generated during the mission, using the current state of the world as the starting condition. This approach requires an accurate digital twin of the battlefield.[49] By generating data in real-time, the decision support system provides mission-specific analysis, reducing generalization possibilities but potentially leading to more accurate outcomes. Secondly, data can be generated ahead of time for various missions to train a generic decision support system. This approach is usually applied for training intelligent agents that show relevant behavior, which can then be utilized during a mission.[50] In both paradigms, the simulator is tasked to find realistic continuations from a given starting state, which drives the synthetic data generation.

2.3 Measuring the reality gap

The reality gap refers to the difference between real and simulated data, often caused by a lack of diversity, poor domain fit, and low fidelity. Measuring this gap is an important first step to assess the quality of your simulated dataset, as a significant reality gap can limit the ability of a model to generalize to real data. Strategies for bridging this gap, once identified, will be further discussed in Section 3.

Several metrics and tools can be used to measure the reality gap, for which Dale et al. (2024) provide a current overview.[21] The mean square error (MSE), Structural Similarity Index (SSIM), binary cross entropy (BCE), and Jensen Shannon Divergence (JSD) can be used to evaluate the domain gap in the image space. For the abstract embedding of features in an AI model, MSE, BCE, and JSD can also be used. In addition to these, the Fréchet Inception Distance (FID) and Kernel Inception Distance (KID) could be used. Lastly, the Inception Score (IS) can be particularly useful for the prediction space.[21] Dale et al. conclude that SSIM is the best measure for the image space and that BCE is the best measure for the latent space.

Adding to the existing measures, Reinhardt et al. (2024) introduce two new metrics tailored to images: the Fréchet Embedding Distance (FED), a slight generalization of FID, and MAUVE scores, distribution comparison measures using the f-divergence frontiers.[22] Besides these measures, the distribution of real-world and simulated datasets can be compared by visualizing their high-dimensional embedding spaces. Projection algorithms like PCA, t-SNE, and UMAP can be used to project these high-dimensional data into lower dimensions, allowing for a visual assessment of the similarity between the two distributions.[22]

3. BRIDGING THE REALITY GAP WITH SIMULATION VARIATION AND FIDELITY

Our goal is to develop AI models with simulated data that perform well when applied to measured data. AI models are known to learn shortcuts for predicting outputs based on the input data[51,52] and those shortcuts are typically not applicable to data from a different domain. Bridging this reality gap requires some form of approximation of the distribution of the measured data, but it remains unclear in what direction or to what extend. In recent literature, authors have studied several methods for bridging the reality gap. This includes diversification of data,[53] variety in composition,[54] occlusions and variations in lighting,[55] and photo-realism.[56]

The first topic of this paper is on simulation variation to bridge the reality gap. With simulation variation we refer to the number of axes in which simulations are varied and to what degree. For example, for simulation of scenes in RGB and infrared, relevant axes can be the number of viewpoints, the orientation of the objects, the composition of the scene, the relation between the objects, the presence of non-targets, and many more. It can even be beneficial to synthesize images with variations that are non-realistic in a process called domain randomization.[57] For mission planning, the speed at which soldiers move could be such an axis. The ease of adding variation to the simulation depends on the use-case and the axes. For example, increasing the number of object orientations can be relatively straight-forward, while increasing the number of appearances or camouflage types can be laborious. With the addition of more variation options in the simulator, the amount of simulated data is also likely to increase, making the process computationally expensive and resulting in more simulation time. In Section 4, we investigate the importance of simulation variation for military vehicle detection and in Section 5 for object detection in sonar use cases.

Closely related to the topic of simulation variation is data augmentation. Here, instead of directly simulating variations, a processing step is applied that augments the data before feeding it to the AI model training process. The variations that are added with data augmentation typically do not provide any additional information, but are used as a trick to prevent an AI model from overfitting. For example, Ruis et al. (2024)[1] used several data augmentation methods on top of well-known 'traditional' augmentations (flipping, cropping, rotating, scaling, blurring) like AutoAugment,[58] MixUp,[59] RandAugment,[60] CutMix,[61] or GAN-based augmentations[62] leading to improved performance for synthetic-to-real detection tasks in RGB images. For sonar imagery, traditional augmentations are not necessarily applicable. Although we do use some form of data augmentations in most of the experiments in this paper, we consider such experiments as part of the AI model improvement and therefore outside of the scope of this paper.

The second topic of this paper is on how to measure and use the simulators fidelity to bridge the reality gap. The term 'fidelity' originates from simulators and equipment that are used for training personnel.[63] In the context of a decision support system for military operations, the term 'fidelity', often in conjunction with the term 'high', is employed to describe the quality of a model.[64,65] So a high fidelity would indicate a small reality gap. It is important to note that 'high fidelity' does not necessarily mean highly detailed. While greater detail can enhance realism, it is not the sole determinant of fidelity. Fidelity is actually defined as the degree to which a virtual environment accurately replicates the real world.[66] Therefore, even an abstract model can exhibit high fidelity if it accurately portrays realistic behaviours. Consequently, the concept of fidelity can also be applied to rudimentary decision support applications and its generated synthetic data.

High-fidelity models typically entail substantial development costs, often necessitating the expertise of specialists.[67] Additionally, simulations with a high level of detail tend to be computationally expensive, resulting in lengthy execution times.[14] Hence, to design a simulation system that demands minimal maintenance and expertise while ensuring high realism and accuracy, it is crucial to investigate how to achieve the optimal level of fidelity necessary for AI model development. However, a persistent challenge in current literature remains how to precisely define and measure the fidelity of a simulator. Intuitively, an AI model trained with high-fidelity data seems desirable. Nonetheless, it has been shown that models defined as high-fidelity were not necessarily better performing than low-fidelity models.[68,69] This assessment has been conducted specifically for models employed in the educational context but can also be generalized to a military application.[70] In Section 5, we investigate the effect of simulation fidelity for target recognition in sonar imagery and in Section 6, we investigate the effect of simulation fidelity for development of a decision support system for mission planning.

A final strategy is based on domain adaptation, for example by finetuning with a small dataset of the target domain.[14] Similarly, training can be done with a mixed set of simulated and real samples.[71] For example, Rizzi et al. (2024)[72] experimented with combining simulated and real data to build AI vision models with various degrees of mixing. Although not the key focus of this paper, we do explore the combination of real and simulated sources in Section 4, by training with mixed batches and in Section 5, by injecting simulated targets in measured backgrounds.

4. VEHICLE DETECTION IN RGB IMAGERY

Automated detection of military vehicles in RGB photo and video data contributes to situational awareness and supports timely decision making on the battlefield. Detection algorithms can be deployed in many systems including stationary sensors, unmanned systems such as UAVs, and manned systems such as armored reconnaissance vehicles. Development of these detection algorithms, however, remains a challenge due to the limited availability of training data. This lack of data occurs for new types of vehicles, but also for well-established vehicles that operate in a new environment, or of which the appearance has changed (e.g. upgrades, paint, camouflage). Simulations can be used to create a dataset of military vehicles that includes a wide range of variations.

In this chapter, based on recent publications by Eker et al. (2023)[24] and Heslinga et al. (2024),[23] simulated data is used to train a deep learning model for automated detection and classification of military vehicles. We do this for 4 and 15 object classes, including armored personnel carriers, reconnaissance vehicles, battle tanks, howitzers, and military trucks. We evaluate the performance on real photos of military vehicles, scraped from the internet. The main research questions for this use-case are related to simulation variation: *How much variation is needed when simulating training data, and which axes of variation are important?* In addition, we investigate the effect of mixing a large set of simulated images with a small set of real samples.[71] We do so for simulated data that contains a lot of variations as well as a set that contains limited variation.

4.1 Approach

Our strategy to determine the important axes of simulation variation is based on experimenting with the amount of variation in several axes individually, training a deep learning-based detector for each simulated data set, and evaluating the detection performance on measured data.

4.1.1 Experiments

We first experiment with a relatively easy detection challenge consisting of four vehicles classes: Fennek (scout car), Boxer (armoured personnel carrier), Panzerhaubitze 2000 (Howitzer), and DAF YA 4440 (military truck). A *baseline* data set of simulated images (800 images per class) is created that encompasses extensive variation along several axes, including image background, object position, and object appearances. In separate experiments for each axis, the amount of variation is reduced in comparison to the baseline set and the change in detection performance is considered as the relative importance of that variation axis. A full overview of the experiments is provided by Eker et al. (2023).[24] Here, we only report the results for a single reduction step per axis of variation.

Next, we extend the concept of training with simulated data to an object detector for 15 classes. For this fine-grained task, using only our simulated data results in an mAP of ~0.34, which we consider insufficient for practical deployment. Therefore, we mix in a small amount of real data, which is combined with a *full variation* dataset that contains maximal variation (similar to the *baseline*) for 15 classes. For an additional set of experiments, we also mix the small amount of real samples with a *low variation* simulated dataset that contains the reduced amount of variation along each axes described in the previous experiments.

We evaluate the models trained on various combinations of simulated and real samples using a test set of 449 images (21-50 per class). In each image, the object classes and locations are manually annotated. The evaluation metric is the mean average precision (mAP), a popular metric for object detection tasks. It combines classification accuracy with localisation accuracy, based on the Intersection over Union (IoU).

Figure 1. Examples of simulated military vehicle RGB images.

4.1.2 Data Simulation

3D models of military vehicles were placed in high resolution High Dynamic Range (HDRI) scenes in Blender simulation software.[73] HDRI scenes are 360° images that are wrapped against a distant sphere and include a light source, typically from the position of the sun, that illuminates the object placed in the scene. Virtual pictures were acquired from various viewpoints and some examples are shown in Figure 1.

