MultiFloodSynth: Multi-Annotated Flood Synthetic Dataset Generation

YoonJe Kang[1*], Yonghoon Jung[1*], Wonseop Shin[1*], Bumsoo Kim[1*], Sanghyun Seo[1†]

[1]Chung-Ang University, Republic of Korea
{bluejay100, dydgns2017, wonseop218, bumsookim, sanghyun[†]}@cau.ac.kr

Abstract

In this paper, we present synthetic data generation framework for flood hazard detection system. For high fidelity and quality, we characterize several real-world properties into virtual world and simulate the flood situation by controlling them. For the sake of efficiency, recent generative models in image-to-3D and urban city synthesis are leveraged to easily composite flood environments so that we avoid data bias due to the hand-crafted manner. Based on our framework, we build the flood synthetic dataset with 5 levels, dubbed *MultiFloodSynth* which contains rich annotation types like normal map, segmentation, 3D bounding box for a variety of downstream task. In experiments, our dataset demonstrate the enhanced performance of flood hazard detection with on-par realism compared with real dataset.

Introduction

Deep learning algorithm requires high quality, diverse and large training dataset. However, gathering good dataset is labor-intensive and requires substantial cost, especially on hyper-scale situations like wildfire recognition (Hong et al. 2024), pine wilt disease detection (Jung et al. 2024) and so on. To mitigate this issue, a common solution today is to generate synthetic data and use them as training or reference dataset (Kim et al. 2024). Recently, generative models (Hamza et al. 2024; Islam et al. 2024; Khullar et al. 2023) or real-time engine (Delussu, Putzu, and Fumera 2024) are leveraged for more plausible and efficient generation.

However, there is a still concern that editability is insufficient (that is few control parameter) to composite the final scene. Existing dataset has ambiguous label and one or two types of annotation. Furthermore, from the fact that synthetic dataset should reflect the real-world data, some domains have a significant difficulty due to the absence of high quality real data for reference, definition of label and inconsistent annotation problem. One of them is flood hazard situation which makes it difficult to collect the dataset. It stems from their specific situation where flood accidents frequently paralyze the digital system and physically collapse the surveillance device. Despite of this reason, existing works had explored to make real flood dataset with hand-crafted labeling (Wan et al. 2024; Wu et al. 2024b; Gao et al. 2024). They inevitably confront label-inconsistent problem like 2D bounding box. In addition, they only consider one or two types of annotation as ground truth, hindering their applicability to various computer vision tasks.

In this paper, we present a novel framework that utilizes a 3D engine to generate urban flood synthetic dataset, dubbed *MultiFloodSynth*. For fidelity, we faithfully attribute the flood hazard situation as several properties and components (e.g., layout (Shang et al. 2024), lighting, flood-level (Chaudhary et al. 2020; Wan et al. 2024), 3D object, camera view) for scene composition by exploring urban flood situation. Moreover, considering various computer vision tasks, our system includes a variety of annotation types such as normal map, instance/semantic/fine-grained segmentation map, camera 3D pos, 2D/3D bounding box, etc. Thanks to such editable attributes, our *MultiFloodSynth* improved the performance of object-localized flood level detection, while alleviating large dataset requirements for model training.

Related Works

Synthetic Dataset Generation

Recently, generating non-real dataset, synthetic dataset is common technique in a variety of field which requires hyper-scale (Shang et al. 2024; Greff et al. 2022; Zhang et al. 2024; Xie et al. 2024; Wu et al. 2024a; Schieber et al. 2024; Shang et al. 2024; Wang et al. 2024b; Hao et al. 2024; Zhu et al. 2024; Valvano et al. 2024), impossible scenarios (Jung et al. 2024; Greff et al. 2022; Hummel and van Kooten 2019; Mittal et al. 2023; Kokosza et al. 2024; Amador Herrera et al. 2024), and so on. Synthetic dataset generation resolve such issues by constructing scene in virtual world and alleviate vexing manual process with auto labeling. Diverging from conventional way to generate synthetic data, it has been explored to reflect the characteristics of real-world object to enhance the high fidelity and appropriateness (Richter, Al-Haija, and Koltun 2022; Lee, Shin, and Lee 2024; Ebadi et al. 2022). Since these strategy enhance the robustness and realism, it is crucial to consider these attributes for faithful synthesized data.

