2022-06-09

# Generating Reality-Analogous Datasets for Autonomous UAV Navigation using Digital Twin Areas

Thomas Lee
*TU Dublin*, thomas.lee@tudublin.ie

Susan Mckeever
*TU Dublin*, susan.mckeever@tudublin.ie

Jane Courtney
*TU Dublin*, jane.courtney@tudublin.ie



## Recommended Citation

T. Lee, S. Mckeever and J. Courtney, "Generating Reality-Analogous Datasets for Autonomous UAV Navigation using Digital Twin Areas," 2022 33rd Irish Signals and Systems Conference (ISSC), Cork, Ireland, 2022, pp. 1-6, doi: 10.1109/ISSC55427.2022.9826198


Funder: Scientific Foundation Ireland

# Generating Reality-Analogous Datasets for Autonomous UAV Navigation using Digital Twin Areas


Thomas Lee
*School of Electrical & Electronic Eng.*
*TU Dublin*
Dublin, Ireland
thomas.lee@tudublin.ie

Susan Mckeever
*School of Computer Science*
*TU Dublin*
Dublin, Ireland
jane.courtney@tudublin.ie

Jane Courtney
*School of Electrical & Electronic Eng.*
*TU Dublin*
Dublin, Ireland
susan.mckeever@tudublin.ie



***Abstract*—In order for autonomously navigating Unmanned Air Vehicles(UAVs) to be implemented in day-to-day life, proof of safe operation will be necessary for all realistic navigation scenarios. For Deep Learning powered navigation protocols, this requirement is challenging to fulfil as the performance of a network is impacted by how much the test case deviates from data that the network was trained on. Though networks can generalise to manage multiple scenarios in the same task, they require additional data representing those cases which can be costly to gather. In this work, a solution to this data acquisition problem is suggested by way of the implementation of a visually realistic, yet artificial, simulated dataset. A method is presented for the creation of a ”Digital Twin Area” inside of a modern high fidelity game engine using 3D scanned models of physical locations, and a realistic dataset of each area is created to showcase this concept.**



***Index Terms*—simulation, drones, deep learning, image sampling, autonomous aerial vehicles, digital twin**


## I. Introduction

In the past 5 years the UAV research area has had a wealth of projects involving the development of Autonomous functions [1], of those projects Deep Learning based solutions have been featured in many [2], [3]. Image Classification, Object Detection and Image Segmentation are all tasks which utilize Deep Learning and have been employed in the context of Autonomous Navigation. However, in order to achieve good performance in these trained networks, large datasets are required to provide the adequate variation needed for generalisation [4]. By generating some of this data artificially via a simulation, and using it to reinforce existing data, it has been shown to improve the accuracy of a network [5], however the simulation used in these projects typically do not derive from reality and are instead generated procedurally from virtual assets or are virtual spaces built by hand as a representation of a real space. For this project's simulation the use of a Digital Twin was proposed, which uses a quantitatively linked simulation of a physical object in order to analyse and test that object in ways that would be unfeasible through traditional means [6]. Additionally, this project aims to expand on the concept by using a 3D scanned and textured model of a physical location as a Digital Twin otherwise referred to as a ”Digital Twin Area” for use as a visual base for the generation of an artificial dataset. Information regarding the state of the art and core project concepts is summarised in Section II. Details regarding the selection or creation of adequate 3D Scan models, the construction of this area inside of the Unity game engine, and the simulated sampling methodology, are covered in Section III. The dataset results, analysis and a demonstration of potential applications are covered in Section IV.

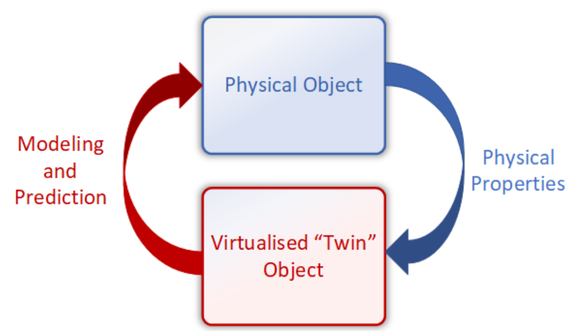


Fig. 1. A simplified description of the ”Digital Twin” concept.