The simulations were generated with variation in the HDRI background scenes (100 scenes), object-camera distance (10-100 meter), object yaw (0-360°), pitch, roll, model subtypes (up to 7), model textures (3 or 4), and model configuration (50% standard, 50% random). These settings were used to create the baseline set (4 classes, 800 images per class). We experimented with reducing the number of images per class to 100; reducing the number of HDRI scenes to 4; reducing the number of textures to 1; only using a single model subtype per class; only using standard configurations; removing pitch and roll; only using 4 horizontal viewing angles; and reducing the number camera-object distances to 3 (Figure 2).

For the experiments with 15 classes, a *full variation* dataset was generated using the same simulation settings as used for the baseline set. A *low variation* dataset was constructed with the reduction steps mentioned above, except that all available model subtypes were used, as we noticed a large standard deviation between experiments with only a single subtype. The number of images was also kept constant (800 per class) to make sure that the number of iterations per epoch was the same across experiments.

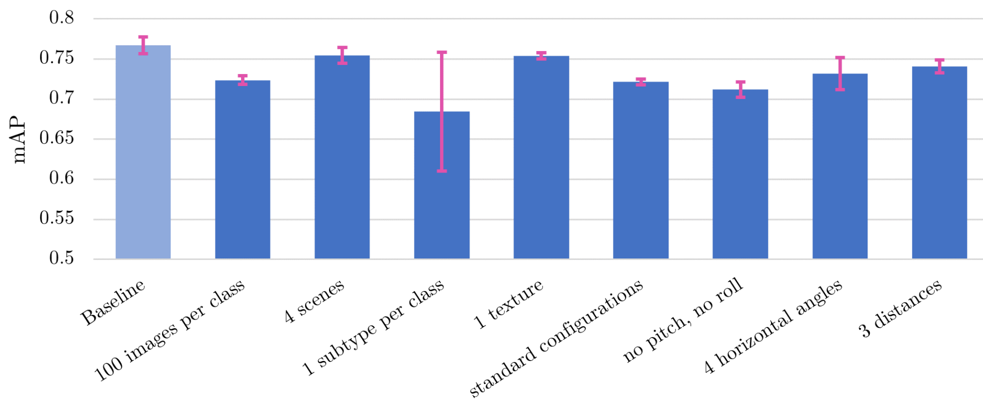


Figure 2. Effect of simulation variation on the performance of object detection in real images. Each bar represent a single axis along which the variation is reduced. Error bars represent the observed standard variation based on three experiments with newly simulated datasets. mAP = mean Average Precision.

4.1.3 AI methods

Simulation variation experiments were performed by training a deep learning-based object detector with a Mask R-CNN architecture[74] with a transformer backbone,[75] pre-trained on COCO.[76] The transformer backbone had been identified as beneficial when training with simulated data because of its sensitivity to shape over texture compared to typical convolutional neural networks.[1] Training details are provided in Eker et al. (2023).[24]

Another object detection model was used in addition to Mask R-CNN for the experiments with 15 classes. Grounding DINO[77] is a foundation model that integrates detection and grounding by combining vision and language models, using contrastive loss to align image features with textual descriptions. Unlike typical object detection models that rely solely on visual inputs, Grounding DINO is trained on large datasets with visual and text annotations.

Besides training the Mask R-CNN and Grounding DINO detectors only with simulated images, we combined the full variation and low variation simulated datasets with small numbers of real samples (2, 8 or 24 images per class). Joint training is performed by using mixed batches instead of sequential training, to encourage the models to learn generalizable features across both datasets and reduce the risk of overfitting on the simulated data.[71]

4.2 Results

The results of the simulation variation experiments are shown in Figure 2. Interestingly, all considered axes of variation seem important and reducing the variation in each single axis results in a lower mAP score than obtained with a detector trained on the baseline dataset. The observed standard variation is small, except for the experiments where only a single model subtype is used per class. This discrepancy is probably related to the difference in the quality of the 3D models, which was found to differ across subtypes.

Object detection performances for 15 classes are displayed in Figure 3. When the object detectors are only trained with simulated data, the mAP scores are lower than for the 4-class problem. The *low variation* simulation set scores substantially lower than the *full variation* set, substantiating the importance of simulation variation. A large increase in mAP is obtained when the simulated data sets are combined with a small number of real samples. For example, Mask R-CNN trained with only simulated data leads to an mAP of 0.338 [$\pm$0.007] compared to a mAP of 0.559 [$\pm$0.026] when two real samples per class are added to training data.

The positive effect of mixing both sources is consistent even when a larger number of real samples is available, although the benefit of simulated data becomes smaller as the number of real images increases. Moreover, even when real samples are mixed in, the simulated data set with *full variation* leads to a better mAP than the *low variation* set. We note similar trends for both Mask R-CNN and Grounding DINO detectors, however when more real samples are available ($\geq$ 8 samples per class) Grounding DINO outperforms Mask R-CNN.

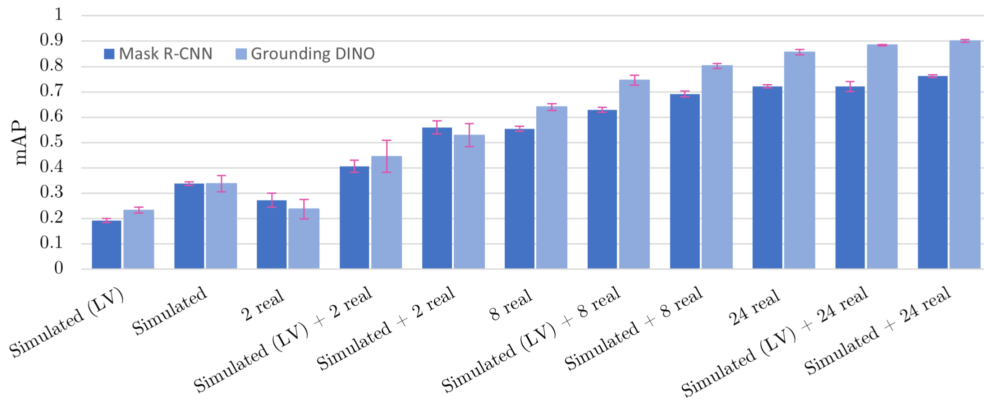


Figure 3. Results for the experiments with 15 object classes, increasing number of real samples, combined with simulated data, and either with full variation or low variation (LV). Error bars represent the observed standard deviation based on three training repetitions.

Figure 4 shows more detailed results for a single model: the Mask R-CNN trained on 800 *full variation* simulated images and 2 real images per class. The confusion matrix is relatively balanced and most confusion occur between similar looking classes (e.g. the DAF and Patria military trucks, and the TPz Fuchs and Patria armoured personnel carriers). The visual examples in Figure 4 show examples of predicted bounding boxes.

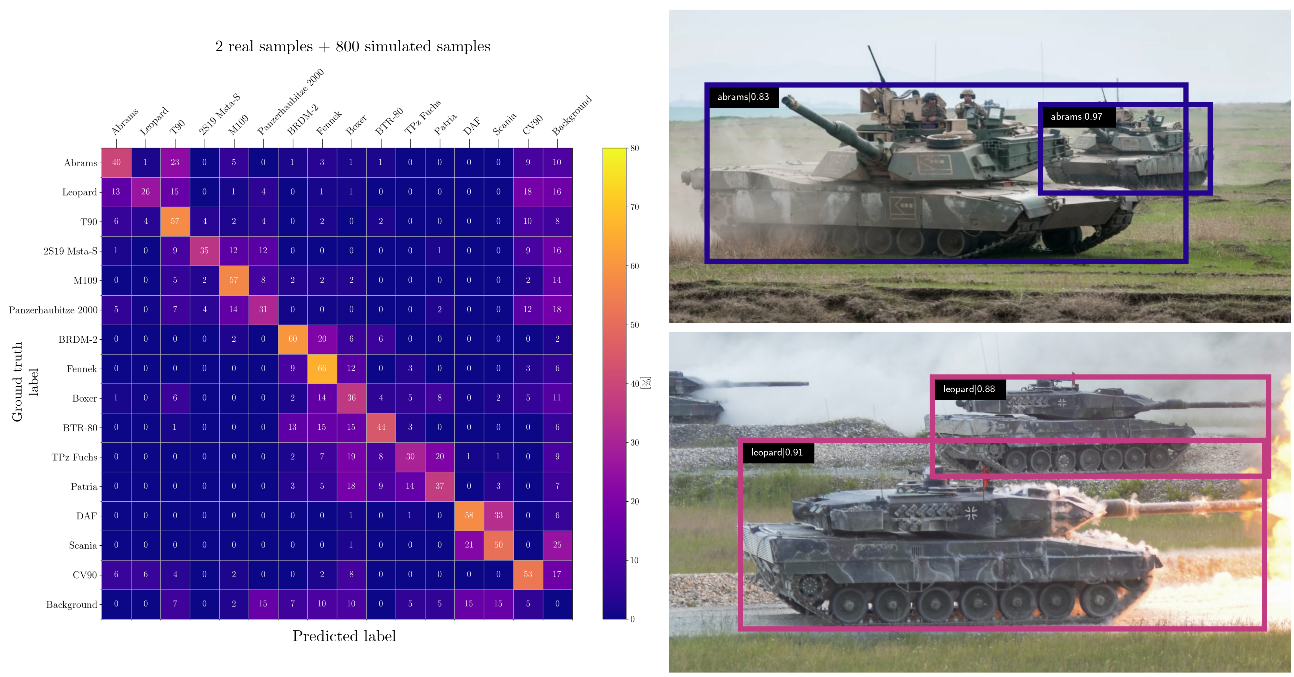


Figure 4. Results for Mask R-CNN trained on 800 simulated and 2 real images per class. Left: Confusion matrix. Right: Examples of predicted bounding boxes for two main battle tank classes. Top: M1A2 Abrams (photo by Staff. Sgt. Matthew Keeler/U.S. Army). Bottom: Leopard 2A6 (photo by 7th Army Training Command/Flickr). The lines of the bounding boxes were thickened as a postprocessing step to enhance visibility.

