

Development of Hybrid Artificial Intelligence Training on Real and Synthetic Data Benchmark on Two Mixed Training Strategies

Paul Wachter¹[0000–0002–6224–6140], Lukas Niehaus²[0009–0009–9978–6851], and
Julius Schöning¹[0000–0003–4921–5179]

¹ Faculty of Engineering and Computer Science, Osnabrück University of Applied
Sciences, Osnabrueck, Germany

{p.wachter,j.schoening}@hs-osnabrueck.de

² Institute of Cognitive Science, Osnabrück University, Osnabrueck, Germany
luniehaus@uni-osnabrueck.de

Abstract. Synthetic data has emerged as a cost-effective alternative to real data for training artificial neural networks (ANN). However, the disparity between synthetic and real data results in a *domain gap*. That gap leads to poor performance and generalization of the trained ANN when applied to real-world scenarios. Several strategies have been developed to bridge this gap, which combine synthetic and real data, known as mixed training using hybrid datasets. While these strategies have been shown to mitigate the domain gap, a systematic evaluation of their generalizability and robustness across various tasks and architectures remains underexplored. To address this challenge, our study comprehensively analyzes two widely used mixing strategies on three prevalent architectures and three distinct hybrid datasets. From these datasets, we sample subsets with varying proportions of synthetic to real data to investigate the impact of synthetic and real components. The findings of this paper provide valuable insights into optimizing the use of synthetic data in the training process of any ANN, contributing to enhancing robustness and efficacy cf. ^a

Keywords: Hybrid Datasets · Mixed Training · Artificial Intelligence · Benchmark · Synthetic Data · Domain Gap · Reality Gap

1 Introduction

Synthetic data has recently gained considerable attention as a cost-effective way to generate training data [14] for artificial neural networks (ANN). Synthetic data, defined as information not derived from real-world sources, inherently lacks realism. This absence creates a discrepancy between synthetic and real examples, which can be termed the *reality gap* [25]. The reality gap is a subset of the

^a Supplementary code and results: <https://hs-osnabrueck.de/prof-dr-julius-schoening/ki2025>

broader *domain gap*, which limits the exclusive use of synthetic data in ANN training since the target domain usually is the real-world. As a result, applying ANNs, which are solely trained on synthetic data, to real-world scenarios requires domain adaptation to transfer knowledge from one domain to the other [17].

One key area of domain adaptation research focuses on enhancing the quality and diversity of synthetic data to bridge the domain gap [14]. This can be achieved by increasing the realism of synthetic data either during its generation or through post-processing techniques. Enhancing realism during generation requires an advanced simulation framework that accurately models real-world conditions [10]. Such frameworks often prove costly and labor-intensive [28]. Alternatively, image transformation methods, such as neural style transfer, improve realism in post-processing [6], although these methods often suffer from training instabilities and may introduce artifacts [32].

Another research focus targets the architecture of ANNs, aiming to enable them to learn domain-invariant features, thereby reducing the domain gap [30]. However, designing such architectures is inherently challenging because they must effectively abstract away domain-specific details to maintain generality. Moreover, many of these architectures are developed for multi-domain adaptation [16], including domains with no available data. These assumptions may limit their effectiveness in addressing the specific reality gap, where real data is usually accessible.

Consequently, this paper examines a mixed training approach for domain adaptation, which integrates synthetic and real data into a unified hybrid dataset. By integrating both types into a hybrid set, mixed training allows models to learn features from synthetic part, while real-world data introduces authentic visual and contextual variations, thereby mitigating the lack of realism in the synthetic data and bridging the domain gap. This approach can be used with every ANN architecture, thus making it broadly applicable, while omitting the need for highly specialized ANNs when relaying on synthetic sources. Mixed training prominently follows two strategies: *simple mixed* (SM), where both data types are used simultaneously, and sequentially *fine-tuned* (FT), in which ANNs are pretrained on synthetic data and afterwards fine-tuned with real data. Although these methods are widely applied, their underlying mechanisms and performance under varying conditions remains only partially understood.

In response to these limitations, this study systematically evaluates both mixed training strategies across three structurally distinct datasets, varying ratios of synthetic-to-real data and three different ANN architectures on image classification tasks. Unlike previous studies that boost model performance by expanding the hybrid dataset with additional synthetic examples, our approach maintains a constant dataset size while gradually increasing the synthetic-to-real proportion. Thus, our goal is not limited to improving the overall performance, but to conduct a precise assessment of each strategies efficacy under varying conditions. This design offers actionable insights to guide the practical application of synthetic data in real-world settings.

2 Related Work

Although many publications have successfully applied mixed training to reduce the domain gap [9,23,21,19], most studies concentrate on the generation of synthetic data rather than examining the mixed training strategies themselves [7,29].

Nowruzi et al. [15] evaluated an object detector for cars and pedestrians using three synthetic and three real datasets. They created hybrid training sets by combining synthetic data with four different, low ratios of real data. Moreover, they compared the SM strategy with the FT approach for their tasks and concluded that FT more effectively reduces the domain gap. However, the study did not consider the effects of varying dataset sizes and class distributions. In contrast, Burdorf et al. [3,18] did not confirm the advantages of FT for a very similar task as reported by Nowruzi et al. [15].

In [26], Vanherle et al. compared the SM and FT strategies across subsets with the same overall size but different synthetic-to-real ratios; notably, the subsets were imbalanced. They employ the DIMO dataset [4], which contains real and synthetic images, to train and test a Mask-Region Based Convolutional Neural Networks (Mask R-CNN) for object detection. Although the FT strategy yielded better performance, the study involved pretraining on the COCO dataset before introducing synthetic examples. Moreover, synthetic data was used solely to train the heads of the pretrained model. These methodological choices impact network performance [12] and may interact with the mixed training strategies, potentially confounding true differences.

Together, these studies leave several questions unresolved. Reported gains could be due to uncontrolled factors, like dataset size, class imbalance, or pre-training on large scale datasets with freezing of layers, that obscure the true effect of the mixing strategy. Most experiments rely on a single hybrid dataset and one ANN, making it unclear whether conclusions generalize across synthetic data types or ANN architectures. A more controlled, cross-dataset, cross-architecture analysis is therefore required, which we address in this study.

3 Methodology

The complete code, configurations and numerical results of our study can be found at 1. In general, mixed training using real and synthetic data can be applied to any neural network and hybrid dataset, regardless of the architecture, synthetic data type or synthetic-to-real ratio. Consequently, this study covers a broad range of possible configurations of these key factors to evaluate their influence on the two training strategies SM and FT. Our study is designed to expose and isolate the effects on model performance, instead of aiming for maximum accuracy in each configuration. Accordingly, we minimize other factors which can influence the training process, such as advanced optimization algorithms, regularization heuristics, data augmentation or hyperparameter tuning. In the following, we lay out the choices of our ANN architectures, hybrid datasets and synthetic-to-real proportions, as well as the implementation of the SM and FT training strategies.