### A. Autonomous Navigation

The development of Autonomous Navigation is challenging, due to the complexity of the tasks involved. Perception [7], [8], state estimation [9], [10], and rule-based navigational protocol [11], [12] are some examples of these tasks. Many of these projects sought to use a monocular camera system and Convolutional Neural Networks (CNN) for the prediction of task-relevant information from images. One key issue with such trained networks is the specificity of the training data used, visual data especially carries many environmental elements which can affect the performance of the model when in an untrained environment [2]. Conversely, to generalise the network by training on an excessively varied dataset with many


This research was funded by the Scientific Foundation Ireland (SFI) ADVANCE Centre for Research Training.

different environment variations would lead to a reduction in the performance of the network when approached with specific use cases, even those that are present in the training data [13], [14]. When considering an image of a given environment, there are many dimensions where variation can occur, as an example, it is typical in most places for the average light levels of an area to change based on the time of day, the time of year, and the weather. Understanding how each of these phenomena affects the certainty of model prediction is likely to be critical for safe Autonomous Navigation in these environments [15]. By using a Digital Twin approach applied over an area, it is possible to generate data with these affects simulated such that this issue is mitigated to a degree.

### *B. Digital Twins*

The Digital Twin concept, described in Fig. 1, was initially only considered in terms of a single subject or product [16], this allowed researchers to model the behaviour of a physical object, such as the forces on said object without risk by imparting the same forces on the simulated twin object. The simulated twin could in turn reveal information regarding the behaviour of the physical object, for example: by linking the simulated twin object's velocity with the physical object's measured velocity one can determine the relative position of the real object without having to measure it physically [17]. This concept is then expanded upon to include multiple elements simulated in tandem into a Digital Twin "domain" [18].

## II. Background

It is commonplace in UAV research to begin with a simulation of the project, this approach is beneficial as initial tests in the physical space can be costly and dangerous if unpredicted events occur during the experiment. Recent Literature [19], [20] suggests two common simulation interfaces used in the robotics space from which UAV research inherits many tools, such as the Robotic Operating System(ROS) framework for use in the development of robotics. Gazebo specifically is an open source simulator designed to interface with ROS and is the more popular of the two, as an alternative which is also open source, the Microsoft-developed AirSim simulator is more directed toward UAV research [21]. These simulators are orientated more towards physical simulation rather than visual quality, as a result it is not recommended to use these simulators for the purposes of simulating visual data for convolutional neural networks [22], [23]. However, engines designed for powering modern Video Game titles such as Unity or the Unreal engine could potentially suit the task of realistic visual simulation better, as an added benefit, engines such as these are highly optimised in terms of runtime and development allowing for rapid prototyping and lower operational overhead.

### *A. Trained Autonomous Features*

In a previous review of literature, the functions of Deep Neural Network(DNN) based autonomous UAV projects were developed into an informative taxonomy [1]. Of the conclusions drawn from this review, it was posited that one of the most commonly researched functions was Collision Avoidance and said research typically involved CNN based, Residual Network optimised solutions [24]. As an example, [2] uses this approach to train a network for steering angle and crash probability from footage gathered from ground-based vehicles and employs this network in an autonomous UAV. Object Distinction was also revealed to be another popular solution in recent UAV research and was typically achieved through fine-tuning generic classification models trained on very large datasets [4] to recognise specific elements in an image [25], [26]. Additionally, The review author noted a lack of the Environmental Distinction feature among projects.


Fig. 2. Simulated image samples of several 3D scanned locations, rendered in the Unity engine.

### *B. Data Collection*

The most major determining factor of the performance of a learned model is the data that it is trained on. Datasets are most commonly either collected or annotated manually by experts [4], this reliance on human interaction for the generation of data causes a number of issues with not just research efficiency, but the quality of the data itself. Such a manual approach to data collection results in uncontrollable elements of the environment such as time, or weather varying between, or during sampling sessions which can be detrimental to the usefulness of image-based datasets especially. Additionally, if the intention was to collect a gamut of data that varies by one of these environmental elements, a manual collection approach is simply unfeasible given the exponential increase in the amount of data required for each dimension of variance added.

### *C. Simulating Data*

It is easy to assume that in order to accurately predict something in the physical world using a learned network, that the data used to train the network must also be derived from the physical world. Though some connection is required, it has been shown on multiple accounts that simulation can be used in certain circumstances, such as improving image segmentation [27], [28], or for the initial training of networks [5]. An underrepresented benefit of simulated data collection is the control the researcher has over the parameters of the simulation [29]. As an example, the runtime element of a simulation is able to be controlled, if the sampling time of a sensor is undesirable the researcher can pause the simulation,

Fig. 3. A 3D Scanned scene containing a door, scaled to the 0.8m reference object (pictured in white).