As a post-hoc analysis step, we looked at the t-SNE embedding of the simulated and real images. Figure 5 shows a t-SNE visualisation of the features from the final layer of the feature pyramid of a Mask R-CNN trained with simulated data and 24 real samples per class. Despite a strong detection performance on real images by this model (mAP = 0.762), a clear distinction can be seen between simulated and real data in the t-SNE plot. This indicates that while the features extracted from simulated and real samples are quite distinct, the model

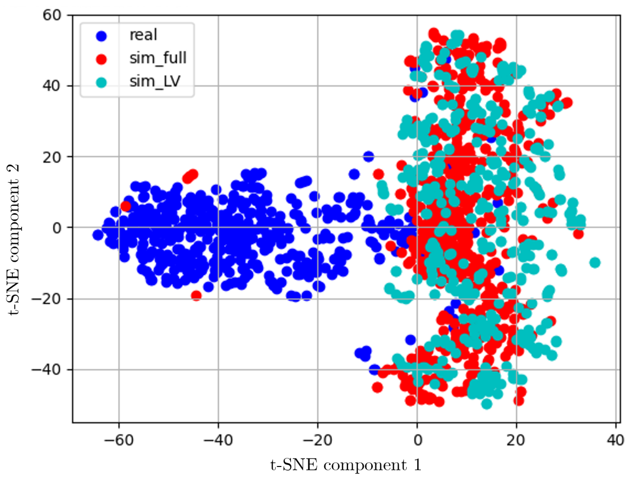


Figure 5. t-SNE plot for features extracted from the final feature pyramid layer of the a Mask R-CNN trained on a mix of simulated data and 24 real samples per class. Features computed for the whole *real* dataset of 449 samples are shown, and for both the full variation (*sim_full*) and the low variation (*sim_LV*) simulated sets, 449 images were sampled randomly.

generalizes nonetheless. Future work could focus on investigating the differences in feature space in more detail.

5. TARGET RECOGNITION IN SONAR

Automated interpretation of sensor data is highly relevant in sonar applications, and becomes even more important as a consequence of two recent developments: Both the amount of sensor data that is collected and the quality increases. Furthermore, Maritime Unmanned Systems (MUS) are being introduced. To obtain actionable information in-mission, there needs to be an effective automated perception capability at the MUS. Owing to communication constraints, it is often not feasible to have a human in-the-loop in the perception process.

In this chapter, which is based on recent publications by van de Sande et al. (2023)[78] and Kuijf et al. (2024),[79] we consider perception in different modalities of low-frequency (LF) sonar data. Effective automation of target recognition of these types of data is challenging for several reasons. First, the characteristics of LF sonar data substantially differ from optical imagery. The resolution and Signal-to-Noise Ratio (SNR) of these data are much lower compared to optical imagery. Second, it is anticipated that the information captured both in amplitude and phase data (frequency content) needs to be made available as input to support target recognition. A third challenge is scarcity of threat data. Although large volumes of sonar data can be made available as input to training, they are unbalanced and hardly contain data with relevant threat signatures. A fourth challenge is that the threat signature varies with environmental conditions and with relative position and orientation with respect to the sonar system. In low-frequency active sonar (LFAS), for example, the sound emitted by a sonar system propagates through a waveguide, to a potential target, interacts with the target (target scattering), and subsequently propagates back to the receiver system. The properties of the waveguide vary both in space and in time.

To address these challenges, several solution directions are being proposed. It is advocated that the neural network size should be tailored to the availability of data.[80,81] Furthermore, it is hypothesized that environmentally-adaptive target recognition techniques need to be developed to realize effective and reliable target recognition.[82]

We investigate whether simulations can be used to address the scarce threat data and open-world/dynamic environment challenges encountered in sonar data. This is done by considering two case studies. The first one is the detection and classification of Unexploded Ordnance (UXO) in Low-Frequency Synthetic Aperture Sonar Data (LF-SAS).[78] The second case study investigates the classification of targets in a waveguide in LFAS data.[79]

5.1 Classification of Unexploded Ordnance in LF-SAS data

The analysis presented in this section uses data collected by TNO's Mine Underground Detection (MUD) system[83] as input. The MUD system has been developed to detect and localize buried objects in the seabed in inshore environments, such as harbours. It uses an interferometric LF-SAS as primary sensor to detect proud and buried objects, and operates at frequencies below 30 kHz.

During several field experiments, recordings have been made of mine-like targets. Among those are two dummy UXO: NL-REF and NL-CYL. NL-REF is a 0.5 m x 1 m aluminium cylinder with internal structure and filled with water. NL-CYL is a 0.3 x 0.6m solid aluminium cylinder, which is a replica of the target used by Williams et al.[84] We generate high-fidelity simulations of these targets by using a finite-element modelling (FEM) approach[85] and subsequently inject synthetically generated targets in real data measured by TNO's MUD system. With this approach, the full elastic response of axisymmetric objects is obtained, which comprises both a geometric response and target resonances. More than 1000 synthetic realizations of NL-CYL and NL-REF were generated by changing target ranging (grazing angle random between 25 and 65 degrees), target orientation (azimuth angle random between -90 and +90 degrees), burial depth and sediment type (sand and mud). The training set is complemented by approximately 400 clutter contacts extracted from measured data.

In this use case, we investigate whether there is a reality gap between simulated NL-REF and NL-CYL targets and the corresponding target signatures LF-SAS data. This will be done by training a CNN classifier on synthetic targets and measured clutter, and subsequently evaluating the performance both on synthetic and measured targets and clutter contacts. Refer to van de Sande et al.[78] for a more elaborate description of the experiment.

The goal of the classification is to discriminate the dummy UXO objects NL-REF and NL-CYL from the clutter contacts. The architecture of the CNN used is described in van de Sande et al.[78] The model is trained on roughly 1000 synthetic NL-REF and NL-CYL views and 500 clutter data snippets. The classifier is evaluated on independent synthetic and/or measured target and clutter data sets. The set of measured targets consists of 72 NL-REF and 15 NL-CYL images and the evaluation set for clutter consists of 153 data snippets.

As already indicated in the introduction, there are different representations that can be used for sonar data. One representation is LF-SAS imagery.[86–89] It is a spatial representation aiming to provide insight in the spatial distribution of scatterers. A purely geometrical interpretation of LF-SAS imagery, however, is not appropriate since objects excited by sound will resonate at specific frequencies. This complicates the interpretation of LF-SAS imagery. On the other hand, these resonances are of specific interest as they contain clues not only on the size of objects, but also on the composition.[90] Another representation is referred to as Multi-Aspect Acoustic Color (MAAC). It focuses on the spectral response of objects and the variation of this response with the angle of incidence. This representation can provide detailed information on the existence and the mechanism responsible for resonances.[91] An example of LF-SAS and corresponding MAAC imagery for NL-REF for different aspect angles is presented in Figure 6. It illustrates the variability of the SAS images with aspect angle. The variations in MAAC should predominantly consist of a translation along the aspect angle axis. The MAAC imagery, however, also contains substantial differences as a result of a limited opening angle of the SAS aperture.

Figure 7 shows receiver operator characteristics (ROC) curves for the experiments conducted on SAS and MAAC imagery. For SAS imagery, there is only a small performance degradation on real targets compared to synthetic targets, suggesting that there is hardly a reality gap between simulated and real target signatures. For the corresponding MAAC results, a substantial performance degradation occurs when the classifier trained on synthetic targets is applied to the measurements. These results indicate that the reality gap between simulations and measurements is influenced by the data representation that is chosen, i.e. some information content is more susceptible to differences between the simulated scenario and measurements.

This observation suggests that the reality gap between simulations and measurements is variable and depends on the information used by the classifier to distinguish between targets and clutter. We found that the SNR of the SAS imagery is higher compared to MAAC, indicating that the information content in SAS imagery is more robust. Additionally, the information entropy of SAS imagery is lower compared to MAAC.

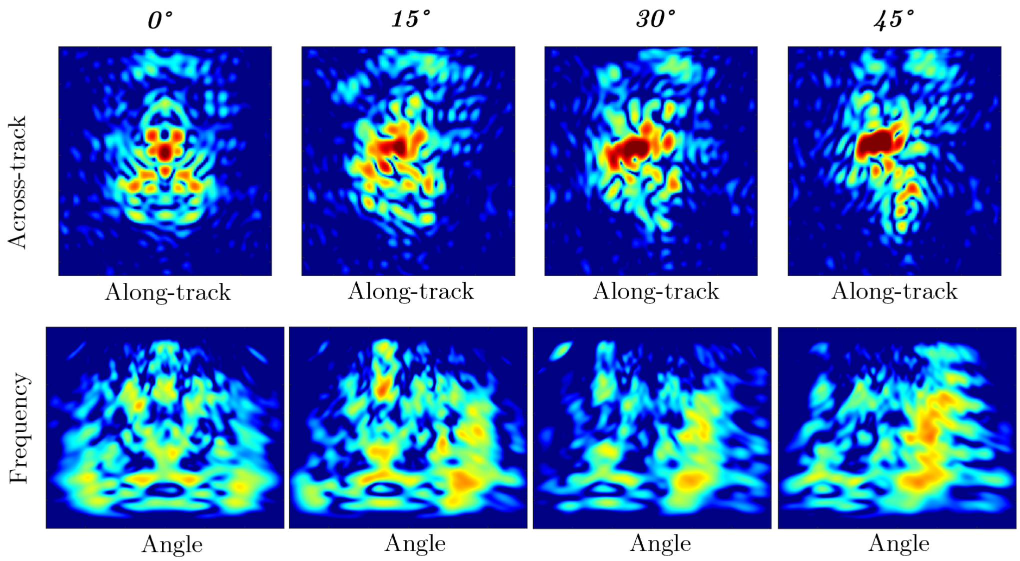


Figure 6. Examples of LF-SAS simulations for NL-REF for different aspect angles. The top row shows LF-SAS images, the bottom row the corresponding MAAC images (aspect angle vs frequency).

