Reality Analagous Synthetic Dataset Generation with Daylight Variance for Deep Learning Classification

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

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

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

Funder: Scientific Foundation Ireland

Reality Analagous Synthetic Dataset Generation with Daylight Variance for Deep Learning Classification

Thomas Lee, Susan Mckeever and Jane Courtney

TU Dublin, School of Electrical & Electronic Engineering

Abstract

For the implementation of Autonomously navigating Unmanned Air Vehicles (UAV) in the real world, it must be shown that safe navigation is possible in all real world scenarios. In the case of UAVs powered by Deep Learning algorithms, this is a difficult task to achieve, as the weak point of any trained network is the reduction in predictive capacity when presented with unfamiliar input data. It is possible to train for more use cases, however more data is required for this, requiring time and manpower to acquire. In this work, a potential solution to the manpower issues of exponentially scaling dataset size and complexity is presented, through the generation of artificial image datasets that are based off of a 3D scanned recreation of a physical space and populated with 3D scanned objects of a specific class. This simulation is then used to generate image samples that iterates temporally resulting in a slice-able dataset that contains time varied components of the same class.

Keywords: Simulation, Drones, Deep Learning, Autonomous Aerial Vehicles

1 Introduction

Figure 1: 'FIDIM' Scene view (wireframe enabled)

Over the years, there has been a notable increase in the activity within the UAV autonomy research space [Lee et al., 2021]. Many projects have a specific focus on neural network based solutions [Rojas-Perez and Martinez-Carranza, 2020]. These projects aim to infer a safe solution from a learned model to autonomous UAV tasks rather than engineer one. These tasks include image classification, object detection, and image segmentation in order for the drone to detect and identify obstacles, safe movement vectors, or unsafe situations. These tasks often require access to training datasets that are of considerable size and variation to achieve good generalisation [Deng et al., 2009], but also have specific elements through which quality training for that task can be assured [Udacity, 2016]. The challenge of training for autonomous UAV navigation is a matter of data availability, due to the high degree of variability in even common environments such as urban, suburban, forested, or rural areas it is not feasible to manually gather a dataset on all variations. Furthermore, such environments are defined arbitrarily and do not have binary transitions, but gradually shift from one environment to the other. This issue is compounded when considering other environmental factors such as light level or weather variation. It is possible to generate artificial data to make up for a lack of samples and doing so is considered to prevent overfitting [Shorten and Khoshgoftaar, 2019]. This is typically done through the augmentation of an existing, manually collected dataset [Takahashi et al., 2020]. However, these solutions are not robust, and vary from simple image processing transforms with some examples being flipping, cropping, and noise injection on existing data within the set to the synthesis of entirely samples based on the output of trained adversarial networks. in both cases, images are modified but new image subjects are not created. Augmentation can generate samples but cannot generate new sample content including information that does not already exist within the original set, simulated data with new sample content is used in reinforcement learning to create unknown situations which can assist in generalisation, however this process is time consuming and costly, requiring a simulation to be run over the course of the model's training. The information in Section 2 introduces the background concepts and previous work which lead to the creation of the simulator, while the method outlined in Section 3 details the construction of the simulated dataset. A goal of this project is to generate not just photo realistic image samples with unique content, but the same image content under unique environmental contexts, in the case of this project a cyclical time axis is introduced to mimic a Day-Night cycle which is then analysed using a pretrained YOLOv3 network in Section 4 [Redmon and Farhadi, 2018].

1.1 Autonomous Navigation

Autonomous navigation tasks tend to be complex, incorporating elements of physics calculation, state estimation [Al-Sharman et al., 2020], perception [Yang and Wang, 2020], and decision making in various degrees. Many projects aim to achieve these tasks through a monocular camera and the utilization of a Convolutional Neural Network to allow for the prediction of task-related data from input images. However, Neural Networks have accuracy issues regarding operation in environments which the autonomous navigator is not trained for [Loquercio et al., 2018] which may not cripple autonomous UAV projects mechanically but could legislatively, given that the EASA Special Condition guidelines [European Union Aviation Safety Agency, 2020] state that: "Certification of light UAS with highly integrated systems will be fundamentally based on a safety assessment that includes thrust/lift/power systems and also interaction with structures". These accuracy issues are considered to be the trade-off of performance in a Deep Learned solution based on how specific the task is [Alshehri et al., 2019]. Consider a specific task such as the identification of a specific brand of item in an image. Generalisation of that same task, such as identifying objects of a similar type (ie: other brands of that item) is likely to result in a reduction of the model's capability to identify the original brand, if even by a small amount. The reduction of task-specific performance in generalised models is exacerbated with perception based problems such as collision avoidance in UAVs [Loquercio et al., 2020]. There are many variables that bring additional complexity to the task of interpreting the image beyond what the subject is, for example: When and where the image was taken are two dimensions with massive impact on the resulting image, and the perception of said image makes up the majority of modern Deep Learned autonomous navigation solutions.