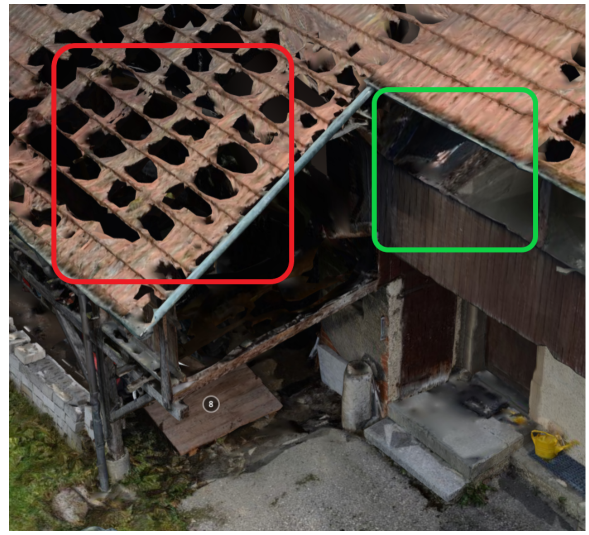

Fig. 4. Example of undesirable features in a scanned scene, holes in the mesh pictured in red, and error due to narrow perspective sampling pictured in green.

collect all samples from all sensors as needed, and there will be no need to account for time deviations caused by hardware limitation or code execution delay. This is similarly true for the relocation of sampling equipment, which can be moved instantly in simulation. Additionally the simulation area can also change as needed, either being generated dynamically by using a library of individual objects such as street corners, houses, lamps and buildings, or set specifically by the researcher for the achievement of specific goals, such as being equivalent spatially to a physical location. Fig. 2 visually compares several reconstructed 3D scan areas that were used for the development and testing of the simulation, for this proof of concept, acquired 3D Scans [30]–[34] were used in place of a bespoke Digital Twin Area.

## III. Method

The process of finding, selecting or creating suitable scenes for use in sample generation is challenging. The 3D scan must be resolute enough to not cause artifacts in the resultant captured mesh. However, the scan must also have enough physical size to be able to gather a valid dataset of the scene without overexposure to the model's edges, while also being of an acceptable filesize to be loaded into system memory for the duration of the sampling run. Additionally, as it is with image data, the sampled scene will be effected by the environmental parameters outlined in Section II.B. It is recommended that an overcast day at midday is chosen when scanning an area for 3D reconstruction as this will result in a bright, softly lit model with fewer sharp shadows which can affect the accuracy of simulation.

### A. 3D Scans

Ideally, the scan should be captured off of a location the researcher has physical access to for appropriate ground-truth measurement. For the purposes of this research, resolution quality was preferred over area size. Problematic scans that contained a large number of holes in the mesh, and narrow perspective lines (see Fig. 4) and shadows were avoided as much as possible. Five 3D Scans were selected where these features were minimised, while showcasing an array of different locations. It is preferable to fabricate a 3D Scan using a known permissible area. Though doing so requires equipment and technical expertise, there are several benefits to be considered. Most notably, having access to the physical area which was scanned allows for better scale calibration in the simulation, additionally any measurements made in that area using ground truth sensors also applies to the digital recreation of the scan and vice versa (assuming that the quality of the scan can be verified). For this reason, bespoke scene creation for quantitatively accurate simulation is intended to be approached in future work. Even with bespoke scene creation, it is unrealistic to expect to gather high resolution scans of a large area, in most cases the scenes are composed of high resolution point clouds which diminish in resolution the further they are from the source of the sampling device. Though it is not impossible to create a scan of an area with a consistent high resolution by using multiple sampling points and techniques, longer scan times increases the deviation in lighting and texture (for outdoor scans), which may require post-processing to repair (potentially affecting the connection between the simulation and the real scene).