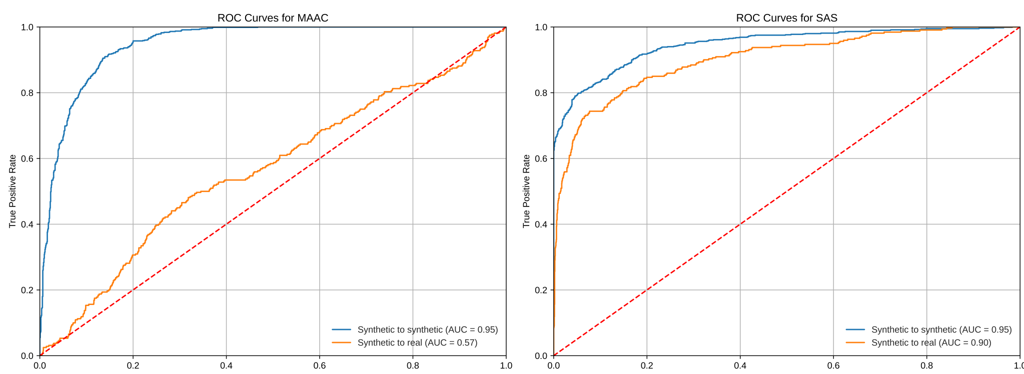


Figure 7. ROC curves both for synthetic and measured UXO and clutter objects for CNN classifier based on MAAC (left) and LF-SAS (right) imagery.

5.2 Target classification in a waveguide in LFAS data

This section considers target classification in Low-Frequency Active Sonar (LFAS) data.[92,93] The goal is to distinguish target echoes from air-filled objects positioned in the water column from echoes from clutter that originate from back-scattering of sound at the seabed. This is investigated by considering air-filled spheres as targets and granite spheres as clutter objects. It was hypothesized that the different types of objects can be discriminated based on their target scattering response.[94] The target echo, however, has traveled from the emitter to the object of interest, and back to the receiver array through the water column. The properties of the water column, which acts as a waveguide, have a substantial effect on the received target echo, i.e. the target signature strongly varies with environment and with its relative position and orientation with respect to the sonar (sonar depth, target depth and target range). Furthermore, the environmental conditions change with space and time. This is a substantial challenge that needs to be addressed when designing solutions for LFAS target classification. Measurements are conducted in specific environmental conditions and measurement geometries, posing questions to the validity of the training when applied in different conditions.

To resolve this challenge, Kuijf et al.[79] hypothesized that target simulations can be exploited to develop target classification approaches robust to environmental differences. This is demonstrated by comparing the performance of classifiers trained on different data using three approaches:

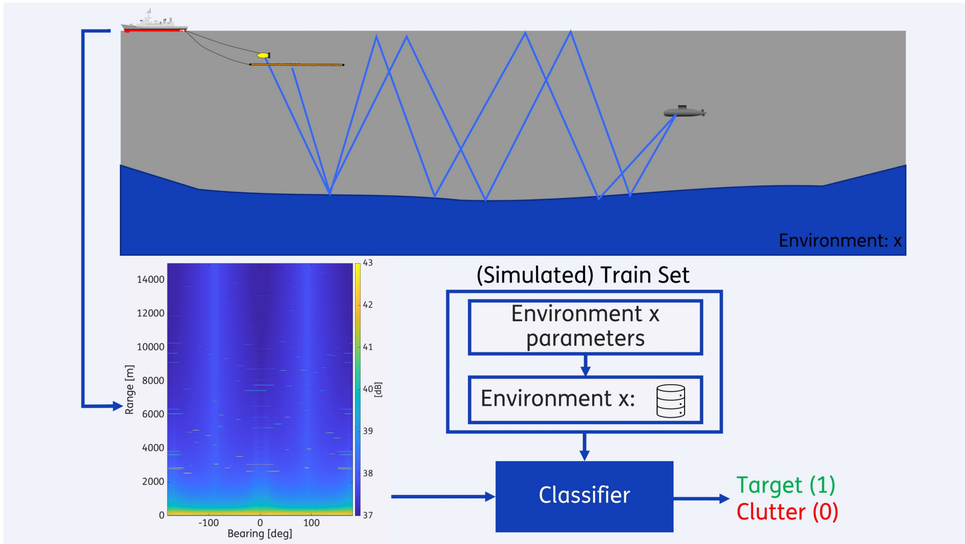


Figure 8. Illustration of LFAS use case geometry, synthetic target injection, and classification problem.

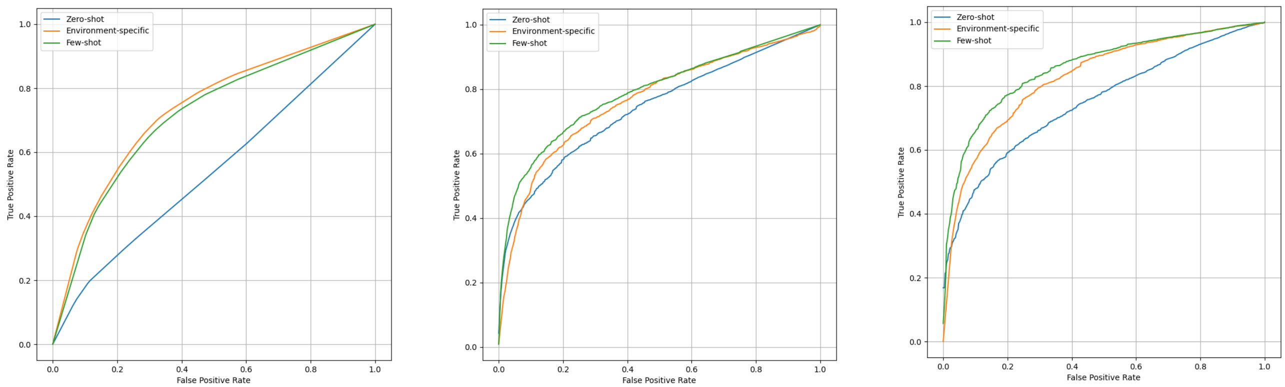
- Zero-shot/generalization approach: a classifier is trained without using any data corresponding to the environmental conditions in which the classifier is evaluated; It is trained using data collected in different environmental conditions;
- Environment-specific approach: only the limited subset of simulated data is used for training corresponding to the same environmental conditions in which the classifier will be evaluated.
- Few-shot approach: the zero-shot classifier is fine-tuned using a limited subset of simulated data corresponding to the environmental conditions in which the classifier will be evaluated. The few shot approach uses the training data of the zero-shot combined with the environment-specific approach.

An artificial environment is combined with simulated target and clutter data, that are injected into the environment Figure 8. Four different environments were simulated, corresponding to different environmental conditions: Weston 1 (shallow water, iso sound speed profile), Weston 2 (shallow water, winter profile), Weston 3 (shallow water, summer profile), and Weston 7 (deep water, Munk profile).[95,96] These environments are simulated with different water depths (50–2500 m).

Simulated targets (air-filled sphere of steel) and simulated clutter (solid sphere of granite) are injected at different locations (range, bearing, depth) into the simulated environments. For every unique environment, 12 'pings' are simulated, of which six include 100 steel spheres as a proxy for targets and the other six include 100 granite spheres as a proxy for clutter. Targets are placed at a random position in the water column and clutter is placed at the bottom of the sea. All objects are placed randomly between 2–15 km from the source.

Three different kinds of AI approaches are implemented: a support vector machine (SVM), a fully-connected neural network (FCN), and a convolutional neural network (CNN).[79] These approaches are validated in a cross-validation setting, where they are trained on a subset of the data (60 % of the data of an environment) and tested on a separate subset (40 % of the data of an environment). In addition, the effects of varying the sound speed profiles (the different Weston environments), the sediment types (coarse clay, medium silt, medium sand, very coarse sand), and water depths are studied.

Results show that the zero-shot approach, which does not contain information about the environment in which the system will be deployed, consistently has the worst performance. An example when varying the Weston sound speed profiles is shown in Figure 9; and similar trends are observed when varying the sediment types or water depths (see Kuijf et al.[79]). The few-shot and environment-specific approaches show the best performance for all three machine learning approaches. Overall, the CNN consistently outperforms the SVM and FCN by having higher receiver operator characteristics (ROC) area-under-the-curve (AUC) values.



(a) SVM (b) FNN (c) CNN

Figure 9. Results of the three machine learning approaches for the Weston 1 sound speed profile. From left to right: the SVM, FCN, and CNN methods. Colours indicate the three approaches: zero-shot/generalization (blue), few-shot (orange), and environment-specific (green). The CNN outperforms the SVM and FCN, which is also confirmed by higher AUC values.

These results indicate that it is feasible to fine-tune a previously trained machine learning system with data of simulated targets that are injected into the new environment. This outperforms the zero-shot approach, which is unable to generalize to unseen environments. In practice, underwater environment information is available and by injecting in-situ simulations of targets, training data can be generated for a target classification system. The few-shot approach shows slightly better performance than the environment-specific approach, likely because it is trained on more data. Both approaches can be considered in future applications for underwater target classification.

6. MISSION PLANNING

This chapter aims to explore the relationship between the realism of the simulator (its fidelity) and the predictive quality of a decision support system (DSS) that utilizes such a simulator. The study examines the significance of various model parameters and further investigates the manoeuvre fidelity, extending earlier work.[97] Furthermore, the relation between the manoeuvre fidelity and the performance of a decision support model is analysed in greater depth. Accurately assessing different aspects of simulator fidelity can support the development of future DSS. Instead of repeatedly altering simulator parameters, generating synthetic data, and training the DSS, parameters might be fine-tuned to enhance a fidelity measure, accelerating the feedback loop for the developer and avoiding costly re-training of the DSS. The methods investigated are applied to a practical use case: a DSS focused on the scenario of liberating a village from occupying forces.

