Conditioned Generative AI for Synthetic Training of 6D Object Pose Detection

Mathijs Lens[a], Aaron Van Campenhout[b] and Toon Goedemé[c]
EAVISE-PSI, KU Leuven Campus De Nayer, Sint-Katelijne-Waver, Belgium
{*mathijs.lens, aaron.vancampenhout, toon.goedeme*}*@kuleuven.be*

Keywords: Pose Estimation, 6D Pose Estimation, Pacel Detection, ControlNet, Stable Diffusion.

Abstract: In this paper, we propose a method to generate synthetic training images for a more complex computer vision task compared to image classification, specifically 6D object pose detection. We demonstrate that conditioned diffusion models can generate unlimited training images for training an object pose detection model for a custom object type. Moreover, we investigate the potential of (automatically) filtering out ill-produced images in the dataset, which increases the quality of the image dataset, and show the importance of finetuning the trained model with a limited amount of real-world images to bridge the remaining sim2real domain gap. We demonstrate our pipeline in the use case of parcel box detection for the automation of delivery vans. All code is publicly available on our GitLab https://gitlab.com/EAVISE/avc/generative-ai-synthetic-training-pose-detection.

1 INTRODUCTION

Training deep learning models for computer vision tasks typically requires extensive manually annotated datasets. As the scaling laws predict, the more challenging a task, the more complex the neural network architecture must be, and consequently, the greater the amount of data required for training. Collecting large quantities of relevant images is already time-consuming and costly process. However, the fact that all of these images need to be manually labeled, renders this in many cases infeasible. Especially when the labels are complex and demand significant manual effort, as in the task we focus on.

Indeed, this paper's vision task is 6D object pose determination. Compared to the simple image labels required for classification, the labels for 6D object pose detection are significantly more expensive to produce. The effort to annotate an object's 6D $(x, y, z, \alpha, \beta, \gamma)$ pose is clearly more time-demanding than the actual image acquisition.

A solution we explore in this paper to overcome this bottleneck, is synthetic data generation. This paper introduces the first pipeline, to the best of our knowledge, that uses Generative AI for the creation of realistically looking synthetic datasets for 6D object pose detection. We propose to use wireframe conditioning as control input to a combination of Stable Diffusion and ControlNet, enabling to exactly steer the image creation process with precise 6D object pose labels. Stable Diffusion is utilized to create realistic synthetic images, and ControlNet conditions these images based on specific 6D pose information.

By randomly modifying the input prompts, we generate diverse object instances under various lighting, texture, and environmental conditions, enriching the variety and robustness of the synthetic dataset.

Additionally, we incorporate a filter network that automatically evaluates the quality of the generated images, identifying and discarding poorly rendered or inaccurate examples. This filtering process ensures that only high-quality synthetic data, accurately reflecting the target 3D poses, is used for model training. By leveraging this synthetic data pipeline, we drastically reduce the dependency on expensive real-world datasets, enabling more efficient and scalable 3D object pose detection. We demonstrate that our pipeline is effective for parcel box detection, but by modifying the prompt inputs, it can be easily adapted to generate annotated data for a wide range of other objects, making it a versatile solution for various applications.

In this paper, we showcase our pipeline on a parcel box detection use case. Since last-mile delivery is the most time-consuming part of the delivery process, investments in making this process more efficient pay

[a] https://orcid.org/0009-0005-4798-3555
[b] https://orcid.org/0009-0006-4193-3650
[c] https://orcid.org/0000-0002-7477-8961

Lens, M., Van Campenhout, A. and Goedemé, T.
Conditioned Generative AI for Synthetic Training of 6D Object Pose Detection.
DOI: 10.5220/0013130600003912

In *Proceedings of the 20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2025) - Volume 2: VISAPP*, pages 324-331
ISBN: 978-989-758-728-3; ISSN: 2184-4321

Figure 1: Illustration of our parcel recognition use case. This paper focuses on the first phase, 6D pose detection of parcel boxes. Using the 6D pose, we can extract and rectify the box faces, a crucial step for identifying specific parcels.