### B. Scene Configuration

The 3D Scans were imported using the Unity GUI Import function, these scans were then configured for lossless texture resolution and initially were self-lit to ignore all dynamic scene lights (since the texture would have shadows baked into it from the initial scan), this setting was reverted back to the default to allow for the day/night cycle to affect the scan, instead, a gradient emission texture was used to align the level of self-illumination with the time of day. Additionally, since the scans were acquired from an online repository, physical size could not be verified, therefore in order to align the scale of the mesh

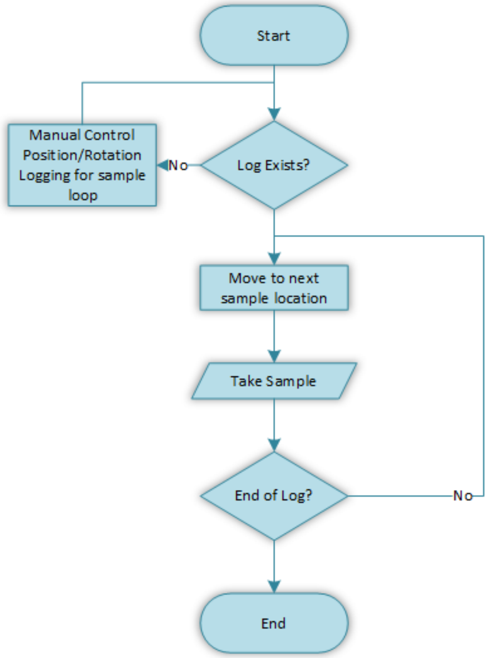


Fig. 5. General strategy used for the simulation and sampling of artificial datasets.

to a realistic estimation, an object matching the approximate width of an average European door (0.8 meters wide) was used as a reference and doors visible in the scan were scaled to match (see Fig. 3). A basic quad-rotor helicopter UAV control scheme was implemented in C# using the Unity scripting API, with features to allow for sample recording, which would later be modified to record positions and rotations. For ease of use, inputs were mapped to both a standard game pad as well as a keyboard for control. Although this feature was not necessary for the simulation running automatic generation, it is a recommended quality-of-life feature for the rapid creation of new datasets in different environments.

### C. Sampling

As previously stated, one of the benefits of simulated data generation is the ability to control the simulation runtime in order to ensure there is no deviation between samples taken at the same position. Fig. 5 below, describes the strategy employed in order to maximise this effect.

Rather than sample using direct control of the UAV in the simulation, the position, rotation, and runtime of the UAV at each sample is recorded and stored as a csv file for use in between simulations. This allows the sampler to vary the parameters of the simulation, while still capturing images at the exact same perspectives to assist in ensuring that the only difference between these sampling cycles are the chosen variables. Where traditional sampling would have either manual or automatic relocation between points and sampling that would include a minimum sample time, which would be based on the specifics of the hardware used for the sampling. In a dataset that is generated from simulation however, the hardware can be ideal with a sample time of zero. This is normally an undesirable feature of simulation, but in the context of data creation it can be a benefit. Once the logs have been created, the simulation can be left to run automatically. The filename of each image is generated to contain the scene and position number for reference, and can also contain other appropriate metadata as need be.


Fig. 6. "Farmhouse" scene, iterated over 4 points on a cyclical day/night axis (6am to 6pm)

TABLE I
SAMPLE GENERATION RESULTS PER AREA

| Location | Number of Samples |
|---|---|
| Farmhouse | 5598 |
| Alleyway | 2010 |
| Street | 2001 |
| Indoor 1 | 1097 |
| Indoor 2 | 2013 |

## IV. RESULTS

As a test of the simulation concept and sampling strategy, position and rotation logs were created for 5 areas with various geographic qualities, these logs were used to create individual datasets of each area automatically, details of these results can be seen in Table I.

### A. Time Variation

To demonstrate of the concept of using simulated dataset generation to create varied environmental samples of the same image, the sampling workflow was modified to iterate the light level and simulated solar angle over a cyclical axis to emulate the day/night cycle. The results of which are shown in Fig. 6. This is potentially the most impactful feature of the proposed approach, it allows for not only the control of one environmental variable, but could be used to control multiple in tandem to generate a spectrum of environmental variance. As such, the generation and analysis of a dataset that iterates over a number of time and weather configurations is expected to be a topic for future research.

## V. CONCLUSION

This research was intended to investigate Two concepts. The First being the viability of using a high fidelity rendering engine in conjunction with 3D Scans of a physical space to create a Digital Twin version of that space. The Second is the viability of using said Digital Twin Areas to generate image samples for the inference of Neural Network based

Autonomous UAV functions. To this end, it was discovered that not only are both concepts viable, but the idea of an artificially generated dataset that is visually comparable with a physical counterpart has considerable potential in assisting with the development of generalised Autonomous Navigation solutions.