6.1 Approach

To investigate the influence of the synthetic data quality on the performance of decision support applications, the referent-abstract model concept is used.[98] In other words, to assess the fidelity of an abstract model, a comparison to a referent model is performed. The referent model is considered the most accurate model available, closely mimicking the reality. The referent model serves as a benchmark against which the abstract model's performance and fidelity can be evaluated.

This comparison provides insights into how closely the model approximates the real-world, and the use of real-world data in the referent model is encouraged. In order to facilitate a meaningful comparison between the referent and abstract models, it is essential that both models simulate the same scenario. Hence, for this research, the chosen scenario revolves around the liberation of a village from occupying forces as it is being performed in the Mobile Combat Training Centre (MCTC) data.

6.1.1 Experiments

The MCTC[99,100] was introduced in 2003 by the Dutch ministry of Defence and enables soldiers to practice combat in a realistic setting, but without using real ammunition. Lasers and sensors are utilised to simulate firing weapons. The system tracks the location of soldiers and vehicles, used ammunition, and health status. A variety of weapons (e.g., rifles, heavy machine guns, indirect fire), vehicles (e.g., Fennek, Boxer) and terrain (e.g., cross-country, urban) may be included in the exercise. Figure 10(a) shows a soldier training with the MCTC. Note the laser sensors on the helmet that registers hits, and the laser on the rifle for firing at opposing forces.

(a) A soldier training in the Mobile Combat Training Centre (MCTC).[100]

(b) Simulated forces are liberating the training village Marnehuizen in an Agent-Based Model (ABM).

Figure 10. Illustrations of the data used in the mission planning experiments.

All the data that the systems generate are logged, so that they are available for the after-action review and to provide a rough overview of the current state of the battlefield to the trainer. The logged MCTC data contains the location of soldiers and vehicles at regular intervals. Also, fire events, hit events, kill events, and vehicle associations (when a soldier enters or exits a vehicle) are recorded.

An exercise was selected that took place in the Dutch training village Marnehuizen, which was entirely built to train Military Operations on Urban Terrain.[101] The Blue forces entered the village at the bridge in the north-east and were tasked to clear the village of enemy forces. A house-to-house battle was fought, which lasted 20 hours, until the last houses on the west side of the village were declared free of enemies. Figure 10(b) shows an excerpt of the village map, in which agent-based soldiers execute the same mission. The Blue forces occupy several buildings in the east, and move forward building by building, eliminating any red forces (west) they encounter. Even though only data from a single exercise was obtained, the manoeuvring of all soldiers of the company in training during 20 hours of exercise is available.

The consistency of the data is somewhat lacking in several aspects, which is not a problem for gaining a rough overview of the state of the operation for which the data was intended, but do form an additional hurdle for training models.[102] In spite of the deficiencies, the recorded data is one of few examples in which actual data is available of a mission execution. It accurately reflects the behaviour of the soldier, such as movement speed and manoeuvre choices, as the soldier had to make such choices in the exercise.

One might argue that a single mission execution is not sufficient to avoid overfitting. This is in essence correct, and more data will lead to better results. In this single exercise there are however 96 different participants that generate data (i.e., including vehicles and red forces), over a period of 20 hours, providing ample information on locations and movements of a soldier. Therefore we believe that data to be sufficient for the presented exploration into simulation fidelity.

6.1.2 Data Simulation

For the simulation model, an Agent-Based Model (ABM) was created (see Figure 10(b)). It was not the aim of this study to make this model perfect in terms of tactical military movements, as we want to study the differences between the real world and the simulation. The abstract ABM consists of an entity-level model, where two opposite forces are facing each other. The red forces adopt a random walk with minor fluctuations in the position, whereas the blue forces are moving towards the occupier.

The blue commander assigns a target zone for each blue unit to move towards to. The target zone is defined by being between the centroid of the red forces and the centroid of the blue unit. On the way to the targeted zone, the forces have to secure the closest building. The securing process functions similar to a progress bar, with each soldier's presence in the house incrementally increasing the value until it reaches 100%, at which point the house is considered secure. However, this only holds true if there are no enemy occupants in the house. If any unit (blue or red) has direct line of sight with the adversary, the unit opens fire.

The simulation ends when either no enemy forces are left or all the buildings have been cleared. During the simulation, random events may occur, leading to different continuations even from the same starting state. Moreover, the model includes three adjustable parameters (axes) - sensory capacity, velocity and firing success - each of which influences the model's behavioural dynamics. The sensory capacity is defined as the distance, measured in meters, within which a soldier can perceive an enemy soldier. The velocity quantifies the speed at which soldiers traverse their environment, expressed in kilometres per hour (km/h). The firing success is the likelihood that a firing attempt directed at the adversary results in success.

6.1.3 AI methods

In this research, we further elaborate on a new fidelity aspect for computer generated forces, namely the *manoeuvre fidelity*,[97] which describes the degree to which the movements of agents in a simulation match those of agents in a real-world setting.

It can be quantified by measuring the disparity between the paths followed in a simulation (abstract model) and those observed in an actual scenario (referent). For measuring the manoeuvre fidelity, different measurement methods can be used. In earlier work,[97] we have shown that modified Dynamic Time Warping,[103] here referred to as the Dynamic Time Dependent Warping (DTDW), works best. DTDW can be conceptualised as choosing the minimum-cost path matching points of both manoeuvre paths. The costs of each edge in the graph are determined by both the euclidean distance between the points as well as a difference in timing of when the movement occurs. The exact formula is described by Weyland et al. (2024)[97] and contains a parameter α that balances the importance between euclidean distance and timing aspects, which was set to 0.5. In this work, we aim to optimize this α parameter to create the highest correlation between the fidelity measure and the performance of a DSS.

We have implemented a rudimentary DSS that supports the user in estimating the remaining duration for liberating the village. A linear regression model is used to fit the synthetic data and predict the remaining mission time based on a given timestamp of the mission. The model is trained with one independent variable, the number of houses secured, and the dependent variable, the remaining mission time according to the simulators.

6.2 Results

Figure 11 presents a thorough investigation into the relationship between the manoeuvre fidelity score, based on the formulated DTDW formula, and the simulator's generated mission time. In order to see what the best weight α is between the distance term and the time term of the DTDW formula, we vary the weight between 0 and 1 with 0.1 step sizes*. A low α value puts more weight on the distance term, making it more important that the same route is followed, while a large α puts more weight on the timing of the movement.

It is observed that progressively shifting the importance from the timing term to the distance term makes the relationship between manoeuvre fidelity and simulated mission time more linear. It is interesting to point out that in the bottom sub-figures, the patterns seems to indicate that if the fidelity score would further decrease, the simulated mission's time would further increase. This brings us to the conclusion that the corresponding weights are not suitable to measure the manoeuvre fidelity.

The upper graphs of Figure 11 are in line with the expectations: when the DTDW measure shows lowest distance (i.e., thus highest fidelity), the simulated mission time is closest to the actual mission time, and when

*$\alpha = 0.5$ is omitted as it was already tested in earlier work.

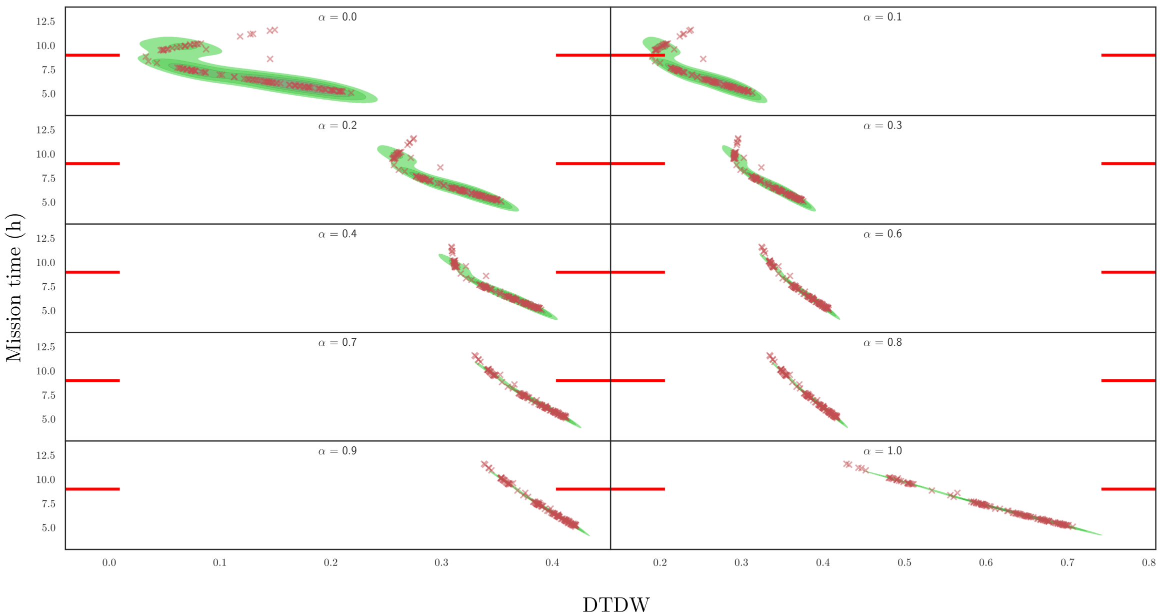


Figure 11. Investigating the impact of the temporal term and spatial term of the DTDW formula on the relation between the fidelity score and the ABM's accuracy in simulating the correct mission time. The fidelity was computed using the ABM as abstract model and the MCTC data as referent. α correspond to the weight given to the temporal term, consequently $1 - \alpha$ corresponds to the weight given to the spatial term. A red cross corresponds to the average of one distinct simulator with its unique parameter combination as described in Section 6.1.2. The underlying green blob corresponds to the kernel density estimation of the red crosses. The horizontal lines correspond the the actual mission's time of 9 hours.

the DTDW measure increases (i.e., the fidelity decreases), the simulated mission time either is too long or too short. From these results, an α weight of somewhere in the range including 0 and 0.2 seems to work best.[†]

The next experiment investigates how the performance of the classifier is affected when it is trained on data originating from simulators of different fidelities.[‡] The performance of the classifier is measured as the mean absolute distance between predicted mission time versus actual mission time from the MCTC data, using a variety of states that occurred in the real-world data. We calculate the Spearman coefficient between the simulator's fidelity and the performance of the classifier trained on data generated by the respective simulator, for different α parameter settings. The results are depicted in Figure 12.