[*]Equal contributors
[†]Corresponding author

Presented at the Workshop on Preparing Good Data for Generative AI: Challenges and Approaches (Good-Data) in conjunction with AAAI 2025. The authors retain the copyright.

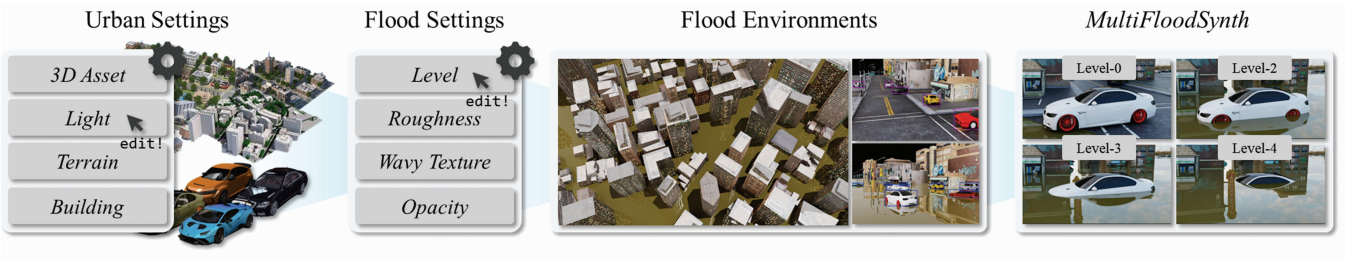


Figure 1: Overview of virtual flood scene composition and synthetic dataset generation pipeline.

Figure 2: Sample of *real* flood data (Wan et al. 2024).

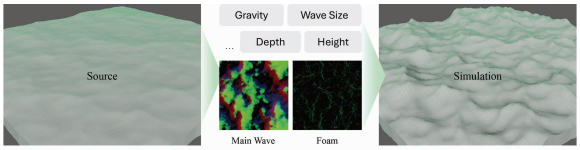


Figure 3: Components of flood simulation and results.

Parameter	Attribute	Type
Urban Settings		
Position	Transform	Constant
Lighting	Light Intensity	Constant
Background	Texture	Image
Layout	-	Image
Flood Settings		
Level (Scale)	Transform	Constant
Roughness	-	Constant
Wavy Texture	-	Image
Opacity	Material	Constant
Specular	Material	Constant
Main Wave	-	Image
Wave Foam	-	Image

Table 1: Control parameters to composite the virtual flood hazard scene. – denotes no corresponding attribute.

Flood Hazard Detection

Detecting flood situation can be considered as object detection using the level of flood of object. A multitude of studies on classifying and detecting objects based on deep learning algorithms has been continuously conducted to address abovementioned requirement (Lo et al. 2021; Pally and Samadi 2022; Karanjit, Pally, and Samadi 2023; Zhong et al. 2024; Wu et al. 2024b). However, most studies face challenges such as a lack of data sharing, ethical concerns, absence of abundance and limited labels. To do that, in this paper, we intent to address these issues in following section.

Proposed Method

Main objective is to synthesize urban-scale flood hazard situation in virtual scene and enhance the performance of flood-level detection system with our *MultiFloodSynth* by alleviating a problem of dataset collection. To commence, we describe the real-world flood hazard situation with some considerations and how our *MultiFloodSynth* promise the fidelity. Overall pipeline is shown in Fig. 1.

Challenges of Real-World Flood Hazard Scenarios

Existing real-world dataset (Gao et al. 2024) was obtained on vehicle-based flood detection. This dataset includes label information divided into five levels based on the percentage of a vehicle submerged in water (Wan et al. 2024). Fig. 2 shows some samples of the data included in the real-world dataset. However, they have inconsistent problem with incoherent bounding box by human-hand. Meanwhile, to simulate the flood situation, it should be considered to appropriately composite several components including camera view, lighting condition and flood-level (*i.e.*, flood height).