3.1 Application Task

We choose image classification as the application task of our study. Image classification is considered one of the most fundamental tasks within the domain of AI-based and classical computer vision [22]. It was selected, because its significance extends beyond mere categorization: more intricate tasks, like object detection, semantic segmentation, and image generation, are derived from the same principles [24]. This allows to assume a similar behavior of the mixed training strategies on a broad range of tasks in computer vision.

3.2 Artificial Neural Networks

As noted before, mixed training can be used irrespective of the ANN architecture. Therefore, we decided to assess the impact of three commonly used architectures in AI-based computer vision. Namely, the multilayer perceptron (MLP) [22], the convolutional neural network (CNN) [8], and the vision transformer (ViT) [5]. Each of these ANNs is implemented in the original form, without additional mechanisms, like normalization or regularization layers. This reduces the number of confounding variables and allows to focus on the respective ANN’s core mechanism. However, normalization layers were incorporated into the transformer blocks and the patch encoder of the ViT, given their critical role in preserving the fundamental functionality and performance of transformer-based networks [31]. The exclusion of these layers would constitute a deviation from the fundamental design principles of transformers [27]. The general design choices, such as the width of the layers, the activation functions, the initialization, etc., reflect common practices. A detailed overview of the network configurations can be found in 1.

3.3 Datasets

Synthetic data can be generated through several distinct mechanisms [14]. For this comprehensive study on mixed training strategies, we curated three hybrid datasets, each created using a different synthetic data generation mechanism. These are generative AI (GenAI), computer aided design (CAD) and hand drawings, which results in distinct structures and characteristics of each dataset.

Cifar-10 and CiFake comprise the first dataset in which, as illustrated in Fig. 1 (a) and (d), **Cifar-10** [11] represents the real and **CiFake** [1] the synthetic part. **Cifar-10** and **CiFake** both comprise 60,000 32×32 RGB images across ten object classes, each with 6,000 images. The **CiFake** dataset is generated using the *CompVis Stable Diffusion* model [20], an open-source latent diffusion model. Additional prompt modifiers enhanced the diversity, while still producing images that closely resemble **Cifar-10**’s characteristics. Importantly, 668 duplicate image pairs were found in the **CiFake** dataset, and one from each was removed to ensure dataset integrity.

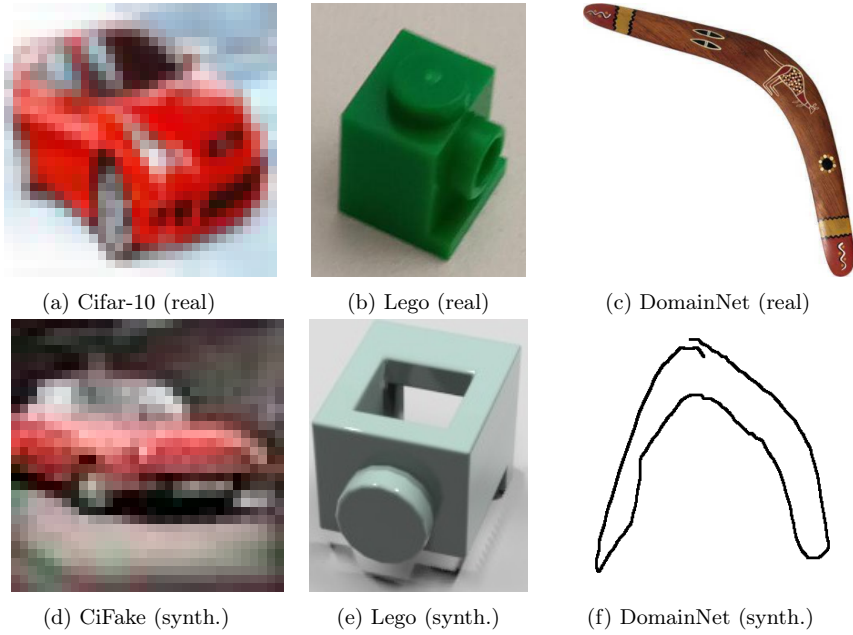


Fig. 1: Real and synthetic example images of the datasets.

LEGO Bricks [2] for training classification networks was selected as the second dataset. It's real part is shown in Fig. 1 (b), while the synthetic images, shown in Fig. 1 (e) were generated using the computer-aided design (CAD) program LDraw [13]. The synthetic images introduce additional colors and angles, but still try to mimic the real images accurately. The dataset features 447 unique LEGO brick classes with a total of 50,000 real and 560,000 synthetic images. The classes are identified by the official LEGO IDs based on shape, regardless of color or decorations. All images are RGB with varying sizes, and both datasets are highly, yet differently, unbalanced, with class samples ranging from 6 to over 600.

DomainNet [16] is the third dataset used in our study. In it's original form, it consists of six domains: Clipart, Infograph, Painting, Quickdraw, Real, and Sketch, with a total of 596,000 samples across 345 categories. We selected the Real domain as our real, and the Quickdraw domain as our synthetic data, as visualized in Fig. 1 (c) and (f). The former consists of roughly 176,000 RGB images of varying sizes and class distributions, while the latter is composed of 500 hand-drawn, black-and-white images per class, each with a size of 300×300 . This provides a third, distinct hybrid dataset compared to CiFake (GenAI) and LegoBricks (CAD), because it's synthetic part is not designed to be as similar to the real part as possible.

3.4 Dataset Creation

To examine the effect of synthetic-to-real image ratios, we generated 11 equally sized subsamples from each of the three datasets introduced in Section 3.3. For every original dataset we produced one subset made entirely of real images, one made entirely of synthetic images, and nine hybrid subsets whose composition ranges from 90 % synthetic / 10 % real to 10 % synthetic / 90 % real in 10-percentage-point steps. To eliminate potential confounding factors, we matched the class distribution across all subsets. For the nine hybrid subsets, the same class balance was enforced separately within the real and synthetic parts. The 11 synthetic-to-real ratios on the three original datasets resulted in 33 subsampled datasets for this study, with the attributes summarized in Table 1.

Furthermore, the images were rescaled to the same spatial dimensions, to match the ANN’s fixed input sizes. No rescaling was necessary for Cifar-10 and CiFake, since all images have the shape 32×32 . The LegoBricks images were rescaled to 256×256 , and all DomainNet images to 300×300 . Additionally, the grayscale images from the synthetic part of the DomainNet dataset were converted into the 3-channel RGB format, for the same reason. Lastly, all images were normalized to lie within the range $[0, 1]$, in order to ensure consistency across all RGB channels and image sources.