For a lower α value, i.e. more weight on the spatial term, the Spearman's coefficient increases. An α value close to 0 seems optimal, which is in line with the observations from Figure 11. In future research, to determine a well-working α value that generalizes well, this test should be performed on multiple simulators. Preliminary results with a different simulator suggest that well working α values range between 0 and 0.4.

As all Spearman's coefficients of Figure 12 are positive, we can draw the conclusion that the manoeuvre fidelity is a suitable fidelity measure to predict the performance of a DSS that is trained on synthetic data from that simulator. Simulations can vary greatly after certain randomized events occur, which partly explains why coefficients are still quite low. Even though the results are promising, it has yet to be determined whether the current level of correlation is sufficient to aid the developer during the creation of novel simulators.

[†]Please note, that when multiple missions with different mission times would be available, Figure 11 could be generated for each individual mission. Possibly a different α value works best for different missions, and the results would have to be aggregated to make a choice for the best working α value.

[‡]We regard the agent-based simulator with a different parameter set as a different simulator, as it may have a large difference in fidelity.

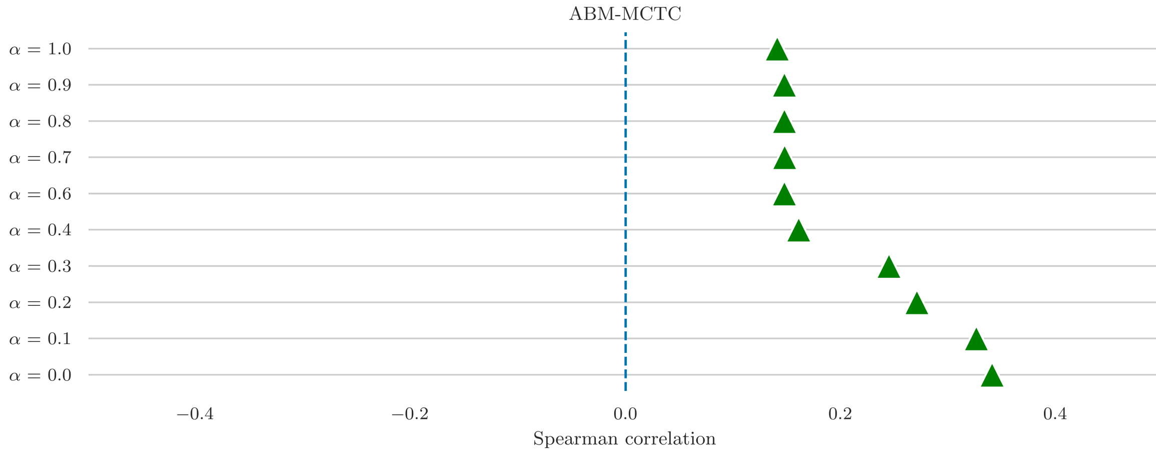


Figure 12. The Spearman's correlation coefficient that compares a simulator's fidelity and the performance of a DSS trained on the respective simulator's data. Results are shown for different values of α, which determines the weight between the distance and time term in the Dynamic Time Dependent Warping (DTDW) formula. The DTDW is used to measure the maneouvre fidelity of the simulation.

7. DISCUSSION

Development of AI tools for military applications is challenging due to the lack of high-quality measured data in a diverse set environments of scenarios. The use of simulated data for AI model training is a promising alternative. In this paper, we investigated the relevance of simulation variation and simulation fidelity for bridging the reality gap.

With the use case on military vehicle detection, we were able to show the importance of simulation variation. Interestingly, all axes of variation that were explored were found to be relevant, indicating that it is beneficial to define a broad range of variations. At the same time, each individual axis can only bring a limited amount of variation and at some point the performance of the AI system will saturate because of it.[23] Similarly, in the use case on target classification in a waveguide in LFAS data, we found that the addition of extra variation underwater environments can help some classifiers, even when these environments are different than the target environment.

Despite including plenty of simulation in the military vehicle detection, it still proved insufficient for the more fine-grained challenge with similar-looking vehicles. This could mean we missed key axes of variation, but the fact that adding a small amount of real samples boosts the AI detection model performance strongly indicates that some important details were lacking in the simulated data.[104] We hypothesize that these details can also be incorporated by improving the fidelity of the simulations.

The value of simulation fidelity also follows from the other use cases. On the topic of mission planning, we found that the quality of a decision support system is directly related to the fidelity of the simulation. A higher manoeuvre fidelity indicates that the system is able to predict the total duration of the mission with higher accuracy. In the low-frequency active sonar use case results we see the importance of including the target environment in the training set. If the target environment is left out (as is the case in the zero-shot method), the classification performance is substantially reduced. We also learned that fidelity does not only refer to the quality of the simulated objects, but also to the data representation that is chosen as input for an AI model: the classifier trained on simulated LF-SAS generalizes well to real data, while the classifier trained on simulated MAAC data does not.

In conclusion, AI tools are likely to play a role in various military use cases and simulated data will be important for AI model development. Incorporation of sufficient simulation variation is essential to make AI tools effective in real-world situations. As an added benefit, simulations allow for quick modifications to new environments and mission goals. Furthermore, the measuring of the fidelity provides valuable insights in the relation between specific aspects and the performance of a classifier or decision support system. The envisioned use case of such systems pose requirements on the fidelity on certain aspects of the simulator or data, and not every aspect is required to have high fidelity to create a well-working classifier or decision support system.

REFERENCES

[1] Ruis, F. A., Liezenga, A. M., Heslinga, F. G., Ballan, L., Eker, T. A., den Hollander, R. J., van Leeuwen, M. C., Dijk, J., and Huizinga, W., "Improving object detector training on synthetic data by starting with a strong baseline methodology," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 333–345, SPIE (2024).

[2] Steiniger, Y., Kraus, D., and Meisen, T., "Survey on deep learning based computer vision for sonar imagery," *Engineering Applications of Artificial Intelligence* **114**, 105157 (2022).

[3] Chen, J., Sun, J., and Wang, G., "From unmanned systems to autonomous intelligent systems," *Engineering* **12**, 16–19 (2022).

[4] Soleyman, S. and Khosla, D., "Multi-agent mission planning with reinforcement learning," in [*AAAI Symposium on the 2nd Workshop on Deep Models and Artificial Intelligence for Defense Applications: Potentials, Theories, Practices, Tools, and Risks*], 51–57, AAAI (2020).

[5] Man, K. and Chahl, J., "A review of synthetic image data and its use in computer vision," *Journal of Imaging* **8**(11), 310 (2022).

[6] Jordon, J., Szpruch, L., Houssiau, F., Bottarelli, M., Cherubin, G., Maple, C., Cohen, S. N., and Weller, A., "Synthetic data–what, why and how?," *arXiv preprint arXiv:2205.03257* (2022).

[7] Sarıyıldız, M. B., Alahari, K., Larlus, D., and Kalantidis, Y., "Fake it till you make it: Learning transferable representations from synthetic imagenet clones," in [*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*], 8011–8021 (June 2023).

[8] Pichler, A. and Hueber, N., "Method for training deep neural networks in vehicle detection using drone-captured data and background synthesis," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 150–162, SPIE (2024).

[9] Hueber, N. and Pichler, A., "Comparison of hybrid real and synthetized image sources for training an object detector," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 1303505, SPIE (2024).

[10] Moate, C. P., Hayward, S. D., Ellis, J. S., Russell, L., Timmerman, R. O., Lane, R. O., and Strain, T. J., "Vehicle detection in infrared imagery using neural networks with synthetic training data," in [*Image Analysis and Recognition: 15th International Conference, ICIAR 2018, Póvoa de Varzim, Portugal, June 27–29, 2018, Proceedings 15*], 453–461, Springer (2018).

[11] Uplinger, J., Schesser, D., Meyer, C., Conroy, J., and de Melo, C., "Adversarial learning using synthetic ir imagery," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications*], **12529**, 103–108, SPIE (2023).

[12] Pichler, A. and Hueber, N., "Automated method for generating datasets of infrared or visible images for context-specific training of deep neural network-based object detectors," in [*Image and Signal Processing for Remote Sensing XXIX*], **12733**, 141–160, SPIE (2023).

[13] Vo, D. T., Duc, P. A., Thao, N. N., and Ninh, H., "An approach to synthesize thermal infrared ship images," in [*Synthetic Data for Computer Vision Workshop - CVPR*], (2024).

[14] Heslinga, F. G., Uysal, F., van Rooij, S. B., Berberich, S., and Caro Cuenca, M., "Few-shot learning for satellite characterisation from synthetic inverse synthetic aperture radar images," *IET Radar, Sonar & Navigation* **18**(4), 649–656 (2024).

[15] Rizaev, I. G. and Achim, A., "Synthwakesar: A synthetic sar dataset for deep learning classification of ships at sea," *Remote Sensing* **14**(16) (2022).

[16] Ochal, M., Vazquez, J., Petillot, Y., and Wang, S., "A comparison of few-shot learning methods for underwater optical and sonar image classification," in [*Global Oceans 2020: Singapore – U.S. Gulf Coast*], 1–10 (2020).