### A. Future Work

The 3D scans used in this project demonstrated the validity of this sampling approach and concept. However, the creation of a bespoke, linked mesh of a known accessible area is considered the most optimal approach. As such, future research aims to create a Digital Twin space that is linked to a known physical space for quantitative testing of this approach. Additionally, further testing of Neural Network responsiveness to the artificial datasets is required, which is likely to involve research done in the area of Uncertainty Estimation [35], [36]. Furthermore, research into the generation of Simultaneous time and weather varied datasets using this simulation strategy is of particular interest.

## References


[1] T. Lee, S. Mckeever, and J. Courtney, "Flying Free: A Research Overview of Deep Learning in Drone Navigation Autonomy," *Drones*, vol. 5, no. 2, p. 52, jun 2021.

[2] A. Loquercio, A. I. Maqueda, C. R. Del-Blanco, and D. Scaramuzza, "DroNet: Learning to Fly by Driving," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1088–1095, apr 2018.

[3] D. Gandhi, L. Pinto, and A. Gupta, "Learning to fly by crashing," in *IEEE International Conference on Intelligent Robots and Systems*, vol. 2017-Septe. Institute of Electrical and Electronics Engineers Inc., dec 2017, pp. 3948–3955.

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, no. June. IEEE, jun 2009, pp. 248–255. [Online]. Available: https://ieeexplore.ieee.org/document/5206848/

[5] A. Vierling, T. Sutjaritvorakul, and K. Berns, "Dataset Generation Using a Simulated World," in *International Conference on Robotics in Alpe-Adria Danube Region*, 2020, pp. 505–513. [Online]. Available: http://link.springer.com/10.1007/978-3-030-19648-6_58

[6] M. Grieves, "Digital Twin : Manufacturing Excellence through Virtual Factory Replication - A Whitepaper by Dr . Michael Grieves," *White Paper*, no. March, pp. 1–7, 2014.

[7] M. T. Matthews and S. Yi, "Model Reference Adaptive Control and Neural Network Based Control of Altitude of Unmanned Aerial Vehicles," in *2019 SoutheastCon*, no. 1. IEEE, apr 2019, pp. 1–8. [Online]. Available: https://ieeexplore.ieee.org/document/9020447/

[8] X. Dai, Y. Zhou, S. Meng, and Q. Wu, "Unsupervised Feature Fusion Combined with Neural Network Applied to UAV Attitude Estimation," in *2018 IEEE International Conference on Robotics and Biomimetics, ROBIO 2018*. IEEE, 2018, pp. 874–879.

[9] A. Giusti, J. Guzzi, D. C. Ciresan, F.-L. He, J. P. Rodriguez, F. Fontana, M. Faessler, C. Forster, J. Schmidhuber, G. D. Caro, D. Scaramuzza, and L. M. Gambardella, "A Machine Learning Approach to Visual Perception of Forest Trails for Mobile Robots," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 661–667, jul 2016.

[10] Y. Zhang, X. Xiao, and X. Yang, "Real-Time object detection for 360-degree panoramic image using CNN," in *Proceedings - 2017 International Conference on Virtual Reality and Visualization, ICVRV 2017*. Institute of Electrical and Electronics Engineers Inc., jul 2017, pp. 18–23.

[11] H. Shiri, J. Park, and M. Bennis, "Remote UAV Online Path Planning via Neural Network-Based Opportunistic Control," *IEEE Wireless Communications Letters*, vol. 9, no. 6, pp. 861–865, jun 2020.

[12] X. Han, J. Wang, J. Xue, and Q. Zhang, "Intelligent Decision-Making for 3-Dimensional Dynamic Obstacle Avoidance of UAV Based on Deep Reinforcement Learning," in *2019 11th International Conference on Wireless Communications and Signal Processing, WCSP 2019*. IEEE, 2019.

[13] A. Alshehri, S. Member, Y. Bazi, and S. Member, "Deep Attention Neural Network for Multi-Label Classification in Unmanned Aerial Vehicle Imagery," *IEEE Access*, vol. 7, pp. 119 873–119 880, 2019.

[14] O. Csillik, J. Cherbini, R. Johnson, A. Lyons, and M. Kelly, "Identification of Citrus Trees from Unmanned Aerial Vehicle Imagery Using Convolutional Neural Networks," *Drones*, vol. 2, no. 4, p. 39, nov 2018.

[15] A. Loquercio, E. Kaufmann, R. Ranftl, A. Dosovitskiy, V. Koltun, and D. Scaramuzza, "Deep Drone Racing: From Simulation to Reality With Domain Randomization," *IEEE Transactions on Robotics*, vol. 36, no. 1, pp. 1–14, feb 2020.