2 Background

Since UAV research in the physical space is costly (both in time and budget), projects often test the UAV design in simulation during the initial stages of project development, only moving to physical experimentation for final validation stages. A review of recent literature [Loquercio et al., 2020] identified two common simulation interfaces which are used throughout the UAV research space. Gazebo is an open source robotics simulator which can seamlessly interface with ROS, a software framework for the development and operation of robotics. An alternative to Gazebo is AirSim, an open source robotics simulator developed by Microsoft. Although a simulator like Gazebo supports high fidelity rendering the main focus of the simulator is not visual quality but realistic device control through ROS and the accurate simulation of physical forces. This can lead to discrepancies between the simulation results and physical experiment results in image based tasks like Computer Vision and more modern Deep Learning RCNN tasks. Given that these networks are trained on image inputs, it is proposed that the visual quality is more valuable than the physical accuracy in the context of image data generation. By using a modern graphical rendering engine designed for real time interaction, it is possible to develop simulations that are more analogous to the physical world in visual aspects [Wang et al., 2021]. A 2019 survey on data augmentation [Shorten and Khoshgoftaar, 2019] presents a taxonomy on several methods of data augmentation for machine learning applications, these methods include color space manipulation, noise injection, translations and similar image processing tasks for the use of expanding a limited dataset. While these approaches are helpful, even more complex solutions are noted to simply prevent overfitting [Takahashi et al., 2020], they do not increase the amount of information actually contained within the dataset.

2.1 Trained Autonomous Features

Previous work sought to categorise the functions in DNN based autonomous UAV projects [Lee et al., 2021], and organize them into a usable taxonomy. Based on the results of that literature review, the most common function projects sought to train was that of autonomous collision avoidance. Typically through CNN based, ResNet optimised strategies [Palossi et al., 2018]. It was determined that [Loquercio et al., 2018] was the most notable and uses steering angle and crash probability captured from ground based vehicles to train an autonomous collision avoidance policy in a quad-rotor helicopter UAV. Additionally, many projects intend to utilise transfer learning of generic image classification models such as the "ImageNet" dataset [Deng et al., 2009] for the function of Object Distinction, such as locating civilians in a collapsed building [Hartawan et al., 2019], or anomalies in natural environments [Yong and Yeong, 2018]. While not as popular in the UAV space, Classification as a task is an immensely popular application of Deep Learning in general, and in certain contexts Classification can serve as the basis for a Collision Avoidance solution. One of the key conclusions drawn from [Lee et al., 2021] was noting a distinct lack or papers that approached classifying, distinguishing. or generalising multiple different environments in a neural network. Environmental Distinction is considered by the author to be a key feature in the development path of the next generation of autonomous navigation.

2.2 Environment Specificity

One of the fundamental issues with using a neural network to solve a general task is the restriction placed on the network by its predefined complexity. When a network structure is defined in code it is also bounded to that structure which has a maximum mathematical complexity with which it can reach. Because of these limits, it is logical to assume that, given that the complexity of a task scales exponentially with the number of dimensions being considered on input, trying to generalize a complex task (for many use cases) using a bounded network can result in a decrease of specific (single use case) performance due to reaching that limit of the network's complexity as it is defined. This trade-off can be mitigated in design, through careful architecture decisions in the network or by tailoring the data being fed into the network into a more general context (converting an RGB image into grey scale before using it for training is an example of this). This is not usually an issue for rudimentary tasks, but when considering different environments in a CNN this becomes a much more distinct problem. A model's predictive capability is primarily dependant on the data being fed into it, specific datasets are often gathered or annotated manually by experts for use in the training of that task [Deng et al., 2009], this process is typically done in sessions as befits the researcher's schedule and as a result of this, uncontrollable environment parameters can change from session to session. This naturally biases the information to these uncontrollable parameters such as time, location, and weather and can lead to a less accurate, state-based representation of a dynamic area. While it is theoretically possible to sample the entire spectrum of an area's change and create a dataset containing all reasonable permutations of environmental parameters, it is unfeasible to do so manually, given that the addition of just one dimension leads to an exponential gain in the amount of data to be collected.