Table 1: Attribute overview of the (hybrid) datasets

Name	# Images	# Classes	Image Dimensions	Synth. Source
Cifar-10 [11]/Cifake [1]	55,000	10	$32 \times 32 \times 3$	GenAI
LegoBricks [2]	20,100	134	$256 \times 256 \times 3$	CAD
DomainNet [16]	30,000	60	$300 \times 300 \times 3$	Drawing

3.5 Mixed Training Strategies

Two mixed training strategies are compared in this study, which we termed *simple mixed* (SM) and *fine-tuned* (FT). The former treats the synthetic and real part entirely equivalent. It does not distinguish between synthetic and real data during training; data from both domains are utilized in the same manner, as if they were part of a single dataset. The inputs and their corresponding labels are sampled randomly from both datasets, regardless of their size or distribution. This provides the ANN with a broader range of examples, increasing its ability to adapt to various scenarios and improving its robustness.

Conversely, the FT strategy utilizes the real and synthetic parts sequentially. Only the synthetic part is used to train the network in the first step. This pretraining is stopped when there is no more improvement on the real evaluation set, implemented through validation-based early stopping. In the second step,

the real part is used to retrain the network obtained after step one. The FT strategy acknowledges the differences between data types and their relation to the real test data, and adjusts the training process accordingly.

3.6 Training Procedure

All models were training using vanilla Stochastic Gradient Descent (SGD) with a learning rate of 0.01 and mini-batches of 64 samples. Datasets were split 60%/20%/20% into training, validation, and test sets. The validation and test sets are sampled exclusively from the real part of the dataset and remained unchanged. In both SM and FT training strategies, the MLP, CNN and ViT were trained for 100 epochs, ensuring convergence. After training, the model was restored to the state where it showed the highest validation accuracy. For the FT strategy, the first training step proceeded until the validation accuracy did not improve for 10 epochs; at that moment, early stopping triggered, and the weights from the best-performing model were preserved. In the second step, the network was trained for the remaining number of epochs. All parameters remained unchanged throughout this procedure.

To summarize, we trained three neural network architectures, each with two training strategies, on 27 hybrid datasets and six non-hybrid datasets (all-real or all-synthetic). One full training cycle therefore produced 162 models trained on hybrid data and 18 models trained on non-hybrid data, yielding 180 trained networks in total. To obtain statistically robust results, we repeated this cycle ten times, resulting in 1,800 trained networks overall. At the beginning of each training cycle, all 33 datasets were resampled.

4 Results

The two baselines that guide the evaluation of the mixed training strategies are the purely synthetic and purely real dataset settings. While the former provides information about the quality of the synthetic data, the latter shows the performance when no domain gap exists between the training and test data. Table 2 reports the test accuracy of the baselines, averaged over ten repetitions. All purely synthetic setups achieve performance above chance level. This result indicates their applicability as a proxy for the real-world, although a large gap to the performance of the purely real setting is evident, highlighting the domain gap. The domain gap can be quantified as the difference between the purely real and purely synthetic results. The goal of mixed training strategies is to minimize this gap, which would result in a comparable performance to the purely real setups.

In the mixed training setting, the two training strategies, SM and FT, have resulted in divergent performances. Across all combinations of datasets, synthetic-to-real proportions, and ANN architectures, FT has outperformed SM in 635 out of 810 cases; when averaged over the ten repetitions, in 69 out of 81.

Table 2: MLP, CNN, and ViT average test accuracy without mixing (0.0=100% real, 1.0=100% synthetic) averaged over ten runs.

	Cifar-10/CiFake		LegoBricks		DomainNet	
	Real	Synthetic	Real	Synthetic	Real	Synthetic
MLP	0.5090	0.1328	0.6699	0.1423	0.2064	0.0244
CNN	0.6413	0.1430	0.5582	0.0130	0.2870	0.0353
ViT	0.5165	0.1255	0.6296	0.0125	0.2660	0.0345

The synthetic-to-real ratio of the train data strongly influences the performance of the ANNs. This influence is two-fold: it impacts performance in general and determines the magnitude of the difference between the FT and SM strategies. The general impact on the performance was observed throughout all settings. Every increase in the real proportion leads to an improved performance. This finding was expected since more real data in the training set leads to a reduction of the domain gap to the real test data. Notably, the most significant improvement was observed for the first 10% increase. The improvement made through the next increments gradually decreased. In other words, even a small amount of real data in the hybrid dataset leads to a drastic improvement in performance, and the bigger the proportion of real data in the dataset, the less improvement is made by increasing the real data proportion further.

In addition to the general role of the synthetic-to-real ratio on the performance, it also determines the magnitude of the difference between the FT and SM strategies. Fig. 2 shows a ViT trained on the Cifar-10/Cifake dataset using both strategies. Here, the difference between the two strategies increases parallel to the synthetic data proportion. In the setup with only 10% synthetic data, FT only marginally outperforms SM. With every increase of the synthetic data proportion, this difference increases as well, up to the point of 90% synthetic data, where the FT strategy performs roughly twice as good as the SM strategy.

What follows from this is that although the performance of the ANNs decrease with an increase of the synthetic proportion, this degradation of performance is less drastic for the FT strategy. Fig. 3 depicts the test accuracy gradients of the ViT trained on the Cifar-10/CiFake dataset with both strategies. The gradient shows the rate at which the performance is changing in relation to the synthetic-to-real ratio. The gradient of the FT strategy is constantly less negative compared to the SM strategy, highlighting the slower degradation of performance. In other words, the higher the proportion of the synthetic data in the dataset, the bigger is the gain in performance of the FT over the SM strategy. This tendency caused by the synthetic-to-real ratio was observed for all networks trained on the Cifar-10/CiFake and LegoBricks datasets, although less pronounced for the CNN.

In the setups that included the DomainNet dataset, FT did not clearly outperform SM, as it was the case for Cifar-10/CiFake and LegoBricks. Here, the architecture of the ANN played a crucial role. For the ViT, the FT strategy

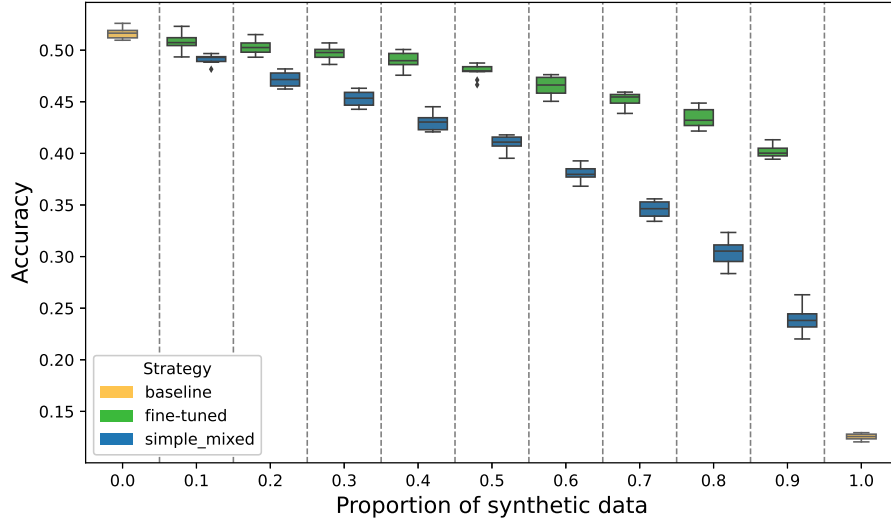


Fig. 2: ViT test accuracy on the Cifar-10/CiFake dataset (0.0=100% real, 1.0=100% synthetic) averaged over ten runs shown as boxplots.