off quickly. Currently, the delivery person manually loads the vehicle using his "mental map" for sorting, enabling him to retrieve the parcels later on. If we can automate this, the loading (incl. sorting) of the parcels can be done a lot quicker. This would require automatic recognition and localisation of the parcels in the van. We develop a camera-based system which can be installed in the delivery van. The delivery van operator can use this camera system to quickly identify any parcel box by just holding it in the field-of-view. Once identified, the in-van computer system can then indicate where to optimally store that parcel in the van during the loading phase, or what the exact destination address is of the parcel in the delivery phase. Our envisioned system must be capable of detecting parcels and performing pose estimation, such that the box faces can be cut out from the image, rectified and compared with a database recorded in the parcel facility, as demonstrated in figure. High accuracy in pose estimation is essential to ensure the reliable extraction and rectification of these box faces. Figure 1 illustrates this. While the box face recognition part is not covered in this paper, the first step of this application, i.e. 6D parcel box pose estimation, is an ideal use case for the presented pipeline. A key challenge, however, lies in creating a large and diverse dataset of 6D pose-annotated parcel images, as manual annotation is both time-consuming and costly.

This paper presents several key contributions. First, up to our knowledge, we propose the first automatically annotated synthetic dataset production pipeline for 6D object pose detection datasets. We demonstrate it is a viable solution to the challenge of creating large, annotated real-world datasets and that a high-quality synthetic dataset can achieve decent performance while requiring less data. Furthermore, we show that combining synthetic data with real-world data results in even better performance than using either data type alone. Finally, the pipeline we introduce enables the creation of diverse synthetic datasets for a wide range of objects, wherever it is feasible to create a wireframe of the object.

2 RELATED WORK

In this section, first we review the state-of-the-art on 6D object pose estimation techniques, focusing on advancements in deep learning-based methods. Next, we discuss recent developments in synthetic data generation, highlighting its growing role in addressing the challenges of large-scale annotated dataset creation.

2.1 6D Pose Estimation

Understanding the position and orientation of objects is crucial in various fields such as robotics, autonomous driving, and augmented reality, as it enables machines to comprehend and interact with their environment. A key task in this domain is 6D pose estimation, which determines an object's translation (x, y, z) and rotation (α, β, γ) within a scene.

Many state-of-the-art 6D pose estimation methods utilize depth sensors or CAD models for training, as these provide precise 6D annotations like (Zhao et al., 2020; Josifovski et al., 2018; Su et al., 2015). However, since our approach does not rely on such resources, we focus on monocular pose estimation techniques. These methods predict object poses from single RGB images, offering a more flexible and accessible solution that aligns with our synthetic data generation pipeline. These techniques often aim to estimate the projection of a 3D object and rely on the PNP algorithm to solve the translation and rotation. By identifying keypoints or correspondences between the 3D model and its 2D projection, the PnP algorithm computes the object's pose in the scene (Marullo et al., 2023). Monocular methods often employ deep learning to predict these keypoints directly from RGB images, making them a versatile approach for scenarios where depth data or CAD models are unavailable.

A commonly used monocular dataset for pose estimation is Google's Objectron dataset (Ahmadyan et al., 2021), which consists of 9 object classes annotated with 3D bounding boxes. This dataset has become a key benchmark for 6D object pose estimation,

enabling models to learn object-centric features in a variety of real-world scenes.

Several notable models utilize 3D bounding box annotations for 6D pose estimation. BB8 (Rad and Lepetit, 2017) employs a multistage approach using a cascade of convolutional neural networks (CNNs), where each network progressively refines the keypoint predictions from the previous stage. This multistage refinement improves accuracy, especially in challenging conditions such as occlusions or cluttered backgrounds. YOLO6D (Tekin et al., 2018), in contrast, adopts a single-shot approach that simultaneously detects objects in RGB images and predicts their 6D pose in real time. By eliminating the need for multiple stages, YOLO6D achieves greater computational efficiency while maintaining reliable performance in complex environments.