Depicting Flood Scenarios in Virtual Simulator

In contrast to existing synthetic generation works, our framework is capable of controlling some parameters to composite final virtual scene as discussed in former subsection. It allows the user to control the scene for user-wanted structure. Based on our exploration with several real-world data (Zhong et al. 2024; Gao et al. 2024; Wu et al. 2024b), we observed that following settings play a crucial role to determine the virtual scene: *urban setting*, *flood settings* which heavily affects to semantic feature in neural network. Detailed parameters are listed in Table 1.

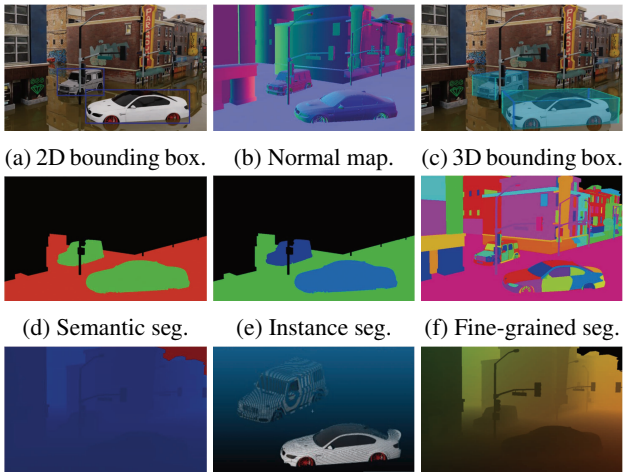
Furthermore, since it is quite cumbersome to search several 3d objects for scene composition, we adopt image-to-3d model (Wu et al. 2024a) to generate 3D objects by inputting web-crawled car image. To avoid quality degradation and blurry texture, image-to-3d is used than text-to-3d (Lin et al. 2023). In the case of layout and building, we utilize 3D city generation (Xie et al. 2024) results for base layout.

Data type	(Zhong et al. 2024) *real*	(Gao et al. 2024) *real*	(Wu et al. 2024b) *real*	(Wan et al. 2024) *real*	*MultiFloodSynth* *synthetic*
Flood-Level	4	✗	3	5	5
Availability	✗	✗	✗	✓	✓†
Frames	1,177	45,199	19,062	2,000	70,117
Labeling	Manual (Inconsistent)	Manual (Inconsistent)	Manual (Inconsistent)	Manual (Inconsistent)	Auto (Consistent)
Normal Map	✗	✗	✗	✗	✓
Camera Pos	✗	✗	✗	✗	✓
3D BBox	✗	✗	✗	✗	✓
2D BBox	✓	✓	✗	✓	✓
Depth Map	✗	✗	✗	✗	✓
Semantic Seg	✗	✓	✗	✗	✓
Instance Seg	✗	✗	✗	✗	✓
Fine-grained Seg	✗	✗	✗	✗	✓

Table 2: Comparison between our *MultiFloodSynth* and related datasets. †Dataset will be available under the acceptance.

(a) Sample of *flooded* case. (b) Sample of *non-flooded* case.

Figure 4: Sample of our *MultiFloodSynth*.



(a) 2D bounding box. (b) Normal map. (c) 3D bounding box.

(d) Semantic seg. (e) Instance seg. (f) Fine-grained seg.

(g) Depth map. (h) Point cloud. (i) Camera 3D pos.

Figure 5: Richness of annotation type of our *MultiFloodSynth*.

Simulating Flood Wave

In flood hazard situation, flood simulating is pretty important factor which determine the flood level (Wan et al. 2024) and annotation-level. Some attributes (*e.g.*, reflection, roughness, opacity, specular, texture) of flood object will directly affect to extract training feature by neural network. In this regards, we also simulate flood dynamics and visual appearance as shown in Fig. 3. Three factors, *main wave, wave foam, gravity*, decide the visual magnitude of level-of-wave as dynamics. Other factors (*e.g.*, wave size, depth, height) change the appearance of flood as static component.