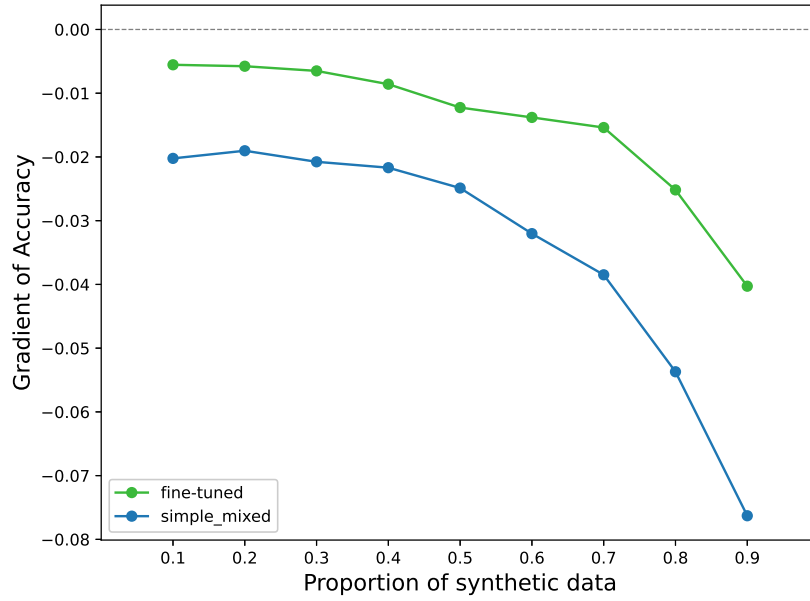


Fig. 3: Test accuracy gradients of the ViT trained on Cifar-10/CiFake datasets. The proportion of 0.0 represents 100% real, and 1.0 represents 100% synthetic data.

consistently resulted in marginally better performances. In the setups, including the MLP shown in Fig. 4, the FT strategy was better only for cases with a higher synthetic proportion. The CNN consistently performed better when trained with the SM strategy, as illustrated in Fig. 5. Notably, the more synthetic data is present, the smaller is the advantage of the SM strategy.

In addition to this preference for the SM strategy, the CNN had a higher variation throughout the 10 repetitions. The interquartile range for the SM strategy showed a high variation around the mean and the FT strategy showed significant outliers. This was observed here, as well as in the case of CNN trained on Cifar-10/Cifake and LegoBricks datasets. In Section A of the Appendix, the results of all setups can be found.

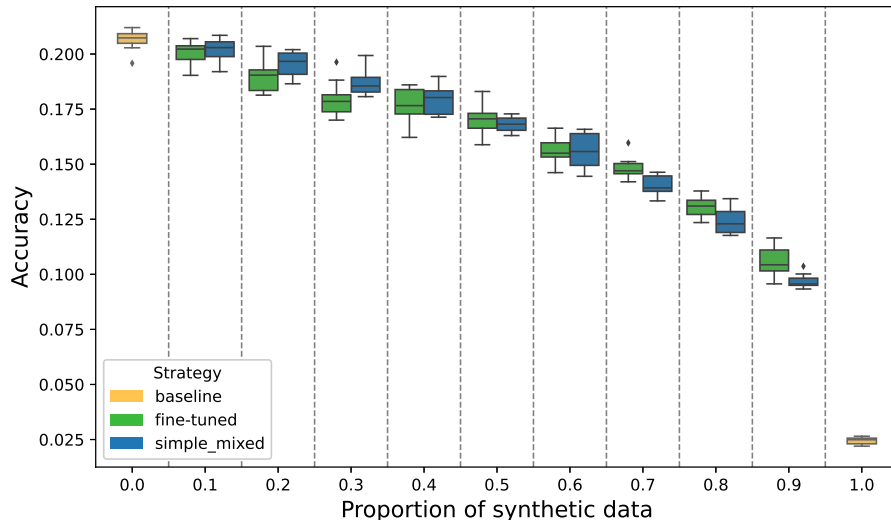


Fig. 4: MLP test accuracy on the DomainNet dataset (0.0=100% real, 1.0=100% synthetic) averaged over ten runs shown as boxplots.

5 Discussion

The results of our study indicate that FT is a preferable strategy for mixed training using real and synthetic data. Nevertheless, the CNN/DomainNet setting constitutes a clear counter-example: here simultaneous exposure to both domains (SM) yielded better performance. In this section we are going to elaborate our theories for the reasons leading to this outlier.

A convolutional layer extracts localized patterns with learnable kernels; each kernel sees only its receptive field, a region whose size equals the kernel dimensions. Stacking convolutional layers (with larger kernels, strides, or pooling)

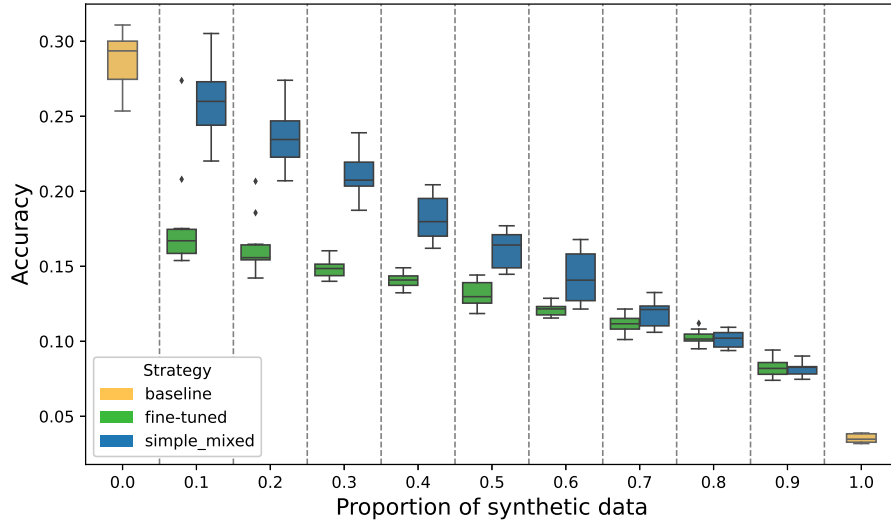


Fig. 5: CNN test accuracy on the DomainNet dataset (0.0=100% real, 1.0=100% synthetic) averaged over ten runs shown as boxplots.

expands the receptive field, such that deeper layer can capture increasingly complex structures. This, however, relies entirely on the patterns captured by earlier layers; if those first layers miss important structures, later layers have nothing to build on. High-quality, diverse features at the network’s outset are therefore critical to the overall performance.