CatTrack (Yu et al., 2024) leverages vision transformers for detecting and tracking object keypoints, providing a robust solution for multi-object tracking in dynamic and cluttered scenes. The combination of tracking and keypoint detection with vision transformers enables CatTrack to accurately estimate 6D object poses in video streams.

2.2 Synthetic Data

Over recent years, synthetic data has become instrumental in computer vision research due to its ability to overcome the challenges of manual generation and ethical concerns associated with real-world data acquisition (Man and Chahl, 2022).

Among the various methods for generating synthetic data, gaming engines like Unity or Unreal have emerged as a powerful tool for creating virtual environments that facilitate the visualization of complex scenes. Example application domains are: human faces (Wood et al., 2021; Delussu et al., 2024), pedestrian detection (Hattori et al., 2018), manufacturing industry (Moonen et al., 2023) and automotive (Deschaud, 2021; Zhang et al., 2019; Zhao et al., 2024). Despite the improving rendering quality of these engines, the resulting images can still look artificial. Hence, a model trained solely on synthetic images will typically perform not as good on real images, i.e. the so-called *sim2real domain gap*. Moreover, one needs a complete and realistic 3D model of all objects in the scene to render.

Alternatively, generative AI techniques can infer missing information in images, bypassing the need for fully explicit input to create synthetic data. For instance, GAN (Generative Adversarial Networks) models have been employed to generate novel, realistic images(Nikolenko, 2021). In (Abbas et al., 2021), the authors extend a limited real dataset of diseased tomato leaves with synthetic images generated using DCGAN, demonstrating improved generalization during training. Another application of GANs is presented in (Rajagopal et al., 2023), in which Unreal Engine-generated images are enhanced using CycleGAN to improve lighting and textures, thus increasing realism. The authors observed that while the performance was initially hindered by the artificial appearance of the data, CycleGAN effectively closed this realism gap.

More recently, high-quality image generation has become achievable with models like Stable Diffusion. (Lomurno et al., 2024) uses Stable Diffusion 2.0 to generate synthetic datasets for image classification tasks. Notably, in one-third of the experiments, models trained on synthetic data outperformed those trained on real data, illustrating the potential of this approach.

The previously mentioned methods primarily focus on generating images without explicit control over the conditions. In (Tran, 2024), a novel method is introduced that generates high-quality segmentation masks alongside synthetic data. By extracting candidate point prompts from attention maps, these prompts guide a vision model to yield fine-grained segmentation masks. While this method provides segmentation masks, it lacks direct control over the image generation process, a feature that our method addresses.

Lastly, in (Valvano et al., 2024), a diffusion model is trained to generate training images of custom objects based on textual prompts and associated conditions. However, this approach cannot explicitly express 6D poses using text prompts, highlighting a limitation that ControlNet seeks to overcome.

3 METHOD

In this section, we will first describe the pipeline used for synthetic dataset generation and provide details about the CenterPose model and the PnP algorithm for object pose estimation.

3.1 Synthetic Data Pipeline

Figure 6 provides an overview of the pipeline we developed for generating synthetic datasets. The pipeline we present starts from randomly chosen 6D pose parameters $(x, y, z, \alpha, \beta, \gamma)$ and randomly chosen box dimensions (w, h, d), which are used as input for a program called the Wireframe Generator, which leverages OpenGL to render a wireframe image of the

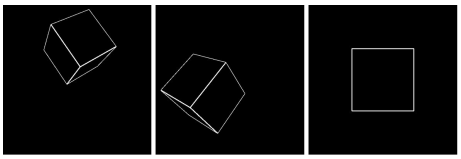

Figure 2: The first component of our pipeline for generating synthetic datasets conditioned on wireframes for 6D pose estimation is the wireframe generator. This component randomly renders 3D wireframes in OpenGL, including 6D pose annotations.

Figure 3: Comparing images generated with different Stable Diffusion models.

box at the specified location. These images are generated with a fixed resolution of 512x512 pixels. The annotated wireframes in the image are automatically saved in a JSON file. A few examples of these wireframes are shown in Figure 2.

