From Sources to Solutions: Enhancing Object Detection Models through Synthetic Data

Eduard Bartolovic
Munich University of Applied Sciences
Germany, Munich, Bavaria
eduard.bartolovic@hm.edu

Tobias Höfer
Munich University of Applied Sciences
Germany, Munich, Bavaria
tobias.hoefer@hm.edu

Clemens Hage
BSH Hausgeräte GmbH
Germany, Munich, Bavaria
clemens.hage@bshg.com

Alfred Nischwitz
Munich University of Applied Sciences
Germany, Munich, Bavaria
alfred.nischwitz@hm.edu

Abstract

Object detection, a fundamental task in computer vision, plays a crucial role in various applications such as autonomous driving, surveillance, and robotics. However, training models for this task require vast amounts of high-quality data, often involving labor-intensive manual labeling. Synthetic data, a promising alternative, remains an active area of research. This paper presents a comprehensive exploration of different object sources for the use of synthetic data in enhancing object detection models. We investigate various synthetic data generation techniques to implant objects into a scene, with a focus on enhancing training data diversity. These objects are either gathered from the training dataset itself using SegmentAnything as a new supervised self augmentation technique or imported from external sources, including a photobox with a rotating table and web scraping of online shops. Moreover, our study delves into the development of a placement logic that gradually evolves from placing objects randomly to placing objects in physically correct orientations to mimic the real world data. We investigate the use of different blending techniques. The outcome of our study demonstrates that synthetic images, when integrated with an existing real training set, substantially improve the object recognition accuracy of the model without compromising inference time. Our code can be found at `https://github.com/EduardBartolovic/synthetic-data-generation`.

Keywords

Synthetic Data Generation, Data Augmentation, Domain Randomization, Object Detection, SegmentAnything, YOLOv5

1 INTRODUCTION

In the current landscape, object detection algorithms and model architectures stand as remarkably powerful tools [1]. However, the efficacy of object detection models is profoundly influenced by the availability and diversity of training data. Traditional labeling relies heavily on manual labor, which is often time-consuming, error-prone and expensive. Occasionally, data privacy regulations further complicate the collection of substantial data volumes. Moreover, obtaining real-world data that covers a wide range of scenarios can be challenging or even impractical, especially in niche areas where there is no large publicly-accessible dataset readily available. In some cases, object detectors need to be trained for future scenarios before real training data is available, such as when introducing a new product. This has led researchers to explore alternatives, such as the creation of synthetic data. Several synthetic data generation techniques have gained popularity, including 3D Rendering [2]–[4], Generative Models [5]–[9] and 2D Image Implantation [10], [11]. While 3D rendering can produce more realistic scenes and objects, it requires significant computational resources and modeling effort compared to the more straightforward 2D image implantation. Generative models, such as GANs or diffusion models, are known to be challenging to train and can be computationally expensive, and they are not yet fully capable of generating realistic images across all categories [8]. 2D implantation with our proposed refinement techniques emerges as a practical solution for easy synthetic data generation. We propose a workflow involving the gathering and cropping of objects from different sources and their implantation into a scene. This approach, detailed further in Section 4, is notably simpler than other methodologies, making it suitable for a wide range of use cases. In our study, we enhance the 2D image im-

plantation technique from [10], [11], focusing more intensely on the object sources and also applying it to a more modern object detection algorithm.
One key aspect of synthetic data generation is the origin of placed objects. Notably, prior research [10], [11] has frequently overlooked the origin of objects employed in synthetic data generation, with a tendency to favor a single object source. In contrast, this study seeks to provide comprehensive insights into this crucial aspect of the generation process. From examining the origins of objects used in synthetic data generation, we find that they typically stem from either 3D CAD models or photographic representations. We conducted an assessment of both internal and external object sources to augment the scope of our training data. To extract objects from the training data we used SegmentAnything [12] as a self augmentation method. This technique takes advantage of the existing annotated training dataset to extract objects, thereby contributing to the creation of synthetic data that closely adheres to the distribution of real-world objects. This approach is a unique form of supervised self augmentation. In addition to the self augmentation method, we also explore the incorporation of foreign objects sourced from external sources. These foreign objects are gathered via web scraping or a 360° photobox. By introducing objects from other sources, we aim to enhance the adaptability of the object detection model to new scenarios and unforeseen objects. This cross-domain synthesis holds the potential to imbue the model with a broader perceptual scope, facilitating its performance in scenarios beyond those present in the original training dataset. In summary, our contributions are as follows:

- **A Scalable Method for Synthetic 2D Data Integration:** We introduce an easy, comprehensive, and highly scalable method for seamlessly integrating synthetic 2D data into the training of object recognition models. This approach significantly reduces the need for human labeling, making it more efficient and adaptable to enlarge the training dataset without affecting inference time. This method is more accurate and less error-prone than traditional labeling. This is a solution, particularly in scenarios where there is a constraint on the availability of training images and labeling resources.
- **Overview of suitable object sources:** Additionally, we provide a comprehensive overview of suitable object sources, like reusing training dataset objects, adding objects from webscraped webshops, and a photobox with a rotating table. This information equips practitioners to choose the most suitable source based on their specific use case.
- **Integration of Synthetic data into realistic data:** We explore how to effectively mix generated data with real data, aiming to discover the ideal balance between the two.

2 RELATED WORKS

In an ideal scenario, synthetic data would seamlessly merge with real-world data, creating a high-fidelity blend with a minimal reality gap. However, achieving this perfection across the board is often hindered by technological limitations or becomes feasible only through an impractical allocation of resources. The question arises: Is it even necessary to make synthetic data perfectly realistic? According to [13], absolute realism is not always essential. A workaround called domain randomization can be employed. This technique introduces random variations to the training data, including changes in lighting, backgrounds, object placements, and more. The concept behind domain randomization is to expose the model to a diverse set of situations during training, ensuring that the synthetic domain encompasses a wide range of possibilities. This approach aims to equip a model trained on synthetic data to perform well in real-world scenarios [13]. However, a minimum level of realism remains necessary and beneficial for success. To achieve this, it is valuable to explore past research in synthetic image data generation. While this study primarily focuses on 2D image composition, insights from other methodologies can provide valuable perspectives. All methods can be categorized into four main approaches:

2D Image Composition: In this approach, 2D images are incorporated by implanting them into another image. It stands out as the simplest among the considered methods. This approach has been used by some studies [10], [11]. While [10] uses a placement logic [11] only places objects randomly. Drawing inspiration from these established workflows, our research places a special emphasis on the sources of the implantation objects. Moreover, our research undertakes the challenge of working with a highly complex dataset, characterized by significant variance within a single object category.

Full 3D Rendering: This approach aims to render complete 3D scenes through the utilization of 3D assets and the modeling of entire 3D environments. Notably, it has been employed in studies [13] and has attracted attention from major companies like NVIDIA, signifying a current trend in the field [2]–[4]. While this method is capable of generating high-quality synthetic data, it is also the most challenging in terms of design and implementation. These efforts often involve complex procedures to simulate real-world environments and object interactions, facilitating the creation of diversified training data.

3D rendered objects into a 2D Image: This approach is a combination of the previous two. It involves the integration of 3D rendered objects into a real scene. The studies [14]–[16] try to enhance the realism of synthetic data by embedding rendered 3D objects seamlessly into 2D scenes, offering a middle ground between complexity and simplicity in the generation process. One potential issue in this method, however, is ensuring the availability of detailed 3D CAD models.

Generative Models: Recent advancements in generative AI, such as GANs or diffusion models, have introduced a novel approach to synthetic data generation. Models like DALL-E 4 by OpenAI [5], [6] or Stable Diffusion XL by Stability AI [7] can create images based on descriptions or natural language. However, these models still face challenges in consistently generating realistic images across all categories [8]. It's crucial to note that synthetic data produced by generative models might unintentionally replicate or intensify existing biases [9].

All of these approaches try to handle the reality gap differently. Some papers [10], [11], [15] try to narrow this gap by generating more realistic images. This process can be called system identification [13] which is the process of tuning the parameters to match the distribution of the real world. For example, this is done by placing objects with a realistic object arrangement into a scene or using blending techniques to reduce boundary artifacts. Additionally, the concept of domain adaptation is introduced, particularly through the use of Generative Adversarial Networks (GANs) for image enhancement. This further contributes to reducing the gap between synthetic and real-world data. The majority of the referenced papers attempt to use the aforementioned domain randomization. While most studies have focused on domain-specific data, restricting broader application, they collectively show that synthetic imagery can enhance model performance. Our research draws insights from these studies. Considering the challenges in our study, it is crucial to re-evaluate methods for optimal applicability.