DomainNet exhibits a larger domain gap compared to Cifar-10/CiFake and LegoBricks. The latter two contain colored, texture-matched synthetic images, meticulously designed to mimic their real counterpart, while DomainNet’s synthetic subset consists solely of black-and-white sketches with only two pixel values, sharp edges and no texture (Fig. 3.3).

Pre-training a CNN on these simple, synthetic images biases its early layers toward extreme black-white edges. Kernels aligned with these simple pattern receive large gradients during training, increasing a few weights rapidly, while suppressing the rest. The result is comparable to Sobel or Prewitt kernels, whose few large coefficients emphasize a single-oriented edge, whereas kernels that would encode more fine grained patterns need many smaller, finely balanced coefficients. Once dominated by such coarse detectors, the early layers fail to capture more complex patterns, leaving deeper layers without the information necessary to form complex features. The resulting kernels make subsequent fine-tuning difficult: gradients become ill-conditioned and struggle to reshape the early layers once real data are introduced in the fine-tuning step. In summary, the network is effectively trapped in a local minimum shaped by oversized and diminishing small weights.

6 Conclusion

Our study reveals three findings that refine the current understanding of mixed training using real and synthetic data. First, the fine-tuning strategy usually, but not always, outperforms the simple mixed strategy. Across 89 individual configurations FT surpassed SM in 69 (78 %), particularly when synthetic data comprised a large portion of the dataset. This indicates that it is more effective in utilizing real data to bridge the domain gap. Nevertheless, the CNNDomainNet setting constitutes a clear counter-example. Thus, architecture-data interactions, as outlined in 5, influence the strategies’ success.

Second, diminishing returns of real data. The largest performance gains occurred with an initial 10% increase in real data; further increases yielded progressively less. Practically, adding a modest subset of real data to a synthetic set can markedly boost performance, providing a possible cost-effective solution in many cases.

Third, Interaction of strategy with domain gap For the two synthetic sources that explicitly mimic the real domain (GenAI and CAD), FT was consistently advantageous. Conversely, hand-drawn sketches of the DomainNet dataset constitute large discrepancy to the real images, semantically as well as visually. Under such conditions the SM strategy can outperform FT. This suggests that the “optimal” strategy could be chosen based on a quantifiable domain gap measure.

Future work should test mixed training on richer tasks, like object detection, instance or semantic segmentation, and with production-grade networks that include modern regularization and optimization methods. Furthermore, it should develop a robust domain-gap metric that combines low-level visual statistics with high-level semantic information. A reliable score would enable the prediction of the required real-data fraction and select the appropriate mixing strategy, turning today’s trial-and-error process into a more systematic workflow.

Acknowledgments. This work is part of the AgrifoodTEF-DE project. AgrifoodTEF-DE is supported by funds of the Federal Ministry of Agriculture, Food and Regional Identity (BMLEH) based on a decision of the Parliament of the Federal Republic of Germany via the Federal Office for Agriculture and Food (BLE) under the research and innovation program ‘Climate Protection in Agriculture’.

The computation of this research was done, using computing resources of the High-Performance Computing (HPC) cluster of the Osnabrück University of Applied Sciences, which were provided by the German Federal Ministry of Research, Technology and Space (BMFTR) within the HiPer4All@HSOS project.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

A Appendix

A.1 Results on Cifar-10/CiFake

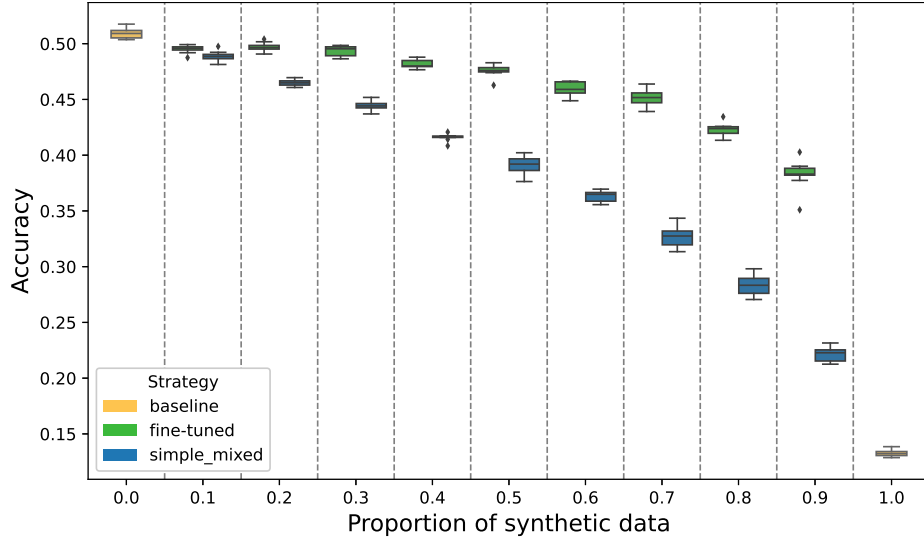


Fig. 6: MLP test accuracy on the Cifar-10/Cifake dataset (0.0=100% real, 1.0=100% synthetic) averaged over ten runs shown as boxplots.

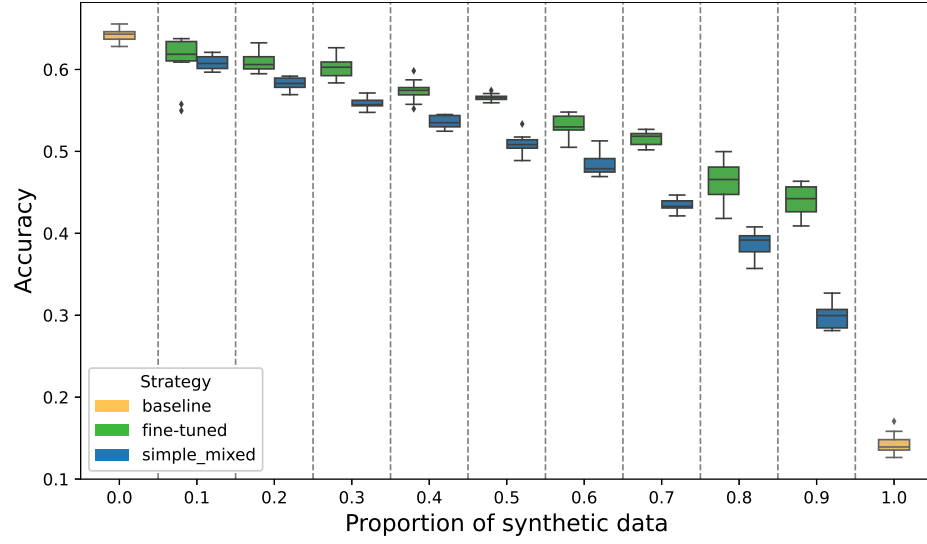


Fig. 7: CNN test accuracy on the Cifar-10/Cifake dataset (0.0=100% real, 1.0=100% synthetic) averaged over ten runs shown as boxplots.