[17] Koo, S., Youm, S., and Shin, J., "Cycle-gan-based synthetic sonar image generation for improved underwater classification," in [*Ocean Sensing and Monitoring XVI*], **13061**, 69–83, SPIE (2024).

[18] Denos, K., Ravaut, M., Fagette, A., and Lim, H.-S., "Deep learning applied to underwater mine warfare," in [*OCEANS 2017 - Aberdeen*], 1–7 (2017).

[19] Wang, X. and Fang, X., "A multi-agent reinforcement learning algorithm with the action preference selection strategy for massive target cooperative search mission planning," *Expert Systems with Applications* **231**, 120643 (2023).

[20] Huerten, C., Sieberg, P. M., and Schramm, D., "Determining required simulation model fidelity for developing an advanced driver assistance system for automated lane change decision making," in [*AmE 2022 - Automotive meets Electronics; 13. GMM-Symposium*], 1–6 (2022).

[21] Dale, A. S., Reindl, W. R., Sanchez, E., William, A., and Christopher, L., "Mind the (domain) gap: metrics for the differences in synthetic and real data distributions," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 312–321, SPIE (2024).

[22] Reinhardt, C. N., Brockman, S., Blue, R., Clipp, B., and Hoogs, A., "Toward quantifying the real-versus-synthetic imagery training data reality gap: analysis and practical applications," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 261–274, SPIE (2024).

[23] Heslinga, F. G., Eker, T. A., Fokkinga, E. P., van Woerden, J. E., Ruis, F. A., den Hollander, R. J., and Schutte, K., "Combining simulated data, foundation models, and few real samples for training object detectors," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 44–55, SPIE (2024).

[24] Eker, T. A., Heslinga, F. G., Ballan, L., den Hollander, R. J., and Schutte, K., "The effect of simulation variety on a deep learning-based military vehicle detector," in [*Artificial Intelligence for Security and Defence Applications*], **12742**, 183–196, SPIE (2023).

[25] de Melo, C., Conover, D., Poster, D., Leung, S., Nguyen, R., and Conroy, J., "Synthetic data for automatic target recognition from small drones," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications*], **12529**, 191–196, SPIE (2023).

[26] Westlake, S. T., Volonakis, T. N., Jackman, J., James, D. B., and Sherriff, A., "Deep learning for automatic target recognition with real and synthetic infrared maritime imagery," in [*Artificial intelligence and machine learning in defense applications II*], **11543**, 41–53, SPIE (2020).

[27] McKenzie, K., Jacobs, E., Ramirez, A., Conroy, J., and Watson, T., "Acoustic sensing on multi-rotor uav for target detection using a convolutional neural network," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 1303504, SPIE (2024).

[28] Xian, R., Vogel, B. I., De Melo, C. M., Harrison, A. V., and Manocha, D., "Real-time human action recognition from aerial videos using autozoom and synthetic data," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 119–129, SPIE (2024).

[29] Tolk, A., [*Engineering Principles of Combat Modeling and Distributed Simulation*], Wiley (3 2012).

[30] Ören, T., Zeigler, B., and Tolk, A., [*Body of Knowledge for Modeling and Simulation*], Simulation Foundations, Methods and Applications, Springer International Publishing, Cham (2023).

[31] van Oijen, J. and Toubman, A., "Are We Machine Learning Yet? Computer Generated Forces with Learning Capabilities in Military Simulation," in [*2021 Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)*], (2021).

[32] Spell, G. P., Tran, M., Torrione, P., Jeiran, M., Bahhur, B., and Manser, K., "Integration of synthetic data into real world computer vision pipelines," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 275–283, SPIE (2024).

[33] Ozdemir, M. R., Cevher, L., and Ertekin, S., "Ai-based air-to-surface mission planning using predictive launch acceptability region approach," in [*2021 International Conference on Military Technologies (ICMT)*], 1–8 (2021).

[34] Scholl, K., Briggs, G. J., Zhang, L. A., and Salmon, J. L., [*Understanding the Limits of Artificial Intelligence for Warfighters: Volume 5, Mission Planning*], RAND Corporation, Santa Monica, CA (2024).

[35] Kenul, E., Black, M., Massey, D., Havelka, Z., Henkai, M., Gavin, K., and Shellhorn, L., "Replicanttm framework for synthetic data generation," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 445–459, SPIE (2024).

[36] Polat, Ö. M. and Özer, Y. C., "Generating physics-based synthetic data for infrared/electro-optical system analysis and design," in [*Optical Modeling and Performance Predictions XII*], **12215**, 19–29, SPIE (2022).

[37] Williams, G., Lecakes Jr, G. D., Almon, A., Koutsoubis, N., Naddeo, K., Kiel, T., Ditzler, G., and Bouaynaya, N. C., "Dyvir: dynamic virtual reality dataset for aerial threat object detection," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications*], 129–139, SPIE (2023).

[38] Nadell, C. C., Spell, G. P., Jeiran, M., and Manser, K. E., "Real-data performance evaluation of unreal engine synthetic ir data," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications*], **12529**, 19–33, SPIE (2023).

[39] Dabbiru, L., Goodin, C., Carruth, D. W., Aspin, Z., Carrillo, J., and Kaniarz, J., "Simulation fidelity analysis using deep neural networks," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 346–352, SPIE (2024).

[40] Soloviov, A., Anderson, D. T., Buck, A. R., and Alvey, B., "Mizsim: a headless open-source simulation framework for training and evaluating artificial intelligence," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 247–260, SPIE (2024).

[41] Azizi, S., Kornblith, S., Saharia, C., Norouzi, M., and Fleet, D. J., "Synthetic data from diffusion models improves imagenet classification," *arXiv preprint arXiv:2304.08466* (2023).

[42] Khullar, D., Shah, Y., Kulkarni, N., and Sokhandan, N., "Synthetic data generation for scarce road scene detection scenarios," in [*NeurIPS 2023 Workshop on Synthetic Data Generation with Generative AI*], (2023).

[43] Yuan, L., Hong, J., Sarukkai, V., and Fatahalian, K., "Learning to place objects into scenes by hallucinating scenes around objects," in [*NeurIPS 2023 Workshop on Synthetic Data Generation with Generative AI*], (2023).

[44] Trabucco, B., Doherty, K., Gurinas, M., and Salakhutdinov, R., "Effective data augmentation with diffusion models," *arXiv preprint arXiv:2302.07944* (2023).

[45] Clement, N., Schoen, A., Boedihardjo, A., and Jenkins, A., "Synthetic data and hierarchical object detection in overhead imagery," *ACM Transactions on Multimedia Computing, Communications and Applications* **20**(4), 1–20 (2024).

[46] Eilertsen, G., Tsirikoglou, A., Lundström, C., and Unger, J., "Ensembles of gans for synthetic training data generation," *arXiv preprint arXiv:2104.11797* (2021).

[47] Pulakurthi, P. R., De Melo, C. M., Rao, R., and Rabbani, M., "Enhancing human action recognition with gan-based data augmentation," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 194–204, SPIE (2024).

[48] Kerley, J., Anderson, D. T., Buck, A. R., and Alvey, B., "Generating simulated data with a large language model," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 25–43, SPIE (2024).

[49] Metcalf, J. G. and Laffey, J. A., "Integrating Digital Twin Concepts to Enhance Agility of the United States Marine Corps' Decision Support Framework," tech. rep., Naval Postgraduate School, Monterey, CA (2023).

[50] Goecks, V. G., Waytowich, N., Asher, D. E., Park, S. J., Mittrick, M., Richardson, J., Vindiola, M., Logie, A., Dennison, M., Trout, T., Narayanan, P., and Kott, A., "On games and simulators as a platform for development of artificial intelligence for command and control," (2021).

[51] Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A., "Shortcut learning in deep neural networks," *Nature Machine Intelligence* **2**(11), 665–673 (2020).

[52] Ong Ly, C., Unnikrishnan, B., Tadic, T., Patel, T., Duhamel, J., Kandel, S., Moayedi, Y., Brudno, M., Hope, A., Ross, H., and McIntosh, C., "Shortcut learning in medical ai hinders generalization: method for estimating ai model generalization without external data," *NPJ Digital Medicine* **7**(124) (2024).

[53] Wang, Q., Gao, J., Lin, W., and Yuan, Y., "Learning from synthetic data for crowd counting in the wild," in [*2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*], 8190–8199 (2019).

[54] Dwibedi, D., Misra, I., and Hebert, M., "Cut, paste and learn: Surprisingly easy synthesis for instance detection," in [*2017 IEEE International Conference on Computer Vision (ICCV)*], 1310–1319 (2017).

[55] Rajpura, P. S., Hegde, R. S., and Bojinov, H., "Object detection using deep cnns trained on synthetic images," *ArXiv* **abs/1706.06782** (2017).

[56] Movshovitz-Attias, Y., Kanade, T., and Sheikh, Y., "How useful is photo-realistic rendering for visual learning?," in [*Computer Vision - ECCV 2016 Workshops*], 202–217 (2016).

[57] Tremblay, J., Prakash, A., Acuna, D., Brophy, M., Jampani, V., Anil, C., To, T., Cameracci, E., Boochoon, S., and Birchfield, S., "Training deep networks with synthetic data: Bridging the reality gap by domain randomization," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* , 1082–10828 (2018).

[58] Cubuk, E. D., Zoph, B., Mané, D., Vasudevan, V., and Le, Q. V., "Autoaugment: Learning augmentation policies from data," *CoRR* **abs/1805.09501** (2018).

[59] Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D., "mixup: Beyond empirical risk minimization," (2018).

[60] Cubuk, E. D., Zoph, B., Shlens, J., and Le, Q. V., "Randaugment: Practical automated data augmentation with a reduced search space," (2019).