3 DATASET

In this study, a domain-specific dataset is used to investigate the efficacy of synthetic data in enhancing object detection models. We focused testing the methods on a closed dataset of detecting milk and milk alternatives stored in tetrapaks and bottles within refrigerators. This specialized dataset encompassed a variety of scenarios, lighting conditions, orientations, and clutter levels commonly encountered within fridge interiors. This dataset is collected with smart refrigerators equipped with a camera system. We focused on the camera looking at the fridge door. Mentioned cameras produce high resolution images with a resolution of 1920x2560 pixels. It's important to note that our dataset is geographically limited to locations within Germany, and the image capture period spans a single year. Additionally, this dataset exhibited strong imbalances. Both the distribution of objects and the variety of refrigerator models present in the dataset were notably skewed. This imbalance is a natural consequence of the dataset's real-world origin. Notably, larger refrigerators are less prevalent, yet they present a more challenging detection environment due to their pronounced viewing angles. The milk training dataset comprises 1190 images, while an additional 425 images are reserved for validation purposes. For the test dataset, 1042 images are used. This dataset, in terms of diversity and size, is more limited compared to other object classification datasets like COCO [17]. Furthermore, the images in the test dataset were captured on distinct refrigerators, reducing the potential for knowledge transfer from the training dataset.

4 METHODOLOGY TO GENERATE SYNTHETIC IMAGES

Figure 1 provides a comprehensive overview of the key steps involved in the synthetic image generation process. Additionally, it contextualizes the generation of synthetic images within the broader framework of training and evaluation, offering a holistic perspective on the integration of synthetic data into the model development pipeline.

1. **Gathering of Implantation Objects:** The first step is to collect instances which can be implanted into a scene.
2. **Gathering of Implantation Backgrounds:** Scenes where these objects can be situated are gathered.
3. **Placement Logic:** With both objects and backgrounds at hand, decisions are made regarding the placement of objects within the scenes.
4. **Blending:** This step focuses on implanting an object into a specified location on a background.

4.1 Gathering of Implantation Objects

The acquisition of implantation objects, crucial for the synthetic data generation process, involves strategic choices to ensure both diversity and relevance. In this study, three distinctive approaches were explored. In the exploration of suitable implantation objects, we deliberately avoided the use of 3D rendered objects due to their inherent complexity.

Self augmentation:
The self augmentation methodology was applied to

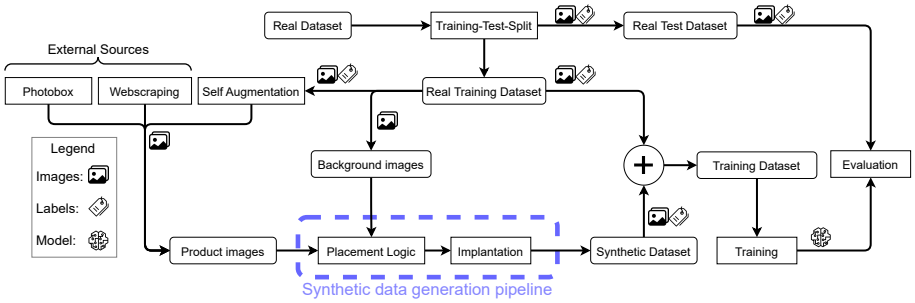

Figure 1: Illustration of the workflow as a flowchart.

extract objects directly from the pre-existing training dataset. For this approach, we used SegmentAnything [12] as a segmentation algorithm. The bounding box information from the training dataset was used to guide the segmentation process. This approach aligns with the statistical distribution of real-world objects in the dataset and introduces a unique augmentation technique. A noteworthy advantage of this technique lies in its efficiency, requiring relatively small manual intervention. SegmentAnything performs especially well on the milk dataset because its objects are relatively simple, making it a particularly effective approach in this context. An additional benefit of this technique is that it introduces significant variability into the dataset by placing objects randomly in new positions. In cases where the dataset contains a high percentage of incorrect or mislabeled objects, self augmentation may inadvertently reproduce these inaccuracies.

Web scraping:
By web scraping images from online retail platforms, it is possible to incorporate foreign objects from external sources to enhance the diversity of the dataset. Web scraping can be performed either manually or automatically using libraries like Beautiful Soup [18]. Some images may not even require cropping because they already have an alpha channel, which simplifies the incorporation of objects into our dataset. The scraping approach introduced a wider range of object variations, supplementing our dataset with new objects that might not have been adequately covered in the initial training data. An interesting aspect of this technique is its ability to easily add regularization objects. These are synthetic objects without labels placed into the scene alongside the labeled instances. This encourages the learning algorithm to focus on more than just the object boundaries when detecting objects. Furthermore, the inclusion of regularization objects can help mitigate issues related to false positive classifications. For example, in the milk dataset, juice tetrapaks are added as regularization objects. This methodology potentially expands the model's ability to recognize new objects that it has not been previously exposed to. This approach is considerably less labor-intensive compared to manual labeling. In the context of our study, we scraped webshops which sell groceries to extract product photos of fridge related objects.