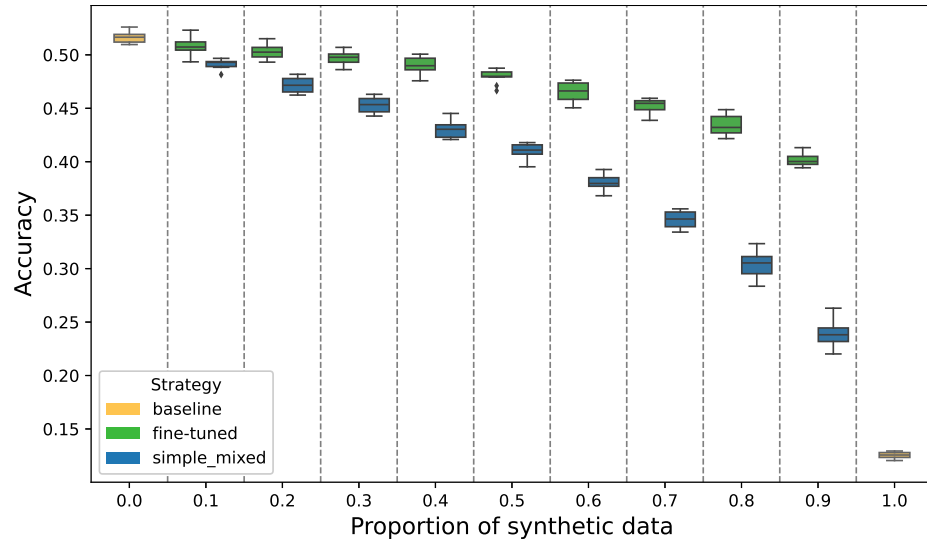


Fig. 8: ViT test accuracy on the Cifar-10/Cifake dataset (0.0=100% real, 1.0=100% synthetic) averaged over ten runs shown as boxplots.

A.2 Results on LegoBricks

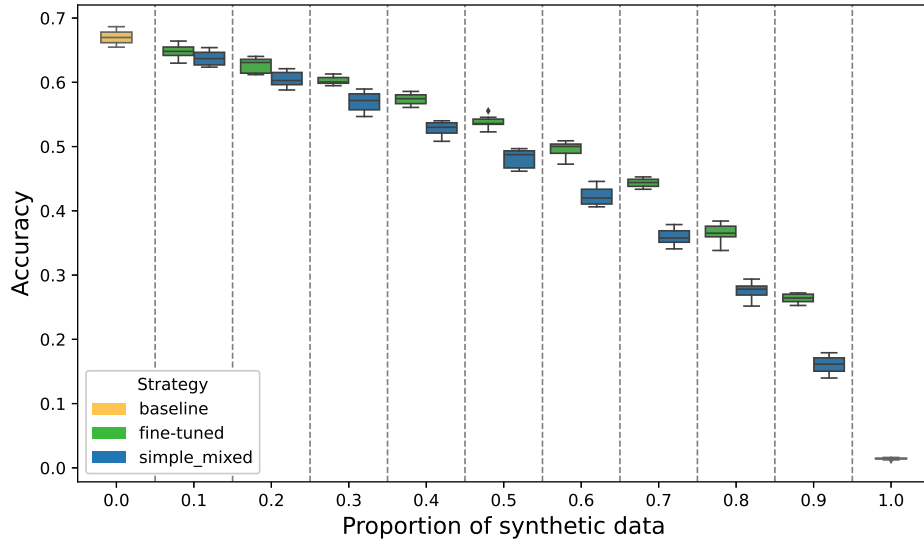


Fig. 9: MLP test accuracy on the LegoBricks dataset (0.0=100% real, 1.0=100% synthetic) averaged over ten runs shown as a boxplot.

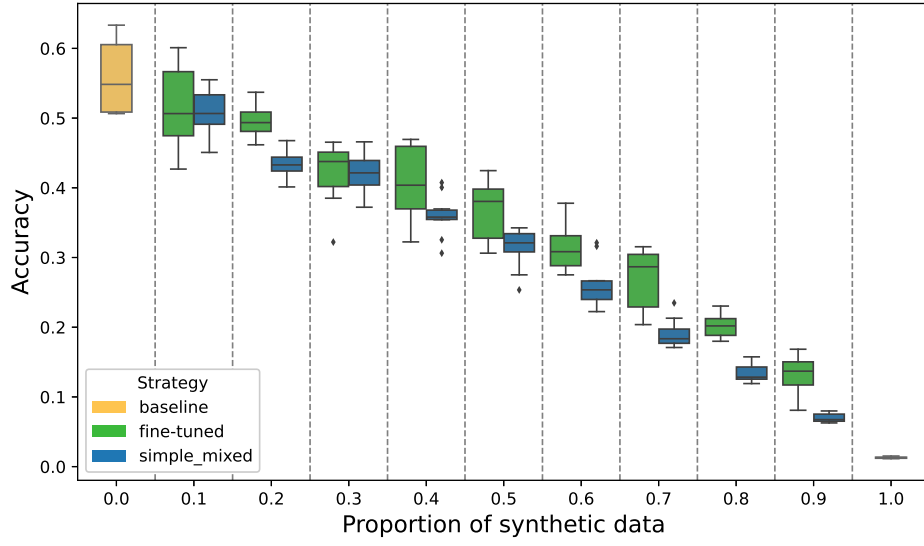


Fig. 10: CNN test accuracy on the LegoBricks dataset (0.0=100% real, 1.0=100% synthetic) averaged over ten runs shown as a boxplot.