The crux of the matter is that we can use these wireframe images as a means to control the synthetic image generation process. ControlNet (Zhang et al., 2023) indeed enables to guide the Stable Diffusion (Rombach et al., 2022) image generation with a graphical input. It enforces spatial consistency between the input image and generated synthetic image. as we want the box object to be synthesized exactly at the 6D pose corresponding to the parameters we generated the wireframe image with, we chose to use the *scribble* condition modality ControlNet offers. We observed that this indeed controls the pose of the box in the generated image well, while the rest of the image is filled in with realistic fantasized content. This behaviour is not possible when a depth map of the box was used as input instead. Because of the lack in depth input data in the background, only flat scenes were synthesized with that modality.

We chose a Stable Diffusion model fine-tuned on human figures, which results in higher quality images when generating depictions of a person holding a parcel. As shown in Figure 3, the *Realistic Vision v20* model generates high-quality images featuring humans.

During the image generation process, we employ a prompt template to generate a variety of different text prompts, as illustrated in Figure 5. In this template, bold words in the sentences are randomly substituted with terms from predefined lists. This approach al-

Figure 4: Poorly generated images.

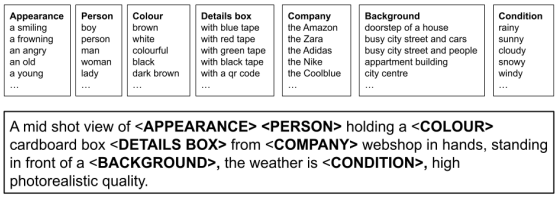

Figure 5: Prompt-template used for generating diverse prompt texts.

lows to create a wide variety of detailed prompts, leading to the generation of diverse images.

By cascading these components, we create a pipeline capable of generating a diverse, realistic-looking synthetic dataset. This process can be repeated indefinitely until a sufficiently large dataset is accumulated for training, without the need for human intervention.

3.2 Synthetic Image Filtering

However, during our experiments, we observed that some images appeared unrealistic or poorly generated, as illustrated in Figure 4. To minimize the gap between real and synthetic data, we propose a filtering neural network that detects and removes poorly generated images. For this classifier network, we use a ResNet-18 model, trained on a seperate set of synthetic images which we manually annotated as “good“ or “bad“. This classifier achieves 73.87% accuracy on the test set.

3.3 6D Pose Detection Model

Once the pipeline presented above is executed, we end up with a large filtered synthetic dataset suitable for training CenterPose. CenterPose (Lin et al., 2022) is a single-stage keypoint-based approach for category-level object pose estimation. It uses three branches to determine the 6-DoF (degrees of freedom) pose of an object. First, a keypoint detection branch identifies the center of the object. Once the center is detected, the model estimates the object’s vertices. Finally, the

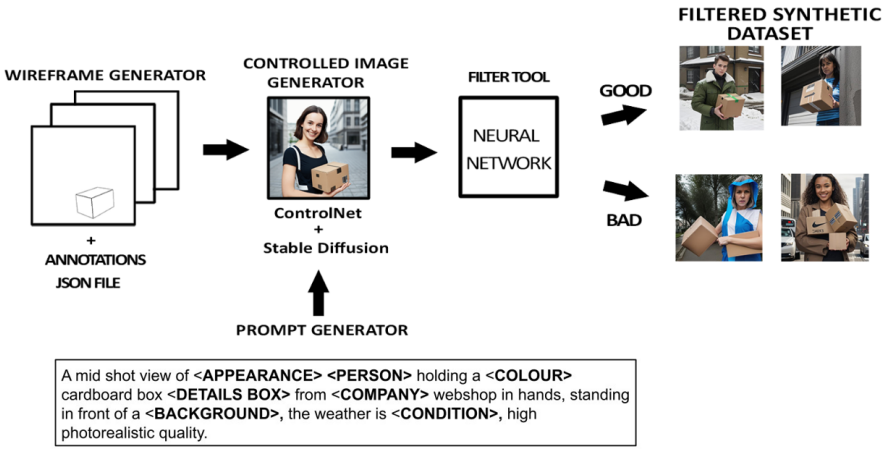

Figure 6: An overview of all the components in the synthetic dataset generation pipeline.

cuboid dimension branch is used to estimate the relative dimensions of the object in the scene.