360° Photobox:
In image creation, we utilize a 360° photobox featuring a rotating table. This setup captures object photographs from all perspectives, requiring some human effort. However, it provides a distinct advantage by offering multi-angle views of objects. Following a similar rationale, it is also possible to incorporate foreign objects from external sources that could potentially address gaps in our initial training data coverage. Furthermore, this approach grants us the ability to include specific objects that have historically performed poorly in our model. This is also the case for the inclusion of objects that may emerge in the future. This technique also has the ability to easily add regularization photos.

Figure 3 shows examples of all different object sources. In this study, we conducted a comparative analysis of the aforementioned data gathering techniques to offer guidance for future projects. Depending on the project's setting, one or a combination of these techniques can prove useful in enriching the dataset and improving the model's performance.

4.2 Gathering of Implantation Backgrounds

The implantation backgrounds serve as the canvas upon which objects are implanted. The selection and design of implantation backgrounds is important. The inclusion of a substantial number of backgrounds is a critical element for the domain randomization [13]. In the context of our milk dataset, the selection of implantation backgrounds consists of training images that depict empty fridges, or at the very least, fridges with some free space. The process of selecting these backgrounds can be executed manually or alternatively, by using a simple algorithm to automate the selection. The algorithm would analyze the presence or absence of labels, enabling it to identify which images depict empty spaces suitable for use as implantation backgrounds. An illustrative example image is provided in Figure

2. Furthermore, different refrigerator models are used. The use of different refrigerator models serves a dual purpose. It not only enhances background diversity, but also addresses potential imbalances in the dataset, ensuring a more representative and comprehensive training environment for our model. A crucial question to ask is how many implantation backgrounds should be used, in order to strike the right balance between diversity and the effort of collecting backgrounds.

4.3 Placement Logic

The strategic placement of the aforementioned implantation objects within scenes serves as the most important part of the system identification process, as it significantly contributes to the creation of more realistic images. In our comparison, we investigate two approaches: random placement and the proposed placement logic. While random placement lacks realism and coherence, the placement logic aims to emulate real-world spatial relationships and interactions, observed in reality. This involved implementing techniques that consider object size, occlusions, free space, foreground, and viewing angles, enhancing the verisimilitude of the generated data. For example, in the milk dataset, we put the objects where they would be in a real fridge, not just floating in the air. They go on the shelf like they do in a real fridge. It's important to note that certain aspects of this placement logic require manual labor for every background image. For example, defining a placement area, which dictates where objects are allowed to be positioned to ensure physically accurate placement. Additionally, for the reconstruction of foreground elements, the fridge holding bar needs to be accurately masked. One of these background images can be seen in the figure 2. A visual comparison between random placement and our placement logic can be seen in figure 3.

(a) (b) (c)

Figure 2: Example input background image to generate synthetic images: (a) Example background image; (b) with placement area mask: the white stripe in the middle row; (c) with foreground fridge holding bar.

4.4 Blending

The integration of synthetic objects into scenes involves a nuanced process of blending that directly impacts the visual cohesiveness and realism of the generated data. Blending can be crucial, especially because convolutional networks pay attention to edges and boundaries when recognizing objects. In this study, a list of blending techniques are evaluated. Beginning with a baseline "no blending" approach, where objects are inserted into scenes without any subsequent blending adjustments, often resulting in undesirable image artifacts. Subsequently, we tried seamless blending techniques, which prioritize the natural integration of synthetic objects into scenes. This process entails careful adjustment of object colors and lighting, creating a more visually coherent result. The application of seamless blending aims to mitigate the discernible boundaries between inserted objects and their surroundings, cultivating a more genuine representation that aligns with human visual perception [19]. Furthermore, our investigation explored the utilization of pyramid blending, a sophisticated technique that capitalizes on multi-resolution image processing to achieve a seamless fusion of objects into scenes. This approach involves creating hierarchical image pyramids that progressively refine object integration at different scales, resulting in a harmonious blend that accommodates diverse scene complexities and object scales [20], [21]. By systematically examining these diverse blending strategies, the study aims to find the most suitable blending technqiue. Figure 3 shows an image generated by the synthetic image pipeline.

(a) (b) (c)

Figure 3: Examples of synthetic images: (a) Web-sourced tetrapak centrally placed using placement logic, seamless blending and reblending of the holding bar; (b) Self augmented milk bottle centrally placed with placement logic, seamless blending and reblending of the holding bar; (c) Photobox-sourced objects seamlessly blended centrally but lacking placement logic and holding bar reblending.