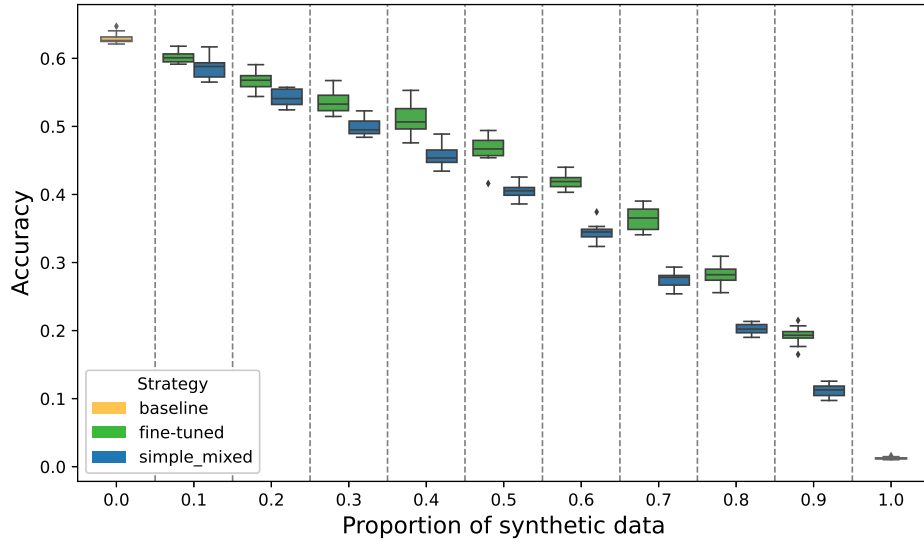


Fig. 11: ViT test accuracy on the LegoBricks dataset (0.0=100% real, 1.0=100% synthetic) averaged over ten runs shown as a boxplot.

A.3 Results on DomainNet

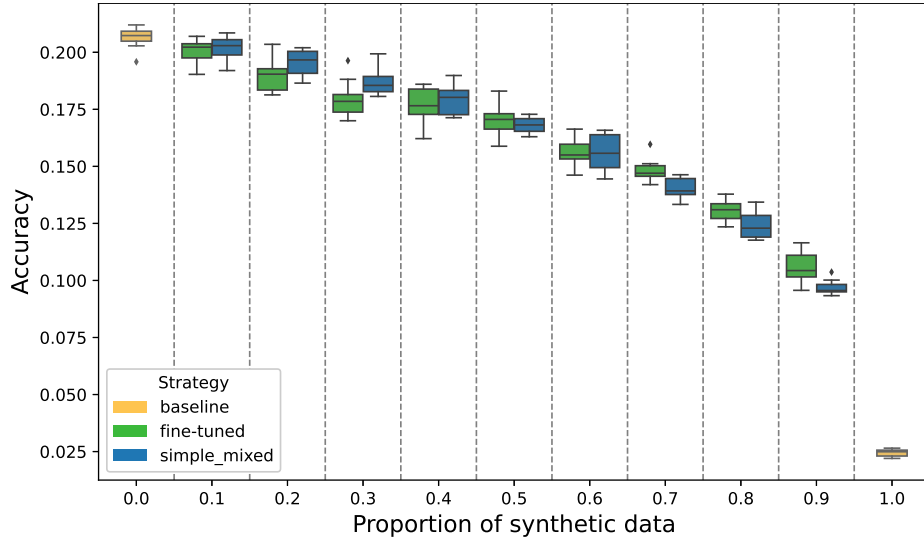


Fig. 12: MLP test accuracy on the DomainNet dataset (0.0=100% real, 1.0=100% synthetic) averaged over ten runs shown as a boxplot.

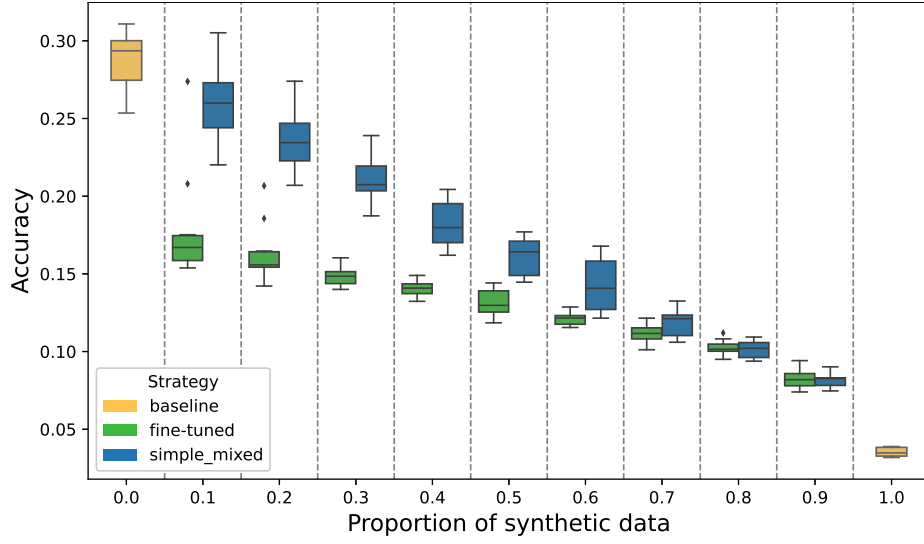


Fig. 13: CNN test accuracy on the DomainNet dataset (0.0=100% real, 1.0=100% synthetic) averaged over ten runs shown as a boxplot.

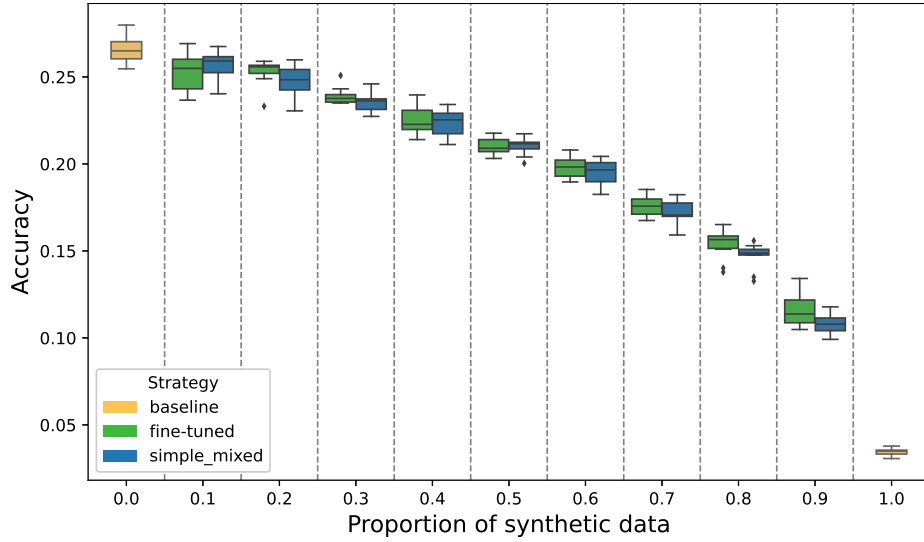


Fig. 14: ViT test accuracy on the DomainNet dataset (0.0=100% real, 1.0=100% synthetic) averaged over ten runs shown as a boxplot.