After these branches have been executed, the PnP (Perspective-n-Point) algorithm is applied. This algorithm uses the detected keypoints, estimated dimensions and the camera's intrinsic parameters to compute the rotation and translation of the object.

4 RESULTS

4.1 Datasets

For our experiments, we use various different datasets, which allows us to gain better insight into the usefulness of synthetic data for training 6D object pose detection and the effectiveness of our pipeline.

Table 1 gives an overview of the different datasets we collected for the parcel box detection use case.

First of all, for our synthetic pipeline we annotated 1,080 generated synthetic images, which we used to train and test the image filtering network described in Section 3.2. For this, we manually inspected all images and annotated them as "good" or "bad". From this annotated dataset, 969 images were set aside for training, while 111 images were used for testing.

To conduct the actual experiments, we put our pipeline in full throttle and let it freely generate a big dataset of 5600 synthetic images. We used our filtering network to filter out poorly generated images, as shown in Figure 4, reducing the initial synthetic dataset to 3,300 training images and 100 test images. Both of these datasets were used to train the object pose detection network, as described below.

Finally, to better understand the sim2real domain gap and evaluate the real-life performance, we also collected a real dataset consisting of 299 images. These 299 images are then split into 199 for training and 100 for testing. The dataset includes pictures photographed from 29 different cardboard parcel boxes, with approximately 10 pictures taken from different angles for each parcel. Half of these pictures are mid-shots of a person holding a parcel, while the other half are parcels lying in various locations, as illustrated in Figure 7. Dimensions of each parcel box were measured with a ruler. Annotation of the 6D pose of each box in each image is done manually. For this, we developed an annotation tool [1], that uses the Perspective-n-Point (PnP) algorithm on manually clicked corner points of the box, yielding a quick initial estimate of the pose. In a second step, this pose can be manually refined by finetuning each of the six dimensions.

4.2 Evaluation Metrics

In this section, we describe the metrics used for evaluating the performance of the model, specifically the 3D Intersection over Union (IoU) and two types of 2D IoU, which are illustrated in fig. 8.

Table 1: Overview of the datasets used in our use case.

		Number of images	
Dataset name	**Creation details**	**Training**	**Test**
Filter training synthetic	manually labeled good/bad	969	111
unfiltered Synthetic training	automatically generated	5600	
Filtered synthetic training	automatically generated and filtered	3300	100
Real	manually annotated 6D pose	199	100

The 3D IoU metric measures how well the predicted 3D bounding box overlaps with the ground truth 3D bounding box, providing insight into the accuracy of the 3D pose estimation. Because a 2D image always has scale ambiguity, CenterPose can't

[1] https://gitlab.com/EAVISE/avc/generative-ai-synthetic-training-pose-detection

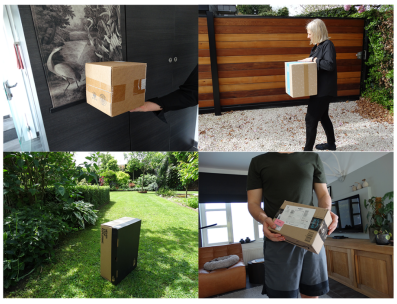

Figure 7: Example of images from the manually annotated test dataset.

Figure 8: Illustration of the evaluation metrics: 3D IoU (left), 2D IoU of bounding boxes (middle) and 2D mask IoU (right).

determine the depth from a single RGB image. We hence do not know if the detected box is a small box close to the camera, or a very large box far away. To alleviate this, we normalize the depth dimension by translating the center point of the detected 3D bounding box to the center point of the ground truth over its threedimensional line-of-sight axis.