MultiFloodSynth Generation

For flood synthetic generation, we construct flood environments based on each objects by varying some parameters (Tab. 1) for diverse data distribution. Flood level is attributed into 5 level as multi-classes. To enhance the quality and domain similarity, we adopt domain randomization (Rawal, Sompura, and Hintze 2023) in all the objects (e.g., light, camera view, object position, etc). Each object is randomly located in every generation pipeline to avoid bias and sparsity of dataset and to include some crucial corner cases. As a result, our *MultiFloodSynth* consists of a total of 70,117 images, with 14,593 *non-flooded* images and 55,524 *flooded* images. Samples are illustrated in Fig. 4. Total image and instance for each class is listed in Tab. 3. For multi-type of annotations, we extract 9 types (*i.e.*, semantic/instance/fine-grained segmentation, 2D/3D bounding box) of paired synthetic scene as shown in Fig. 5. For segmentation map, we allocate the label into car and flood. Differences between competing flood datasets are listed in Tab. 2.

Experiments

Environmental Settings

For detection model, we choose YOLOv10(Wang et al. 2024a). For training hyperparameters, we set batch size as 256, learning rate as 0.001 with 100 epochs. Image-to-3D model is used with Unique3D (Wu et al. 2024a). Our virtual environment is based on NVIDIA Omniverse simulator.

Class		# of Images	Instance
Non-flooded	Level 0	14,593	37,662
Flooded	Level 1	17,485	55,624
	Level 2	14,541	36,141
	Level 3	12,837	61,132
	Level 4	10,661	24,476
Total		70,117	215,035

Table 3: Summary of our *MultiFloodSynth* composition.

Training	PR(%)↑	RC(%)↑	mAP_{50} ↑	$mAP_{50\text{-}95}$ ↑
$\mathcal{D}_{real}$	37.61	**43.81**	33.21	22.92
$\mathcal{D}_{synth}$	19.36	7.21	7.41	4.32
$\mathcal{D}_{real} + \mathcal{D}_{synth}$	**40.97**	41.75	**35.78**	**23.16**

(a) Results on YOLOv10-N.

Training	PR(%)↑	RC(%)↑	mAP_{50} ↑	$mAP_{50\text{-}95}$ ↑
$\mathcal{D}_{real}$	58.92	**58.12**	56.66	40.64
$\mathcal{D}_{synth}$	15.17	14.41	6.94	3.97
$\mathcal{D}_{real} + \mathcal{D}_{synth}$	**61.16**	57.77	**58.61**	**42.71**

(b) Results on YOLOv10-B.

Table 4: Classification performance according to the training dataset. PR, RC, mAP denote precision, recall, mean average precision, respectively. Best score is denoted as **boldfont**.

Comparison on Detection Performance

To evaluate the superiority of our *MultiFloodSynth*, we compare the flood detection performance by varying the training dataset. Based on (Wan et al. 2024), we denotes previous real dataset as $\mathcal{D}_{real}$ and our dataset as $\mathcal{D}_{synth}$. Then, we train the model with different training configuration as: (1) $\mathcal{D}_{real}$, (2), $\mathcal{D}_{synth}$, (3) $\mathcal{D}_{real} + \mathcal{D}_{synth}$. For evaluation metric, we include precision, recall, mAP at 50 and 50-95. In evaluation, we train two size models, YOLOv10-N (*small size*) and YOLOv10-B (*large size*).

The results are shown in Tab. 4. It show that the real-world data training outperformed the synthetic data training. However, training mixing two dataset ($\mathcal{D}_{real} + \mathcal{D}_{synth}$) demonstrated improved performance. To conclude, our *MultiFloodSynth* boost the detection performance in flood hazard recognition task with consistent annotation while also alleviating the cost of burden data collection process.