5 EXPERIMENTS AND RESULTS

For the experiments we used a YOLOv5 [22] model, a widely recognized platform well-suited for object detection tasks. To speed up the training process we used a model pretrained on the COCO dataset [17]. Employing early stopping, we ensured that the training process halts once the model's performance ceases to improve on the validation set.
To establish a robust baseline for our experimentation,

the model underwent training on non-synthetic data across six distinct runs, each initiated with different seeds. To evaluate a synthetic data generation configuration, six different synthetic datasets were generated, each comprising approximately 1000 images. The repetition was necessary to account for the inherent randomness in the data generation process. This randomness includes factors like the choice of background, the selection and quantity of objects, and the positioning of these objects. Following the generation phase, these datasets were randomly combined with the real dataset. We conducted the following main experiment groups:

1. An explorative analysis of various object sources, incorporating different blending techniques, and evaluating the impact of placement logic. We tested sources like self augmentation, webscraping, photobox, and combinations of these to create a diverse object pool.

2. An investigation into the influence of the number of backgrounds used during image generation. We systematically increased the number of background images from 1 to 50.

3. An analysis of the number and ratio of synthetic images relative to the real dataset. We explored multiple configurations with artificially reduced real datasets and significantly increased synthetic images.

The outcomes of the experiments are systematically evaluated and compared based on their mean Average Precision (mAP) scores. The results are shown in figure 4 and table 1. A model trained solely on real world data exhibited an average mAP of 54.4, reflecting the challenging nature of the environment. The incorporation of synthetic data demonstrated, on average, a 2.32% increase in mAP. However, a deeper analysis is important to distinguish the specific synthetic data generation configurations that proved to be helpful and those that didn't.

5.1 Object source, Placement logic and Blending

Self augmentation: The use of self augmentation objects in synthetic datasets increased the mAP by an average of 1.0%. Incorporating a placement logic had a positive impact on mAP, while blending techniques unexpectedly seemed to lower the results. This highlights the importance of a good synthetic data generation, as incorrect methodologies can potentially degrade the model's performance. The best result is achieved by using a placement logic and simple stamping resulting in a 3.55% improvement.

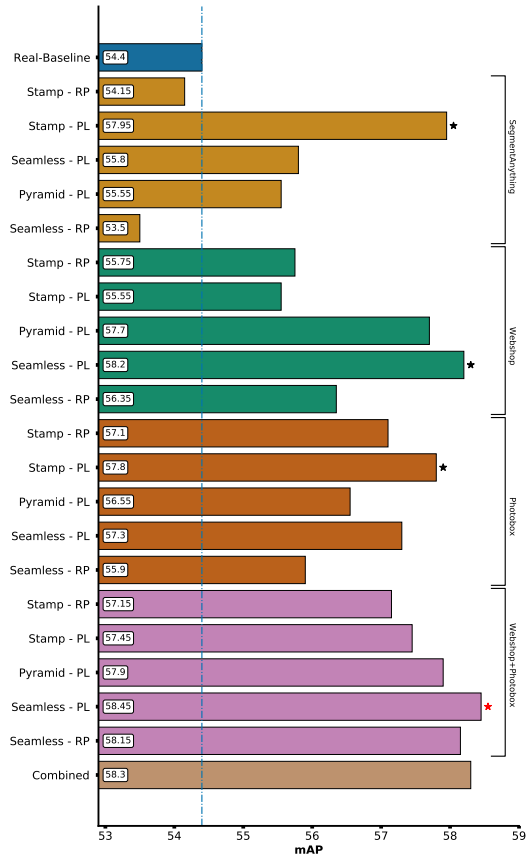

Figure 4: The bars illustrate the median mAP@50 of the experiment runs. The first bar represents the baseline performance achieved using only real data. The best overall result is achieved with a combination of Webscraping and Photobox as object sources together with seamless blending and a placement logic (marked with a red star). PL stands for placement logic. RP stands for random placement.

Webscraping: The use of webscraped object from webshop in synthetic datasets increased the mAP by an average of 2.3%. Including a placement logic enhanced the mAP. Blending techniques showed an overall improvement in results. The best result is achieved by using a placement logic and seamless blending, resulting in a 3.8% improvement.

360° Photobox: The use of the photobox objects in synthetic datasets increased the mAP by an average of 2.5%. On average, adding a placement logic improved the mAP. The impact of blending techniques on the result is uncertain and might even have a negative impact. The best result is achieved by using a placement logic and simple stamping, resulting in a 3.4% improvement.