The 3D IoU is used in two ways. We both compute the mean IoU over the entire test set, as is done in semantic segmentation evalutions. But, next to that, we threshold the IoU of the detections to count it as a true positive or not. The resulting average precision for two different thresholds (0.5 and 0.75) is also reported in our experiments, giving insight to the accuracy of the estimated poses.

Since CenterPose uses the Perspective-n-Point (PnP) algorithm to estimate the 6-DoF pose, it requires knowledge of the intrinsic camera matrix used to capture the images. However, for synthetic data generated using Stable Diffusion, the intrinsic camera parameters are not available, making 3D IOU evaluation infeasible for synthetic data. To address this, we evaluate the accuracy on synthetic data using the 2D IoU metric. We calculated the 2D IoU in two different ways:

- Computing the intersection and overlap of the detected and ground truth 2D bounding boxes of the object in the image.
- Computing the intersection and overlap of the segmentation masks of both detected and ground truth boxes in the image, produced by the convex hull of the object points in the image.

These two methods offer a clearer understanding of how accurately synthetic objects are detected in 2D, despite the absence of 3D camera parameters. Additionally, we report the Miss Rate metric, which indicates the percentage number of objects the model failed to detect. When an object is missed, the PnP algorithm is not executed, resulting in both the 2D and 3D IoU metrics being zero.

4.3 Results Overview

Table 2 presents the results obtained after training on the different datasets. It is important to note for the evaluation results, that all trainings started from the CenterPose model pre-trained on the Google's Objectron dataset (Ahmadyan et al., 2021), more specifically the cereal box category. The datasets consist of images containing a parcel, annotated by determining the eight vertices of each parcel and the normalized dimensions of these parcels. The table compares the performance of the model using the 2D and 3D IoU metrics defined above on the respective datasets.

We evaluated the performance of the models on both synthetic and real test sets. For the training dataset, we compared the unfiltered and filtered versions. The bottom two models were initially trained on a synthetic dataset, followed by further fine-tuning on the real-life training dataset. For the fine-tuning process on real data, we used the same learning rate and selected the number of epochs that yielded the best results.

4.4 Discussion

Upon reviewing the results, it is evident that the model trained solely on cereal box data from Google's Objectron dataset performed the worst, as expected. The limited training on cereal boxes constrained the model's ability to generalize to parcel objects. Training the model exclusively on 199 real training images resulted in improved performance on both synthetic and real metrics, with a significant reduction on the miss rate.

Training solely on the unfiltered synthetic dataset of 5,600 images yielded the best performance when tested on the synthetic test dataset. It also had a positive impact on the real test dataset, demonstrating that unfiltered synthetic data can help the model generalize better to real-world scenarios.

In contrast, training on the filtered synthetic dataset of 3,300 images resulted in slightly worse performance on the synthetic test dataset. However, the filtered synthetic dataset achieved a marginally higher average IoU compared to the unfiltered dataset when

Table 2: Object pose detection results of the model after training on different datasets. Grey cells report the results on the synthetic test set, white cells on the real-life test set.

Training dataset	Test dataset	2D IoU		3D IoU			Miss Rate
		Bounding box	Mask	AP@0.5IoU	AP@0.75IoU	*meanIoU*	
Cereal boxes	Synthetic	0.3832	0.3293	x	x	x	x
	Real	0.2016	0.1922	0.0800	0.0100	0.1058	0.7100
Real	Synthetic	0.6796	0.6122	x	x	x	x
	Real	0.7402	0.6816	0.3000	0.0200	0.3940	0.1000
Unfiltered synthetic	Synthetic	**0.9321**	**0.8919**	x	x	x	x
	Real	0.5083	0.4680	0.2000	0.0200	0.2325	0.3400
Filtered synthetic	Synthetic	0.9293	0.8771	x	x	x	x
	Real	0.5094	0.4647	0.2000	0.0200	0.2384	0.3500
Unfiltered synthetic + real	Synthetic	0.9100	0.8706	x	x	x	x
	Real	**0.7720**	**0.7221**	**0.4400**	**0.0600**	**0.4332**	0.1000
Filtered synthetic + real	Synthetic	0.8974	0.8327	x	x	x	x
	Real	0.7568	0.7016	0.3800	0.0400	0.4247	0.1000

evaluated on the real test dataset. This improvement can be attributed to the higher quality information in the filtered dataset. The higher miss rate observed in the filtered dataset may be due to its smaller size compared to the unfiltered one.