[61] Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J., and Yoo, Y., "Cutmix: Regularization strategy to train strong classifiers with localizable features," in [*Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*], (October 2019).

[62] Motamed, S., Rogalla, P., and Khalvati, F., "Data augmentation using generative adversarial networks (gans) for gan-based detection of pneumonia and covid-19 in chest x-ray images," *Informatics in Medicine Unlocked* **27**, 100779 (2021).

[63] Hays, R. T. and Singer, M. J., [*Simulation fidelity in training system design: Bridging the gap between reality and training*], Springer Science & Business Media (2012).

[64] Nunez, P., Jones, R., and Shoop, S., "High-Fidelity Ground Platform and Terrain Mechanics Modeling for Military Applications Involving Vehicle Dynamics and Mobility Analysis," tech. rep., US Army RDECOM Tank Automotive Research Development and REPORT NUMBER Engineering Center (TARDEC) (2004).

[65] Baqar, S., *Low-cost PC-based high-fidelity infrared signature modelling and simulation*, PhD thesis, Dept. of Aerospace, Power and Sensors, Defence College of Management and Technology, Cranfield University (2007).

[66] Alexander, A. L., Brunyé, T., Sidman, J., Weil, S. A., et al., "From Gaming to Training: A Review of Studies on Fidelity, Immersion, Presence, and Buy-in and Their Effects on Transfer in PC-Based Simulations and Games," *DARWARS Training Impact Group* **5**, 1–14 (2005).

[67] Roza, Z. C., *Simulation Fidelity Theory and Practice: A Unified Approach to Defining, Specifying and Measuring the Realism of Simulations*, PhD thesis, Technische Universiteit Delft, Delft, The Netherlands (2004).

[68] Feinstein, A. H. and Cannon, H. M., "Constructs of Simulation Evaluation," in [*Simulation in Aviation Training*], Jentsch, F. and Curtis, M., eds., 459–474, Routledge, Oxfordshire, England, UK (2017).

[69] Mania, K., Wooldridge, D., Coxon, M., and Robinson, A., "The Effect of Visual and Interaction Fidelity on Spatial Cognition in Immersive Virtual Environments," *IEEE Transactions on Visualization and Computer Graphics* **12**(3), 396–404 (2006).

[70] de Reus, N. M., Kerbusch, P., Schadd, M. P. D., and de Vos., A., "Geospatial analysis for machine learning in tactical decision support," in [*STO-MP-MSG-184*], NATO (2021).

[71] Seib, V., Lange, B., and Wirtz, S., "Mixing real and synthetic data to enhance neural network training - A review of current approaches," *arXiv preprint arXiv:2007.08781* (2020).

[72] Rizzi, P., Gormish, M., Kovarskiy, J., Reite, A., and Zeiler, M., "Mixing synthetic and real data to build ai vision models," in [*Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications II*], **13035**, 360–370, SPIE (2024).

[73] Community, B. O., *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam (2018).

[74] He, K., Gkioxari, G., Dollár, P., and Girshick, R., "Mask R-CNN," *arXiv preprint arXiv:1703.06870* (2018).

[75] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B., "Swin transformer: Hierarchical vision transformer using shifted windows," (2021).

[76] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L., "Microsoft COCO: Common objects in context," in [*Computer Vision – ECCV 2014*], Fleet, D., Pajdla, T., Schiele, B., and Tuytelaars, T., eds., 740–755 (2014).

[77] Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., and Zhang, L., "Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection," (2023).

[78] van de Sande, J. J. M., Huizinga, W., den Hollander, R. J. M., van den Burg, D. W., and van Vossen, R., "Application of a convolutional neural network trained with simulated low-frequency synthetic aperture sonar data for classification of buried uxo," in [*Proceedings of Underwater Acoustics Conference and Exhibiition (UACE), Greece*], (2023).

[79] Kuijf, H. J., van Heteren, A. M., Oppeneer, V. O., van Leeuwen, M. C., and van Vossen, R., "Evaluation of machine learning approach for underwater target classification with low-frequency active sonar robust to environmental differences," in [*Proceedings of International Confernce on Underwater Acoustics (ICUA), Bath, United Kingdom*], (2024).

[80] Williams, D. P., "On the use of tiny convolutional neural networks for human-expert-level classification performance in sonar imagery," *IEEE journal of oceanic engineering* **46**(1), 236–260 (2020).

[81] Williams, D. P., "Underwater target classification in synthetic aperture sonar imagery using deep convolutional neural networks," in [*2016 23rd international conference on pattern recognition (ICPR)*], 2497–2502, IEEE (2016).

[82] Stack, J., "Automation for underwater mine recognition: current trends and future strategy," in [*Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XVI*], **8017**, 205–225, SPIE (2011).

[83] van Vossen, R., van de Sande, J., Duijster, A., van der Burg, D., Mulders, I., and Beckers, G., "Seabed characterization with multi-band interferometric sonar," in [*Proceedings of Meetings on Acoustics*], **40**(1), AIP Publishing (2020).

[84] Williams, K. L., Kargl, S. G., Thorsos, E. I., Burnett, D. S., Lopes, J. L., Zampolli, M., and Marston, P. L., "Acoustic scattering from a solid aluminum cylinder in contact with a sand sediment: Measurements, modeling, and interpretation," *The Journal of the Acoustical Society of America* **127**(6), 3356–3371 (2010).

[85] Zampolli, M., Tesei, A., Jensen, F. B., Malm, N., and Blottman, J. B., "A computationally efficient finite element model with perfectly matched layers applied to scattering from axially symmetric objects," *The Journal of the Acoustical Society of America* **122**(3), 1472–1485 (2007).

[86] Hansen, R. E., Callow, H. J., Sabo, T. O., and Synnes, S. A. V., "Challenges in seafloor imaging and mapping with synthetic aperture sonar," *IEEE Transactions on geoscience and Remote Sensing* **49**(10), 3677–3687 (2011).

[87] Hansen, R. E., "Synthetic aperture sonar technology review," *Marine Technology Society Journal* **47**(5), 117–127 (2013).

[88] Vossen, R. v., Hunter, A. J., Duijster, A. J., and Beckers, A. L. D., "Uxo detector for underwater surveys using low-frequency sonar," tech. rep., Netherlands Organisation for Applied Scientific Research (TNO) (2015).

[89] van Vossen, R., van de Sande, J., van der Burg, D., Duijster, A., Mulders, I., and Beckers, G., "Area and depth coverage assessment for low-frequency synthetic aperture sonar unexploded ordnance detection surveys," in [*Proceedings of Meetings on Acoustics*], **44**(1), AIP Publishing (2021).

[90] Hoang, T., Dalton, K. S., Gerg, I. D., Blanford, T. E., Brown, D. C., and Monga, V., "Resonant scattering inspired deep networks for munition detection in 3d sonar imagery," *IEEE Transactions on Geoscience and Remote Sensing* (2023).

[91] Kargl, S. G., Williams, K. L., Marston, T. M., Kennedy, J. L., and Lopes, J. L., "Acoustic response of unexploded ordnance (uxo) and cylindrical targets," in [*OCEANS 2010 MTS/IEEE SEATTLE*], 1–5 (2010).

[92] Tyler, G. D., "The emergence of low-frequency active acoustics as a critical antisubmarine warfare technology," *Johns Hopkins APL Technical Digest* **13**(1), 145–159 (1992).

[93] Beerens, S. P. and van der Spek, E., "The interim removable low frequency active sonar system," in [*Proceeding of the European conference on Undersea Defence Technology, UDT Europe 2006, June, 27-29, Hamburg, Germany*], Nexus Media, Ltd (2006).

[94] Dragonette, L. R. and Gaumond, C. F., "Transient and steady-state scattering and diffraction from underwater targets," *Handbook of Acoustics* , 373 (1998).

[95] Ainslie, M., "Editorial: Validation of sonar performance assessment tools," in [*Proceedings of the Institute of Acoustics*], **32** (2010).

[96] Zampolli, M., Ainslie, M., and Schippers, P., "Scenarios for benchmarking range-dependent active sonar performance models," in [*Validation of Sonar Performance Assessment Tools*], (11 2010).

[97] Weyland, L. F., Schadd, M. P. D., and Henderson, H. C., "Exploring the fidelity of synthetic data for decision support systems in military applications," in [*International Conference on Military Communication and Information Systems, ICMCIS 2024, Koblenz, Germany, April 23-24, 2024*], 1–8, IEEE (2024).

[98] Schricker, B. C., Franceschini, R. W., and Johnson, T. C., "Fidelity Evaluation Framework," in [*Proceedings. 34th Annual Simulation Symposium*], 109–116, IEEE Computer Society, Los Alamitos, California (2001).

[99] Groen, S., "De Genie versus het Simulatiecentrum Landoptreden," *Promotor* **39**(4), 33–39 (2015).

[100] de Boer, A., "Naar een hoger level," *Landmacht* **13**(3) (2015).

[101] de Jong, J., Burghouts, G., Hiemstra, H., te Marvelde, A., van Norden, W., and Schutte, K., "Hold your fire!: Preventing fratricide in the dismounted soldier domain,," in [*Proceedings of the 13th International Command and Control Research and Technology,*], Bellevue, WA, USA (2008).

[102] Schadd, M. P. D., de Reus, N. M., Uilkema, S., and Voogd, J. M., "Data-driven behavioural modelling for military applications," *Journal of Defence & Security Technologies* **4**(2), 12–36 (2022).

[103] Berndt, D. J. and Clifford, J., "Using dynamic time warping to find patterns in time series," in [*Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*], *AAAIWS'94*, 359–370, AAAI Press (1994).

[104] Eker, T. A., Fokkinga, E. P., Heslinga, F. G., and Schutte, K., "Balancing 3d-model fidelity for training a vehicle detector on simulated data," in [*Artificial Intelligence for Security and Defence Applications II*], **13206**, in press, SPIE (2024).