Webscraping + 360° Photobox: Using the combined image pools of webscraped and photobox objects improved the mAP on average by 3.38%. The use of

	Self augmentation	Webscraping	360° Photobox	Webscraping + 360° Photobox
Stamp-RP	54.15	55.75	57.1	57.15
Stamp-PL	**57.95**	55.55	**57.8**	57.45
Seamless-PL	55.8	57.7	56.55	57.9
Pyramid-PL	55.55	**58.2**	57.3	**58.45**
Seamless-RP	53.5	56.35	55.9	58.15

Table 1: Experimental results as median mAP@50 scores. Baseline with solely real world training data is mAP@50 = 54.4. RP stands for random placement and PL stands for placement logic

blending techniques and placement logic improved the results. The best result is achieved by using a placement logic and seamless blending, resulting in a 4.05% improvement. This is also the best overall score.

Combined: Using the combined image pools of webscraping, photobox and self augmentation data improved the mAP by 3.9%. This dataset represents the combination of the best individual results among all object sources, where objects from self augmentation and the Photobox were implanted using stamping, and webscraped images from webshops were blended with seamless blending. The slightly lower performance compared to the combined webscraped and 360° Photobox category may be attributed to the potential negative influence of self augmentation on the overall results or just some degree of uncertainty.

Perspective Tranformation: We tried using perspective transformations on objects added to scenes, hoping to boost realism by aligning their orientation with the scene. However, despite extensive manual adjustments, this didn't lead to significant performance gains in our model, indicating that the practical benefits of these perspective adjustments might be minimal at the moment.

5.2 Number of Backgrounds

The quantity of backgrounds employed in synthetic data generation is a critical factor that contributes to enhancing the overall quality and effectiveness of the synthetic dataset. To investigate this, multiple datasets were generated with varying sizes of the available background image pool, using webscraped objects as the image pool due to being the best single-source approach. As illustrated in Figure 5, an increased number of backgrounds correlates positively with improved results. This aligns with the established concept of domain randomization found in prior research [13].

5.3 Number and Ratio of Synthetic Images

In our investigation, we explored the impact of the number and ratio of synthetic images within the training dataset. We systematically generated and incorporated synthetic images, varying both the quantity and the proportion in relation to the non-synthetic data. This allowed us to determine the optimal balance between real

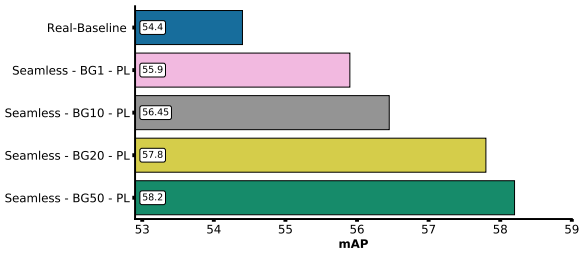

Figure 5: The bars illustrate the median mAP@50 of the experiment runs comparing different amounts of backgrounds used for the data generation. Webscraped images are used as object source. PL stands for placement logic.

and synthetic data to achieve the best results and also shows how synthetic data could improve the results when real data is extremely in short supply. Our experiments involved training the model using solely 2000 synthetic images and gradually increasing the amount of real-world data from zero to 100, along with the addition of varying numbers of synthetic images (zero, 500, 1000, 1900). Subsequently, we increased the real-world data to 500, repeating the process of adding synthetic images (zero, 500, 1000, 1500). Finally, we utilized the original 1190-sized real-world dataset, incorporating different quantities of synthetic images (zero, 500, 1000, 2000, 4000). Throughout these experiments, we employed webscraped objects as the image pool, along with seamless blending and a placement logic, as this combination yielded the best results among single-source approaches. Refer to Figure 6 for a visual representation of our findings. Our experiments revealed a discernible pattern, unveiling a "sweet spot" in the quantity of synthetic data. Both insufficient and excessive infusion of synthetic data were identified as detrimental factors impacting the final results. So, while 1000 was the ideal number in our case, this number might not be the same across different projects and should be considered as an important parameter for a hyperparameter optimization. Furthermore, our analysis highlighted the significance of an increased number of real-world data, showcasing a strong positive correlation with model performance. However, we observed that the improvements with increasing amounts of only real data tended to stagnate. We observed that synthetic

data is particularly valuable in scenarios where a shortage of real data is encountered. It's important to note that our experiments demonstrated that relying solely on synthetic data proved to be insufficient.