Finally, when fine-tuning both the unfiltered and filtered synthetic datasets on real data, the best overall performance was surprisingly achieved with the unfiltered synthetic + real training dataset. This suggests that the larger volume of data in the unfiltered set (almost double the amount of images) contributed more than the data quality itself. But, despite having fewer images, the filtered dataset performed nearly as well, indicating that the higher quality of data in the filtered set compensated somehow for its smaller size. However, when fine-tuning further, the impact of filtering out poor-quality images was minimal. The randomness introduced by the unfiltered dataset may have been sufficient for effective pre-training.

4.5 Non-Cuboid Objects

In our test use case on parcel detection, the object at hand is a simple cuboid object. However, our pipeline can be used for any non-cuboid object of which a 3D wireframe model is available. Figure 9 demonstrates synthetically generated images using our pipeline, after rendering these none-cuboid wireframe models in random 6D poses.

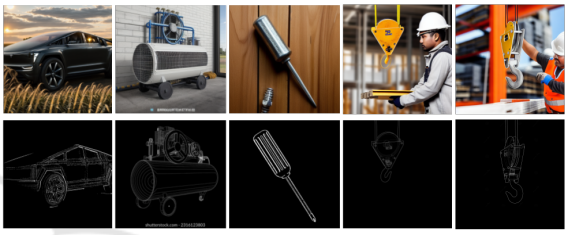

Figure 9: Non-cuboid objects generated with the same pipeline starting from a 3D object wireframe.

5 CONCLUSION

In this paper, we proposed a pipeline for the controlled generation of 6D annotated image datasets for object pose detection training. with this, we explored the potential of automatic generation of a large annotated synthetic dataset, as well as the effectiveness of using synthetic data to train a 6D object pose detection network.

The proposed pipeline for generating annotated synthetic datasets provides a cost-efficient solution to reduce the reliance on extensive manual annotation. The results on a real-life application example, i.c. parcel box pose detection, demonstrate that models trained on synthetic datasets produced by this pipeline exhibit strong generalization capabilities when applied to real-world test data. However, a key challenge remains in generating high-quality, realistic images with rich information. Ensuring that prompts are diverse and well-aligned with the conditioned generation process is crucial to enhancing the overall quality and variability of the synthetic data.

We explored the potential of automatic filtering out of poorly produced synthetic images, increasing the data quality in the generated dataset. Our findings indicate that when training solely on synthetic data, the filtered dataset achievable slightly better performance due to the higher quality of the data. However, the unfiltered dataset exhibited a lower miss rate, likely because it exposed the model to a larger volume of synthetic data during training. This highlights the importance of balancing data quality and quantity for optimal model training.

Additionally, training on synthetic data alone demonstrated a positive transfer effect when tested on real-world data. While the results were not as strong as those achieved through training solely on real data,

combining synthetic and real datasets produced the best overall results. Interestingly, when fine-tuning the model with real data after initial training on synthetic data, the impact of filtering became less significant. The randomness introduced by the unfiltered dataset improved generalization during fine-tuning. This hybrid approach suggests that synthetic data can be a valuable supplement in situations where real-world data is limited or difficult to annotate. We also demonstrated the general applicability on objects beyond simple cuboid shapes.

REFERENCES

Abbas, A., Jain, S., Gour, M., and Vankudothu, S. (2021). Tomato plant disease detection using transfer learning with c-gan synthetic images. *Computers and Electronics in Agriculture*, 187:106279.

Ahmadyan, A., Zhang, L., Ablavatski, A., Wei, J., and Grundmann, M. (2021). Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Delussu, R., Putzu, L., and Fumera, G. (2024). Synthetic data for video surveillance applications of computer vision: A review. *IJCV*, pages 1–37.