Evaluation of *MultiFloodSynth*

Furthermore, we intend to evaluate our *MultiFloodSynth* in the perspective of realism compared with real dataset. To do that, we borrow the recent synthetic dataset evaluation metric, *Realistic Score* which is proposed in urban world generation in (Shang et al. 2024). By randomly selecting 1K samples in each dataset, we average the score. We normalized the scores of $\mathcal{D}_{synth}$ based on the scores of the real dataset. As shown in Tab. 5, our dataset showed 93.17% plausibility which is similar level of realism compared with real dataset.

Data	Realistic Score(%)↑
$\mathcal{D}_{real}$	100 (7.18)
$\mathcal{D}_{synth}$	93.17 (6.69)

Table 5: Realism of synthetic data compared with real data.

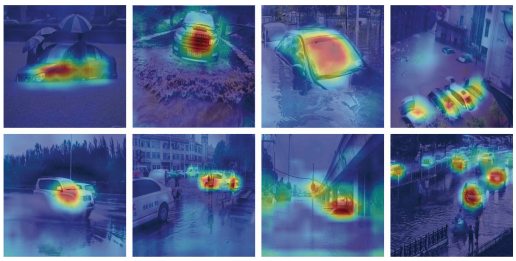


Figure 6: Visualization of EigenCAM for explainability. A red part indicates that the model considers that part to be important evidence for decision.

Explanation of Flood Detection Model

For explainability of detection, we adopt recent XAI method, EigenCAM(Muhammad and Yeasin 2020) which can analyze some specific parts of an input image that play a crucial role in the model decision. As shown in Fig. 6, our *MultiFloodSynth* played a crucial role in training and thus enables model to recognize important evidence from images.

Conclusions

In this paper, we have presented synthetic dataset generation pipeline for urban flood detection into 5 levels and evaluated the utility of our *MultiFloodSynth*. For faithful synthetic dataset, we charactersize several parameters including not only environmental settings, *e.g.*, lighting color, camera position, but also flood simulation factors, *e.g.*, roughness, texture, opacity, etc. To mitigate the data bias and domain gap, we adopt domain randomization and vary the above parameters in each generation pipeline. Moreover, for the sake of efficiency of pipeline, we leverage the recent generation techniques, image-to-3D generation and urban city generation, for 3D object and base layout of virtual flood world, respectively. Experimental results demonstrated that the model trained with our synthetic dataset and real-world dataset show enhanced detection performance in object-localization based flood-level recognition. In addition, our *MultiFloodSynth* showed on-par realism compared with real dataset in terms of *Realistic Score* metric. To conclude, our *MultiFloodSynth* generated from our parameter-controllable flood environment can serve as a valuable training dataset, alternating data requirements of real-world dataset.

Acknowledgments

This research was supported by Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by Ministry of Culture, Sports and Tourism in 2024 (Project Name : Developing Professionals for R&D in Contents Production Based on Generative AI and Cloud, Project Number : RS-2024-00352578, Contribution Rate: 100%) and Artificial intelligence industrial convergence cluster development project funded by the Ministry of Science and ICT(MSIT, Korea) & Gwangju Metropolitan City.

References

Amador Herrera, J. A.; Klein, J.; Liu, D.; Pałubicki, W.; Pirk, S.; and Michels, D. L. 2024. Cyclogenesis: Simulating Hurricanes and Tornadoes. *ACM Transactions on Graphics (TOG)*, 43(4): 1–16.

Chaudhary, P.; D'Aronco, S.; Leitão, J. P.; Schindler, K.; and Wegner, J. D. 2020. Water level prediction from social media images with a multi-task ranking approach. *ISPRS Journal of Photogrammetry and Remote Sensing*, 167: 252–262.

Delussu, R.; Putzu, L.; and Fumera, G. 2024. Synthetic Data for Video Surveillance Applications of Computer Vision: A Review. *International Journal of Computer Vision*, 1–37.