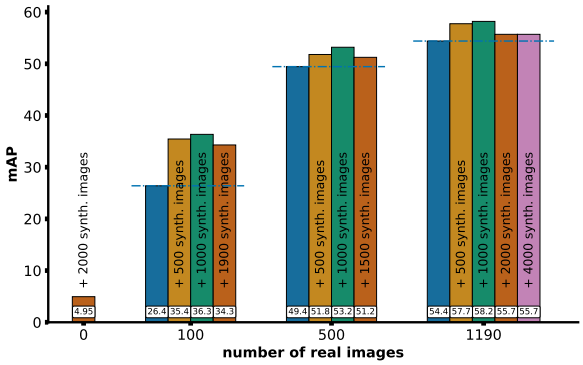

Figure 6: The bars illustrate the median mAP@50 of the experiment runs comparing different ratios between real or synthetic images. The blue bars are models trained on only real world data.

5.4 Bias in the Dataset

The synthetic data generation process notably reduced the bias in the dataset, contributing to a more representative distribution of fridge types. Consequently, the most significant mAP increase was observed in the case of larger fridge types, which were initially underrepresented in the baseline dataset. This resulted in an average mAP increase of 3.87% on larger fridges.

6 CONCLUSION

This study explores different ways to improve existing datasets through the integration of synthetic data. Our approach encompasses the incorporation of internal objects, harnessed via self augmentation, and external objects obtained through two distinct avenues: web scraping and the photobox method. We showed that all investigated object sources are useful for the synthetic data generation. Leveraging self augmentation showcased the smallest improvement that was still significant. This is a good sign because this kind of data can easily be generated. Furthermore, the incorporation of objects from the photobox and web scraping exhibited even more substantial enhancements. While web scraping proves to be a less time-consuming technique, its applicability may be limited in certain scenarios, making the photobox method a valuable alternative, especially for scenarios where manual inclusion of specific objects is essential. The integration of a placement logic proves to be a significant contributor to substantial improvements across various cases. Exploring blending techniques, while not universally applicable, showed potential in enhancing results. It's important to note that the success of blending depends on the image source. For example, self augmentation data benefits from simple stamping, while web scraped images benefit from better blending. The study demonstrated that synthetic data does not need to be ultra-realistic to deceive object detection algorithms. Synthetic data, derived from both internal and external sources, successfully addressed imbalances in the dataset, particularly in scenarios involving larger fridge types. This harmonization of the dataset led to a significant mAP boost on larger fridges, affirming the effectiveness of synthetic data in bridging gaps in real-world dataset disparities. The most notable performance improvements of 9.9 mAP@50 were observed when the training dataset was extremely limited. Moreover, this study corroborated the findings of previous works [10], [11] utilizing an updated object detection network. In conclusion, our approach not only underscores the significance of diverse object sources but also highlights the utilization of placement logic and blending techniques, collectively contributing to a better dataset.

7 FUTURE WORK

A promising direction for further exploration involves the evolution of a dynamic placement logic. Such an adaptive system would intelligently respond to different scenes, reducing the dependency on manual labor. Exploring image enhancement is another compelling direction. The integration of advanced techniques, such as CycleGANs [23], could improve the realism of generated images. CycleGANs, by learning the translation of images between domains, offer a sophisticated means to bridge the gap between synthetic and real data. While our initial experiments with a Masked-CycleGAN have shown promise, a more comprehensive evaluation is required.

Acknowledgments: We are thankful to Conrad Smith for his proofreading assistance and anonymous reviewers for their suggestion

DECLARATIONS

- **Authors' Contributions** Eduard Bartolovic led the majority of the work. Tobias Höfer, Clemens Hage, and Alfred Nischwitz contributed equally. All authors have approved the final manuscript.
- **Funding** The BSH-Group granted computational resources and data access.
- **Conflicts of interest:** Clemens Hage is employee of the BSH group. The other authors declare no conflicts of interest.
- **Availability of Data:** Due to privacy considerations, we are unable to share the data.
- **Code Availability:** The source code is available at https://github.com/EduardBartolovic/synthetic-data-generation.

REFERENCES

[1] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digital Signal Processing*, vol. 126, p. 103 514, 2022, ISSN: 1051-2004. DOI: https://doi.org/10.1016/j.dsp.2022.103514.

[2] A. Prakash, S. Boochoon, M. Brophy, *et al.*, "Structured domain randomization: Bridging the reality gap by context-aware synthetic data," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 7249–7255. DOI: 10.1109/ICRA.2019.8794443.

[3] NVIDIA, *Nvidia omniverse*. [Online]. Available: https://developer.nvidia.com/blog/tag/omniverse/.