Deschaud, J.-E. (2021). Kitti-carla: a kitti-like dataset generated by carla simulator. *arXiv preprint arXiv:2109.00892*.

Hattori, H., Lee, N., Boddeti, V. N., Beainy, F., Kitani, K. M., and Kanade, T. (2018). Synthesizing a scene-specific pedestrian detector and pose estimator for static video surveillance - can we learn pedestrian detectors and pose estimators without real data? *Int. Journal of Computer Vision*, 126(9):1027–1044.

Josifovski, J., Kerzel, M., Pregizer, C., Posniak, L., and Wermter, S. (2018). Object detection and pose estimation based on convolutional neural networks trained with synthetic data. In *2018 IEEE/RSJ IROS*, pages 6269–6276.

Lin, Y., Tremblay, J., Tyree, S., Vela, P. A., and Birchfield, S. (2022). Single-stage keypoint-based category-level object pose estimation from an RGB image. In *IEEE ICRA*.

Lomurno, E., D'Oria, M., and Matteucci, M. (2024). Stable diffusion dataset generation for downstream classification tasks. *arXiv preprint arXiv:2405.02698*.

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

Marullo, G., Tanzi, L., Piazzolla, P., and Vezzetti, E. (2023). 6d object position estimation from 2d images: A literature review. *Multimedia Tools and Applications*, 82(16):24605–24643.

Moonen, S., Vanherle, B., de Hoog, J., Bourgana, T., Bey-Temsamani, A., and Michiels, N. (2023). Cad2render: A modular toolkit for gpu-accelerated photorealistic synthetic data generation for the manufacturing industry. In *Proceedings of WACV*, pages 583–592.

Nikolenko, S. (2021). *Synthetic data for deep learning*, volume 174. Springer.

Rad, M. and Lepetit, V. (2017). Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3848–3856.

Rajagopal, B. G., Kumar, M., Alshehri, A. H., Alanazi, F., Deifalla, A. F., Yosri, A. M., and Azam, A. (2023). A hybrid cycle gan-based lightweight road perception pipeline for road dataset generation for urban mobility. *Plos one*, 18(11):e0293978.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.

Su, H., Qi, C. R., Li, Y., and Guibas, L. J. (2015). Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views. In *2015 IEEE ICCV*, pages 2686–2694.

Tekin, B., Sinha, S. N., and Fua, P. (2018). Real-time seamless single shot 6d object pose prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Tran, T. (2024). Synthesizing image with high-quality segmentation mask by prompting large vision model. In *CVPR Workshop*.

Valvano, G., Agostino, A., De Magistris, G., Graziano, A., and Veneri, G. (2024). Controllable image synthesis of industrial data using stable diffusion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5354–5363.

Wood, E., Baltrušaitis, T., Hewitt, C., Dziadzio, S., Cashman, T. J., and Shotton, J. (2021). Fake it till you make it: face analysis in the wild using synthetic data alone. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3681–3691.

Yu, S., Zhai, D.-H., Xia, Y., Li, D., and Zhao, S. (2024). Cattrack: Single-stage category-level 6d object pose tracking via convolution and vision transformer. *IEEE Transactions on Multimedia*, 26:1665–1680.

Zhang, H., Tian, Y., Wang, K., He, H., and Wang, F.-Y. (2019). Synthetic-to-real domain adaptation for object instance segmentation. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7.

Zhang, L., Rao, A., and Agrawala, M. (2023). Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847.

Zhao, H., Wang, Y., Bashford-Rogers, T., Donzella, V., and Debattista, K. (2024). Exploring generative ai for sim2real in driving data synthesis. *arXiv preprint arXiv:2404.09111*.

Zhao, W., Zhang, S., Guan, Z., Luo, H., Tang, L., Peng, J., and Fan, J. (2020). 6d object pose estimation via viewpoint relation reasoning. *Neurocomputing*, 389:9–17.