2.3 Simulating Data

Though traditional thinking would have it that in order to model decisions for use in the physical world, the dataset used for training must contain physical world data. However, artificial data has already been used as a means of base network training for adjustment later using real data [Vierling et al., 2020], and simulation as a means of generating datasets for learning tasks has been used for tasks such as Image Segmentation [Richter et al., 2016]. [Perri et al., 2022] presents a novel benefit and rationale for the use of simulation for data

Figure 2: Various bike assets used for dataset generation

Figure 3: Single position/rotation sliced by time axis in synthetic dataset

generation, the main concept being that using a simulation allows for greater control aspects than is possible in the traditional gathering process. Notably, that it is possible during simulation to simply pause the simulation and acquire all the necessary samples instantaneously (from a runtime perspective), without having to account for minor deviations in time or position due to hardware or human limitations. This effect also applies between samples, rather than wait for the sampling hardware to relocate to a new position as would be required in a physical sampling session, it is possible to move the camera and any attached sampling hardware to the next position immediately without having to ensure that the position is correct or make readjustments.

3 Method

Previous work [Lee et al., 2022] detailed the method regarding the acquisition of the 3D Scene and development of the simulator used for this experiment. To summarise, a 3D Scan of a physical location and a simulated quadcopter UAV camera is imported into a prototype simulator, referred to developmentally as "Fine, I'll Do It Myself" (FIDIM). FIDIM is being developed for the generation of visually realistic synthetic image data for Deep Learning using the Unity 3D Game engine which is common in Game Development as a framework due to the engines ease of use and robust package library. This scene was augmented using a small library of 3D Scanned bicycle assets (5 in total) to be used as classification subjects for the position/rotation log (see Fig. 2). Bicycle objects were chosen due the availability of models and "Bicycle" being a classifiable label by the base Yolov3 network used for the analysis in Section 4.

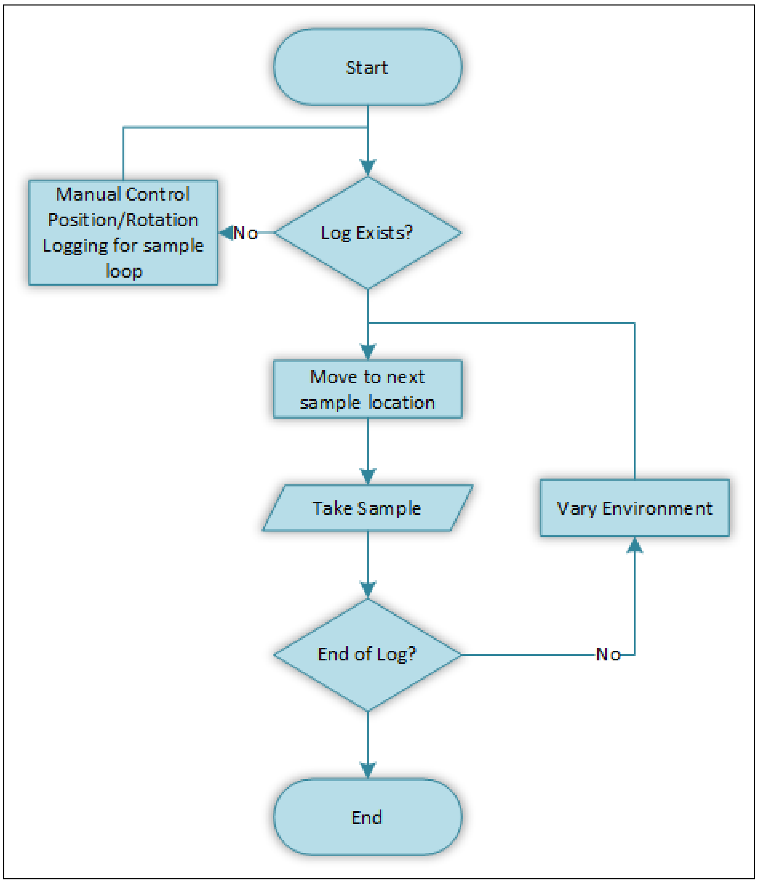

Figure 4: Automated sampling strategy flowchart, with environmental variance

3.1 Time Axis Simulation

Exhibiting different time effects in a visually realistic manner is key to the value of this dataset. Since the average light level in an given outdoor scene is primarily dictated by the time of day, with the season, and location having less significant impact. By using sufficiently realistic light simulation combined with adjustments to the texture lighting and skybox colour the transition from day to night can be simulated as seen in Fig. 3. From this simulated axis, 6 time configurations were selected. Further development into time axis simulation will account for geographic variation, seasonal change, and solar angle but were considered unnecessary for the initial simulation, as the time configurations are only relative to the simulations initial parameters at this time.