Ebadi, S. E.; Dhakad, S.; Vishwakarma, S.; Wang, C.; Jhang, Y.-C.; Chociej, M.; Crespi, A.; Thaman, A.; and Ganguly, S. 2022. PSP-HDRI +: A Synthetic Dataset Generator for Pre-Training of Human-Centric Computer Vision Models. arXiv:2207.05025.

Gao, K.; Yang, Z.; Gao, X.; Shao, W.; Wei, H.; and Xu, T. 2024. Measuring urban waterlogging depths from video images based on reference objects. *Journal of Flood Risk Management*, 17(1): e12948.

Greff, K.; Belletti, F.; Beyer, L.; Doersch, C.; Du, Y.; Duckworth, D.; Fleet, D. J.; Gnanapragasam, D.; Golemo, F.; Herrmann, C.; Kipf, T.; Kundu, A.; Lagun, D.; Laradji, I. H.; Liu, H.-T.; Meyer, H.; Miao, Y.; Nowrouzezahrai, D.; Oztireli, C.; Pot, E.; Radwan, N.; Rebain, D.; Sabour, S.; Sajjadi, M. S. M.; Sela, M.; Sitzmann, V.; Stone, A.; Sun, D.; Vora, S.; Wang, Z.; Wu, T.; Yi, K. M.; Zhong, F.; and Tagliasacchi, A. 2022. Kubric: A scalable dataset generator. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 3739–3751.

Hamza, A.; Lojo, A.; Núñez-Marcos, A.; and Atutxa, A. 2024. Ali-AUG: Innovative Approaches to Labeled Data Augmentation using One-Step Diffusion Model. arXiv:2410.18678.

Hao, S.; Han, W.; Jiang, T.; Li, Y.; Wu, H.; Zhong, C.; Zhou, Z.; and Tang, H. 2024. Synthetic data in AI: Challenges, applications, and ethical implications. arXiv:2401.01629.

Hong, Z.; Hamdan, E.; Zhao, Y.; Ye, T.; Pan, H.; and Cetin, A. E. 2024. Wildfire detection via transfer learning: a survey. *Signal, Image and Video Processing*, 18(1): 207–214.

Hummel, M.; and van Kooten, K. 2019. Leveraging nvidia omniverse for in situ visualization. In *High Performance Computing: ISC High Performance 2019 International Workshops, Frankfurt, Germany, June 16-20, 2019, Revised Selected Papers 34*, 634–642. Springer.

Islam, K.; Zaheer, M. Z.; Mahmood, A.; and Nandakumar, K. 2024. DiffuseMix: Label-Preserving Data Augmentation with Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 27621–27630.

Jung, Y.; Byun, S.; Kim, B.; Amin, S. U.; and Seo, S. 2024. Harnessing synthetic data for enhanced detection of Pine Wilt Disease: An image classification approach. *Computers and Electronics in Agriculture*, 218: 108690.

Karanjit, R.; Pally, R.; and Samadi, S. 2023. FloodIMG: flood image DataBase system. *Data in brief*, 48: 109164.

Khullar, D.; Sokhandan, N.; Kulkarni, N.; and Shah, Y. 2023. Synthetic data generation for scarce road scene detection scenarios.

Kim, H.-C.; Lam, H.-K.; Lee, S.-H.; and Ok, S.-Y. 2024. Early Fire Detection System by Using Automatic Synthetic Dataset Generation Model Based on Digital Twins. *Applied Sciences*, 14(5): 1801.

Kokosza, A.; Wrede, H.; Gonzalez Esparza, D.; Makowski, M.; Liu, D.; Michels, D. L.; Pirk, S.; and Palubicki, W. 2024. Scintilla: Simulating Combustible Vegetation for Wildfires. *ACM Transactions on Graphics (TOG)*, 43(4): 1–21.

Lee, K.; Shin, U.; and Lee, B.-U. 2024. Learning to Control Camera Exposure via Reinforcement Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2975–2983.

Lin, C.-H.; Gao, J.; Tang, L.; Takikawa, T.; Zeng, X.; Huang, X.; Kreis, K.; Fidler, S.; Liu, M.-Y.; and Lin, T.-Y. 2023. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 300–309.