References

1. Bird, J.J., Lotfi, A.: Cifake: Image classification and explainable identification of ai-generated synthetic images. *IEEE Access* **12**, 15642–15650 (2024). <https://doi.org/10.1109/access.2024.3356122>
2. Boiniski, T., Zaraziński, S., Śledź, B.: Lego bricks for training classification network (2021). <https://doi.org/10.34808/RCZA-JY08>
3. Burdorf, S., Plum, K., Hasenklever, D.: Reducing the amount of real world data for object detector training with synthetic data (2022). <https://doi.org/10.48550/ARXIV.2202.00632>
4. De Roovere, P., Moonen, S., Michiels, N., Wyffels, F.: Dataset of industrial metal objects (2022). <https://doi.org/10.48550/ARXIV.2208.04052>
5. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (ICLR) (2021), <https://openreview.net/forum?id=YicbFdNTTy>
6. Ettedgui, S., Abu-Hussein, S., Giryas, R.: ProCST: Boosting semantic segmentation using progressive cyclic style-transfer (2022). <https://doi.org/10.48550/ARXIV.2204.11891>
7. Eversberg, L., Lambrecht, J.: Generating images with physics-based rendering for an industrial object detection task: Realism versus domain randomization. *Sensors* **21**(23), 7901 (Nov 2021). <https://doi.org/10.3390/s21237901>
8. Fukushima, K.: Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics* **36**(4), 193–202 (Apr 1980). <https://doi.org/10.1007/bf00344251>
9. Kim, J., Kim, D., Lee, S., Chi, S.: Hybrid dnn training using both synthetic and real construction images to overcome training data shortage. *Automation in Construction* **149**, 104771 (May 2023). <https://doi.org/10.1016/j.autcon.2023.104771>
10. Klein, J., Waller, R., Pirk, S., Paľubicki, W., Tester, M., Michels, D.L.: Synthetic data at scale: a development model to efficiently leverage machine learning in agriculture. *Frontiers in Plant Science* **15** (Sep 2024). <https://doi.org/10.3389/fpls.2024.1360113>
11. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Tech. rep., University of Toronto, Toronto, Ontario (2009), <https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf>
12. Lambrecht, J., Kästner, L.: Towards the usage of synthetic data for marker-less pose estimation of articulated robots in rgb images. In: 2019 19th International Conference on Advanced Robotics (ICAR). pp. 240–247. IEEE (Dec 2019). <https://doi.org/10.1109/icar46387.2019.8981600>
13. LDraw.org: Content and licensing information (2020), <https://www.ldraw.org/>, accessed: 2025-04-24
14. Nikolenko, S.I.: Synthetic Data for Deep Learning. Springer International Publishing (2021). <https://doi.org/10.1007/978-3-030-75178-4>
15. Nowruz, F.E., Kapoor, P., Kolhatkar, D., Hassanat, F.A., Laganieri, R., Rebut, J.: How much real data do we actually need: Analyzing object detection performance using synthetic and real data (2019). <https://doi.org/10.48550/ARXIV.1907.07061>
16. Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., Wang, B.: Moment matching for multi-source domain adaptation. In: 2019 IEEE/CVF International Conference

- on Computer Vision (ICCV). pp. 1406–1415. IEEE (Oct 2019). <https://doi.org/10.1109/iccv.2019.00149>
17. Peng, X., Usman, B., Kaushik, N., Wang, D., Hoffman, J., Saenko, K.: Visda: A synthetic-to-real benchmark for visual domain adaptation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 2102–21025. IEEE (Jun 2018). <https://doi.org/10.1109/cvprw.2018.00271>
 18. Poucin, F., Kraus, A., Simon, M.: Boosting instance segmentation with synthetic data: A study to overcome the limits of real world data sets. In: 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). pp. 945–953. IEEE (Oct 2021). <https://doi.org/10.1109/iccvw54120.2021.00110>
 19. Rajpura, P.S., Bojinov, H., Hegde, R.S.: Object detection using deep cnns trained on synthetic images (2017). <https://doi.org/10.48550/ARXIV.1706.06782>
 20. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10674–10685. IEEE (Jun 2022). <https://doi.org/10.1109/cvpr52688.2022.01042>
 21. Ros, G., Sellart, L., Materzynska, J., Vazquez, D., Lopez, A.M.: The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (Jun 2016). <https://doi.org/10.1109/cvpr.2016.352>
 22. Schmidhuber, J.: Annotated history of modern ai and deep learning (2022). <https://doi.org/10.48550/ARXIV.2212.11279>
 23. Staniszewski, M., Kempinski, A., Marczyk, M., Socha, M., Foszner, P., Cebula, M., Labus, A., Cogiel, M., Golba, D.: Searching for the ideal recipe for preparing synthetic data in the multi-object detection problem. *Applied Sciences* **15**(1), 354 (Jan 2025). <https://doi.org/10.3390/app15010354>
 24. Szeliski, R.: *Computer Vision: Algorithms and Applications*. Springer International Publishing, 2nd edn. (2022). <https://doi.org/10.1007/978-3-030-34372-9>
 25. Tremblay, J., Prakash, A., Acuna, D., Brophy, M., Jampani, V., Anil, C., To, T., Cameracci, E., Boochoon, S., Birchfield, S.: Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 1082–10828. IEEE (Jun 2018). <https://doi.org/10.1109/cvprw.2018.00143>
 26. Vanherle, B., Moonen, S., Reeth, F.V., Michiels, N.: Analysis of training object detection models with synthetic data. In: 33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21–24, 2022. BMVA Press (2022), <https://bmvc2022.mpi-inf.mpg.de/0833.pdf>
 27. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) *Advances in Neural Information Processing Systems*. vol. 30. Curran Associates, Inc. (2017), https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
 28. Vuletic, J., Polic, M., Orsag, M.: Procedural generation of synthetic dataset for robotic applications in sweet pepper cultivation. In: 2022 International Conference on Smart Systems and Technologies (SST). pp. 309–314. IEEE (Oct 2022). <https://doi.org/10.1109/sst55530.2022.9954643>
 29. Wachter, P., Kruse, N., Schöning, J.: Synthetic fields, real gains: Enhancing smart sgriculture through hybrid datasets. In: *Informatik in der Land-, Forst-und Ernährungswirtschaft-Fokus: Biodiversität fördern durch digitale Landwirtschaft*.

- pp. 437–442. Gesellschaft für Informatik e.V. (2024). https://doi.org/10.18420/GILJT2024_44
30. Wang, M., Deng, W.: Deep visual domain adaptation: A survey. *Neurocomputing* **312**, 135–153 (Oct 2018). <https://doi.org/10.1016/j.neucom.2018.05.083>
 31. Xiong, R., Yang, Y., He, D., Zheng, K., Zheng, S., Xing, C., Zhang, H., Lan, Y., Wang, L., Liu, T.Y.: On layer normalization in the transformer architecture. In: Proceedings of the 37th International Conference on Machine Learning. ICML’20, JMLR.org (2020), <https://dl.acm.org/doi/10.5555/3524938.3525913>
 32. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 2242–2251. IEEE (Oct 2017). <https://doi.org/10.1109/iccv.2017.244>