[16] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, "Characterising the Digital Twin: A systematic literature review," *CIRP Journal of Manufacturing Science and Technology*, vol. 29, pp. 36–52, 2020.

[17] R. Stark, T. Damerau, I. Technology, D. Virtual, P. Creation, D. Twin, and C. Components, "Digital Twin," 2019.

[18] F. J. Kahlen, S. Flumerfelt, and A. Alves, "Transdisciplinary perspectives on complex systems: New findings and approaches," *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches*, pp. 1–327, 2016.

[19] S. Dionisio-Ortega, L. O. Rojas-Perez, J. Martinez-Carranza, and I. Cruz-Vega, "A deep learning approach towards autonomous flight in forest environments," in *2018 International Conference on Electronics, Communications and Computers (CONIELECOMP)*. IEEE, feb 2018, pp. 139–144. [Online]. Available: http://ieeexplore.ieee.org/document/8327189/

[20] G. Muñoz, C. Barrado, E. Çetin, and E. Salami, "Deep Reinforcement Learning for Drone Delivery," *Drones*, vol. 3, no. 3, p. 72, sep 2019.

[21] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles," *Springer Proceedings in Advanced Robotics*, vol. 5, pp. 621–635, 2018.

[22] R. Codd-Downey, P. M. Forooshani, A. Speers, H. Wang, and M. Jenkin, "From ROS to unity: Leveraging robot and virtual environment middleware for immersive teleoperation," *2014 IEEE International Conference on Information and Automation, ICIA 2014*, no. July, pp. 932–936, 2014.

[23] Z. Wang, K. Han, and P. Tiwari, "Digital twin simulation of connected and automated vehicles with the unity game engine," *Proceedings 2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence, DTPI 2021*, pp. 180–183, 2021.

[24] D. Palossi, A. Loquercio, F. Conti, E. Flamand, D. Scaramuzza, and L. Benini, "A 64mW DNN-based Visual Navigation Engine for Autonomous Nano-Drones," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8357–8371, may 2018.

[25] D. R. Hartawan, T. W. Purboyo, and C. Setianingsih, "Disaster victims detection system using convolutional neural network (CNN) method," in *Proceedings - 2019 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology, IAICT 2019*, 2019, pp. 105–111.

[26] S. P. Yong and Y. C. Yeong, "Human Object Detection in Forest with Deep Learning based on Drone's Vision," in *2018 4th International Conference on Computer and Information Sciences: Revolutionising Digital Landscape for Sustainable Smart Society, ICCOINS 2018 - Proceedings*. IEEE, aug 2018, pp. 1–5. [Online]. Available: https://ieeexplore.ieee.org/document/8510564/

[27] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9906 LNCS, pp. 102–118, 2016.

[28] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 3234–3243, 2016.

[29] D. Perri, M. Simonetti, and O. Gervasi, "Synthetic data generation to speed-up the object recognition pipeline," *Electronics (Switzerland)*, vol. 11, no. 1, pp. 1–19, 2022.

[30] Nedo, "Small Town Draft modus," 2017. [Online]. Available: https://sketchfab.com/3d-models/small-town-draft-modus-da3b5831716b459299b699cd2c6d3972

[31] Z. Perrino, "Street - Miguel de Benavides," 2020. [Online]. Available: https://sketchfab.com/3d-models/street-miguel-de-benavides-d71a888417974d88945775ed23e9e380

[32] A. Spognetta, "Via S. Gemini (Viterbo - Italy)," 2018. [Online]. Available: https://sketchfab.com/3d-models/via-s-gemini-viterbo-italy-608f8339c95f410ea42f2849dd272ce2

[33] 9of9, "New Year's on the Verge of Stokes Croft," 2017. [Online]. Available: https://sketchfab.com/3d-models/new-years-on-the-verge-of-stokes-croft-a327090936db421ba966c96152a8c72a

[34] Y. Martin, "Living room," 2021. [Online]. Available: https://sketchfab.com/3d-models/living-room-5621e1375b2d4ef1bdbba34aefd3fd36

[35] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, V. Makarenkov, and S. Nahavandi, "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," *Information Fusion*, vol. 76, pp. 243–297, dec 2021.

[36] A. Loquercio, M. Segu, and D. Scaramuzza, "A General Framework for Uncertainty Estimation in Deep Learning," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3153–3160, apr 2020.