3.2 Sampling

The strategy for the generation of synthetic data samples is outlined in Fig. 4. After manual collection, the simulation will automatically generate the iterated dataset once the logs have been created and the chosen time variations are set. Additionally, each image that is generated contains the time iteration number and position number embedded in the filename for later reference, it is possible that other useful outputs can be contained in samples this way such as virtual lidar values or similar sensor data.

4 Results

For the first round of sampling, the virtual UAV was used to manually define a position and rotation log containing 411 entries was iterated over 6 time settings to create a dataset which contains a total of 2466 images. Each

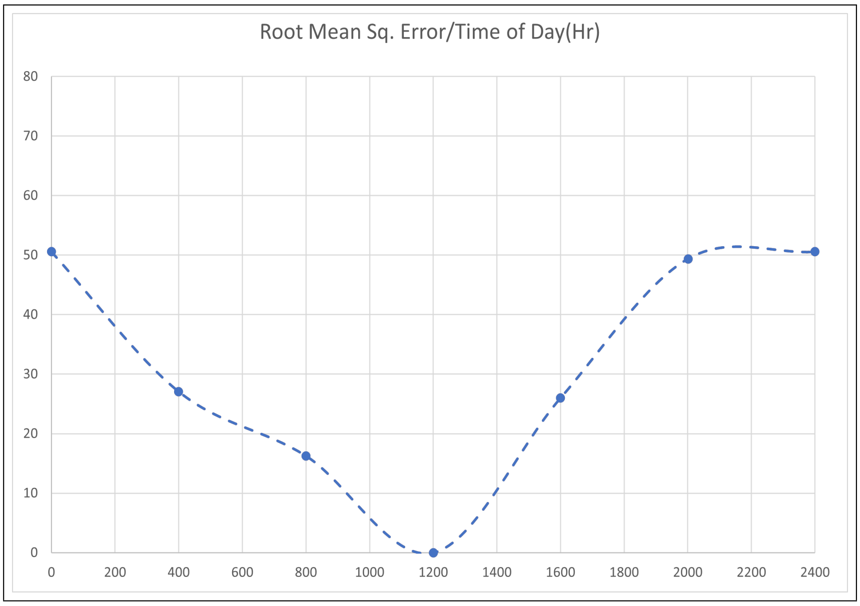

Figure 5: Linear graph of RMSE by Time of Day, represented as a percentage of deviation from ideal conditions, using base Yolov3 network classification

new position in the log creates 6 unique images related to that position, this allows for the data to be sliced in ways that are unfeasible in traditional sampling. These samples were then classified using the YOLOv3 pretrained network [Redmon and Farhadi, 2018], note that the pretrained network was used as is and no training was done using the simulated data. The highest average light in the simulation occurs at 1200hrs (noon) which is used as an ideal comparison for the rest of the time configurations. Average Root Mean Squared Error(RMSE) is then calculated by comparing the classification results from a given time variation sample to the ideal time sample.

4.1 Comments

For initial sampling, four equidistant points on the time axis; 6 hours, 12 hours, 18 hours and 24 hours respectively. However, these time configurations were too distant to provide adequate insight. In order to generate more informative samples, the number of configurations was increased to ensure a 4 hour gap between points along the time axis (see Fig. 5). It is important to note that since the graph is represented linearly but is cyclical in reality, '0' and '2400' on the x axis are the same point and thus have the identical Average RMSE, additionally, noon was expected to yield an RMSE of 0 since it is the point of comparison from which the RMSE of the 5 other time configurations are calculated.

5 Conclusion

This work sought to investigate the use of custom, photo realistic simulation and synthetic data generation as method of measuring deviation in the classification probability of the commonly used YOLOv3 pretrained network. The dataset, which is generated across a simulated time axis, when evaluated showed an increase in the Average RMSE of the network response peaking at 50.56% at the darkest point on the time axis being 2400hrs (simulated midnight), which is to be expected based on how the average light level (and thus visibility) in a scene changes as the cyclical time axis progresses from day into night. Since object visibility is critical to the task of classification, daylight level is shown to be of high impact to the performance of the classification network even when the change in simulated time is quite minimal.

5.1 Future Work

This project is an initial part of a series of investigations on the creation of a spatially linked area which is then simulated in a modern 3D engine that has been purpose-built for the task of creating reality-analogous synthetic image samples. Future development is aimed at generating simulated data that iterates over other environmental axes such as weather, the creation of a locally accessible "Digital Twin Area" to serve as a test-bed for Autonomous Navigation Projects, and increasing the overall quality of the simulation. Additionally, further analysis into methods of verifying trained network performance, such as Uncertainty estimation is being considered.