[4] S. Iqbal, J. Tremblay, A. Campbell, *et al.*, "Toward sim-to-real directional semantic grasping," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 7247–7253. DOI: 10.1109/ICRA40945.2020.9197310.

[5] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, *Hierarchical text-conditional image generation with clip latents*, 2022. DOI: https://doi.org/10.48550/arXiv.2204.06125. arXiv: 2204.06125 [cs.CV].

[6] J. Betker, G. Goh, L. Jing, *et al.*, "Improving image generation with better captions." [Online]. Available: https://api.semanticscholar.org/CorpusID:264403242.

[7] D. Podell, Z. English, K. Lacey, *et al.*, *Sdxl: Improving latent diffusion models for high-resolution image synthesis*, 2023. DOI: https://doi.org/10.48550/arXiv.2307.01952. arXiv: 2307.01952 [cs.CV].

[8] A. Stöckl, "Evaluating a synthetic image dataset generated with stable diffusion," in *Proceedings of Eighth International Congress on Information and Communication Technology*, X.-S. Yang, R. S. Sherratt, N. Dey, and A. Joshi, Eds., Singapore: Springer Nature Singapore, 2023, pp. 805–818, ISBN: 978-981-99-3243-6.

[9] A. Jadon and S. Kumar, "Leveraging generative ai models for synthetic data generation in healthcare: Balancing research and privacy," in *2023 International Conference on Smart Applications, Communications and Networking (SmartNets)*, 2023, pp. 1–4. DOI: 10.1109/SmartNets58706.2023.10215825.

[10] G. Georgakis, A. Mousavian, A. C. Berg, and J. Kosecka, "Synthesizing training data for object detection in indoor scenes," *CoRR*, vol. abs/1702.07836, 2017. DOI: 10.1109/ICCV.2017.146. arXiv: 1702.07836. [Online]. Available: http://arxiv.org/abs/1702.07836.

[11] D. Dwibedi, I. Misra, and M. Hebert, "Cut, paste and learn: Surprisingly easy synthesis for instance detection," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1310–1319. DOI: 10.1109/ICCV.2017.146.

[12] A. Kirillov, E. Mintun, N. Ravi, *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2023, pp. 4015–4026. DOI: DOI:10.48550/arXiv.2304.02643.

[13] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 23–30. DOI: 10.1109/IROS.2017.8202133.

[14] H. Su, C. R. Qi, Y. Li, and L. J. Guibas, "Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 2686–2694. DOI: 10.1109/ICCV.2015.308.

[15] W. Liu, J. Liu, and B. Luo, "Can synthetic data improve object detection results for remote sensing images?" *CoRR*, vol. abs/2006.05015, 2020. arXiv: 2006.05015. [Online]. Available: https://arxiv.org/abs/2006.05015.

[16] S. Hinterstoisser, V. Lepetit, P. Wohlhart, and K. Konolige, "On pre-trained image features and synthetic images for deep learning," in *Computer Vision - ECCV 2018 Workshops: Munich, Germany, September 8-14, 2018, Proceedings, Part I*, Munich, Germany: Springer-Verlag, 2019, pp. 682–697, ISBN: 978-3-030-11008-6. DOI: 10.1007/978-3-030-11009-3_42. [Online]. Available: https://doi.org/10.1007/978-3-030-11009-3_42.

[17] T. Lin, M. Maire, S. J. Belongie, *et al.*, "Microsoft COCO: common objects in context," *CoRR*, vol. abs/1405.0312, 2014. DOI: https://doi.org/10.1007/978-3-319-10602-1_48. arXiv: 1405.0312.

[18] L. Richardson, *Beautiful soup documentation*, 2007.

[19] P. Perez, M. Gangnet, and A. Blake, "Poisson image editing," SIGGRAPH 03, pp. 313–318, 2003. DOI: 10.1145/1201775.882269. [Online]. Available: https://doi.org/10.1145/1201775.882269.

[20] A. Nischwitz, M. Fischer, P. Haberaecker, and G. Socher, *Bildverarbeitung: Band II des Standardwerks Computergrafik und Bildverarbeitung*. Boston: Springer Vieweg, 2020, ISBN: 978-3-658-28704-7.

[21] P. Burt and E. Adelson, "The laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 532–540, 1983. DOI: 10.1109/TCOM.1983.1095851.

[22] G. Jocher, *Ultralytics/yolov5: V7.0 - yolov5 sota realtime instance segmentations*, Zenodo, Nov. 2022. DOI: 10.5281/zenodo.7347926..

[23] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.