Lo, S.-W.; Wu, J.-H.; Chang, J.-Y.; Tseng, C.-H.; Lin, M.-W.; and Lin, F.-P. 2021. Deep sensing of urban waterlogging. *IEEE Access*, 9: 127185–127203.

Mittal, M.; Yu, C.; Yu, Q.; Liu, J.; Rudin, N.; Hoeller, D.; Yuan, J. L.; Singh, R.; Guo, Y.; Mazhar, H.; et al. 2023. Orbit: A unified simulation framework for interactive robot learning environments. *IEEE Robotics and Automation Letters*, 8(6): 3740–3747.

Muhammad, M. B.; and Yeasin, M. 2020. Eigen-cam: Class activation map using principal components. In *2020 international joint conference on neural networks (IJCNN)*, 1–7. IEEE.

Pally, R.; and Samadi, S. 2022. Application of image processing and convolutional neural networks for flood image classification and semantic segmentation. *Environmental modelling & software*, 148: 105285.

Rawal, P.; Sompura, M.; and Hintze, W. 2023. Synthetic data generation for bridging Sim2Real gap in a production environment. arXiv:2311.11039.

Richter, S. R.; AlHaija, H. A.; and Koltun, V. 2022. Enhancing photorealism enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2): 1700–1715.

Schieber, H.; Demir, K. C.; Kleinbeck, C.; Yang, S. H.; and Roth, D. 2024. Indoor synthetic data generation: A systematic review. *Computer Vision and Image Understanding*, 103907.

Shang, Y.; Lin, Y.; Zheng, Y.; Fan, H.; Ding, J.; Feng, J.; Chen, J.; Tian, L.; and Li, Y. 2024. UrbanWorld: An Urban World Model for 3D City Generation. arXiv:2407.11965.

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*, 5354–5363.

Wan, J.; Qin, Y.; Shen, Y.; Yang, T.; Yan, X.; Zhang, S.; Yang, G.; Xue, F.; and Wang, Q. J. 2024. Automatic detection of urban flood level with YOLOv8 using flooded vehicle dataset. *Journal of Hydrology*, 639: 131625.

Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; and Ding, G. 2024a. Yolov10: Real-time end-to-end object detection. arXiv:2405.14458.

Wang, Z.; Draghi, B.; Rotalinti, Y.; Lunn, D.; and Myles, P. 2024b. High-fidelity synthetic data applications for data augmentation.

Wu, K.; Liu, F.; Cai, Z.; Yan, R.; Wang, H.; Hu, Y.; Duan, Y.; and Ma, K. 2024a. Unique3D: High-Quality and Efficient 3D Mesh Generation from a Single Image. arXiv:2405.20343.

Wu, L.; Liu, Y.; Zhang, J.; Zhang, B.; Wang, Z.; Tong, J.; Li, M.; and Zhang, A. 2024b. Identification of flood depth levels in urban waterlogging disaster caused by rainstorm using a CBAM-improved ResNet50. *Expert Systems with Applications*, 124382.

Xie, H.; Chen, Z.; Hong, F.; and Liu, Z. 2024. Citydreamer: Compositional generative model of unbounded 3d cities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9666–9675.

Zhang, S.; Zhou, M.; Wang, Y.; Luo, C.; Wang, R.; Li, Y.; Yin, X.; Zhang, Z.; and Peng, J. 2024. CityX: Controllable Procedural Content Generation for Unbounded 3D Cities. arXiv:2407.17572.

Zhong, P.; Liu, Y.; Zheng, H.; and Zhao, J. 2024. Detection of urban flood inundation from traffic images using deep learning methods. *Water Resources Management*, 38(1): 287–301.

Zhu, J.; Li, S.; Liu, Y.; Huang, P.; Shan, J.; Ma, H.; and Yuan, J. 2024. ODGEN: Domain-specific Object Detection Data Generation with Diffusion Models. arXiv:2405.15199.