6 Acknowledgments

This project was funded by the ADVANCE CRT PhD Program.

References

[Al-Sharman et al., 2020] Al-Sharman, M. K., Zweiri, Y., Jaradat, M. A. K., Al-Husari, R., Gan, D., and Seneviratne, L. D. (2020). Deep-learning-based neural network training for state estimation enhancement: Application to attitude estimation. *IEEE Transactions on Instrumentation and Measurement*, 69(1):24–34.

[Alshehri et al., 2019] Alshehri, A., Member, S., Bazi, Y., and Member, S. (2019). Deep Attention Neural Network for Multi-Label Classification in Unmanned Aerial Vehicle Imagery. *IEEE Access*, 7:119873–119880.

[Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Kai Li, and Li Fei-Fei (2009). ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, number June, pages 248–255. IEEE.

[European Union Aviation Safety Agency, 2020] European Union Aviation Safety Agency (2020). Special Condition for Light Unmanned Aircraft Systems - Medium Risk INTRODUCTORY. Technical report, EASA.

[Hartawan et al., 2019] Hartawan, D. R., Purboyo, T. W., and Setianingsih, C. (2019). 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*, pages 105–111.

[Lee et al., 2021] Lee, T., Mckeever, S., and Courtney, J. (2021). Flying Free: A Research Overview of Deep Learning in Drone Navigation Autonomy. *Drones*, 5(2):52.

[Lee et al., 2022] Lee, T., Mckeever, S., and Courtney, J. (2022). Generating Reality-Analogous Datasets for Autonomous UAV Navigation using Digital Twin Areas. In *2022 33rd Irish Signals and Systems Conference (ISSC)*, pages 1–6. IEEE.

[Loquercio et al., 2020] Loquercio, A., Kaufmann, E., Ranftl, R., Dosovitskiy, A., Koltun, V., and Scaramuzza, D. (2020). Deep Drone Racing: From Simulation to Reality With Domain Randomization. *IEEE Transactions on Robotics*, 36(1):1–14.

[Loquercio et al., 2018] Loquercio, A., Maqueda, A. I., Del-Blanco, C. R., and Scaramuzza, D. (2018). DroNet: Learning to Fly by Driving. *IEEE Robotics and Automation Letters*, 3(2):1088–1095.

[Palossi et al., 2018] Palossi, D., Loquercio, A., Conti, F., Flamand, E., Scaramuzza, D., and Benini, L. (2018). A 64mW DNN-based Visual Navigation Engine for Autonomous Nano-Drones. *IEEE Internet of Things Journal*, 6(5):8357–8371.

[Perri et al., 2022] Perri, D., Simonetti, M., and Gervasi, O. (2022). Synthetic data generation to speed-up the object recognition pipeline. *Electronics (Switzerland)*, 11(1):1–19.

[Redmon and Farhadi, 2018] Redmon, J. and Farhadi, A. (2018). YOLOv3: An Incremental Improvement. *CoRR*, 1804.02767.

[Richter et al., 2016] Richter, S. R., Vineet, V., Roth, S., and Koltun, V. (2016). 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)*, 9906 LNCS:102–118.

[Rojas-Perez and Martinez-Carranza, 2020] Rojas-Perez, L. O. and Martinez-Carranza, J. (2020). DeepPilot: A CNN for Autonomous Drone Racing. *Sensors*, 20(16):4524.

[Shorten and Khoshgoftaar, 2019] Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, 6(1):60.

[Takahashi et al., 2020] Takahashi, R., Matsubara, T., and Uehara, K. (2020). Data Augmentation Using Random Image Cropping and Patching for Deep CNNs. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(9):2917–2931.

[Udacity, 2016] Udacity (2016). An Open Source Self-Driving Car.

[Vierling et al., 2020] Vierling, A., Sutjaritvorakul, T., and Berns, K. (2020). Dataset Generation Using a Simulated World. In *International Conference on Robotics in Alpe-Adria Danube Region*, pages 505–513.

[Wang et al., 2021] Wang, Z., Han, K., and Tiwari, P. (2021). 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*, pages 180–183.

[Yang and Wang, 2020] Yang, R. and Wang, X. (2020). UAV Landmark Detection Based on Convolutional Neural Network. *2nd IEEE Eurasia Conference on IOT, Communication and Engineering 2020, ECICE 2020*, 0(1):5–8.

[Yong and Yeong, 2018] Yong, S. P. and Yeong, Y. C. (2018). 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*, pages 1–5. IEEE.